SPECIAL EDITION | www.ScienceRobotics.org

Science Robotics



Celebrating 10 Years of Publishing Cutting Edge Research in Robotics

AAAS

SPECIAL EDITION: *Science Robotics*



Credit: Christian Ziegler, Máté Nagy, and Liang Li

This booklet features both front matter and research content recently published at *Science Robotics* highlighting advances in wearable robots, motion planning and control, social robots, robot locomotion, as well as robots for understanding basic science. The journal covers traditional disciplines in robotics such as soft robots, grasping and manipulation, but also emerging trends in human robot interaction and ethics in artificial intelligence agents.

Science Robotics | AAAS

Science Robotics publishes original, peer-reviewed, science and engineering-based research articles that advance the field of robotics bearing the quality hallmark of the *Science* family of journals.

Editor-in-Chief
H. Holden Thorp
AAAS, Washington, DC

Editor
Amos Matsiko, PhD
AAAS, Cambridge, UK

Associate Editor
Melisa Yashinski, PhD
AAAS, Washington, DC

Publisher/*Science* family of journals: **Bill Moran**
AD/Business Development: **Justin Sawyers**
Senior Marketing Manager: **Shawana Arnold**
Layout/Design: **Kim Huynh**

Advertising inquiry information:
science_advertising@aaas.org

IN THIS BOOKLET:

FOCUS

3 **Equalizing access: How robotics and AI can transform surgical care worldwide**
Marta Weber *et al.* (Salim Afshar)

VIEWPOINT

5 **Data will solve robotics and automation: True or false?": A debate**
Nancy M. Amato *et al.* (Ken Goldberg)

REVIEW ARTICLES

9 **A review of learning-based dynamics models for robotic manipulation**
Bo Ai *et al.* (Yunzhu Li)

24 **Will your next surgeon be a robot? Autonomy and AI in robotic surgery**
Samuel Schmidgall *et al.* (Axel Krieger)

RESEARCH ARTICLES

41 **Deep domain adaptation eliminates costly data required for task-agnostic wearable robotic control**
Keaton L. Scherpereel *et al.* (Aaron J. Young)

55 **Robotic locomotion through active and passive morphological adaptation in extreme outdoor environments**
Max Polzin *et al.* (Josie Hughes)

71 **Social robots as conversational catalysts: Enhancing long-term human-human interaction at home**
Huili Chen *et al.* (Hae Won Park)

83 **Reverse engineering the control law for schooling in zebrafish using virtual reality | Science Robotics**
Liang Li *et al.* (Iain D. Couzin)

93 **Explosion-powered eversible tactile displays**
Ronald H. Heisser *et al.* (Robert F. Shepherd)

101 **RoboBallet: Planning for multirobot reaching with graph neural networks and reinforcement learning**
Matthew Lai *et al.* (Jonathan Scholz)

Scientific Advisory Board Members

Henrik I. Christensen
University of California San Diego, USA

Steve Collins
Stanford University, USA

Marco Dorigo
Université Libre de Bruxelles, Brussels, Belgium

Pierre E. Dupont
Boston Children's Hospital, Harvard Medical School, USA

Peer Fischer
Engineering and Advanced Materials, Heidelberg University, Germany

Dario Floreano
EPFL, Switzerland

Toshio Fukuda
Nagoya University, Meijo University, Japan and Beijing Institute of Technology, China

Robert J. Full
University of California at Berkeley, USA

Ken Goldberg
University of California, Berkeley, USA

Daniel Goldman
Georgia Institute of Technology

Ayanna Howard
The Ohio State University, USA

Robert D. Howe
Harvard University, USA

Koji Ikuta
University of Tokyo, Japan

Rachael Jack
University of Glasgow, UK

Lydia E. Kavraki
Rice University, USA

Danica Kragic
Royal Institute of Technology, Stockholm, Sweden

Cecilia Laschi
National University of Singapore

Hod Lipson
Columbia University, USA

Robin R. Murphy
Texas A&M University, USA

Bradley Nelson
ETH Zurich, Switzerland

Allison Okamura
Stanford University, USA

Selma Šabanović
Indiana University, USA

Yulia Sandamirskaya
Zurich University of Applied Sciences

Hannah Stuart
University of California at Berkeley, USA

Yu Sun
University of Toronto, Canada

Douglas Weber
Carnegie Mellon University, USA

Li Wen
Beihang University

Robert J. Wood
Harvard University, USA

Guang-Zhong Yang
Shanghai Jiao Tong University, China

Learn more and submit your research today: ScienceRobotics.org

© 2026 by The American Association for the Advancement of Science. All Rights Reserved.

MEDICAL ROBOTS

Equalizing access: How robotics and AI can transform surgical care worldwide

Marta Weber[1]*, Kee B. Park[2]†, Salim Afshar[2,3]†

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

The integration of robotics and artificial intelligence holds promise for improving access to surgical care worldwide.

Nine of 10 people in low- and middle-income countries (LMICs) lack safe and affordable surgical and anesthesia services when needed (*1*). Of the 313 million surgical procedures performed annually, only 6% occur in these areas, and this lack of access contributes to an estimated 17 million deaths annually (*1*). The need for affordable, timely medical care in LMICs is more critical than ever given that morbidity and mortality continue to rise.

Meanwhile, in high-income regions, the use of robotic surgery increased from 20 to 49% between 2012 and 2024 (*2*), shifting from traditional open procedures to less invasive laparoscopic methods, resulting in reduced pain, bleeding, shorter hospital stays, and faster recovery times while improving surgical precision.

Similarly, artificial intelligence (AI) continues to revolutionize the medical field by optimizing patient care through predictive analytics, improving diagnostic accuracy, personalizing treatment, and identifying high-risk patients. In addition, the ongoing advancements in large language models and multimodal large-scale AI can streamline health care delivery by optimizing resource allocations, automating appointments, processing billing, and managing medical records (*3*). At the Mayo Clinic, AI tools have been integrated into neurology and cardiovascular specialties to analyze computerized tomography (CT) data after strokes and interpret electrocardiograms (ECGs) to predict heart failure or detect atrial fibrillation (*4*).

Despite advancements, access to surgical care in LMICs continues to worsen, and these technologies remain limited. Implementing robotics and AI in underserved regions could help bridge the health care gap by reducing workforce shortages, automating administrative tasks, and enhancing surgical education through simulation-based training. In addition, AI-augmented tools such as digital image processing on mobile devices can offer cost-effective, rapid, and accurate diagnoses while also predicting complications and tailoring care plans through preventive measures (*5*, *6*). Moreover, AI-powered telemedicine platforms facilitate remote consultations, allowing distant specialists to assist local surgeons in real time (*5*). Meanwhile, robotic telesurgery allows experienced surgeons to operate remotely, giving LMICs access to expert care without requiring full on-site infrastructure. Minimally invasive robotic surgery can also reduce infection risk, discomfort, hospitalization time, and costs (*7*).

Some initiatives have shown promising results, although they remain limited in scale, and widespread adoption in LMICs has yet to materialize. Proximie, a telepresence surgical platform, has facilitated remote guidance during surgical procedures in Tanzania, a potential game changer for educational and training purposes. However, broader implementation is still lacking. Similarly, transcontinental robotic telestenting for cardiac procedures has been successfully demonstrated (*8*), and the use of robots for telesurgery, even over 5G networks, is currently being validated (*9*). In addition, projects such as the SAVE (Surgery: Assess/Validate/Expand) Initiative, which focuses on AI-driven surgical innovations for global health, highlight the importance of addressing the gap in essential surgery services through scalable, technology-driven solutions (*10*).

We believe it is only a matter of time before telesurgery becomes an acceptable mode of providing surgical care. The critical question remains: Who will benefit?

CHALLENGES

The fundamental challenges to implementing robotics and AI in LMICs are well known: high costs, insufficient infrastructure, limited technical expertise, and ethical concerns. The initial investment for a new surgical system can exceed $1 million, with additional costs per procedure and ongoing maintenance expenses (*7*). Moreover, training gaps and infrastructure constraints, including unreliable electricity, network issues, and inadequate operating rooms, further limit the adoption of these technologies. Telesurgery, in particular, relies on stable high-speed internet, which is often inconsistent in underserved areas. In addition, the lack of professionals trained in maintaining complex robotic systems further hinders implementation. Addressing AI bias and ethical concerns is equally critical, given that algorithms may underrepresent minority populations and overlook ethnic variations in disease predisposition, potentially resulting in disparities in decision-making and outcomes. Furthermore, language barriers and mistrust in foreign-led technologies may impede acceptance, underscoring the need for clear guidelines on responsibility, liability, and informed consent, along with robust measures to ensure data privacy and security. Overcoming these barriers is essential for the successful integration of AI and robotics in LMICs.

FROM BARRIERS TO SOLUTIONS

Bridging the gap between AI/robotics and global surgery communities through a collaborative and multifaceted approach is key. Expert leaders from both fields can accelerate knowledge exchange by collaborating at events ranging from major conferences to webinars, as exemplified by the 2024 Hamlyn Symposium on Medical Robotics, which aims to strengthen interdisciplinary collaboration. Establishing research partnerships between AI and surgical experts is vital for driving meaningful innovation, and academic journals can

[1]University of Geneva, Geneva, Switzerland. [2]Harvard Medical School, Boston, MA, USA. [3]Harvard School of Dental Medicine, Boston, MA, USA.
*Corresponding author. Email: marta.weber.mdt@gmail.com
†These authors contributed equally to this work.

actively support these efforts by dedicating sections to the intersection of AI, robotics, and global surgery. Moreover, creating joint educational programs between universities from high-income countries and medical schools in LMICs can substantially enhance global knowledge sharing through structured exchange opportunities. International collaborative clinical trials between hospitals in high-income countries and LMICs can effectively evaluate the performance of AI-driven technologies in diverse populations, ensuring their global applicability. Funders play a pivotal role by providing grants for projects aimed at promoting surgical equity in LMICs. Accelerating the implementation of telesurgery in LMICs requires forming international partnerships focused on capacity building, creating scalable training programs, and integrating low-cost robotic alternatives. Strengthening telemedicine and telerobotics through investments in network infrastructure, especially by enhancing 5G access and piloting telerobotic programs in LMIC hospitals, is essential. In addition, implementing fail-safe protocols and ensuring the presence of on-site medical personnel can help manage unexpected complications and system failures. Last, increasing industry investment in research and development (R&D) for innovative AI and robotic solutions tailored to LMICs' unique challenges, such as affordable robotic alternatives and subsidized equipment, can help bridge the gap between advanced technology and underserved populations.

The integration of robotics and AI into global surgery offers promising opportunities to improve health care, particularly in LMICs, by making surgical care more accessible, efficient, and safe. However, to realize this promise, the two communities need to be intentional in collaborating. The future of global surgery relies on our ability to harness these technologies for the benefit of all. As we move forward, one key guiding question should be how technology can be used to better deliver surgical care to those in need.

REFERENCES AND NOTES

1. T. M. Diehl, E. Soto, J. A. Laryea, S. N. Zafar, Surgery as a global health need. *Clin. Colon Rectal Surg.* **35**, 362–370 (2022).
2. National Institute for Health and Care Excellence (NICE), "Cutting-edge robotic surgery gets green light as 11 systems are recommended," (2025); https://nice.org.uk/news/articles/cutting-edge-robotic-surgery-gets-green-light-as-11-systems-are-recommended.
3. M. Shuliak, "AI in healthcare: Examples, use cases & benefits," *Acropolium*, 6 February 2025; https://acropolium.com/blog/ai-in-healthcare-examples-use-cases-and-benefits/.
4. Mayo Clinic, Artificial intelligence (AI) in cardiovascular medicine—overview (2022); https://mayoclinic.org/departments-centers/ai-cardiology/overview/ovc-20486648.
5. K. Malhotra, B. N. X. Wong, S. Lee, H. Franco, C. Singh, L. A. Cabrera Silva, H. Iraqi, A. Sinha, S. Burger, D. S. Breedt, K. Goyal, M. M. Dagli, A. Bawa, Role of artificial intelligence in global surgery: A review of opportunities and challenges. *Cureus* **15**, e43192 (2023).
6. A. D. Oliveira, C. Prats, M. Espasa, F. Zarzuela Serrat, C. Montañola Sales, A. Silgado, D. L. Codina, M. E. Arruda, J. G. I. Prat, J. Albuquerque, The malaria system MicroApp: A new, mobile device-based tool for malaria diagnosis. *JMIR Res. Protoc.* **6**, e70 (2017).
7. A. Mehta, J. Cheng Ng, W. Andrew Awuah, H. Huang, J. Kalmanovich, A. Agrawal, T. Abdul-Rahman, M. M. Hasan, V. Sikora, A. Isik, Embracing robotic surgery in low- and middle-income countries: Potential benefits, challenges, and scope in the future. *Ann. Med. Surg. (Lond.)* **84**, 104803 (2022).
8. R. D. Madder, S. VanOosterhout, J. Parker, K. Sconzert, Y. Li, N. Kottenstette, A. Madsen, J.-M. Sungur, P. Bergman, Robotic telestenting performance in transcontinental and regional pre-clinical models. *Catheter. Cardiovasc. Interv.* **97**, E327–E332 (2021).
9. G. Moustris, C. Tzafestas, K. Konstantinidis, A long distance telesurgical demonstration on robotic surgery phantoms over 5G. *Int. J. Comput. Assist. Radiol. Surg.* **18**, 1577–1587 (2023).
10. Wellcome Leap, "SAVE"; https://wellcomeleap.org/save/.

Competing interests: K.P. owns advisory shares of Hoth Intelligence, a surgical guidance company. All other authors declare that they have no competing interests.

10.1126/scirobotics.adt6471

ARTIFICIAL INTELLIGENCE

"Data will solve robotics and automation: True or false?": A debate

Nancy M. Amato[1], Seth Hutchinson[2], Animesh Garg[3], Aude Billard[4], Daniela Rus[5], Russ Tedrake[5], Frank Park[6], Ken Goldberg[7]*

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

Leading researchers debate the long-term influence of model-free methods that use large sets of demonstration data to train numerical generative models to control robots.

INTRODUCTION

Nancy M. Amato, Seth Hutchinson, and Ken Goldberg

The robotics and automation research community is experiencing what historian of science Thomas Kuhn would call a "paradigm shift" (*1*). This was the subject of a featured debate at the IEEE International Conference on Robotics and Automation (ICRA), the largest research conference for the field, where many presentations celebrated algorithmic ("model-based") breakthroughs over the decades. However, since 2012, advances in deep "neural" networks that combine unprecedented amounts of example data, advances in stochastic gradient descent techniques, and advances in computing, in particular GPUs (graphics processing units), have produced remarkable results in computer vision and speech recognition. And with the emergence of ChatGPT and associated large vision language models (VLMs) in 2022, with associated advances in natural language processing and denoising diffusion methods, the paradigm of "end-to-end" (also known as "model-free") approaches to perception and control suggest an entirely new, data-driven approach to robotics and automation.

Some researchers believe, at one extreme, that the data-driven paradigm is overrated, and some at the other extreme believe that model-based approaches are outdated and should be retired. Many researchers take less extreme positions, believing both paradigms have merit and that the model-based and model-free paradigms can coexist (*2*).

As cochairs of the 2025 IEEE International Conference on Robotics and Automation (ICRA), held in Atlanta GA, USA, 26 to 29 May 2025, N.M.A. and S.H. decided to host a conference-wide keynote debate in the last session of the conference. To moderate the debate, we invited K.G., who has been very active in the past 40 years of model-based robotics and automation research and also active in robot learning (recently elected president of the Robot Learning Foundation). K.G. formulated a catchy title and short description: "Will the future of robotics and automation be written in code or in data? Is the *Handbook of Robotics* obsolete? Are home humanoids overhyped? Join us for a high-voltage debate about physics vs pixels, theory vs terabytes."

The three of us then worked to identify a panel of six debaters, inviting three to represent each side in the spirit of the classic Oxford debates. Six highly respected researchers accepted our invitation: A.G., D.R., and R.T. advocating for "true," and A.B., Leslie Kaebling (of MIT), and F.P. advocating for "false." We asked each debater to take extreme positions and prepare 3-min introductory presentations, responses to questions, and closing remarks.

The debate was very well attended—more than 1000 audience members cheering or jeering as debaters alternated between the two positions. The debaters focused on software and control and did not address the many unsolved issues related to hardware, materials, and design, nor did they have time to discuss how such massive datasets can be collected and processed.

Interestingly, most of the debaters conveyed a shared respect for hybrid methods that ultimately combine model-free and model-based methods. Ultimately, a show of hands indicated that not many audience members were persuaded to change their initial positions, but everyone was pushed to think more deeply about the new data paradigm.

DATA ARE NOT MERELY BENEFICIAL; RATHER, THEY ARE INDISPENSABLE AND FOUNDATIONAL

Animesh Garg

Humanity has frequently developed "know-how" before the "know-why." We engineered technologies mainly using observational understanding—such as metalworking, sailing ships, internal combustion engines, and airplanes (all foundational to civilization)—yet preceding a thorough understanding of their underlying sciences.

We are in the midst of building next-generation technology—"computational and physical intelligence" with data-driven empirical methods. Modern computing has only recently emerged as a tool for scientific discovery, yet it fundamentally redefines both the acquisition of know-how and the pursuit of know-why. We can now create and collect planetary-scale data and perform comprehensive computations to compress them into a single model. This unprecedented speed and scale has upended our approach to scientific inquiry compared with any other point in human history.

In our quest to understand intelligence and "solve" robotics, I contend that data are not merely beneficial; rather, they are indispensable and foundational. This necessity stems from several critical perspectives:

[1]University of Illinois, Urbana, IL, USA. [2]Northeastern University, Boston, MA, USA. [3]Georgia Institute of Technology, Atlanta, GA, USA. [4]École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. [5]Massachusetts Institute of Technology, Cambridge, MA, USA. [6]Seoul National University, Seoul, South Korea. [7]University of California, Berkeley, Berkeley, CA, USA.
*Corresponding author. Email: Goldberg@berkeley.edu



The debaters. Top row (left to right): Nancy M. Amato, Seth Hutchinson, and Ken Goldberg. Bottom row (left to right): Animesh Garg, Aude Billard, Russ Tedrake, and Frank Park.

1) General-purpose intelligence, particularly in physical robots, is ambiguous, open ended, and underspecified. Data offer guidance for achieving distributional similarity where a closed-form solution for common sense is absent. Moreover, self-supervised learning—through observation or experience—allows for the encoding of concepts without manual programming. Last, learning with sparse yet critical feedback enhances robustness in observed behaviors and facilitates the discovery of novel ones.

2) Accurate models foster innovation. Conversely, incomplete models can hinder the discovery of true optima, because these might be unrepresentable or unattainable with existing optimization tools. Our understanding of generalizable autonomy in robotics is currently insufficient to define even necessary conditions, let alone sufficient ones.

3) Robotics represents the evolution of computing, moving beyond desktop, mobile, and cloud environments into the physical world. Whereas data-driven approaches are often criticized for lacking elegance, the empirical process of data collection, creation, and curation is inherently a scientific endeavor. Moreover, the construction of complex technologies like foundation models involves intricate processes, despite their seemingly monolithic outputs. Therefore, we must apply principles of abstraction from other computing domains to build sophisticated yet well-behaved robotic systems.

4) Data-driven methods have successfully unified diverse problems in fields like language processing and computer vision. However, robotics remains a collection of disparate communities. A potential solution lies in adopting the foundation model perspective, which could unify many seemingly unrelated problems within the robotics community, given that all robots are governed by the same fundamental laws of physics.

Generalizable autonomy for physical systems is an open-ended problem that requires us to find any necessary solution before we can develop a more elegant, interpretable, and efficient approach. Our endeavors in creating data-driven technological artifacts are merely one step in enhancing our comprehension of the essence of intelligence itself.

MODELS GIVE MEANING TO DATA AND ARE ESSENTIAL FOR UNDERSTANDING ROBOT CONTROL

Aude Billard

All science is grounded in data, but data alone are insufficient. I do not believe that data alone will "solve" robotics and automation.

Consider astronomy: It started off with the simple observation of the sky. But thousands of people look at the sky each day, and millions have before. Although they all marvel at its beauty, most of these people, if asked, would be incapable of explaining the reason why the stars are aligned as they are or distinguishing stars from planets. Astronomers from ancient civilizations ago did not use these millions of human eye observations to decipher the underlying laws of physics. Rather, they used few but meticulously gathered data points. Most importantly, they constantly went back and forth from building hypothetical models to data to test their theories.

Some argue that what was true for hard-core science does not apply to robotics. Robotics, and its deployment through large-scale automation, is an engineering discipline, relying on experimentation and numerical models tailored to the specific platforms being controlled.

Robotics today seems like an impatient child who wants it all, now and at all costs. It dreams of deploying a magic tool that will control a variety of robotic bodies from flying to walking and swimming machines and endow them with large language models' level of cognition in the flesh, enabling them to interact seamlessly with all humans, from children to the elderly, and to operate outdoors and indoors in all weather, with minimal changes to our homes and habits.

If this is what is meant by solving robotics, then the task may seem unsolvable with a traditional scientific approach, given that none of the sciences we know today can capture, through mathematical laws, all of the world's complexity—let alone the complexity of human beings. It is hence understandable to seek an alternative road.

The success of large language models relies on the fact that humans had put at the disposal of computers billions of pieces of information—from entire encyclopedias to largely uncurated material—all of which provided a view of a large chunk of human knowledge that can be put into words. Should robotics approach the problem in a similar way to achieve similar levels of knowledge, it needs to encapsulate all human and other

animals' knowledge of perceiving and doing things. This would require embodying those creatures. However, existing interfaces for transferring knowledge from humans to robots are still cumbersome and intrusive and offer only a crude approximation of human sensing, and we are still a long way from embodying birds, fishes, or worms.

Yet, suppose robotics does succeed in this endeavor, but only through sheer computational power applied to vast datasets, without the development of any underlying general theoretical principles. Then, the [Springer] *Handbook of Robotics* would not become obsolete. On the contrary, it might become our only cherished resource, because we may no longer be able to produce a third edition, or any new handbook for that matter, beyond coding guidelines for how best to use the latest artificial intelligence (AI) toolbox.

We started by contrasting data from models, debating which may matter more. Perhaps what matters is neither alone but rather the interplay between gathering data and refining models that is essential to building an understanding of how robots can act in the world. Failing this, we may just be building another tower of Babel.

I BELIEVE THAT WE NEED BOTH DATA AND MATHEMATICAL MODELS

Daniela Rus

Although I was assigned to argue in favor of data "solving" robotics and automation, my actual position is more nuanced: I believe that we need both data and mathematical models. Robotics has long drawn strength from first principles. Physics gives us elegant models for simple tasks: grasping a rigid object, stabilizing a robot arm, or following a collision-free path. When the environment is structured, the materials known, and the goals well defined, analytical models deliver precision and insight. The moment we leave the lab and enter the real world, those assumptions collapse. Grasping becomes "pick the right object out of a cluttered drawer." Navigation becomes "move through a crowded room without bumping into people—or social norms." In these domains, first principles alone are no longer enough.

The challenge is complexity. Real-world tasks are multimodal, context-dependent, and filled with ambiguity. They involve soft, deformable objects; occluded or partial views; unpredictable humans; and environments that shift over time. There is no single equation for how to unload a dishwasher in a new home or to hand a tool to someone who may—or may not—reach out to take it. To function in these settings, robots need more than structure—they need experience. And experience comes from data.

Data allow us to capture the richness and complexity of the real world—complexity that first-principles models alone cannot fully express. They provide breadth, variation, and direct engagement with reality. Through data, we can model tasks that are too intricate to formalize, capture interactions too subtle to simulate, and learn behaviors too diverse to program by hand. Data enable robots to build representations that reflect the ambiguity and variability of real environments—accounting for uncertainty, human intent, and the fluidity of physical execution. Most critically, data allow robots to respond intelligently when things go wrong—not by following a prescripted rule, but by drawing on prior experience. They do not recover because we told them what to do; they recover because they have seen something like it before.

Data also bring resilience and enable robots to operate under noisy perception, ambiguous instructions, and changing dynamics. With enough structured experience, AI systems can learn to filter uncertainty, infer intent, and adapt online. Just as large language models gained generality from vast, varied text, robots will require diverse, richly labeled physical experience to achieve robust intelligence—not just more video, but multisensory, embodied data: joint torques, contact forces, hand trajectories, and human gazes.

That is why, in our lab, we have built a physical kitchen task testbed that goes far beyond passive observation. We instrument people with sensors that record body pose, muscle activity, force interactions, and gaze patterns as they perform real tasks—cutting, pouring, reaching, and handing off. These recordings generate dense, multimodal datasets that capture the what and the how of human movement. From this, we train AI systems that learn to perform those same tasks—gracefully, robustly, and with the flexibility to adapt when the environment changes.

As I noted above, I also believe that mathematical models and first principles still matter. They help structure the learning process, provide constraints, and enforce physical consistency. But without data, we cannot capture the long tail of variation that defines the real world. The path to capable, general-purpose robots is not paved solely in equations—it is built on experience. To solve robotics and automation, we need systems that combine the rigor of physics with the richness of data. That is how we move from brittle to robust, from reactive to adaptive, from lab demos to real-world autonomy.

Ultimately, solving robotics and automation requires a marriage of disciplines. The rigor of physics gives us constraints and structure. The breadth of data gives us coverage and flexibility. One without the other is incomplete. If we want robots that thrive not just in factories but also in homes, hospitals, and everyday human environments, we must invest in the data that capture the richness of those worlds. The future of robotics will be driven by equations alongside experience.

MANIPULATION REQUIRES "COMMON SENSE"; LARGE DATA AND LARGE MODELS CAN PROVIDE IT

Russ Tedrake

I do believe that the next most impactful steps toward "solving" robotics lie in large-scale data collection and large pretrained models. How could I not? The advances in large models for language, vision, and now video are undeniable. For the last few years at Toyota Research Institute (TRI) and other labs, I have seen more and more examples of robots programmed via imitation learning performing dexterous manipulation tasks that would have been impossible to imagine even a few years ago; most recently, we have empirically validated that multitask pretraining improves robustness and/or allows us to add new skills with fewer demonstrations and that performance predictably increases as the scale and diversity of the pretraining data grow (*3*, *4*).

Why are data so fundamental to this success? Large-scale pretraining from diverse multitask data is the best (only?) way that I know to program "common sense." ChatGPT gets this from reading all of the text on the internet, and our robots are slowly getting physical common sense from seeing enough demonstrations of manipulation. Most people agree that large models are useful for manipulation tasks that require common-sense language understanding (e.g., "pick up the extinct animal"), but many people underestimate how essential common sense is for low-level control.

For example, at TRI, we recently taught a bimanual robot with simple grippers to core and slice an apple using kitchen tools. No two apples are the same, and the diversity of rollouts that we observed in this experiment

was impressive. Most impressive to me were the subtle recovery maneuvers that the robot could make when apple pieces would slip or squirt around on the cutting board. Having spent enough time myself manually programming state machines for tasks like this, it is incredibly powerful to see how quickly we can add recovery behaviors to the demonstration dataset to improve the performance; as these recovery maneuvers start to appear automatically through pretraining via transfer from demonstrations in completely different tasks, we start to have what I would call common sense for dexterous manipulation. This suggests a clear path to true open-world robustness.

For me, "solving" robotics is a very long-term agenda. Well after we have general-purpose robots out in the open world performing useful work, we will still have a lot more work to do to deeply understand the laws of robotics. We are in the early phases, where everything feels messy. I am optimistic that we will quickly deepen our empirical understanding of these new capabilities and that we will eventually have theorems that guide us. However, precisely because the world is so messy and because so much of manipulation runs through common sense, I think we will need large-scale data and algorithms that can digest them, in order for humans (or machines) to build the right simple models to help us deeply understand.

A MORE GROUNDED STRATEGY IS NEEDED

Frank Park

There is no doubt that multimodal foundation models represent a transformational moment in human history, but expecting a parallel revolution in robotics is premature at best and wishful thinking at worst. Robotics is a different beast entirely: Real-world data are scarce, simulations remain unreliable, and robots in their many embodiments must contend with an endless variety of environments and tasks. Real-world performance requirements can be unforgiving, to say nothing of the massive computing, energy, and data demands of today's foundation models.

None of this is to say that foundation models for robotics are not worth pursuing—they are, but a more grounded strategy is needed. Increasingly, machine learning researchers are questioning whether scaling deep learning alone is enough; Goyal and Bengio (*5*) have argued that better inductive biases (priors, preferences, or assumptions) are the key to human-level cognition and its corresponding hierarchy of representations. I believe the same is true for robotics.

Fortunately, in robotics, we already have a wealth of such inductive biases. Biology offers insights into how humans control movement through layers of abstraction, as well as the interplay between feedforward versus feedback and internal versus external coordinates. Physics provides models of motion, force, and compliance. Geometry helps to simplify complex movements into lower-dimensional representations and to exploit any underlying symmetries. Programming gives us tools like motion description languages that formalize manipulation tasks in a device-independent manner. Current robot foundation models do not sufficiently exploit these inductive biases.

Instead of collecting endless hours of video footage on how to clean a bathroom or assemble a chair, we would do better to study tasks where the nuances of control are everything—like how a chef slices a carrot or a cucumber. What matters is not just the motion, but the tactile feedback at the fingertips, the compliance in the wrist and elbow, and the complex forces involved at each stage. Manipulation tasks like these are precisely the kinds of challenges where traditional model-based methods fall short; foundation models trained with the right data can truly make a difference.

Let us also not forget: Convolutional neural networks (CNNs) and transformers were not some serendipitous discovery of blind scaling. They were designed by researchers who understood the importance of symmetry and equivariance (translation for CNNs, permutations for transformers). Robotics deserves the same level of intentional design. Rather than force-fitting models developed for vision and language to robots, new models that capture the inductive biases inherent to robotics are needed. It is time we treated robotics not as an extension of deep learning but as its own grand challenge.

REFERENCES AND NOTES

1. T. Kuhn, *The Structure of Scientific Revolutions* (Univ. Chicago Press, 1962).
2. K. Goldberg, How to close the 100,000 year data gap in robotics and automation? *Sci. Robot.* **10**, eaea7390 (2025).
3. R. Tedrake, "Multitask transfer in TRI's large behavior models for dexterous manipulation," Stanford seminar, 25 April 2025; available at https://youtube.com/watch?v=TN1M6vg4CsQ.
4. TRI LBM Team, J. Barreiros, A. Beaulieu, A. Bhat, R. Cory, E. Cousineau, H. Dai, C.-H. Fang, K. Hashimoto, M. Zubair Irshad, M. Itkina, N. Kuppuswamy, K.-H. Lee, K. Liu, D. McConachie, I. McMahon, H. Nishimura, C. Phillips-Grafflin, C. Richter, P. Shah, K. Srinivasan, B. Wulfe, C. Xu, M. Zhang, A. Alspach, M. Angeles, K. Arora, V. Campagnolo Guizilini, A. Castro, D. Chen, T.-S. Chu, S. Creasey, S. Curtis, R. Denitto, E. Dixon, E. Dusel, M. Ferreira, A. Goncalves, G. Gould, D. Guoy, S. Gupta, X. Han, K. Hatch, B. Hathaway, A. Henry, H. Hochsztein, P. Horgan, S. Iwase, D. Jackson, S. Karamcheti, S. Keh, J. Masterjohn, J. Mercat, P. Miller, P. Mitiguy, T. Nguyen, J. Nimmer, Y. Noguchi, R. Ong, A. Onol, O. Pfannenstiehl, R. Poyner, L. Priebe Mendes Rocha, G. Richardson, C. Rodriguez, D. Seale, M. Sherman, M. Smith-Jones, D. Tago, P. Tokmakov, M. Tran, B. Van Hoorick, I. Vasiljevic, S. Zakharov, M. Zolotas, R. Ambrus, K. Fetzer-Borelli, B. Burchfiel, H. Kress-Gazit, S. Feng, S. Ford, R. Tedrake, A careful examination of large behavior models for multitask dexterous manipulation. arXiv:2507.05331 [cs.RO] (2025).
5. A. Goyal, Y. Bengio, Inductive biases for deep learning of higher-level cognition. *Proc. R. Soc. A* **478**, 20210068 (2022).

Acknowledgments: This debate ("Data Will Solve Robotics and Automation: True or False?") was held at the 2025 IEEE International Conference on Robotics and Automation. A video of the debate can be found at https://youtu.be/PfvctjoMPk8?si=Zwpt-QfJ3EiqferR.

10.1126/scirobotics.aea7897

MANIPULATION

A review of learning-based dynamics models for robotic manipulation

Bo Ai[1]*, Stephen Tian[2], Haochen Shi[2], Yixuan Wang[3], Tobias Pfaff[4], Cheston Tan[5], Henrik I. Christensen[1], Hao Su[1,6], Jiajun Wu[2], Yunzhu Li[3]*

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

Dynamics models that predict the effects of physical interactions are essential for planning and control in robotic manipulation. Although models based on physical principles often generalize well, they typically require full-state information, which can be difficult or impossible to extract from perception data in complex, real-world scenarios. Learning-based dynamics models provide an alternative by deriving state transition functions purely from perceived interaction data, enabling the capture of complex, hard-to-model factors and predictive uncertainty and accelerating simulations that are often too slow for real-time control. Recent successes in this field have demonstrated notable advancements in robot capabilities, including long-horizon manipulation of deformable objects, granular materials, and complex multiobject interactions such as stowing and packing. A crucial aspect of these investigations is the choice of state representation, which determines the inductive biases in the learning system for reduced-order modeling of scene dynamics. This article provides a timely and comprehensive review of current techniques and trade-offs in designing learned dynamics models, highlighting their role in advancing robot capabilities through integration with state estimation and control and identifying critical research gaps for future exploration.

INTRODUCTION

Humans have an intuitive grasp of physics that lets us interact with the environment and predict its evolution (*1*). By processing multisensory information, we form mental models that help us anticipate how our actions affect the world (Fig. 1) (*2*). This intuitive understanding of physics does not depend on analytical methods. However, it applies across materials and objects, supporting diverse interactive skills that far exceed those of current robots.

Emulating this intuitive physics understanding in robots equates to deriving predictive models that anticipate action outcomes and support effective planning. Physics-based models (*3*, *4*) generalize well but rely on full-state information, which is often unattainable in real-world manipulation tasks. Learning-based models offer an alternative by learning predictive dynamics directly from raw sensory data, capturing hard-to-model factors (*5*, *6*), reasoning about uncertainty (*6*–*8*), and accelerating high-precision simulations too slow for real-time control (*9*, *10*). Recent advances leverage deep neural networks as function approximators (*8*, *11*).

Despite their promise, learning-based dynamics models face a fundamental challenge: designing inductive biases that ensure sample efficiency and generalization (*12*). This is particularly critical in robotics, where real-world data collection is costly and open-world environments have vast state spaces. Effective models require compact state representations and structural priors to efficiently process this information. However, this introduces trade-offs—although compact state spaces enhance generalization, they may reduce model expressiveness or complicate state estimation. Addressing these challenges requires careful consideration of task requirements, environmental complexity, and sensory modalities.

This Review provides a comprehensive analysis of learning-based dynamics models, examining trade-offs in state representations and model architectures and their implications for robotic capabilities. We discuss perception requirements for state estimation and model architectures for learning state transitions and how different representations influence sample efficiency, generalization, and task suitability. Given increasing integration of learning-based dynamics models with planning for manipulation—spanning object repositioning (*13*–*15*), deformable object handling (*16*–*19*), multimodal perception (*20*, *21*), and multiobject interaction (*14*, *22*, *23*)—a discussion on model design and implications for planning is crucial. Although prior reviews focused on related topics, such as deformable object manipulation (*24*, *25*), physics-based simulation (*26*, *27*), and intuitive physics (*28*), a dedicated review of learning-based dynamics models is lacking. This work fills that gap, providing insights for future research in robotic manipulation and beyond.

This Review focuses on the intersection of learning-based dynamics models and robotic manipulation. Thus, analytical dynamics models [such as (*29*)], differentiable (but not learned) models [such as (*27*)], and hybrid models [such as (*5*, *6*)] are beyond its scope. Similarly, learning-based dynamics models without demonstrated applications to robotic manipulation are not covered comprehensively. Within this scope, we begin by introducing learning-based dynamics models and contrasting them with analytical simulators. We then present a taxonomy of models based on state representations, discussing associated perception and dynamics learning techniques. Subsequently, we explore how planning algorithms and policy learning can integrate these learned dynamics models to enable robotic capabilities. We end with discussions on future directions and challenges in the field.

LEARNING-BASED DYNAMICS MODELS

Learning-based dynamics models predict how the world evolves in response to actions. This article focuses on models of environmental dynamics external to the robot.

[1]Department of Computer Science and Engineering, University of California San Diego, La Jolla, CA 92093, USA. [2]Department of Computer Science, Stanford University, Stanford, CA 94305, USA. [3]Department of Computer Science, Columbia University, New York, NY 10027, USA. [4]Google DeepMind, London N1C 4DN, UK. [5]Centre for Frontier AI Research, A*STAR, Singapore 138632, Singapore. [6]Hillbot, La Jolla, CA 92093, USA.
*Corresponding author. Email: bai@ucsd.edu (B.A.); yunzhu.li@columbia.edu (Y.L.)

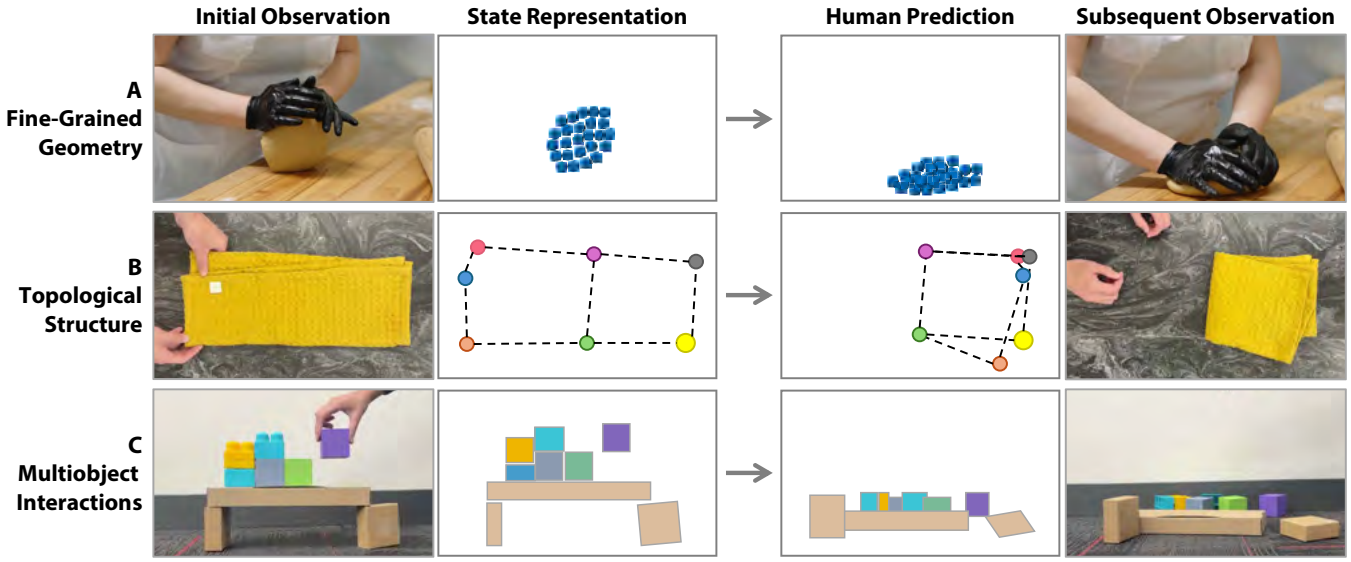


Fig. 1. Human-intuitive physics for manipulation tasks. Humans rely on intuitive physics to perform complex manipulation tasks. Depending on the task and environment, different levels of abstraction may be used in mental representations: (**A**) particles for fine-grained geometry, (**B**) keypoints for structural details, and (**C**) object-centric representations for multiobject interactions. We illustrate mental predictions of manipulation actions based on these abstractions. In robotics, learning-based dynamics models aim to equip robots with similar predictive capabilities using structured state representations.

Background

We use the framework of partially observable Markov decision processes (POMDPs) to formalize the process of perceiving and acting. At time t, the agent is in state $s_t \in \mathcal{S}$, where $\mathcal{S}$ denotes the state space. The agent receives an observation $o_t \in \mathcal{O}$ from the environment and then takes an action a_t based on a policy π, where $a_t = \pi(o_t)$ and $a \in \mathcal{A}$. Conditioned on this action, the environment transitions to the next state $s_{t+1} = \mathcal{T}(s_t, a_t)$, where $\mathcal{T}$ is the environment transition function. The process repeats until the task objective is achieved or the number of time steps reaches the task horizon H. The agent's goal is to find a policy that minimizes the cost function c defined on the state s_t and action a_t over the time horizon, defined as

$$\min_{\pi} \mathbb{E}_{\tau \sim \pi}\left[\sum_{t=0}^{H} c(s_t, a_t)\right] \quad (1)$$

This formulation underlies both classical robotic control and model-based reinforcement learning (RL): In both paradigms, a transition model simulates future trajectories for planning or policy learning. The key question is how to represent and construct the policy function π. Learned model–based approaches approximate the transition model using a learned function $\widehat{\mathcal{T}}$ before using it for control. We examine the core components of this framework below.

Perception module

The perception module g estimates the environment state s_t from past observations $o_{0:t}$ and actions $a_{0:t-1}$, that is, $s_t = g(o_{0:t}, a_{0:t-1})$, which simplifies to $s_t = g(o_t, a_{t-1})$ in fully observable settings. We view s_t as a unified representation of all task-relevant information inferred from raw sensory data, serving as input to downstream processes. A central challenge lies in defining s_t to capture minimal yet sufficient information for manipulation. This Review surveys different choices of s_t and their trade-offs.

Dynamics module

The dynamics model $\widehat{\mathcal{T}}$ predicts the state transition from s_t to s_{t+1} given action a_t. Its design is closely coupled with the structure of s_t, often leveraging inductive biases to improve generalization and data efficiency. For instance, graph neural networks (GNNs) naturally suit particle-based states because of their spatial equivariance. This Review examines model architectures for $\widehat{\mathcal{T}}$ across different state representations.

Control module

The policy π generates control signals to minimize the cost (Eq. 1). It can be implemented via planning or policy learning and can output position-based (such as end-effector poses) or force-based control signals (such as joint torques). The policy's design directly affects computational efficiency and control quality. This Review examines how control algorithms integrate with dynamics models toward solving concrete manipulation tasks. Figure 2 illustrates how dynamics models are learned from physical interaction data and integrated with control for downstream tasks.

Comparison with physics-based dynamics models

Traditionally, dynamics models are defined by analytical solvers that integrate governing equations of motion. In robotics, this includes rigid-body simulators (*30*, *31*) and deformable object solvers such as the material point method (MPM) (*4*). Despite decades of progress, these models often diverge from real-world behavior—a challenge known as the simulation-to-reality (sim-to-real) gap (*32*). This gap arises because many physical effects, such as frictional contact or actuator drift, are difficult to model precisely or require parameters that are hard to measure (*33*). Even with accurate parameters, missing latent factors such as temperature can undermine accuracy. Moreover, real-world deployment depends on accurate state estimation and system identification, where small errors can accumulate over time (*34*).

Learned dynamics models offer an alternative by directly capturing physical processes from interaction data, mitigating the sim-to-real gap. They can compensate for state estimation errors (*35*) or bypass

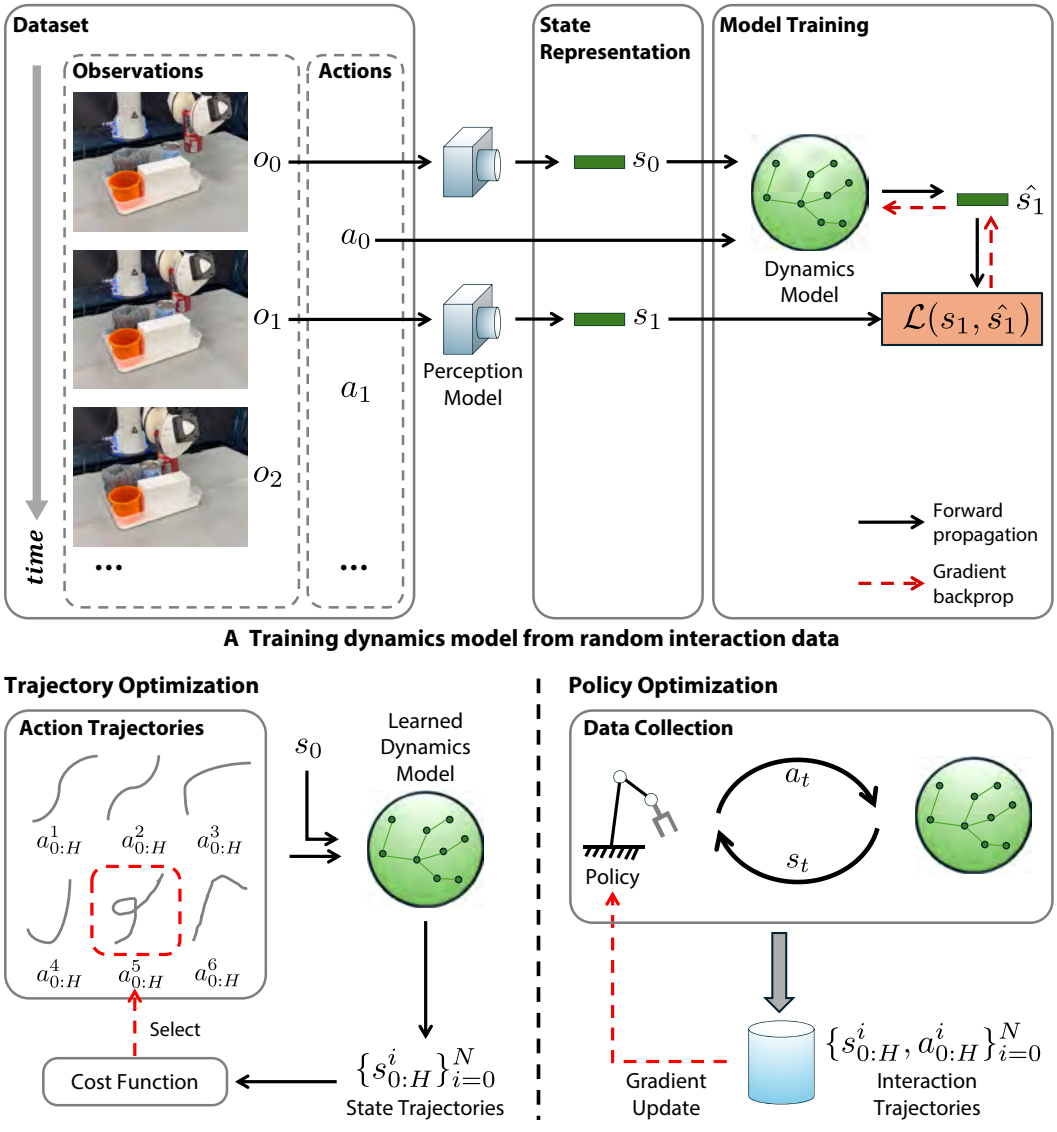


Fig. 2. Robotic manipulation using learning-based dynamics models. (**A**) A dynamics model is trained on interaction data. The perception model extracts state representations s_t from observations o_t. Dynamics are learned in a self-supervised fashion. (**B**) The learned dynamics model is applied for downstream control, either by evaluating action trajectories $\{a^i_{0:H}\}^N_{i=0}$ for planning or by generating interaction data $\{s^i_{0:H}, a^i_{0:H}\}^N_{i=0}$ for policy learning.

state estimation entirely when trained on raw sensory inputs. Beyond bridging this gap, learned models are also end-to-end differentiable, enabling gradient-based planning, control, and online adaptation. Some studies found that learned models offer smoother gradients than analytical solvers (*36*) and can be more computationally efficient, especially for nonrigid systems (*37*).

STATE REPRESENTATIONS

How should we represent the state of the world for learning-based dynamics models? A natural starting point is raw observations, such as pixels encoding RGB (red, green, blue), depth, or force fields. However, the state needs to capture only aspects of the environment relevant for accurate prediction and decision-making. This need motivates compact and structured representations.

One approach is to use latent representations, which compress raw observations into lower-dimensional encodings but often lack explicit three-dimensional (3D) structure. To incorporate geometry, the world can be discretized into particles, representing surfaces and volumes in three dimensions. For many tasks, particles may be overly detailed, and keypoints offer a more abstract alternative by capturing salient task-relevant features. However, these representations often treat scenes as unstructured collections of elements, whereas humans perceive and interact with discrete entities. Object-centric representations explicitly model objects and their interactions, adding structure beyond lower-level elements.

These representations reflect varying levels of abstraction and modeling assumptions. More abstract representations enable reduced-order modeling, improved sample efficiency, and generalization by focusing on task-relevant dynamics but often require stronger perception

PHOTO CREDIT: (20)

priors, such as object segmentation or keypoint detection. This section surveys perception methods and dynamics models for each representation and discusses trade-offs and practical considerations.

2D pixel representations

Perhaps the most straightforward state representation is observed raw pixels, such as two-dimensional feature maps of RGB(D) (where D indicates depth) data. They are typically obtained directly from cameras and usually require only simple postprocessing such as downsampling or cropping.

Dynamics learning

Learning dynamics models in pixel space can be framed as action-conditioned video prediction. Given one or more context frames, $o_{0:t}$ and a sequence of actions $a_{0:H}$, the model predicts future observations $o_{t+1:H}$ conditioned on the agent taking those actions. Although video prediction and generation are widely studied in computer vision and use similar techniques as pixel dynamics models, we focus on models that focus on physical prediction and action-conditioned models for planning.

An early line of work applying pixel-space models to planning with physical robots is visual foresight (*38*). This method trained a flow-based action-conditioned video dynamics model on robotic physical interaction data and used it to plan object-pushing tasks. Extensions of this work demonstrated application to robotic tool use (*39*) and enabled quick adaptation to new objects via metalearning (*40*).

Suh and Tedrake (*41*) showed that a switched-linear pixel-based model can also yield strong performance. Transformer-based models have also been applied to robotic manipulation (*42*) and large-scale autonomous driving data (*43*). Last, recent video diffusion models learned visual dynamics with improved scalability (*44*–*46*). These nonautoregressive models can not only predict future frames but also inpaint intermediate ones. Du *et al.* (*46*) used this property combined with an inverse dynamics model to perform control.

Pixel-prediction methods can be readily applied to other data modalities as long as they can be represented as 3D arrays. These modalities include depth (*47*), percepts from an optical tactile sensor (*21*), and density fields of granular materials (*48*). Pixel-space models are typically trained with maximum likelihood objectives in a self-supervised manner on video sequences, with prediction targets sampled from future frames. Agent action labels for action-conditioned models may be represented as end-effector poses or joint positions. These can be obtained through proprioception. When action labels are absent, some models infer latent action representations (*49*–*51*).

Pixel dynamics models are usually evaluated with metrics from the video generation literature. However, these metrics focus on visual appearance and often do not correlate with planning performance (*52*). Physical prediction–based benchmarks (*53*) partially bridge this gap, but developing additional metrics is an open challenge.

The wide availability of pixels may allow pixel-space models to achieve broad generalization capabilities. Pixel-prediction models have been trained on increasingly diverse datasets, for instance, on robotic interactions across several robots and scenes (*54*). GAIA-1 (*43*) was trained using in-the-wild driving data, and UniSim (*45*) is a single model trained with robotic data, human videos, internet media, and navigation data.

Overall, pixel-based representations do not require explicit state estimation and, in principle, can model arbitrary physical phenomena. They bypass explicit perception pipelines but require large datasets to learn effectively in high-dimensional observation spaces. Convolutional neural networks (CNNs) are commonly used, with recent approaches using transformers and diffusion models. Despite these advances, such models often struggle with object permanence and temporal consistency, even when trained with substantial computational resources (*45*, *46*). For control, pixel-based models are sensitive to partial observability, which can lead to hallucinations, and their high computational cost poses challenges for high-frequency control.

Latent representations

Predicting dynamics from raw observations o_t is challenging because of their high dimensionality and redundancy. A common alternative is to project o_t into a lower-dimensional latent vector z_t first. Although both pixel-based models and latent-state models can be trained on pixel data, they differ in their prediction domain: Pixel-based models autoregressively generate future observations, whereas latent-state models predict in the abstract latent space.

This projection introduces inductive biases by assuming that the state space admits a compact and smooth parameterization. This can substantially enhance learning efficiency and generalization by filtering out irrelevant variations.

Perception and representation learning

A key challenge in learning a latent representation is ensuring that the latent vector z_t encodes task-relevant features rather than collapsing to trivial solutions, such as mapping the set of all inputs to a constant vector. Existing approaches address this challenge by imposing structure via supervision and can be categorized into reconstruction-based and reconstruction-free methods.

Reconstruction-based training is a common approach for learning latent-state representations that ensures that encoded states retain sufficient information to reconstruct raw observations. Early work, such as Embed to Control (*55*), enforced alignment between decoded and ground-truth observations using Kullback-Leibler divergence but with the limiting assumption of linear state dependencies. More expressive models instead learn nonlinear mappings with deep networks, such as variational autoencoders (*56*) and GNNs (*57*). Latent states trained to reconstruct volumetric scenes further impose strong geometric and 3D priors (*58*, *59*). However, these models are computationally expensive and impractical for real-time control.

In partially observable environments, reconstruction-based training extends to inferring occluded states. Action-conditional implicit dynamics (ACID) (*60*) encodes partial RGB-D inputs into a 3D feature field, predicting occupancy probabilities to handle occlusions for deformable object manipulation. When single-step observations are insufficient, recurrent models can aggregate history information (*7*, *61*).

Reconstruction-based training may lead to latents encoding task-irrelevant details. To avoid this, reconstruction-free approaches use alternative learning signals. One alternative is to predict task-relevant features, such as object motion represented by optical flow (*14*). Inverse dynamics learning trains models to predict the action responsible for a state transition, ensuring only action-relevant latent features (*15*). Contrastive learning avoids trivial solutions, pulling predicted next states closer to ground truth states and pushing them away from incorrect encodings (*19*). When rewards are available, predicting rewards from latent representations provides compact, efficient, and task-relevant encodings (*62*, *63*), at the cost of increased task dependence of the learned models.

Dynamics learning

Latent dynamics models can be categorized as probabilistic or deterministic. Probabilistic models predict distributions over future

states, whereas deterministic models estimate a single most likely outcome. In both cases, robot actions are typically incorporated by concatenating them with the estimated latent states as input to the dynamics predictor.

Probabilistic models leverage classical statistical methods or deep neural networks to predict distributions over future states. Linear probabilistic models, such as Gaussian state-space models, are typically trained by optimizing distributional divergence metrics (*55*, *64*). More expressive approaches are based on deep neural networks and often incorporate history information. DayDreamer (*7*) and DeformNet (*59*) apply recurrent state-space models (RSSMs) (*65*), which use recurrence to maintain temporal memory and capture uncertainty, to real-world manipulation. Probabilistic models can output multimodal future predictions by combining mixture density networks with recurrent models (*8*).

When environmental dynamics are relatively predictable, deterministic models provide a simpler alternative. They may use multilayer perceptrons (MLPs) for low-dimensional latent spaces (*15*, *57*, *58*, *62*, *63*) and CNNs for high-resolution feature maps (*14*, *60*).

In summary, latent-state models have been applied to diverse manipulation settings, including rigid bodies, articulated objects, and fluids. Training objectives range from task-agnostic formulations, such as reconstruction, to highly task-specific losses incorporating reward signals. RSSMs and MLPs are commonly used for modeling dynamics in low-dimensional latent spaces, where well-structured representations often lead to sample-efficient learning. Task-specific objectives may produce representations that struggle to generalize, whereas task-agnostic approaches can support cross-task transfer, although generalization to varying object counts or scene configurations remains limited. Compact latent representations make these models computationally efficient, enabling fast closed-loop control.

3D particle representations

Unlike latent and pixel-based representations, particle-based models explicitly encode 3D structure by representing objects as discrete points, capturing both surfaces and volumes. This structure enables precise interaction modeling, and incorporating strong physical priors improves sample efficiency.

Particles have long been used in physics-based simulation methods, such as MPM (*4*). These techniques underpin modern physics-based simulators (*31*, *66*) but rely on approximate modeling, leading to a sim-to-real gap that often requires system identification (*32*). Learned particle dynamics models can predict particle behavior directly from real-world data.

Perception and representation learning

In real-world applications, particles are commonly sampled from observed point clouds (*16*, *17*, *22*, *67*). Single-camera methods can reconstruct point clouds through gradient-based optimization (*68*), although these reconstructions are often noisy. Data are usually downsampled before training (*16*, *17*). However, occlusion remains a challenge in cluttered environments. Some methods incorporate geometric priors to handle occlusion, for example, assuming dough conforms to the shape of a tool interacting with it (*16*). Integrating tactile sensing may also improve particle state estimation in combination with historical observations and recurrent structures (*20*).

Alternatively, volumetric representations can be constructed from multiview images via neural radiance fields (NeRFs) (*69*). Then, particles can be sampled from voxel grids using trilinear interpolation (*4*).

Dynamics learning

Particle dynamics arise from local particle interactions, which models typically capture using inductive biases such as spatial equivariance and locality. To this end, existing approaches primarily use graph-based architectures or convolutional models.

GNNs are widely used for modeling particle interactions. Particles are represented as graph nodes, and node features may include physical parameters or motion and displacement information. Hierarchical relation networks (HRNs) (*70*) introduce a hierarchical graph structure where leaf particles encode local interactions, whereas root nodes provide object-level abstractions to handle rigid and nonrigid transformations. To enhance adaptability, dynamic particle interaction networks (DPI-Nets) (*57*, *71*) update dynamic interaction graphs during simulation, effectively capturing object deformations. This flexibility enabled DPI-Nets to lay the foundation for modeling elastoplastic objects (*17*, *72*), granular material manipulation (*22*), food preparation (*16*), and object packing (*20*, *73*). Graph network–based simulators (GNS) (*74*, *75*) generalize this framework by providing a simpler yet more accurate model for fluids, rigid bodies, and deformable materials.

Alternatively, convolutional architectures model local interactions without explicit graphs. Smooth particle networks (SPNets) (*76*) use specialized convolutions: ConvSP for particle-particle interactions and ConvSDF for differentiable collisions with static geometry. Actions are supplied by updating the poses of controllable objects, which then interact with other objects. Compared with graph-based models, convolutional architectures are often more efficient and parallelizable but less flexible for long-range or irregular interactions.

To summarize, particle-based representations explicitly encode geometric structure, with physical properties preserved through particle interactions. They are particularly well suited to deformable objects but have also been applied to rigid bodies and fluids. Estimating particle states from depth observations is sensitive to occlusions, and point tracking is often used to establish correspondences across frames. Graph-based networks and convolutional architectures are common modeling choices, offering strong inductive biases and sample efficiency. These inductive biases support generalization to unseen object geometries, reflecting a trade-off: More demanding perception enables more accurate and efficient dynamics modeling through structured representations. For control, GNNs may face scalability challenges with dense graphs, whereas convolutional networks are generally lightweight. Particle representations also integrate multimodal inputs, such as vision and touch, to enable fine-grained control.

Keypoint representations

Keypoint representations consist of sparse points that may encode implicit or explicit semantic information. Unlike particle representations, which use dense 3D points to capture object geometries, keypoints offer a more compact state representation that retains only task-relevant points. Typically, keypoints are defined by a set of 2D or 3D coordinates; for instance, a rigid box can be represented by its eight corner points. Unlike unordered particle sets, keypoints are often structured as ordered lists with semantic information, such as visual features, implicitly assigned to specific indices.

Perception and representation learning

The literature presents three common approaches for keypoint extraction: supervised learning with manual labels, unsupervised learning using reconstruction losses, and zero-shot prediction using pretrained vision models. Supervised learning methods train networks

to predict keypoints from labeled datasets, but efficient keypoint annotation remains a challenge. Keypoint affordances for category-level robotic manipulation (kPAM) (*77*) addresses this by labeling keypoints in three dimensions and projecting them into image space. Dense object nets (*78*) introduce dense visual descriptors, tracking keypoints over time via feature similarity. This approach has been extended for keypoint-based object tracking (*13*) and deformable object tracking (*79*).

Unsupervised learning methods extract keypoints using encoder-decoder frameworks where the decoder reconstructs observations from keypoints. Transporter (*80*) is a representative 2D method, extracting keypoints from RGB images through feature inpainting and reconstruction. In 3D, KeypointDeformer (*81*) predicts shape-representative keypoints from object meshes, training on source-target mesh pairs to learn deformation consistency.

Recent work explores zero-shot keypoint detection using visual foundation models. RoboABC (*82*) aligns robot observations with human-object interaction data using contrastive language-image pretraining (CLIP) and diffusion features (*83*, *84*) to identify contact points. Back to 3D (B2-3D) (*85*) projects 2D DINO features into 3D space to detect category-specific keypoints with minimal annotations. The "keypoints + instructions to execute" (KITE) (*86*) method extends this by training a grounding module to localize semantic keypoints on the basis of text inputs.

Dynamics learning

Keypoints can be processed similarly to particles using GNNs because they also represent points in space (*87*–*90*). However, when organized as ordered lists, keypoints can encode additional semantic information. Thus, although GNNs are permutation invariant and well suited for unordered data, ordered keypoints are typically processed with MLPs, which can leverage ordering information (*13*, *91*). In both cases, actions are represented as fixed-dimensional vectors and concatenated with graph node features or keypoint feature vectors for dynamics prediction (*13*, *88*, *89*).

Keypoint-based representations focus on task-relevant features rather than full scene geometry, making them suitable for tasks where specific object regions are salient for control. They have been applied to both rigid-body and deformable object manipulation. Keypoints are typically extracted using learned detectors. Although more compact than particle sets, they are sensitive to occlusion and require consistent detection over time. Lightweight architectures, such as MLPs or graph-based models, are commonly used to capture keypoint dynamics and interactions. Because keypoints correspond to consistent abstract task-relevant structures, models can often generalize across object instances. Their compactness also enables fast inference, real-time planning, and feedback control.

Object-centric representations

A core challenge of scaling dynamics models to diverse scenes is the combinatorial complexity of possible object configurations in the world, which is challenging to handle without compositional generalization abilities. Humans address this by perceiving scenes in an object-centric way: containing discrete entities with boundaries and predictable interactions (*92*).

Motivated by this, some approaches adopt object-centric representations that model dynamics at the level of interacting objects rather than low-level particles or features. These structured representations support generalization to novel object arrangements and are the highest abstraction level that we consider for modeling dynamics.

Perception and representation learning

Techniques for obtaining object-level latent representations from raw observations include segmenting objects from visual inputs and encoding their features, directly mapping multiobject scenes to structured object-centric encodings, or leveraging inverse rendering techniques to infer physical object states. The first approach explicitly segments objects before extracting features. Object-oriented prediction and planning (O2P2) (*93*) assumes access to instance segmentation and encodes each object separately, enforcing meaningful representations through a neural rendering engine. Neuro-symbolic dynamic reasoning (NS-DR) (*94*) extends this technique to video, whereas region proposal interaction networks (RPINs) (*95*) jointly detect and encode objects for dynamics learning. Compositional NeRF (*23*) integrates segmentations across camera viewpoints for 3D consistency.

Alternatively, object-centric representations can be learned from multiobject scenes without explicit segmentation. Visual interaction networks (VINs) (*96*) extract object-wise latent representations from image sequences and decode them into object states, including position and velocity, but require ground-truth supervision. To alleviate this, the object-centric perception, prediction, and planning (OP3) framework (*97*) performs unsupervised object discovery, iteratively refining posterior estimates of object assignments on the basis of interaction data.

A third approach leverages inverse rendering to infer object states from raw observations. Tian *et al.* (*98*) used neural implicit object representations and optimization-based inference to estimate 6D object poses, achieving robust performance under varying lighting conditions.

Dynamics learning

Object-centric dynamics models treat objects as discrete entities and model their interactions. They can be implemented using generic neural networks or graph-based architectures that explicitly leverage relational structures.

To explicitly leverage relational structure, the neural physics engine (NPE) (*99*) introduces a mechanism akin to message passing in GNNs. It iterates over object pairs, predicting their relative motion and aggregating predictions. O2P2 (*93*) and OP3 (*97*) adopt similar object-centric architectures. Although O2P2 incorporates environment actions such as object placement or motion, OP3 further embeds action information as a latent vector to modulate both per-object dynamics and interactions.

GNNs provide a more structured approach to modeling object interactions through iterative message passing. The foundational interaction networks (INs) (*100*) represent objects as graph nodes and encode relational attributes, such as restitution coefficients and spring constants, in the edges. However, the model assumes access to ground-truth physical properties, such as shape and mass. When object states are unavailable, latent representations or robot actions can act as node features (*23*, *94*).

To sum, object-centric representations are well suited for tasks involving multiobject interactions but less effective for modeling continuous materials, such as fluids or highly deformable objects. Perception is typically achieved through instance segmentation, inverse rendering, or object proposal techniques, although it remains challenging in general. Dynamics are often modeled using graph-based architectures, enabling relational reasoning and modularity. With object-level priors, these models support combinatorial generalization across varying numbers and configurations of objects.

They are generally computationally efficient for control, but perception can become a bottleneck, potentially introducing latency.

Comparing and selecting state representations

Each state representation provides distinct trade-offs in modeling capacity, sample efficiency, generalization, task alignment, interpretability, and computational cost (Fig. 3). Less-structured representations, such as latent states and pixels, simplify state estimation but introduce challenges in model bias and generalization. They often require larger datasets and can hallucinate predictions in out-of-distribution scenarios. These inaccuracies can be problematic during downstream control because optimization over unreliable model outputs may degrade control performance.

Conversely, structured representations, such as particles and object-centric models, provide strong inductive biases, such as spatial equivariance, enhancing generalization and prediction accuracy. These models are often trainable within a few graphics processing unit (GPU) hours (*17*, *71*) and more robust for policy and trajectory optimization. However, they pose challenges for state estimation. Particle-based methods require temporally consistent 3D point tracks, which remain difficult to obtain (*90*). Similarly, object-centric representations depend on accurate perception modules, often making state estimation a bottleneck for scaling structured models.

The optimal choice of state representation depends on both the downstream task and hardware constraints. For instance, object-centric representations excel in manipulation tasks involving multiple rigid bodies (*94*, *96*, *101*), such as object rearrangement (*23*), stacking (*93*), and sliding (*14*), because of their high level of abstraction but are unsuited for fluids or granular materials, for which it is unclear how to define the notion of an object. Particle-based representations flexibly capture the dynamics of deformable and nonrigid objects (*71*, *74*), including dough (*17*), clothes (*102*), and soft toys (*20*), although they often require multiview RGB-D sensing for point cloud perception. Latent and pixel-based models (*44*–*46*) can, in principle, handle arbitrary entities. However, they may produce physically inconsistent predictions without physics priors and explicit 3D representations, particularly in contact-rich scenarios such as cutting or splitting objects. In addition, highly specular, transparent, or otherwise visually complex materials can be challenging for RGB image reconstruction.

Interpretability is another key consideration. For some state representations, such as pixels, particles, and keypoints, visualizing predicted trajectories is natural, making it simple to diagnose failure cases and refine models. In contrast, reconstruction-free latent-state models often lack this transparency.

Last, computational cost is crucial for real-world robotic deployment. Pixel-based models require high-capacity architectures, whereas keypoint-based models often operate with smaller networks. Graph-based methods, often used with particle representations, can scale linearly with the number of graph edges or quadratically with the particle count, making inference costly. In addition, backpropagation through models during gradient-based action optimization can add substantial computational overhead.

Table 1 summarizes the key trade-offs across state representations, including sensing modalities, computational requirements, and target applications. Although existing methods excel at handling specific object types and sensor inputs, a unifying representation that generalizes across diverse robotic tasks remains a challenging but crucial direction for future work.

CONNECTION TO ROBOTIC CONTROL

Learning-based dynamics models can be integrated with control modules to generate robot motions for predefined task objectives. We first detail two ways to leverage learned dynamics models and then discuss representative tasks that benefit from this integration.

Control methods

Techniques for performing control using learning-based dynamics models fall into two main paradigms: motion planning and policy learning.

Motion planning

Motion planning searches for a feasible path from an initial state to a goal state while satisfying task constraints. Learned dynamics models enable planning in complex or unknown environments, where analytical models are unavailable, inaccurate, or hard to obtain. Motion planning methods can be broadly categorized into path planning and trajectory optimization.

Path planning focuses on finding a sequence of collision-free states without modeling system dynamics. Sampling-based methods such as rapidly exploring random trees (RRTs) (*103*) and probabilistic roadmaps (PRMs) (*104*) are widely used to search high-dimensional spaces with complex constraints. An RRT incrementally expands a

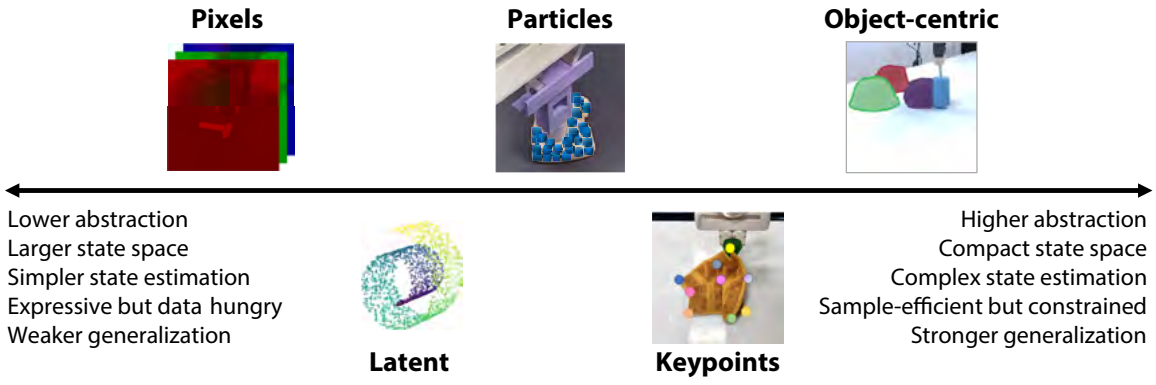


PHOTO CREDITS: PIXELS, (*15*); PARTICLES, (*16*); OBJECT-CENTRIC, (*23*)

Fig. 3. A spectrum of state representations with varying structural priors. State representations in dynamics models range from unstructured (pixels and latent) to structured (particles, keypoints, and object-centric). Increasing structure introduces stronger priors and abstraction, enabling better generalization but requiring more complex state estimation. The "Swiss roll" illustration for latent states was inspired by Tenenbaum *et al.* (*132*) and created using Python.

Table 1. Summary of key studies on dynamics learning. This table categorizes the literature on the basis of the type of state representation, sensors used, dynamics modeled, and object types considered. The dimensions of the dynamics (2D and 3D) refer to the space in which object rotations, translations, and deformations are modeled.

	Sensing				Dynamics		Object type			
References	**RGB**	**Depth**	**Multiview**	**Tactile**	**2D**	**3D**	**Rigid**	**Deformable**	**Fluids**	**Multiobject**
2D pixel representations										
Finn and Levine (*11*)	✓				✓		✓	✓		✓
Suh and Tedrake (*41*)	✓				✓		✓			
Hoque *et al.* (*47*)	✓	✓				✓		✓		
Du *et al.* (*46*)	✓				✓		✓			✓
Yang *et al.* (*45*)	✓				✓		✓			✓
Latent representations										
Agrawal *et al.* (*15*)	✓					✓	✓			
Yan *et al.* (*19*)	✓				✓			✓		
Wu *et al.* (*7*)	✓					✓	✓			
Li *et al.* (*58*)	✓		✓			✓	✓	✓	✓	✓
Shen *et al.* (*60*)	✓	✓				✓	✓	✓		
3D particle representations										
Gonzalez *et al.* (*74*)						✓		✓		✓
Li *et al.* (*71*)		✓				✓		✓	✓	✓
Ai *et al.* (*20*)	✓	✓	✓	✓		✓	✓	✓		✓
Shi *et al.* (*16*)	✓	✓	✓			✓		✓		
Wang *et al.* (*22*)	✓	✓			✓		✓			✓
Keypoint representations										
Manuelli *et al.* (*13*)	✓	✓				✓	✓			
Wang *et al.* (*91*)	✓					✓	✓			✓
Li *et al.* (*87*)	✓				✓			✓		✓
Ma *et al.* (*88*)		✓			✓			✓		
Rezazadeh and Choi (*89*)	✓				✓		✓			✓
Object-centric representations										
Watters *et al.* (*96*)	✓				✓		✓			✓
Janner *et al.* (*93*)	✓					✓	✓			✓
Xu *et al.* (*14*)	✓	✓				✓	✓			✓
Yi *et al.* (*94*)	✓					✓	✓			✓
Driess *et al.* (*23*)	✓	✓				✓	✓	✓		✓

search tree through random sampling. The resulting paths can then be refined into dynamically feasible trajectories through trajectory optimization with learned dynamics models.

Trajectory optimization refines action sequences locally to improve task performance, directly leveraging learned dynamics models to simulate and evaluate outcomes. Sampling-based methods such as the cross-entropy method (CEM) (*105*) and model-predictive path integral (MPPI) (*106*) explore multiple action candidates, whereas gradient-based methods adjust actions using cost gradients enabled by model differentiability. Moreover, learned dynamics models can

be integrated with online system identification to adapt to uncertain dynamics (*14*, *20*, *107*).

Policy learning

In contrast with motion planning, policy learning seeks to directly obtain a map from observations to actions. Learned dynamics models can provide simulated transitions as training data.

One approach is to generate training data in the form $\langle s_t, s_g, a_t \rangle$, where the state s_t and action a_t yield the next state s_g. An inverse dynamics model can then learn to predict the action needed to transition from s_t to s_g, acting as a goal-conditioned policy. However, errors may accumulate over extended rollouts, and multimodal action distributions can be hard to fit, given that multiple actions could achieve the same transition.

Reinforcement learning (RL) optimizes policies through trial-and-error interactions to maximize cumulative rewards. Learned dynamics models facilitate this process by simulating transitions (*108*), allowing policies to be trained with a reduced or negligible number of real-environment interactions. However, inaccuracies in the learned dynamics model can lead to policy exploitation, particularly in state distributions not well supported by training data. This can be mitigated by fine-tuning the policy on real-world data in addition to simulated rollouts (*109*).

Representative robotic tasks

Learning-based dynamics models have been applied across tasks from object pushing to deformable and multiobject manipulation. This section highlights key applications and integrations with motion planning and policy learning techniques.

Object repositioning

Object repositioning is widely used to evaluate learned dynamics models in robotic control. Latent representations (*15*, *60*) and pixels (*21*) have been used to represent single-object scenes, whereas keypoints can serve as a lower-dimensional representation for efficient dynamics learning given the limited degrees of freedom of rigid objects (*13*). On the other hand, multiobject scenarios can be better modeled with object-centric representations (*23*). For control, motion planning methods such as random search (*14*, *38*, *60*), MPPI (*13*, *106*), and CEM (*11*, *105*) can optimize action sequences. Online system identification can help handle objects with unknown physical properties (*20*). Alternatively, inverse dynamics models trained alongside forward models can directly infer actions from current and target states (*15*).

Deformable object manipulation

Deformable object manipulation presents challenges because of high-dimensional shape variations and complex contact dynamics (*24*, *25*). Particle-based representations can capture the arbitrary geometries of deformable objects (*16*, *17*, *20*, *71*, *74*, *107*, *110*) and can be abstracted into keypoints for objects with salient features, such as clothes (*88*). Learned dynamics models have been integrated with trajectory optimization for manipulating rope (*19*, *23*), cloth (*59*), dough (*17*, *111*), and soft toys (*20*). These models also enable training goal-conditioned policies, as demonstrated in long-horizon tasks such as making dumplings (*16*).

Multiobject manipulation

Manipulation involving multiple objects requires efficient planning to manage large state spaces. Particle-based (*20*) and object-centric representations (*23*) perform well in multiobject modeling, whereas pixel-based methods struggle with modeling contact-rich interactions. To perform control using learned dynamics models, RoboPack (*20*) applies MPPI with action priors for object insertion. Latent-space RRT has been combined with model predictive control for long-term planning and real-time corrections (*23*).

Tool-use manipulation

Modeling tool-use dynamics may extend robotic capabilities beyond manipulating objects directly with an end effector. Particles can provide a unified representation for objects, tools, and robot end effectors (*16*, *20*, *22*) but require detailed 3D sensing; pixel-based methods offer a lightweight perception alternative (*39*). Learned models have been used for shaping dough with rollers and punches (*16*), nonprehensile box manipulation with compliant tools (*20*), and granular material manipulation (*22*). For extended tasks requiring tool selection and task execution, action proposal models improve planning efficiency for sampling-based planning (*39*). Figure 4 illustrates these tasks, and a summary of the discussed work is provided in Table 2.

FUTURE DIRECTIONS

Learning-based dynamics models have advanced adaptive control in robotics model-based planning and policy learning. However, current systems remain far from human-level generalization, adaptability, and robustness in unstructured environments. This section discusses key limitations and outlines promising directions for future research.

Opportunities in perception

Partially observable domains

Real-world environments are inherently partially observable because of visual occlusions and unknown physical properties, such as material rigidity and friction. Although passive history (*7*, *57*), active perception (*14*), and multimodal sensing (*20*) improve state estimation, challenges remain in cluttered and unstructured scenes. Structured representations such as particles require precise perception capabilities, whereas less-structured models such as pixels avoid this requirement but often struggle with accuracy and generalization, especially for contact dynamics (*45*). Future work should explore new representations and robust state estimation methods to better handle partial observability.

Multimodal perception

Although most prior work relied on visual sensing, other modalities, including tactile (*20*) and audio sensing (*112*), provide complementary information for perception and control. However, integrating multimodal signals introduces several challenges. Differences in statistical distributions across modalities complicate model training, and mismatched sensing frequencies create deployment-time difficulties. In addition, effectively fusing heterogeneous signals into a unified representation remains an open problem. Addressing these challenges will enable more robust dynamics reasoning and control performance across a wider range of tasks.

Opportunities in dynamics learning

Robust dynamics models

Inaccuracies in learned dynamics models can be exploited by planning and reinforcement learning agents, leading to failures in long-horizon tasks. Ensuring robust predictions across the entire state-action space is challenging because of its combinatorial size. In addition, certain state-action subspaces can be difficult or unsafe to explore, limiting counterfactual reasoning capabilities. Strategies to address this include using simulation data to cover challenging regions of the state-action

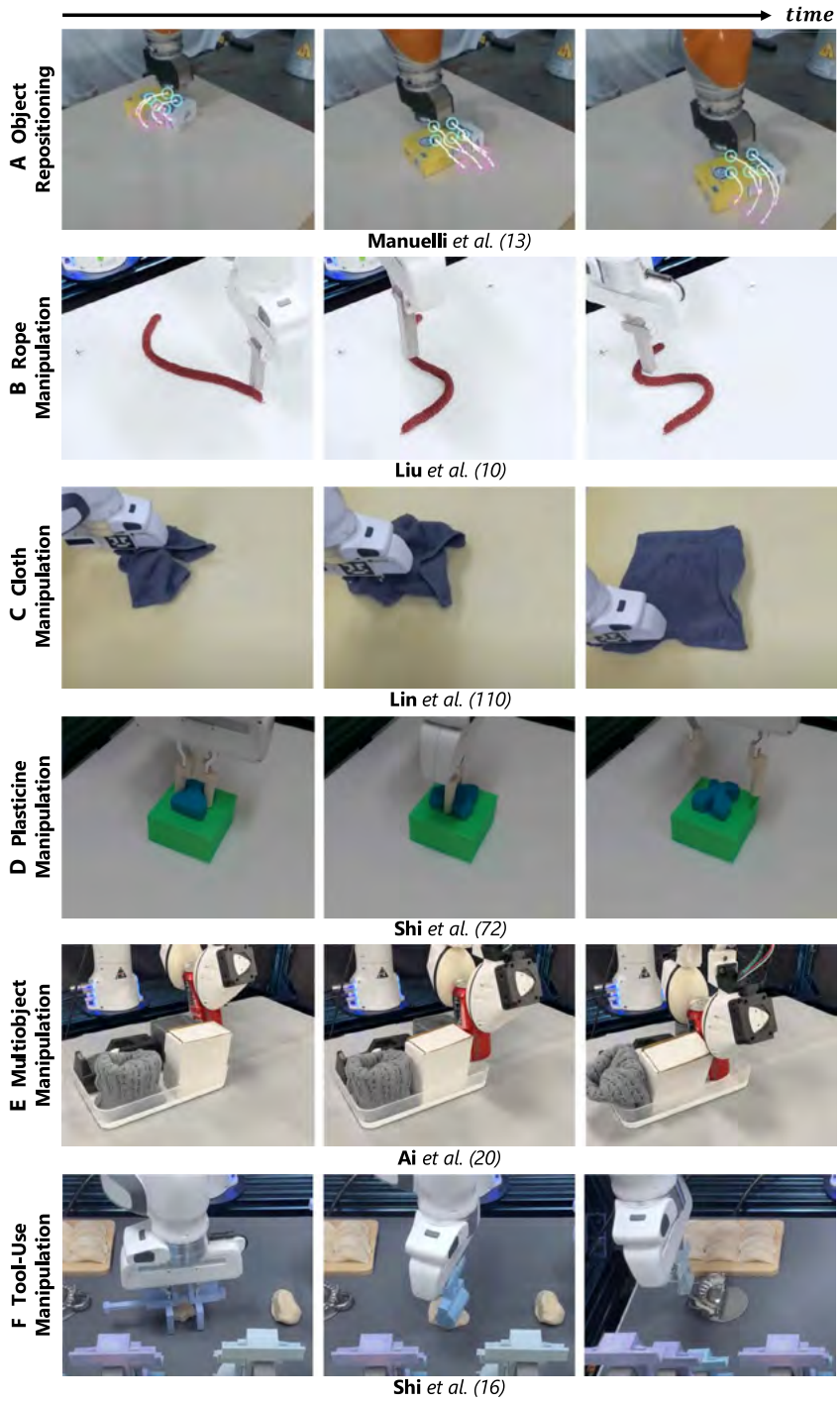


Fig. 4. Robotic manipulation tasks enabled by learning-based dynamics models. (**A**) Object repositioning, (**B**) rope manipulation, (**C**) cloth manipulation, (**D**) plasticine manipulation, (**E**) multiobject manipulation, and (**F**) tool-use manipulation. Examples span rigid and deformable objects, multiobject settings, and tool-assisted operations.

PHOTO CREDITS: (A), (13); (B), (10); (C), (110); (D), (72); (E), (20); (F), (16)

space, introducing physics priors to reduce data requirements (*113*), and using probabilistic models to account for aleatoric uncertainty (*7*, *59*).

Foundation dynamics models

Recent advances in foundation models (*114*) highlight the potential of large-scale training for broadly capable vision and language models. In contrast, most learned dynamics models remain narrow in domain because of the lack of large-scale real-world datasets with action labels (*19*, *102*). Scaling up dynamics models may require inferring actions from unlabeled data such as internet videos. Early efforts, such as learning latent actions (*51*, *115*), suggest promising directions for this goal.

Dynamics priors from foundation models

Estimating physical properties, such as mass, friction, and deformability, is crucial for accurate dynamics modeling yet remains challenging. Prior work has attempted to infer these properties from observations, using visual cues (*98*), tactile sensing (*20*), or multiview depth images (*14*). Recent foundation models demonstrate common-sense reasoning about material properties (*116*, *117*), offering a potential source of priors for estimating system parameters (*107*). By integrating these priors with learned dynamics models, future work could reduce reliance on real-world data and online identification.

Emerging representations from graphics research

Scene representations from computer graphics offer new possibilities for dynamics learning. NeRF-based representations address limitations of direct pixel representations by capturing multiview consistency and 3D structure (*23*, *58*). Particle-based models tend to struggle with smooth continuous deformations, but recent advances in 3D Gaussian splatting (3DGS) (*118*) may help address this by modeling particles as Gaussian functions, producing smoother and more flexible surfaces. Although 3DGS has been applied to dynamic scene reconstruction (*119*, *120*), its integration with action-conditioned dynamics models remains underexplored. Early efforts include tracking objects with 3DGS in particle-based models (*121*), but deeper integration is a promising future direction.

Large-scale scene representations

Most learning-based dynamics models focus on small-scale tabletop environments and local interactions (*10*, *88*, *111*), limiting their applicability to real-world tasks that require reasoning over large, dynamic spaces. Traditional approaches, such as simultaneous localization and mapping (SLAM) (*122*), provide global geometric maps but lack dynamic information. Future directions include developing scene representations at varying levels of abstractions that capture both global structure and local interactions, training dynamics models from local interactions while maintaining scene-level coherence, and designing efficient update mechanisms that modify only affected scene regions (*123*).

Opportunities in robotic control

Hierarchical dynamics modeling and planning

Highly detailed dynamics models are not always ideal for planning, given that they induce large search spaces and high computational

Table 2. Summary of robotic tasks achieved by integrating learning-based dynamics models with planning. The table presents the designs, including representation, dynamics model class, and control methods, used for various robotic tasks. Each row represents a specific task and highlights the combination of approaches used to tackle it.

References	Representation	Dynamics model class	Control
	Object repositioning		
Agrawal *et al.* (*15*)	Latent	CNN	Greedy planner
Shen *et al.* (*60*)	Latent	MLP	Random search
Tian *et al.* (*21*)	Pixel	CNN	CEM
Manuelli *et al.* (*13*)	Keypoint	GNN	MPPI
Driess *et al.* (*23*)	Object-centric	GNN	RRT, random search
	Rope manipulation		
Yan *et al.* (*19*)	Latent	CNN	Random search
Zhang *et al.* (*107*)	Particle	GNN	MPPI
Ma *et al.* (*88*)	Keypoint	GNN	Random search
Liu *et al.* (*10*)	Keypoint	MLP	Mixed-integer programming
Driess *et al.* (*23*)	Object-centric	GNN	RRT, random search
	Cloth manipulation		
Yan *et al.* (*19*)	Latent	CNN	Random search
Hoque *et al.* (*47*)	Pixel	CNN	CMA-ES
Lin *et al.* (*110*)	Particle	GNN	Random search
Ma *et al.* (*88*)	Keypoint	GNN	Random search
	Plasticine manipulation		
Shi *et al.* (*17*)	Particle	GNN	Gradient descent
Shi *et al.* (*16*)	Particle	GNN	Learned policy
Bauer *et al.* (*111*)	Latent	Transformer	CEM
	Multiobject manipulation		
Xie *et al.* (*39*)	Pixel	CNN, LSTM	CEM
Xue *et al.* (*48*)	Pixel	CNN	Gradient descent
Ai *et al.* (*20*)	Particle	GNN	MPPI
Rezazadeh and Choi (*89*)	Keypoint	MLP	GraphMPC
Driess *et al.* (*23*)	Object-centric	GNN	RRT, random search
	Tool-use manipulation		
Xie *et al.* (*39*)	Pixel	CNN, LSTM	CEM
Shi *et al.* (*16*)	Particle	GNN	Learned policy
Wang *et al.* (*22*)	Particle	GNN	Gradient descent
Ai *et al.* (*20*)	Particle	GNN	MPPI

cost. Instead, modeling environments at multiple levels of abstraction can enable more efficient hierarchical planning (*124*) for long-horizon tasks. Low-level models can capture fine-grained physical interactions for motor control, whereas high-level models may represent skill-level transitions or abstract state dynamics to support task planning (*125*). The most effective models are often those that are sufficient for the decision at hand (effective) while remaining minimal in complexity (efficient). Future work may explore constructing such dynamics models across different abstraction levels: spatially, from particle-based to object-centric representations, and temporally, from short-horizon physical transitions to extended skill executions. An exciting direction is to investigate how to learn unified hierarchical dynamics models, or to compose and interface separate models at different abstraction levels, and integrate them with hierarchical planning frameworks to support decision-making across large spatial and temporal scales.

Learning to plan

Existing work typically obtains locally accurate dynamics models and restricts exploration to well-supported regions through engineered action spaces or carefully designed planning costs. Machine learning offers a way to automate this process. One line of work focuses on improving planning efficiency by learning heuristics to guide search (*126*) or optimizing surrogate objectives such as action space selection (*127*). Another approach is to alleviate the need for globally accurate dynamics models by learning action generative models that constrain the sampling space during planning (*128*). Despite these initial explorations, learning to identify reliable regions of learned dynamics models and to plan efficiently and robustly in the presence of model imperfections remains an open challenge.

Performance guarantees

Prediction errors in learned dynamics models can accumulate over time and degrade planning performance. Uncertainty quantification

methods, such as Bayesian neural networks (*129*) and variational inference (*130*), help mitigate this problem by additionally providing confidence estimates. Although uncertainty estimates have been used in model-based RL (*109*) and trajectory optimization (*131*), their use in robust planning remains underexplored. Coupling these techniques with theoretical guarantees from planners is a promising path toward reliable real-world deployment.

CONCLUSIONS

Learning-based dynamics models have substantially advanced robotic capabilities, from simple tasks to more complex scenarios involving long-horizon planning and deformable objects. The choice of state representation critically influences a dynamics model's accuracy, data efficiency, and state estimation requirements. This Review has presented a robotics-centric examination of dynamics models, emphasizing their integration with perception and control. Despite recent advances, key challenges remain in developing robust, generalizable, and scalable dynamics models that can serve as foundational tools for robotic manipulation. Fundamental questions persist: What representations best capture diverse scenes? How can inductive biases balance expressiveness and generalization? Addressing these questions is essential for advancing adaptive, interpretable, and robust robotic systems.

REFERENCES AND NOTES

1. J. B. Tenenbaum, C. Kemp, T. L. Griffiths, N. D. Goodman, How to grow a mind: Statistics, structure, and abstraction. *Science* **331**, 1279–1285 (2011).
2. P. W. Battaglia, J. B. Hamrick, J. B. Tenenbaum, Simulation as an engine of physical scene understanding. *Proc. Natl. Acad. Sci. U.S.A.* **110**, 18327–18332 (2013).
3. M. Müller, B. Heidelberger, M. Hennix, J. Ratcliff, Position based dynamics. *J. Vis. Commun. Image Represent.* **18**, 109–118 (2007).
4. D. Sulsky, S.-J. Zhou, H. L. Schreyer, Application of a particle-in-cell method to solid mechanics. *Comput. Phys. Commun.* **87**, 236–252 (1995).
5. A. Ajay, M. Bauzá, J. Wu, N. Fazeli, J. B. Tenenbaum, A. Rodriguez, L. P. Kaelbling, "Combining physical simulators and object-based networks for control," in *2019 International Conference on Robotics and Automation (ICRA)* (IEEE, 2019), pp. 3217–3223.
6. A. Ajay, J. Wu, N. Fazeli, M. Bauzá, L. P. Kaelbling, J. B. Tenenbaum, A. Rodriguez, "Augmenting physical simulators with stochastic neural networks: Case study of planar pushing and bouncing," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2018), pp. 3066–3073.
7. P. Wu, A. Escontrela, D. Hafner, P. Abbeel, K. Goldberg, "DayDreamer: World models for physical robot learning," in *Proceedings of the 6th Conference on Robot Learning*, K. Liu, D. Kulic, J. Ichnowski, Eds., vol. 205 of *Proceedings of Machine Learning Research* (PMLR, 2022), pp. 2226–2240.
8. D. Ha, J. Schmidhuber, "Recurrent world models facilitate policy evolution," in *Advances in Neural Information Processing Systems 31* (*NeurIPS 2018*), S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnet, Eds. (Curran Associates, 2018).
9. Y. Li, H. He, J. Wu, D. Katabi, A. Torralba, Learning compositional Koopman operators for model-based control, paper presented at the Eighth International Conference on Learning Representations (ICLR), virtual, 29 April 2020.
10. Z. Liu, G. Zhou, J. He, T. Marcucci, F. Li, J. Wu, Y. Li, "Model-based control with sparse neural dynamics," in *Advances in Neural Information Processing Systems 36* (*NeurIPS 2023*), A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine, Eds. (Curran Associates, 2023), pp. 6280–6296.
11. C. Finn, S. Levine, "Deep visual foresight for planning robot motion," in *2017 IEEE International Conference on Robotics and Automation* (*ICRA*) (IEEE, 2017), pp. 2786–2793.
12. L. P. Kaelbling, The foundation of efficient robot learning. *Science* **369**, 915–916 (2020).
13. L. Manuelli, Y. Li, P. R. Florence, R. Tedrake, "Keypoints into the future: Self-supervised correspondence in model-based reinforcement learning," in *Proceedings of the 2020 Conference on Robot Learning*, J. Kober, F. Ramos, C. Tomlin, Eds., vol. 155 of *Proceedings of Machine Learning Research* (PMLR, 2020), pp. 693–710.
14. Z. Xu, J. Wu, A. Zeng, J. B. Tenenbaum, S. Song, "DensePhysNet: Learning dense physical object representations via multi-step dynamic interactions," in *Proceedings of Robotics: Science and Systems*, A. Bicchi, H. Kress-Gazit, S. Hutchinson, Eds. (RSS Foundation, 2019).
15. P. Agrawal, A. Nair, P. Abbeel, J. Malik, S. Levine, "Learning to poke by poking: Experiential learning of intuitive physics," in *Advances in Neural Information Processing Systems 29* (*NeurIPS 2016*), D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, R. Garnett, Eds. (Curran Associates, 2016).
16. H. Shi, H. Xu, S. Clarke, Y. Li, J. Wu, "RoboCook: Long-horizon elasto-plastic object manipulation with diverse tools," in *Proceedings of the 7th Conference on Robot Learning*, J. Tan, M. Toussaint, K. Darvish, Eds., vol. 229 of *Proceedings of Machine Learning Research* (PLMR, 2023), pp. 642–660.
17. H. Shi, H. Xu, Z. Huang, Y. Li, J. Wu, RoboCraft: Learning to see, simulate, and shape elasto-plastic objects in 3D with graph networks. *Int. J. Rob. Res.* **43**, 533–549 (2024).
18. A. Longhini, M. Moletta, A. Reichlin, M. C. Welle, D. Held, Z. Erickson, D. Kragic, "EDO-Net: Learning elastic properties of deformable objects from graph dynamics," in *2023 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2023), pp. 3875–3881.
19. W. Yan, A. Vangipuram, P. Abbeel, L. Pinto, "Learning predictive representations for deformable objects using contrastive estimation," in *Proceedings of the 2020 Conference on Robot Learning*, J. Kober, F. Ramos, C. Tomlin, Eds., vol. 155 of *Proceedings of Machine Learning Research* (PMLR, 2020), pp. 564–574.
20. B. Ai, S. Tian, H. Shi, Y. Wang, C. Tan, Y. Li, J. Wu, "RoboPack: Learning tactile-informed dynamics models for dense packing," in *Proceedings of Robotics: Science and Systems*, D. Kulic, G. Venture, K. Bekris, E. Coronado, Eds. (RSS Foundation, 2024).
21. S. Tian, F. Ebert, D. Jayaraman, M. Mudigonda, C. Finn, R. Calandra, S. Levine, "Manipulation by feel: Touch-based control with deep predictive models," in *2019 International Conference on Robotics and Automation* (*ICRA*) (IEEE, 2019), pp. 818–824.
22. Y. Wang, Y. Li, K. D. Campbell, L. Fei-Fei, J. Wu, "Dynamic-resolution model learning for object pile manipulation," in *Proceedings of Robotics: Science and Systems*, K. Bekris, K. Hauser, S. Herbert, J. Yu, Eds. (RSS Foundation, 2023).
23. D. Driess, Z. Huang, Y. Li, R. Tedrake, M. Toussaint, "Learning multi-object dynamics with compositional neural radiance fields," in *Proceedings of the 6th Conference on Robot Learning*, K. Liu, D. Kulic, J. Ichnowski, Eds., vol. 205 of *Proceedings of Machine Learning Research* (PMLR, 2022), pp. 1755–1768.
24. H. Yin, A. Varava, D. Kragic, Modeling, learning, perception, and control methods for deformable object manipulation. *Sci. Robot.* **6**, eabd8803 (2021).
25. A. Longhini, Y. Wang, I. Garcia-Camacho, D. Blanco-Mulero, M. Moletta, M. Welle, G. Alenyà, H. Yin, Z. Erickson, D. Held, J. Borràs, D. Kragic, Unfolding the literature: A review of robotic cloth manipulation. arXiv:2407.01361 [cs.RO] (2024).
26. C. K. Liu, D. Negrut, The role of physics-based simulators in robotics. *Annu. Rev. Control Robot. Auton. Syst.* **4**, 35–58 (2021).
27. R. Newbury, J. Collins, K. He, J. Pan, I. Posner, D. Howard, A. Cosgun, A review of differentiable simulators. *IEEE Access* **12**, 97581–97604 (2024).
28. J. R. Kubricht, K. J. Holyoak, H. Lu, Intuitive physics: Current research and controversies. *Trends Cogn. Sci.* **21**, 749–759 (2017).
29. K. M. Lynch, H. Maekawa, K. Tanie, "Manipulation and active sensing by pushing using tactile feedback," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 1992), pp. 416–421.
30. E. Todorov, T. Erez, Y. Tassa, "MuJoCo: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (*IROS*) (IEEE, 2012), pp. 5026–5033.
31. V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, G. State, Isaac Gym: High performance GPU-based physics simulation for robot learning. arXiv:2108.10470 [cs.RO] (2021).
32. W. Zhao, J. P. Queralta, T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: A survey," in *2020 IEEE Symposium Series on Computational Intelligence* (*SSCI*) (IEEE, 2020), pp. 737–744.
33. M. Bauza, A. Rodriguez, "A probabilistic data-driven model for planar pushing," in *2017 IEEE International Conference on Robotics and Automation* (*ICRA*) (IEEE, 2017), pp. 3008–3015.
34. S. Pfrommer, M. Halm, M. Posa, "ContactNets: Learning discontinuous contact dynamics with smooth, implicit representations," in *Proceedings of the 2020 Conference on Robot Learning*, J. Kober, F. Ramos, C. Tomlin, Eds., vol. 155 of *Proceedings of Machine Learning Research* (PMLR, 2020), pp. 2279–2291.
35. K. R. Allen, T. L. Guevara, Y. Rubanova, K. Stachenfeld, A. Sanchez-Gonzalez, P. Battaglia, T. Pfaff, "Graph network simulators can learn discontinuous, rigid contact dynamics," in *Proceedings of the 6th Conference on Robot Learning*, K. Liu, D. Kulic, J. Ichnowski, Eds., vol. 205 of *Proceedings of Machine Learning Research* (PMLR, 2022), pp. 1157–1167.
36. K. Allen, T. Lopez-Guevara, K. L. Stachenfeld, A. Sanchez Gonzalez, P. Battaglia, J. B. Hamrick, T. Pfaff, "Inverse design for fluid-structure interactions using graph network simulators," in *Advances in Neural Information Processing Systems 35* (*NeurIPS 2022*), S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh, Eds. (Curran Associates, 2022), pp. 13759–13774.
37. T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez, P. W. Battaglia, Learning mesh-based simulation with graph networks, paper presented at the Ninth International Conference on Learning Representations (ICLR), virtual, 3 to 7 May 2021.

38. F. Ebert, C. Finn, S. Dasari, A. Xie, A. Lee, S. Levine, Visual foresight: Model-based deep reinforcement learning for vision-based robotic control. arXiv:1812.00568 [cs.RO] (2018).
39. A. Xie, F. Ebert, S. Levine, C. Finn, Improvisation through physical understanding: Using novel objects as tools with visual foresight. arXiv:1904.05538 [cs.RO] (2019).
40. L. Yen-Chen, M. Bauza, P. Isola, "Experience-embedded visual foresight," in *Proceedings of the Conference on Robot Learning*, L. P. Kaelbling, D. Kragic, K. Sugiura, Eds., vol. 100 of *Proceedings of Machine Learning Research* (PMLR, 2019), pp. 1015–1024.
41. H. T. Suh, R. Tedrake, "The surprising effectiveness of linear models for visual foresight in object pile manipulation," in *Algorithmic Foundations of Robotics XIV: Proceedings of the Fourteenth Workshop on the Algorithmic Foundations of Robotics*, S. M. LaValle, M. Lin, T. Ojala, D. Shell, J. Yu, Eds., vol. 17 of *Springer Proceedings in Advanced Robotics* (Springer, 2021).
42. A. Gupta, S. Tian, Y. Zhang, J. Wu, R. Martín-Martín, L. Fei-Fei, MaskViT: Masked visual pre-training for video prediction, paper presented at the Eleventh International Conference on Learning Representations, Kigali, Rwanda, 1 to 5 May 2023.
43. A. Hu, L. Russell, H. Yeo, Z. Murez, G. Fedoseev, A. Kendall, J. Shotton, G. Corrado, GAIA-1: A generative world model for autonomous driving. arXiv:2309.17080 [cs.CV] (2023).
44. Y. Du, M. Yang, P. Florence, F. Xia, A. Wahid, B. Ichter, P. Sermanet, T. Yu, P. Abbeel, J. B. Tenenbaum, L. Kaelbling, A. Zeng, J. Tompson, Video language planning, paper presented at the Twelfth International Conference on Learning Representations, Vienna, Austria, 5 to 11 May 2024.
45. M. Yang, Y. Du, K. Ghasemipour, J. Tompson, D. Schuurmans, P. Abbeel, Learning interactive real-world simulators, paper presented at the Twelfth International Conference on Learning Representations, Vienna, Austria, 5 to 11 May 2024.
46. Y. Du, S. Yang, B. Dai, H. Dai, O. Nachum, J. Tenenbaum, D. Schuurmans, P. Abbeel, "Learning universal policies via text-guided video generation," in *Advances in Neural Information Processing Systems 36* (*NeurIPS 2023*), A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine, Eds. (Curran Associates, 2023), pp. 9156–9172.
47. R. Hoque, D. Seita, A. Balakrishna, A. Ganapathi, A. K. Tanwani, N. Jamali, K. Yamane, S. Iba, K. Goldberg, VisuoSpatial Foresight for physical sequential fabric manipulation. *Auton. Robots* **46**, 175–199 (2022).
48. S. Xue, S. Cheng, P. Kachana, D. Xu, "Neural field dynamics model for granular object piles manipulation," in *Proceedings of the 7th Conference on Robot Learning*, J. Tan, M. Toussaint, K. Darvish, Eds., vol. 229 of *Proceedings of Machine Learning Research* (PLMR, 2023), pp. 2821–2837.
49. O. Rybkin, K. Pertsch, K. G. Derpanis, K. Daniilidis, A. Jaegle, Learning what you can do before doing anything, paper presented at the Seventh International Conference on Learning Representations, New Orleans, LA, 6 to 9 May 2019.
50. K. Schmeckpeper, A. Xie, O. Rybkin, S. Tian, K. Daniilidis, S. Levine, C. Finn, "Learning predictive models from observation and interaction," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX*, A. Vedaldi, H. Bischof, T. Brox, J.-M. Frahm, Eds., vol. 12365 of *Lecture Notes in Computer Science* (Springer, 2020), pp. 708–725.
51. J. Bruce, M. D. Dennis, A. Edwards, J. Parker-Holder, Y. Shi, E. Hughes, M. Lai, Mavalankar, R. Steigerwald, C. Apps, Y. Aytar, S. Bechtle, F. M. P. Behbahani, S. C. Y. Chan, N. Heess, L. Gonzalez, S. Osindero, S. Ozair, S. E. Reed, J. Zhang, K. Zolna, J. Clune, N. de Freitas, S. Singh, T. Rocktäschel, "Genie: Generative interactive environments," in *Proceedings of the 41st International Conference on Machine Learning*, R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, F. Berkenkamp, Eds., vol. 235 of *Proceedings of Machine Learning Research* (PLMR, 2024), pp. 4603–4623.
52. S. Tian, C. Finn, J. Wu, A control-centric benchmark for video prediction, paper presented at the Eleventh International Conference on Learning Representations, Kigali, Rwanda, 1 to 5 May 2023.
53. D. M. Bear, E. Wang, D. Mrowca, F. Jedidja Binder, H.-Y. Tung, R. T. Pramod, C. Holdaway, S. Tao, K. A. Smith, F.-Y. Sun, L. Fei-Fei, N. Kanwisher, J. B. Tenenbaum, D. L. K. Yamins, J. E. Fan, Physion: Evaluating physical prediction from vision in humans and machines, NeurIPS 2021 Datasets and Benchmarks Track (Round 1) (2021); https://openreview.net/forum?id=CXyZrKPz4CU.
54. S. Dasari, F. Ebert, S. Tian, S. Nair, B. Bucher, K. Schmeckpeper, S. Singh, S. Levine, C. Finn, "RoboNet: Large-scale multi-robot learning," in *Proceedings of the Conference on Robot Learning*, L. P. Kaelbling, D. Kragic, K. Sugiura, Eds., vol. 100 of *Proceedings of Machine Learning Research* (PMLR, 2019), pp. 885–897.
55. M. Watter, J. T. Springenberg, J. Boedecker, M. A. Riedmiller, "Embed to control: A locally linear latent dynamics model for control from raw images," in *Advances in Neural Information Processing Systems 28* (*NeurIPS 2015*), C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett, Eds. (Curran Associates, 2015).
56. D. P. Kingma, M. Welling, Auto-encoding variational bayes, paper presented at International Conference on Learning Representations 2014, Banff, Canada, 14 to 16 April 2014.
57. Y. Li, J. Wu, J. Zhu, J. B. Tenenbaum, A. Torralba, R. Tedrake, "Propagation networks for model-based control under partial observation," in *2019 International Conference on Robotics and Automation* (*ICRA*) (IEEE, 2019), pp. 1205–1211.
58. Y. Li, S. Li, V. Sitzmann, P. Agrawal, A. Torralba, "3D neural scene representations for visuomotor control," in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, G. Neumann, Eds., vol. 164 of *Proceedings of Machine Learning Research* (PMLR, 2021), pp. 112–123.
59. C. Li, Z. Ai, T. Wu, X. Li, W. Ding, H. Xu, "DeformNet: Latent space modeling and dynamics prediction for deformable object manipulation," in *2024 IEEE International Conference on Robotics and Automation* (*ICRA*) (IEEE, 2024), pp. 14770–14776.
60. B. Shen, Z. Jiang, C. Choy, S. Savarese, L. J. Guibas, A. Anandkumar, Y. Zhu, Action-conditional implicit visual dynamics for deformable object manipulation. *Int. J. Rob. Res.* **43**, 437–455 (2024).
61. M. J. Hausknecht, P. Stone, "Deep recurrent Q-learning for partially observable MDPs," in *AAAI Fall Symposium Series* (Association for the Advancement of Artificial Intelligence, 2015), pp. 29–37.
62. N. Hansen, H. Su, X. Wang, "Temporal difference learning for model predictive control," in *Proceedings of the 39th International Conference on Machine Learning*, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, S. Sabato, Eds., vol. 162 of *Proceedings of Machine Learning Research* (PLMR, 2022), pp. 8387–8406.
63. N. Hansen, H. Su, X. Wang, TD-MPC2: Scalable, robust world models for continuous control, paper presented at the Twelfth International Conference on Learning Representations, Vienna, Austria, 5 to 11 May 2024.
64. B. Lusch, J. Kutz, S. Brunton, Deep learning for universal linear embeddings of nonlinear dynamics. *Nat. Commun.* **9**, 4950 (2018).
65. D. Hafner, T. P. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, J. Davidson, "Learning latent dynamics for planning from pixels," in *Proceedings of the 36th International Conference on Machine Learning*, K. Chaudhuri, R. Salakhutdinov, Eds., vol. 97 of *Proceedings of Machine Learning Research* (PLMR, 2019), pp. 2555–2565.
66. F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, L. Yi, X. Chang, L. J. Guibas, H. Su, "SAPIEN: A SimulAted Part-based Interactive ENvironment," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*) (IEEE, 2020), pp. 11094–11104.
67. X. Lin, C. Qi, Y. Zhang, Z. Huang, K. Fragkiadaki, Y. Li, C. Gan, D. Held, "Planning with spatial-temporal abstraction from point clouds for deformable object manipulation," in *Proceedings of the 6th Conference on Robot Learning*, K. Liu, D. Kulic, J. Ichnowski, Eds., vol. 205 of *Proceedings of Machine Learning Research* (PMLR, 2022), pp. 1640–1651.
68. S. Chen, X. Ma, Y. Lu, D. Hsu, "Ab initio particle-based object manipulation," in *Proceedings of Robotics: Science and Systems*, D. A. Shell, M. Toussaint, M. A. Hsieh, Eds. (RSS Foundation, 2021).
69. B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, R. Ng, NeRF: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* **65**, 99–106 (2021).
70. D. Mrowca, C. Zhuang, E. Wang, N. Haber, L. Fei-Fei, J. B. Tenenbaum, D. L. Yamins, "Flexible neural representation for physics prediction," in *Advances in Neural Information Processing Systems 31* (*NeurIPS 2018*), S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnet, Eds. (Curran Associates, 2018).
71. Y. Li, J. Wu, R. Tedrake, J. B. Tenenbaum, A. Torralba, Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids, paper presented at The Seventh International Conference on Learning Representations, New Orleans, LA, 6 to 9 May 2019.
72. H. Shi, H. Xu, Z. Huang, Y. Li, J. Wu, "RoboCraft: Learning to see, simulate, and shape elasto-plastic objects with graph networks," in *Proceedings of Robotics: Science and Systems*, K. Hauser, D. A. Shell, S. Huang, Eds. (RSS Foundation, 2022).
73. H. Chen, Y. Niu, K. Hong, S. Liu, Y. Wang, Y. Li, K. R. Driggs-Campbell, "Predicting object interactions with behavior primitives: An application in stowing tasks," in *Proceedings of the 7th Conference on Robot Learning*, J. Tan, M. Toussaint, K. Darvish, Eds., vol. 229 of *Proceedings of Machine Learning Research* (PLMR, 2023), pp. 358–373.
74. A. Sanchez-Gonzalez, J. Godwin, T. Pfaff, R. Ying, J. Leskovec, P. W. Battaglia, "Learning to simulate complex physics with graph networks," in *Proceedings of the 37th International Conference on Machine Learning*, H. Daumé III, A. Singh, Eds., vol. 119 of *Proceedings of Machine Learning Research* (PLMR, 2020), pp. 8459–8468.
75. N. Tuomainen, D. Blanco-Mulero, V. Kyrki, Manipulation of granular materials by learning particle interactions. *IEEE Robot. Autom. Lett.* **7**, 5663–5670 (2022).
76. C. Schenck, D. Fox, SPNets: "Differentiable fluid dynamics for deep neural networks," in *Proceedings of the 2nd Conference on Robot Learning*, A. Billard, A. Dragan, J. Peters, J. Morimoto, Eds., vol. 87 of *Proceedings of Machine Learning Research* (PLMR, 2018), pp. 317–335.
77. L. Manuelli, W. Gao, P. Florence, R. Tedrake, "kPAM: Keypoint affordances for category-level robotic manipulation," in *Robotics Research: The 19th International Symposium ISRR*, T. Asfour, E. Yoshida, J. Park, H. Christensen, O. Khatib, Eds., vol. 20 of *Springer Proceedings in Advanced Robotics* (Springer, 2019), pp. 132–157.
78. P. R. Florence, L. Manuelli, R. Tedrake, "Dense object nets: Learning dense visual object descriptors by and for robotic manipulation," in *Proceedings of the 2nd Conference on Robot Learning*, A. Billard, A. Dragan, J. Peters, J. Morimoto, Eds., vol. 87 of *Proceedings of Machine Learning Research* (PLMR, 2018), pp. 373–385.

79. J. Grannen, P. Sundaresan, B. Thananjeyan, J. Ichnowski, A. Balakrishna, V. Viswanath, M. Laskey, J. Gonzalez, K. Goldberg, "Untangling dense knots by learning task-relevant keypoints," in *Proceedings of the 2020 Conference on Robot Learning*, J. Kober, F. Ramos, C. Tomlin, Eds., vol. 155 of *Proceedings of Machine Learning Research* (PLMR, 2020), pp. 782–800.
80. T. D. Kulkarni, A. Gupta, C. Ionescu, S. Borgeaud, M. Reynolds, A. Zisserman, V. Mnih, "Unsupervised learning of object keypoints for perception and control," in *Advances in Neural Information Processing Systems 32* (*NeurIPS 2019*), H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett, Eds. (Curran Associates, 2019).
81. T. Jakab, R. Tucker, A. Makadia, J. Wu, N. Snavely, A. Kanazawa, "KeypointDeformer: Unsupervised 3D keypoint discovery for shape control," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*) (IEEE, 2021), pp. 12778–12787.
82. Y. Ju, K. Hu, G. Zhang, G. Zhang, M. Jiang, H. Xu, "Robo-ABC: Affordance generalization beyond categories via semantic correspondence for robot manipulation," in *Computer Vision–ECCV 2024: 8th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part XLI*, A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, G. Varol, Eds., vol. 15099 of *Lecture Notes in Computer Science* (Springer, 2024), pp. 222–239.
83. A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning*, M. Meila, Tong Zhang, Eds., vol. 139 of *Proceedings of Machine Learning Research* (PLMR, 2021), pp. 8748–8763.
84. L. Tang, M. Jia, Q. Wang, C. P. Phoo, B. Hariharan, "Emergent correspondence from image diffusion," in *Advances in Neural Information Processing Systems 36* (*NeurIPS 2023*), A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine, Eds. (Curran Associates, 2023), pp. 1363–1389.
85. T. Wimmer, P. Wonka, M. Ovsjanikov, "Back to 3D: Few-shot 3D keypoint detection with back-projected 2D features," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*) (2024), pp. 4154–4164.
86. P. Sundaresan, S. Belkhale, D. Sadigh, J. Bohg, "KITE: Keypoint-conditioned policies for semantic manipulation," in *Proceedings of the 7th Conference on Robot Learning*, J. Tan, M. Toussaint, K. Darvish, Eds., vol. 229 of *Proceedings of Machine Learning Research* (PLMR, 2023), pp. 1006–1021.
87. Y. Li, A. Torralba, A. Anandkumar, D. Fox, A. Garg, "Causal discovery in physical systems from videos," in *Advances in Neural Information Processing Systems33* (*NeurIPS 2020*), H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, H. Lin, Eds. (Curran Associates, 2020), pp. 9180–9192.
88. X. Ma, D. Hsu, W. S. Lee, "Learning latent graph dynamics for visual manipulation of deformable objects," in *2012 International Conference on Robotics and Automation* (*ICRA*) (IEEE, 2022), pp. 8266–8273.
89. A. Rezazadeh, C. Choi, KINet: Unsupervised forward models for robotic pushing manipulation. *IEEE Robot. Autom. Lett.* **8**, 6195–6202 (2023).
90. Y. Wang, Z. Li, M. Zhang, K. Driggs-Campbell, J. Wu, L. Fei-Fei, Y. Li, D^3Fields: Dynamic 3D descriptor fields for zero-shot generalizable robotic manipulation. arXiv:2309.16118 [cs.RO] (2023).
91. W. Wang, A. S. Morgan, A. M. Dollar, G. D. Hager, "Dynamical scene representation and control with keypoint-conditioned neural radiance field," in *2022 IEEE 18th International Conference on Automation Science and Engineering* (*CASE*) (IEEE, 2022), pp. 1138–1143.
92. E. S. Spelke, Principles of object perception. *Cognit. Sci.* **14**, 29–56 (1990).
93. M. Janner, S. Levine, W. T. Freeman, J. B. Tenenbaum, C. Finn, J. Wu, Reasoning about physical interactions with object-oriented prediction and planning, paper presented at The Seventh International Conference on Learning Representations, New Orleans, LA, 6 to 9 May 2019.
94. K. Yi, C. Gan, Y. Li, P. Kohli, J. Wu, A. Torralba, J. B. Tenenbaum, CLEVRER: Collision events for video representation and reasoning, paper presented at the Eighth International Conference on Learning Representations (ICLR), virtual, 29 April 2020.
95. H. Qi, X. Wang, D. Pathak, Y. Ma, J. Malik, Learning long-term visual dynamics with region proposal interaction networks, paper presented at the Ninth International Conference on Learning Representations (ICLR), virtual, 3 to 7 May 2021.
96. N. Watters, D. Zoran, T. Weber, P. Battaglia, R. Pascanu, A. Tacchetti, "Visual interaction networks: Learning a physics simulator from video," in *Advances in Neural Information Processing Systems 30* (*NeurIPS 2017*), I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett, Eds. (Curran Associates, 2017).
97. R. Veerapaneni, J. D. Co-Reyes, M. Chang, M. Janner, C. Finn, J. Wu, J. Tenenbaum, S. Levine, "Entity abstraction in visual model-based reinforcement learning," in *Proceedings of the Conference on Robot Learning*, L. P. Kaelbling, D. Kragic, K. Sugiura, Eds., vol. 100 of *Proceedings of Machine Learning Research* (PMLR, 2020), pp. 1439–1456.
98. S. Tian, Y. Cai, H.-X. Yu, S. Zakharov, K. Liu, A. Gaidon, Y. Li, J. Wu, "Multi-object manipulation via object-centric neural scattering functions," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*) (IEEE, 2023), pp. 9021–9031.
99. M. Chang, T. D. Ullman, A. Torralba, J. B. Tenenbaum, A compositional object-based approach to learning physical dynamics, paper presented at the Fifth International Conference on Learning Representations, Toulon, France, 24 to 26 April 2017.
100. P. Battaglia, R. Pascanu, M. Lai, D. J. Rezende, K. Kavukcuoglu, "Interaction networks for learning about objects, relations and physics," in *Advances in Neural Information Processing Systems 29* (*NeurIPS 2016*), D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, R. Garnett, Eds. (Curran Associates, 2016).
101. H. Yu, M. Guo, A. Fathi, Y. Chang, E. R. Chan, R. Gao, T. A. Funkhouser, J. Wu, Learning object-centric neural scattering functions for free-viewpoint relighting and scene composition. *Trans. Mach. Learn. Res.* (2023); https://openreview.net/forum?id=NrfSRtTpN5.
102. Z. Huang, X. Lin, D. Held, "Mesh-based dynamics with occlusion reasoning for cloth manipulation," in *Proceedings of Robotics: Science and Systems*, K. Hauser, D. A. Shell, S. Huang, Eds. (RSS Foundation, 2022).
103. S. M. LaValle, Rapidly-exploring random trees: A new tool for path planning. *Annu. Res. Rep.* (1998); https://api.semanticscholar.org/CorpusID:14744621.
104. L. E. Kavraki, P. Svestka, J.-C. Latombe, M. H. Overmars, Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Rob. Autom.* **12**, 566–580 (1996).
105. R. Y. Rubinstein, D. P. Kroese, *The Cross Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning* (Information Science and Statistics, Springer-Verlag, 2004).
106. G. Williams, A. Aldrich, E. A. Theodorou, Model predictive path integral control using covariance variable importance sampling. arXiv:1509.01149 [cs.SY] (2015).
107. K. Zhang, B. Li, K. Hauser, Y. Li, "AdaptiGraph: Material-adaptive graph-based neural dynamics for robotic manipulation," in *Proceedings of Robotics: Science and Systems*, D. Kulic, G. Venture, K. Bekris, E. Coronado, Eds. (RSS Foundation, 2024).
108. R. S. Sutton, Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bull.* **2**, 160–163 (1991).
109. M. Janner, J. Fu, M. Zhang, S. Levine, "When to trust your model: Model-based policy optimization," in *Advances in Neural Information Processing Systems 32* (*NeurIPS 2019*), H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett, Eds. (Curran Associates, 2019).
110. X. Lin, Y. Wang, Z. Huang, D. Held, "Learning visible connectivity dynamics for cloth smoothing," in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, G. Neumann, Eds., vol. 164 of *Proceedings of Machine Learning Research* (PMLR, 2021), pp. 256–266.
111. D. Bauer, Z. Xu, S. Song, "DoughNet: A visual predictive model for topological manipulation of deformable objects," in *Computer Vision–ECCV 2024: 8th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part XLI*, A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, G. Varol, Eds., vol. 15099 of *Lecture Notes in Computer Science* (Springer, 2024), pp. 92–108.
112. H. Li, Y. Zhang, J. Zhu, S. Wang, M. A. Lee, H. Xu, E. H. Adelson, L. Fei-Fei, R. Gao, J. Wu, "See, hear, and feel: Smart sensory fusion for robotic manipulation," in *Proceedings of the 6th Conference on Robot Learning*, K. Liu, D. Kulic, J. Ichnowski, Eds., vol. 205 of *Proceedings of Machine Learning Research* (PMLR, 2022), pp. 1368–1378.
113. H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, L. J. Guibas, "KPConv: Flexible and deformable convolution for point clouds," in *2019 IEEE/CVF International Conference on Computer Vision* (*ICCV*) (IEEE, 2019), pp. 6410–6419.
114. R. Bommasani, D. A. Hudson, E. Adeli, R. B. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, E. Brynjolfsson, S. Buch, D. Card, R. Castellon, N. S. Chatterji, A. S. Chen, K. Creel, J. Q. Davis, D. Demszky, C. Donahue, M. Doumbouya, E. Durmus, S. Ermon, J. Etchemendy, K. Ethayarajh, L. Fei-Fei, C. Finn, T. Gale, L. Gillespie, K. Goel, N. D. Goodman, S. Grossman, N. Guha, T. Hashimoto, P. Henderson, J. Hewitt, D. E. Ho, J. Hong, K. Hsu, J. Huang, T. Icard, S. Jain, D. Jurafsky, P. Kalluri, S. Karamcheti, G. Keeling, F. Khani, O. Khattab, P. W. Koh, M. S. Krass, R. Krishna, R. Kuditipudi, A. Kumar, F. Ladhak, M. Lee, T. Lee, J. Leskovec, I. Levent, X. L. Li, Xuechen Li, Tengyu Ma, A. Malik, C. D. Manning, S. Mirchandani, E. Mitchell, Z. Munyikwa, S. Nair, A. Narayan, D. Narayanan, B. Newman, A. Nie, J. C. Niebles, H. Nilforoshan, J. Nyarko, G. Ogut, L. Orr, I. Papadimitriou, J. Sung Park, C. Piech, E. Portelance, C. Potts, A. Raghunathan, R. Reich, H. Ren, F. Rong, Y. Roohani, C. Ruiz, J. Ryan, C. Ré, D. Sadigh, S. Sagawa, K. Santhanam, A. Shih, K. Srinivasan, A. Tamkin, R. Taori, A. W. Thomas, F. Tramèr, R. E. Wang, W. Wang, B. Wu, Jiajun Wu, Y. Wu, S. M. Xie, M. Yasunaga, J. You, M. Zaharia, M. Zhang, T. Zhang, X. Zhang, Y. Zhang, L. Zheng, K. Zhou, P. Liang, On the opportunities and risks of foundation models. arXiv:2108.07258 [cs.LG] (2021).
115. D. Schmidt, M. Jiang, Learning to act without actions, paper presented at the Twelfth International Conference on Learning Representations, Vienna, Austria, 5 to 11 May 2024; https://openreview.net/forum?id=rvUq3cxpDF.
116. Y. R. Wang, J. Duan, D. Fox, S. S. Srinivasa, "NEWTON: Are large language models capable of physical reasoning?" in *Findings of the Association for Computational Linguistics: EMNLP 2023* (Association for Computational Linguistics, 2023), pp. 9743–9758.
117. Q. Gao, X. Pi, K. Liu, J. Chen, R. Yang, X. Huang, X. Fang, L. Sun, G. Kishore, B. Ai, S. Tao, M. Liu, J. Yang, C.-J. Lai, C. Jin, J. Xiang, B. Huang, D. Danks, H. Su, T. Shu, Z. Ma, L. Qin, Z. Hu, Do vision-language models have internal world models? Towards an atomic evaluation, paper presented at ICLR 2025 Workshop on World Models: Understanding, Modelling, and Scaling, Singapore, 28 April 2025.

118. B. Kerbl, G. Kopanas, T. Leimkuehler, G. Drettakis, 3D Gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.* **42**, 1–14 (2023).
119. G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, X. Wang, "4D Gaussian splatting for real-time dynamic scene rendering," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*) (IEEE, 2024), pp. 20310–20320.
120. T. Xie, Z. Zong, Y. Qiu, X. Li, Y. Feng, Y. Yang, C. Jiang, "PhysGaussian: Physics-integrated 3D Gaussians for generative dynamics," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*) (IEEE, 2024), pp. 4389–4398.
121. M. Zhang, K. Zhang, Y. Li, "Dynamic 3D Gaussian tracking for graph-based neural dynamics modeling," in *Proceedings of the 8th Conference on Robot Learning*, P. Agrawal, O. Kroemer, W. Burgard, Eds., vol. 270 of *Proceedings of Machine Learning Research* (PMLR, 2024), pp. 1851–1862.
122. C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, J. J. Leonard, Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Trans. Robot.* **32**, 1309–1332 (2016).
123. Y. LeCun, A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27 (2022); https://openreview.net/pdf?id=BZ5a1r-kVsf.
124. C. R. Garrett, R. Chitnis, R. M. Holladay, B. Kim, T. Silver, L. P. Kaelbling, T. Lozano-Pérez, Integrated task and motion planning. *Annu. Rev. Control Robot. Auton. Syst.* **4**, 265–293 (2021).
125. S. Li, Z. Huang, T. Chen, T. Du, H. Su, J. B. Tenenbaum, C. Gan, DexDeform: Dexterous deformable object manipulation with human demonstrations and differentiable physics, paper presented at The Eleventh International Conference on Learning Representations, Kigali, Rwanda, 1 to 5 May 2023; https://openreview.net/forum?id=LIV7-_7pYPl.
126. D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, D. Hassabis, Mastering the game of Go without human knowledge. *Nature* **550**, 354–359 (2017).
127. Y. Lee, P. Cai, D. Hsu, "MAGIC: Learning macro-actions for online POMDP planning," in *Proceedings of Robotics: Science and Systems*, D. A. Shell, M. Toussaint, M. A. Hsieh, Eds. (RSS Foundation, 2021).
128. H. Qi, H. Yin, Y. Du, H. Yang, Strengthening generative robot policies through predictive world modeling. arXiv:2502.00622 [cs.RO] (2025).
129. K. Chua, R. Calandra, R. McAllister, S. Levine, "Deep reinforcement learning in a handful of trials using probabilistic dynamics models," in *Advances in Neural Information Processing Systems 31* (*NeurIPS 2018*), S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnet, Eds. (Curran Associates, 2018).
130. C. Blundell, J. Cornebise, K. Kavukcuoglu, D. Wierstra, "Weight uncertainty in neural networks," in *Proceedings of the 32nd International Conference on Machine Learning*, F. Bach, D. Blei, Eds., vol. 37 of *Proceedings of Machine Learning Research* (PLMR, 2015), pp. 1613–1622.
131. A. Nagabandi, K. Konolige, S. Levine, V. Kumar, "Deep dynamics models for learning dexterous manipulation," in *Proceedings of the Conference on Robot Learning*, L. P. Kaelbling, D. Kragic, K. Sugiura, Eds., vol. 100 of *Proceedings of Machine Learning Research* (PMLR, 2020), pp. 1101–1112.
132. J. B. Tenenbaum, V. de Silva, J. C. Langford, A global geometric framework for nonlinear dimensionality reduction. *Science* **290**, 2319–2323 (2000).

Acknowledgments

Funding: This research is, in part, funded by the NSF AI-Center TILOS and Hillbot Embodied AI Fund. S.T. was supported by NSF GRFP grant no. DGE-1656518. C.T. was supported by A*STAR CRF funding. Y.L. is partially supported by the Toyota Research Institute (TRI), the Sony Group Corporation, Google, Dalus AI, and the DARPA TIAMAT program (HR0011-24-9-0430). This article solely reflects the opinions and conclusions of its authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsors.
Author contributions: B.A. led the project; organized the survey; drafted the introduction, background, representation (latent and object-centric), control, future work, and conclusions; and prepared the tables and figures. B.A. also contributed to correspondence and revisions throughout. S.T. revised the manuscript extensively; drafted the pixels subsection; and assisted with figures, formatting, and reviewer revisions. H. Shi contributed the particle-based subsection and figure guidance. Y.W. contributed the keypoints subsection. T.P. contributed the comparison of physics-based and learning-based models and reviewed the manuscript. J.W. and Y.L. provided guidance, input to selected sections, and comments on revisions, with Y.L. also giving the final pass and handling correspondence. C.T., H.I.C., and H. Su provided high-level advice. **Competing interests:** H.I.C. has equity interests in Robust.AI, which develops warehouse robots, and in Praxis Solutions, which develops financial AI software. H. Su has equity interests in and is the chief technology officer of Hillbot. Y.L. holds equity in and serves as an advisor to SceniX Inc. All other authors declare that they have no competing interests.

Submitted 25 November 2024
Accepted 22 August 2025
Published 17 September 2025
10.1126/scirobotics.adt1497

MEDICAL ROBOTS

Will your next surgeon be a robot? Autonomy and AI in robotic surgery

Samuel Schmidgall[1], Justin D. Opfermann[2], Ji Woong Kim[2]†, Axel Krieger[2]*

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

State-of-the-art surgery is performed robotically under direct surgeon control. However, surgical outcome is limited by the availability, skill, and day-to-day performance of the operating surgeon. What will it take to improve surgical outcomes independent of human limitations? In this Review, we explore the technological evolution of robotic surgery and current trends in robotics and artificial intelligence that could lead to a future generation of autonomous surgical robots that will outperform today's teleoperated robots.

INTRODUCTION

Every year, more than 150,000 patients in the US are diagnosed with colon cancer (*1*), and laparoscopic colectomy stands as their sole curative treatment. However, the surgery's success varies substantially because of differences in surgeons' skills, experience, and techniques, leading to complication rates reported as high as 23% in some studies (*2*). Despite the advancements in teleoperated robotic surgery, it has not notably reduced the risk of complications such as anastomotic leaks (*3*). Given these challenges, there is a growing interest in surgical robots that can perform procedures autonomously. These robots have the potential for both greater consistency and precision than even the most skilled surgeons, potentially lowering complication rates and decreasing patient wait times. Moreover, they could extend essential surgical care to areas lacking expert surgeons, including remote locations on Earth and space missions, ultimately democratizing access to quality surgical interventions.

This Review explores the technological evolution of robotic surgery and current trends in robotics and artificial intelligence (AI) that could lead to a future generation of autonomous surgical robots that will outperform today's teleoperated robots. We expand on previous reviews in the field that focused on clinical applications (*4*), regulatory and ethical considerations (*5*), medical robotics retrospectives (*6*), spine surgery (*7*), glossary terms (*8*), or health ecosystems (*9*) and those that occurred well before the current wave of AI in robotics (*10*–*13*). In this work, we briefly explore the evolution of surgical techniques from ancient times to the present day, highlighting key milestones that have shaped modern surgery. We then discuss the current state of robotic-assisted surgery (RAS) and its limitations, setting the stage for the potential of autonomous surgical robots. We examine the levels of surgical robot autonomy, from teleoperation to full autonomy, and discuss the AI methods driving progress toward increased surgical autonomy. It is important to note that this Review is limited to end-to-end learning methods, such as reinforcement learning and imitation learning, which have shown promising results for achieving autonomous surgical actions. We do not cover broader AI categories—including model-based methods like task planning and motion planning—in depth (*13*–*15*), because our goal is to highlight the recent advances in learning-based surgical autonomy.

Specifically, we focus on two primary approaches: simulation and imitation-based learning. We analyze the advantages and challenges of each method, including the development of sophisticated surgical simulators and the application of reinforcement learning and imitation learning techniques. Throughout the Review, we consider the potential impact of autonomous surgical robots on health care delivery, patient outcomes, and the future of surgical practice. We conclude with a perspective on the current trends in autonomous robotics that will likely shape the future of robotic surgery.

An overview of surgical techniques

Surgical approaches

Despite the history of surgery dating as far back as 3000 BCE (*16*), it was not until the 19th century CE that the innovation of anesthesia and aseptic techniques would lead to a paradigm shift in surgical care. Sedation, sterilization, and sterile techniques reduced infection rates and enabled more complex surgeries. Over the course of the past 60 years, surgeries have evolved from a largely open technique to procedures with no incision at all (Fig. 1). Open surgery refers to those procedures where large incisions are used to expose entire organs and allow the surgeon to directly manipulate the surgical field, such as in open-heart surgery and microvessel reconstruction. Open surgeries were made less invasive by the invention of laparoscopes and long surgical tools that provide visualization and manipulation of tissue through small surgical ports, known as laparoscopic surgery (*17*). These procedures are referred to as single-port surgeries when only a single access point is used. Surgical incisions can be completely eliminated when the single-port surgery is performed through a natural orifice, a procedure called natural orifice transluminal endoscopic surgery (NOTES) (*18*, *19*). NOTES potentially reduces postoperative pain, shortens recovery times, and minimizes the risk of wound-related complications. However, this technique presents unique challenges, including the need for specialized instruments, advanced endoscopic skills, and careful consideration of infection control measures. The development of specialized robotic systems tailored for single-port surgeries is expected to reduce these barriers and expand their application across a broad range of surgical interventions (*20*, *21*).

Robotic-assisted surgery

As surgical approaches become progressively less invasive from open to NOTES procedures, the technical skill and surgical training required to achieve high-quality surgical outcomes drastically increase. The tools used in laparoscopic surgery, for instance, may only have a single degree of freedom, which makes tissue cutting

[1]Department of Electrical and Computer Engineering, Johns Hopkins University, 3400 N. Charles Street, Baltimore, MD 21218, USA. [2]Department of Mechanical Engineering, Johns Hopkins University, 3400 N. Charles Street, Baltimore, MD 21218, USA.
*Corresponding author. Email: axel@jhu.edu
†Present address: Stanford University, Stanford, CA, USA.

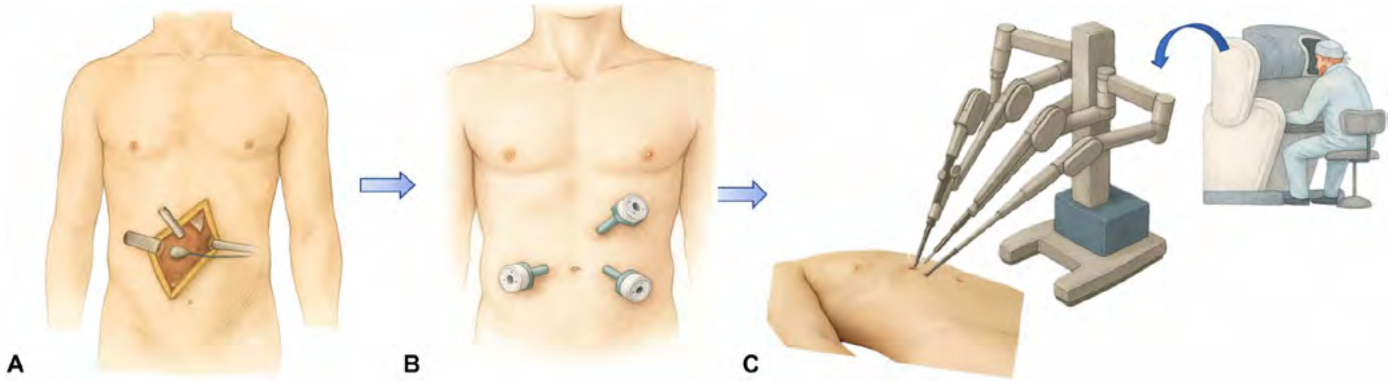


Fig. 1. The evolution of surgical techniques. (**A**) Surgeries began with open interventions where internal organs are exposed with a large incision. (**B**) Clinical and technical innovations led to less-invasive techniques where tools were inserted through multiple ports to conduct laparoscopic surgeries. (**C**) Today's state-of-the-art surgery is performed by human-controlled robotic surgical systems using multi- and single-port approaches.

and suturing challenging for even the most experienced surgeon. These challenges are amplified in single-port and natural orifice surgeries where the working space is further constrained, resulting in tool collisions and small working volumes that restrict surgical motions. Minimally invasive surgery is also notorious for poor ergonomics because of the surgeon's posture, whereas visualization is limited because of two-dimensional (2D) views of the surgical field.

Today's surgical robots are specifically designed to restore the surgeon's surgical dexterity when manipulating tissue and tools while also improving the ergonomics and visualization of minimally invasive procedures (Fig. 1C). These systems use a "leader" and "follower" design whereby a surgeon sits at an operative console (leader) and controls the exact motions of a robotic arm and tool located at the patient's bedside (follower). The tools used in robotic surgery have multiple degrees of freedom at the distal end, which restore dexterity to the surgeon and enable wrist motions of the tools, creating the perception that the surgeon's hands are directly manipulating the distal end of the surgical instrument within the patient. The surgeon's positioning at the console also improves surgical ergonomics because the head and neck rest in a natural position to observe intraoperative video from the surgical camera. The displayed endoscopic video is stereoscopic, providing qualitative 3D visualization and restoring depth perception to the surgeon. Today, RAS is commonly performed for laparoscopic, single-site, and natural orifice procedures. With more than 90% of radical prostatectomies performed with a robotic approach, RAS is the standard of care for urology, but global surgical trends illustrate rapid adoption of RAS in general surgery and gynecology (*22*). In 2023, more than 2.2 million surgeries were performed with da Vinci surgical platforms alone, and global cases are expected to top 2.5 million by the end of 2024 (*23*).

THE NEED FOR AUTONOMOUS ROBOTIC SURGERY

Although RAS has improved ergonomics and introduced technical advantages, its cost-effectiveness and long-term outcome benefits remain subjects of debate (*24*, *25*). Current surgical paradigms rely on the surgeon to control every motion of the robot, and so surgical outcomes are still directly related to the surgeon's training, skill, and decision-making during a procedure. There is a substantial learning curve in robotic surgery (*26*), surgical outcomes vary greatly between hospitals (*27*), and postoperative complications differ by surgeon (*28*).

Surgical robots, which have built-in autonomy, present a promising solution to these challenges, offering the potential to improve efficiency, safety, and consistency in surgical procedures. By providing expert-level care independent of location or surgeon experience, these systems could substantially reduce complication rates in complex operations, such as kidney and bowel surgeries. Moreover, they have the potential to address health care disparities by extending high-quality surgical care to underserved areas, contain rising health care costs, and protect health care workers during pandemics. This becomes increasingly important as surgeon shortages continue to increase across the world (*29*).

The successful integration of autonomous robots could revolutionize surgical practice in several ways. These systems could support surgeons by handling repetitive tasks, enable simultaneous oversight of multiple operations, address projected surgeon shortages (particularly in developing countries), and improve overall surgical safety and consistency. However, the path to fully autonomous surgery is not without obstacles. Key challenges include accurately differentiating between healthy and diseased tissues in complex, variable surgical environments; maintaining robust perception when blood, surgical smoke, fog, or tissue debris occlude the camera (*30*–*32*); compensating for rapid lighting changes caused by endoscope repositioning or electrocautery flashes; accommodating patient-specific anatomical variations; managing real-time tissue deformations during procedures [e.g., bowel wall excursions of ±8 mm at 1 Hz due to respiration and peristalsis (*33*)]; and reconstructing stable, drift-free 3D models of constantly deforming anatomy.

Navigating the complexities of soft tissue surgery presents additional hurdles where preoperative planning is less feasible because of variability and access limitations. To overcome these challenges, autonomous surgical robots will require advanced capabilities in accurate imaging, adaptive planning, and precise execution. In the following sections, we will introduce the varying levels of autonomy (LoAs), as well as discuss the technical paths using AI that have the potential for furthering autonomy in robotic surgery.

LEVELS OF SURGICAL ROBOT AUTONOMY

The integration of robotic surgical systems, augmented with AI, 3D visualization, and specialized tools, has introduced autonomous

capabilities surpassing those of expert surgeons in both synthetic and living tissues, with some aspects already adopted in clinically approved systems. To categorize these advancements, six tiers of LoAs were introduced in 2017 (*5*) and then refined in 2019 (*10*) to better define human supervision requirements and task-specific performance metrics. We can further delineate and clarify differences in the LoAs by adding a secondary metric originally proposed by IEC/TR 60601-4-1 (International Electrotechnical Commission Technical Report; International Organization for Standardization standard for the guidance and interpretation of medical electrical equipment and medical electrical systems) and interpreted by Kaber and Endsley (*34*) called the degree of autonomy (DoA) (Table 1). The DoA classifies the relationship between human and robot interaction during four different operating functions that can be defined in the context of surgery as follows: monitor—collect information about the patient, operator, or surgical task; generate—create possible plans or routines to achieve a surgical goal based on monitoring; select—choose a plan or routine to execute; and execute—carry out the selected plan or routine. In this Review, we differentiate the LoAs of surgical robotic systems on the basis of these four operating functions, because they provide more granularity to better capture the combinations of human and robot decision-making that constitute the autonomous capability of each specific system.

No autonomy (L-0)

With no autonomy, the human surgeon is solely responsible for monitoring, generating, selecting, and executing the surgery. During the surgery, no active robotic equipment is used. No autonomy is the current standard of care for many surgical centers that do not have access to robotic surgical equipment and is commonly referred to as manual surgery. Robotic systems for capsule endoscopy that only enable the operator to visually inspect the surgical scene and not provide intervention, such as the NaviCam (*35*), fall into this category as well.

Robot assistance (L-1)

At this level, monitoring may be shared between the human and robot, whereas the human operator retains control over the generation, selection, and execution of surgical plans. The robot supports low-level mechanical operations to improve human performance without direct involvement in surgical tasks. Examples include teleoperation with systems like the da Vinci surgical robot (*36*–*38*), Senhance Surgical System (*39*), or MUSA-3 robot (*40*), which offer dexterous tool control and tremor filtering. A key to their success is cable-driven actuators enabling wristed tool motions, adding two degrees of freedom for a full six degrees of motion inside the body, intuitively mapped to replicate the surgeon's hand movements. Other robotic

Table 1. Classifying autonomous surgical techniques on the basis of their respective LoAs and DoAs. H represents human actions taken; R represents robot actions taken.

LoA	Description	DoA	Description	Monitor	Generate	Select	Execute
0	No autonomy	0	Full manual: There is no robotic interaction during the surgical procedure, and the human is responsible for all surgical planning, decision making, and execution.	H	H	H	H
1	Teleoperation	1	Teleoperation: The robot assists the human operator during surgery, but the human maintains continuous control of the robot. The robot may perform mechanical tasks to enhance the operator's performance, but it does not perform surgical tasks on the patient.	H/R	H	H	H
2	Task autonomy	2	Preprogrammed execution: The human generates and selects the surgical plan to be performed, and the robot executes that plan.	H/R	H	H	R
		3	Shared decision: Both the human and robot generate possible surgical plans, but the human has the sole decision to select the surgical plan to be executed. The human and robot share execution of the plan.	H/R	H/R	H	H/R
		4	Decision support: The robot generates surgical plans that the human may choose to augment. The robot then performs execution of the surgical plan.	H/R	H/R	H	R
3	Supervised autonomy	5	Blended decision: The human and robot share in the generation and selection of the surgical plan. However, the robot makes an initial selection that the human can choose to confirm or modify.	H/R	H/R	H/R	R
		6	Guided decision: The robot is responsible for generating all of the surgical plans, and the human selects the surgical plan. The robot executes the surgical plan without human assistance.	H/R	R	H	R
4	High-level autonomy	7	Autonomous decision: The robot generates, selects, and executes a surgical plan. The surgical plan can be augmented by the human before it is selected or executed.	H/R	H/R	R	R
		8	Operator monitoring: The robot generates, selects, and executes the surgical plan without human assistance. The human can only monitor the task execution and intervene in cases of emergency.	H/R	R	R	R
5	Full autonomy	9	Full autonomy: The robot executes all steps of a surgical procedure and does not require monitoring from a human.	R	R	R	R

aids include flexible endoscope holders (*41*), the EndoAssist laparoscope holder (*42*), the Jaimy needle grasper (*43*), the Edison system for histrotripsy (*44*), the MagnetoSuture anastomosis robot (*45*), and Micron (*46*) for stabilizing movements in delicate surgeries.

Task autonomy (L-2)

With task-level autonomy, the human performs the selection of a surgical plan, whereas monitoring, generation, and execution of low-risk tasks may be shared with the robot, shifting from continuous to discrete surgeon control. The human's selection of the final plan and shared control during its execution are defining factors of L-2 systems. These plans can range from low-level motions, like needle grasping and insertion, to complex tasks, such as anastomosis. Examples include the AutoLap for camera positioning (*47*), the MAKO robot for arthroplasty (*48*), the KINEVO 900 for microscope positioning (*49*), the Artis Zeego for targeted x-ray imaging (*50*), the Ion endoluminal system for bronchoscopy (*51*), and research systems for needle steering (*52*) (considered L-3 when autonomous). Research with the da Vinci Research Kit (dVRK) has demonstrated L-2 autonomy for laboratory-based debridement (*53*) and blood clearing (*54*). The success of L-2 systems is largely enabled by advancements in image fusion and registration between preoperative imaging and the robot. Furthermore, robust software for tool tracking via color or fiducial segmentation and algorithms for differentiating background from foreground tissues have advanced intraoperative applications.

Supervised autonomy (L-3)

With L-3 autonomy, the human and robot share monitoring and generation of surgical plans, with the robot potentially selecting the final plan. The defining characteristic of L-3 systems is their ability to execute a task without human oversight or intervention, a substantial shift in responsibility from L-2 systems. For instance, the Smart Tissue Autonomous Robot (STAR) (*55*) generates multiple anastomosis plans and autonomously executes the one selected by a surgeon. The critical technical enabler for the STAR was quantitative 3D endoscopic imaging, allowing intraoperative 3D reconstruction of the surgical field to overlay plans with submillimeter accuracy. In addition, machine learning for real-time tissue tracking allows the STAR to compensate for tissue motion. Similarly, autonomous needle steering with (*52*) operates as a blended decision L-3 system. Clinical systems like the TSolution One (*56*) also demonstrate this level by performing surgical drilling routines on the basis of preoperative plans.

High-level autonomy (L-4)

High-level autonomy systems have shared human and robot monitoring and surgical plan generation, but selecting and executing surgical plans are the responsibility of the robot. The defining characteristic of L-4 systems is that once a procedure begins, decision-making is performed by the robot, including potential deviations from preselected surgical plans. L-4 systems have minimal human intervention, which is commonly limited to monitoring the surgery and intervening in emergency situations. The CyberKnife (*57*) and the experimental Veebot (*58*) system represent this, performing tasks like radiosurgery for deep tissue tumor ablation and vein cannulations that autonomously follow a preoperative plan, whereas the VisuMax robotic system is used for corneal refractive surgery (*59*). Most clinical L-4 systems are possible because the procedure does not require tissue dissection to expose the diseased state, and treatment can be applied noninvasively to a targeted region. This approach eliminates technical considerations and complications in automation that arise from soft tissue deformations during surgery, which has simplified automating the procedure.

Full autonomy (L-5)

With full autonomy, robots could complete surgeries independently, even in unforeseen circumstances, a stage not yet realized. The robot would solely perform monitoring, generation, selection, and execution of surgical plans without human intervention. This requires advanced detection, processing, and response capabilities, particularly for soft tissue surgeries. Although a path has been envisioned (*60*), much work remains before L-5 autonomy can be developed.

In summary, a delineation exists for procedural suitability for higher autonomy. Tasks involving rigid structures (orthopedics), noninvasive energy delivery from preoperative plans (radiosurgery), or structured subtasks (supervised anastomosis) are most amenable to near-term levels 3 or 4 autonomy. Conversely, procedures with complex, unpredictable soft tissue interactions, especially with deformation or perception challenges from bleeding or smoke, require continued human oversight (level 1 or 2) or supervisory roles (level 3 or 4). Bridging this gap for complex soft tissue surgery is a primary long-term goal, contingent on major advancements in sensing, planning, and adaptive control.

Autonomy and federal approval

Although the integration of autonomy in surgical robots has made progress in recent years, the regulatory frameworks by the US Food and Drug Administration (FDA) have not progressed at the same rate to match these advancements. A recent systematic review (*61*) indicated that most surgical robots approved by the FDA remain at a basic LoA, requiring direct control from surgeons. The analysis shows that although some surgical robots have achieved conditional autonomy (level 3), most are still at level 1, demonstrating a gap between autonomous capability and regulatory classifications. This delay suggests a need for updated regulatory frameworks that can more accurately manage the increasing autonomy in surgical robotics, especially with the introduction of autonomous AI-based robotic systems.

AI-based surgical autonomy

Figure 2 shows that less invasive robotic systems are inherently less autonomous because of the technical challenges of minimizing surgical invasiveness. Current model-based approaches are reaching their limits, illustrating the need for a paradigm shift to increase surgical autonomy for these procedures. Bridging this gap and shifting the trend line upward require new surgical paradigms.

Intraoperative planning and control must meet millisecond-level deadlines while accommodating large, nonlinear tissue motions. For example, laparoscopic cameras can pivot >90°, and instrument tips can exceed 60 mm s^{-1}. Soft-tissue stiffness may vary by two orders of magnitude, and respiration can shift targets by up to 30% of their diameter. Classical path planners like rapidly exploring random tree (RRT) assume static, rigid workspaces, incurring 3- to 7-mm positional errors when tissue elasticity or kinematic constraints are introduced. Although promising, recent model-predictive controllers require >50 ms per update, an order of magnitude slower than the <5-ms control loop needed for safe suturing. Closing this

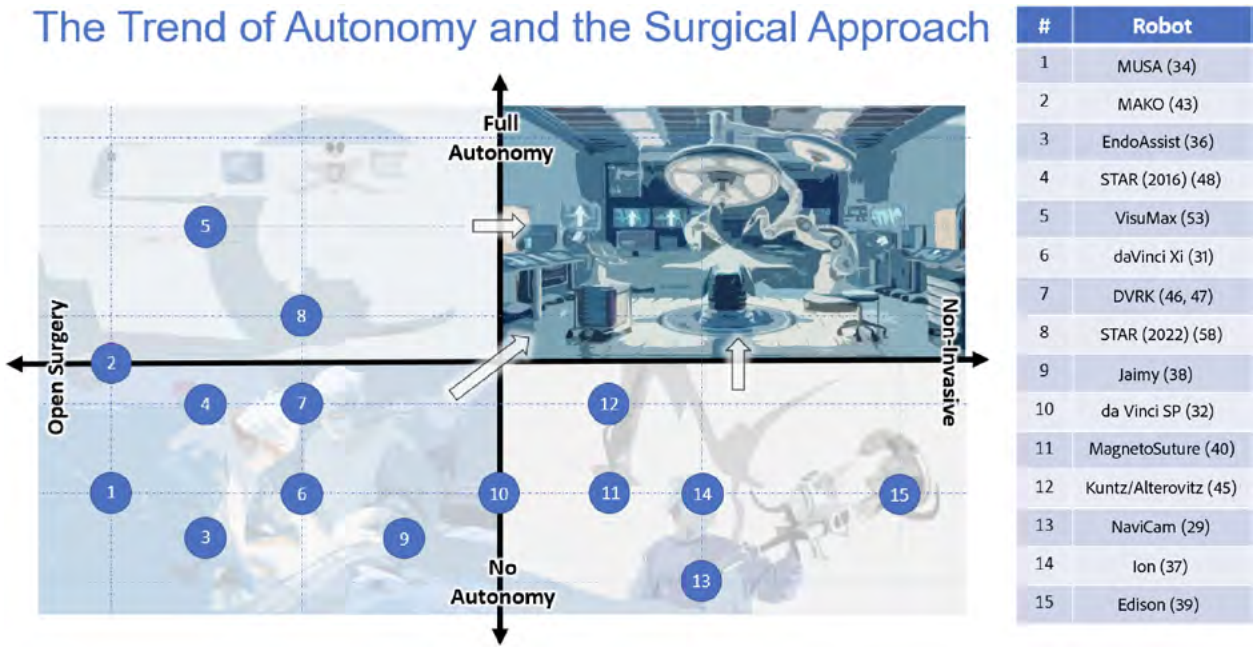


Fig. 2. Illustrating the trend of autonomy of well-known surgical robotics as a function of the surgical approach. The *x* axis represents the measure of invasivness of the surgery, and the *y* axis represents the amount of robotic autonomy. In general, as surgical approaches become less invasive, the need for robotic solutions increases, but the autonomous capability of those systems has not been realized. This is illustrated by the absence of highly autonomous and noninvasive robots in the upper right quadrant. Developments in AI are needed to fundamentally shift the development of robots such that less-invasive surgical approaches can be performed with increased LoAs. Each number represents a notable robotic surgical system plotted at their approximate LoA and invasiveness.

gap demands hybrid methods that fuse fast, sampling-based planners with neural surrogates of soft-body dynamics and closed-loop feedback.

Historically, the greatest advancements in surgical autonomy have been model-based (*55*, *62*–*64*); however, AI now offers a promising direction to learn directly from expert surgeons and from simulated surgical scenes. These methods may be the key to increasing autonomy in less invasive procedures. In the next sections, we outline the key methods toward increasing surgical autonomy with AI and where the future of AI in robotic surgery is moving. Although AI plays a pivotal role in the future of surgical robotics, it is important to recognize that it is not the sole factor driving innovation in surgical robotics. Many other works, such as smaller, softer robotic systems and improved image guidance, contribute to the state of the art and encompass a broad range of interdisciplinary efforts.

AI METHODS TOWARD INCREASED SURGICAL AUTONOMY

Advances in AI are expanding the autonomy of surgical systems, although they are not yet ready to replace expert surgeons. This Review focuses on emerging learning-based AI approaches—namely, reinforcement learning (RL) and imitation learning (IL)—because these methods show great promise in achieving higher LoAs (see Fig. 3). We also explore future trends, highlighting the potential for high-capacity vision-language-action (VLA) models (*65*). Although our Review emphasizes learning-based methods, we recognize the contributions of non–learning-based approaches to robotic autonomy, including task planning, motion planning, and computer vision.

Overview of reinforcement learning

RL (*66*) is a machine learning framework in which a robotic agent learns how to act in an environment by receiving rewards for its actions. Key concepts in RL include the state, action, reward, and policy. A state (denoted s) represents the environment at a given moment—for instance, an endoscopic image or the current joint positions of a surgical robot. An action (denoted a) refers to a possible operation the agent can take while in a given state, such as moving a surgical instrument to grasp tissue. A reward (denoted r) is a numerical signal that reflects the quality of the agent's action in a specific state—for example, how close the robot is to a surgical target. Last, a policy (denoted π) is a strategy that maps states to actions, guiding the agent in selecting which action to take next on the basis of the current state.

For example, imagine a robot learning to position a needle in a tissue sample. The state could be the robot's camera view, the action could be different ways of moving the instrument, and the reward might increase when the robot properly aligns the needle for a precise insertion.

In RL, the behavior of the agent is dictated by a policy $\pi{:}\pi(a|s) = \Pr(a|s)$, which means that for any given state s, the policy assigns a probability to each possible action a. In contrast, the environment's state transition probabilities, denoted by $P(s'|s, a)$, define the likelihood of transitioning to a new state s' given the current state s and the action a taken.

Value function and Q-function

The value function, denoted $V^{\pi}(s)$, estimates how much total reward the robot can gain starting from state s under policy π. The Q-function, $Q^{\pi}(s, a)$, refines this idea further by estimating the total reward for

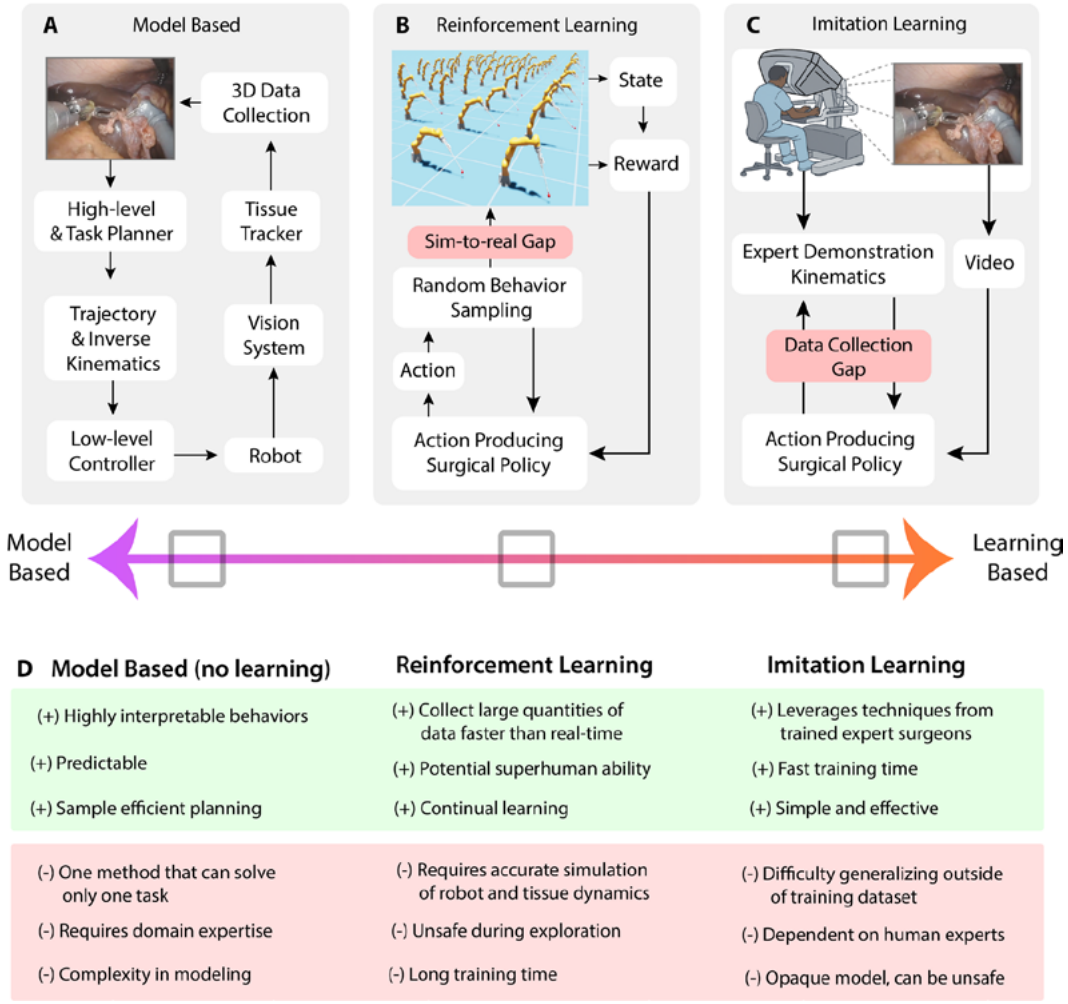


Fig. 3. Comparison of three major control techniques in robotic surgery. (**A**) Demonstration of model-based workflow. (**B**) RL workflow, with the major challenge being translating learned policy from simulation to real hardware. (**C**) IL workflow, with the main challenge being collecting data to perform imitation. (**D**) Comparison of the positive and negative factors of each technique.

taking a specific action *a* in state *s* (*67*). These functions guide the robot in choosing actions that ultimately yield higher accumulated rewards.

Q-learning

A well-known RL method is Q-learning, which can learn an optimal policy without needing a detailed model of the environment (*68*). It updates its Q-function on the basis of the difference between the predicted reward and the actual reward received when an action is taken. Because Q-learning can learn from different data sources, it is often described as an "off-policy" algorithm. In robotic surgery, Q-learning could iteratively improve how an agent handles specific subtasks, such as maneuvering instruments around delicate tissues. For a surgical robot, this is particularly beneficial because accurately modeling the complex interactions in a surgical setting (such as tissue deformation, tool-tissue interactions, and bleeding) is extremely challenging. Instead, the robot learns effective policies directly from its interactions (or simulated interactions) with the environment, thereby adapting to the variability and uncertainty inherent in surgical procedures.

Surgical simulators

The development of surgical simulations has increased in recent years, with most of these simulators focusing on surgical skill tasks, such as picking up pegs or transferring a needle. However, more advanced simulators are just beginning to emerge, more closely resembling real surgical scenes (*69*–*71*). Simulators provide a safe environment for experimentation for robots. This is especially important for surgical robots, where the robot is operating inside real patients who could be harmed if any imprecise actions are taken.

One of the pioneering platforms in this area is the da Vinci Reinforcement Learning (dVRL) platform (*72*). This platform was designed to help engineers without a surgical background create algorithms for autonomous surgery. dVRL offers two basic training tasks: a robotic reach task and a pick-and-place task. A controller trained using dVRL was demonstrated to transition from simulation to actual robotic hardware, using a robot to suck blood in a surgical demonstration. Built on the Unity3D platform, which supports deformable objects, UnityFlexML (*73*) also demonstrates successful policy transfer to the dVRK, similar to dVRL.

CREDIT: (B) ADAPTED FROM (*69*)

LapGym (*74*) is an RL environment that offers a diverse range of tasks for developing and testing automation in laparoscopic surgery. This platform covers skills like spatial reasoning, manipulating and grasping deformable objects, dissection, and thread manipulation. It presents a variety of image-based learning challenges and supports multiple data formats like RGB (red, green, blue), depth, point clouds, and semantic segmentation. CRESSim builds on PhysX (*75*) and enables the simulation of several surgical features, such as soft tissue and body fluids (*70*), focusing on photorealism. SurRoL (*76*) is an open-source RL-focused simulation platform, compatible with the dVRK like its predecessors. It provides environments for different components of the da Vinci system and multiple arms and has been shown to transfer to real-world dVRK problems, such as simple pick-and-place and point tracking. Last, Surgical Gym (*69*) and ORBIT-Surgical (*71*) are open-source graphical processing unit (GPU)–based RL environments that support several surgically relevant training tasks. These works focus on making the simulator more efficient, which made training data much more accessible than previously possible using a GPU-based physics simulator. Overall, this platform (*69*) demonstrates between 100 and 5000 times faster training times compared with previous surgical learning platforms (*70*, *76*). Simulation-based RL methods provide rapid policy development by enabling the exploration of a wide range of scenarios in a safe, virtual environment; however, the sim-to-real gap remains a challenge because of simplified physical dynamics and limited photorealism.

Challenges with simulation-based learning

The simulation-based RL approach has many challenges that prevent it from being used in certain fields, particularly in surgery. One of the greatest challenges thus far is the current design of simulators. Now, most simulators are also not capable of capturing properties that are reflective of the real world in a phenomenon referred to as the sim-to-real gap (*77*–*79*).

The sim-to-real gap

Although simulations can be highly detailed and realistic, they inevitably cannot match certain aspects of real-world complexity. This gap leads to models that perform well in simulation but fail to generalize to real-world conditions (*78*–*80*). There are two primary ways in which simulators will not match reality: physical dynamics and photorealism (*80*, *81*). Physical dynamics in simulations are generally a simplified version of real-world physics. Elements like tissue elasticity, complex tool-tissue interactions, and bleeding are not accurately captured. The visual accuracy, or photorealism, of simulations also falls short of real-world environments (*81*). Fluid light reflection, tissue texture, and bleeding under varying conditions are difficult to replicate precisely (*33*). This is particularly problematic for tasks that require precise visual input, such as complex manipulation tasks encountered during surgery.

Some works have focused on crossing the sim-to-real gap with surgical robots for physical dynamics, such as for grasping tissue (*82*), online registration (*83*), and pose estimation (*84*). Other works have considered using intermediate visual representations, such as segmentation maps, which do not vary substantially between simulation and reality, to perform blood suctioning tasks (*33*). A few works have achieved direct simulation-to-real hardware transfer using RGB visual inputs for rigid (*85*) and soft-body (*80*) manipulation tasks.

Imitation-based autonomy

Imitation-based autonomy in surgical robotics takes inspiration from how human surgeons learn and refine their skills. Just as junior surgeons observe and emulate the techniques of experienced surgeons, models can also be trained to mimic the actions of expert surgeons. This approach involves capturing the movements, decision-making processes, and techniques of skilled surgeons as they perform the procedures.

One approach to this technique is IL (*86*–*88*). In IL, robots are trained on datasets of recorded surgical procedures by expert surgeons. These datasets contain rich information about tool movements, tissue interactions, and decision points throughout various surgical tasks (*89*–*91*). By learning from these data, robotic systems can develop policies that aim to replicate expert-level performance across a range of surgical scenarios (*86*).

One of the main advantages of imitation-based over reward-based learning is that this method can often bypass the need for a highly accurate simulator, thereby avoiding sim-to-real issues. In addition, IL can potentially adapt to different surgical styles, because it can be trained on demonstrations from multiple users with diverse behaviors (*92*). However, this approach also faces challenges, such as the need for large, high-quality datasets of surgical demonstrations and the potential to replicate suboptimal behaviors if present in the training data. In the next section, we provide background of two major techniques in IL and discuss specific examples from prior work.

Overview of imitation learning

IL can be particularly valuable in surgical contexts where defining explicit reward signals is challenging because the desired behavior is complex. IL can be considered a form of supervised learning, with the surgeon acting as the expert, which the model must learn to imitate.

The core concept of IL involves learning a policy that maps states to actions. The states can often be defined as visual observations of the surgical scene (e.g., images from an endoscope) or sensorimotor information (e.g., current effector pose of the robot arms) (*86*, *93*). Actions can often be defined as desired goal poses that the end effectors must reach (*86*, *93*). The primary objective is to develop a policy that accurately replicates the expert surgeon's decision-making process on the basis of a dataset of state-action pairs. There are two main approaches to IL: direct imitation, which learns to replicate the expert's policy by outputting similar commands given the same state, and indirect imitation, which learns a reward function that captures the expert's behavior.

In IL, the learning objective is typically formalized as minimizing a loss function. A common example is the mean-squared error, which measures the difference between the learned policy's output and the expert's action for each state in the demonstration dataset (*86*, *93*).

Behavior cloning

Behavior cloning (BC) is a direct approach to IL that closely aligns with supervised learning principles. It involves training a policy function from a dataset of expert demonstrations. The optimization problem in BC aims to minimize the difference between the learned policy's actions and the expert's actions across all demonstrations.

BC faces several challenges in surgical robotics. The trained policy may encounter states not present in the training set, leading to potentially unpredictable behavior. This is particularly problematic

when the demonstrations come from sequential states and actions, resulting in a mismatch in distributions between the learned policy and expert demonstrations (*94*). Creating a comprehensive dataset often requires manually collecting data over many possible set of states, which can be labor intensive.

To address this challenge, a common approach is to implement some form of DAgger (Dataset Aggregation) (*94*–*96*), a procedure where the operator corrects the policy mistakes at the time of deployment and the corrections become part of the training dataset. Over several iterations of this procedure, the amount of intervention decreases and the policy performance improves.

Inverse reinforcement learning

Inverse reinforcement learning (IRL) (*97*) is another approach to IL that aims to deduce the underlying reward function that the expert surgeon is assumed to be optimizing. This approach offers several advantages: It provides insight into the reasons behind expert behavior, it can accommodate suboptimal expert demonstrations, and it allows for differences in robot-surgeon dynamics or capabilities. IRL typically assumes a parameterized reward function, often as a linear combination of nonlinear features.

IRL faces several challenges in surgical robotics. Multiple reward functions can often explain the same set of expert demonstrations, leading to ambiguity. IRL methods often operate under the assumption that the expert's behavior is optimal, which can be problematic when human surgeons exhibit suboptimal or inconsistent behavior. In addition, IRL methods need to be scalable to complex surgical environments and capable of generalizing from limited data.

Imitation learning in surgery

Thus far, IL approaches with surgical robots (*91*) are generally performed either in simulation or on tabletop tasks, such as fundamentals of laproscopic surgery (FLS) tasks. These environments provide a controlled setting that allows for the detailed capture and analysis of model behaviors, which are essential for model development (Fig. 4). Below, we discuss examples of relevant prior works.

Simulation

In simulation, IL methods have been used to enable surgical robots to refine their skills in a safe and controlled environment. Some works focus on IL on realistic tissue simulations, such as LapGym (*74*), which includes a gallbladder, abdominal cavity, and tissue dissection model. LapGym (*74*, *93*), Orbit-Surgical (*71*), and SuRoL (*76*, *98*, *99*) have tabletop and FLS skill simulations for surgically relevant tasks such as peg pick-and-place and rope threading. The primary challenge with these methods is that they face the same sim-to-real challenge as the RL simulated solution.

Tabletop tasks

Outside of simulation, tabletop and FLS tasks are considered the first step toward teaching robots the necessary skills required during surgical procedures and have thus far received the most attention in IL. These tasks typically involve simpler operations, such as cutting, suturing, or manipulating tissues, which are fundamental to many surgical procedures.

Tanwani *et al.* (*100*) learned an embedding feature space from video frames by minimizing a metric learning loss (i.e., images from similar actions are clustered while pushed away from random images), which was used to perform end-effector pose imitation. In work from Kawaharazuka *et al.* (*101*), Panda robot arms with laparoscopic attachments were trained to perform peg transfer using monocular images. The Surgical Robot Transformer (SRT) (*102*) solved three challenging surgical tasks, including knot tying, tissue manipulation, and suture pick-up. Wrist cameras were used to improve performance, and a relative action representation was used to improve kinematic errors on the dVRK. Motion primitive diffusion was introduced by Scheikl *et al.* (*93*) to solve a tissue deformation problem. This method adapts motion primitives to existing diffusion-based techniques for gentle manipulation of deformable objects. Kim *et al.* (*103*) demonstrated reliable navigation to desired target points in retinal surgery by combining optimal control and imitation learning (*104*).

The primary challenge for these techniques is acquiring sufficient data to train an imitation policy. At the time of writing, kinematic data are not easily acquirable from existing surgical robots because of company regulations. Existing systems that are able to collect kinematic data (e.g., the dVRK or adapted industrial robots) are not approved for use in humans; thus, it is challenging to acquire sufficient expert surgical data. One possible direction for future work is to develop an accurate kinematics estimator from video frames, as has been done in prior works (*105*, *106*), that could be used to collect kinematic data from real surgeries. Although IL leverages high-quality expert demonstration data to replicate sophisticated surgical behaviors, its effectiveness is limited by the availability and consistency of such data; furthermore, the risk of replicating suboptimal human actions necessitates careful curation and validation of demonstration datasets.

Combining imitation learning and reinforcement learning

RL and IL can be combined in several ways to leverage expert demonstrations for improved sample efficiency, exploration, and

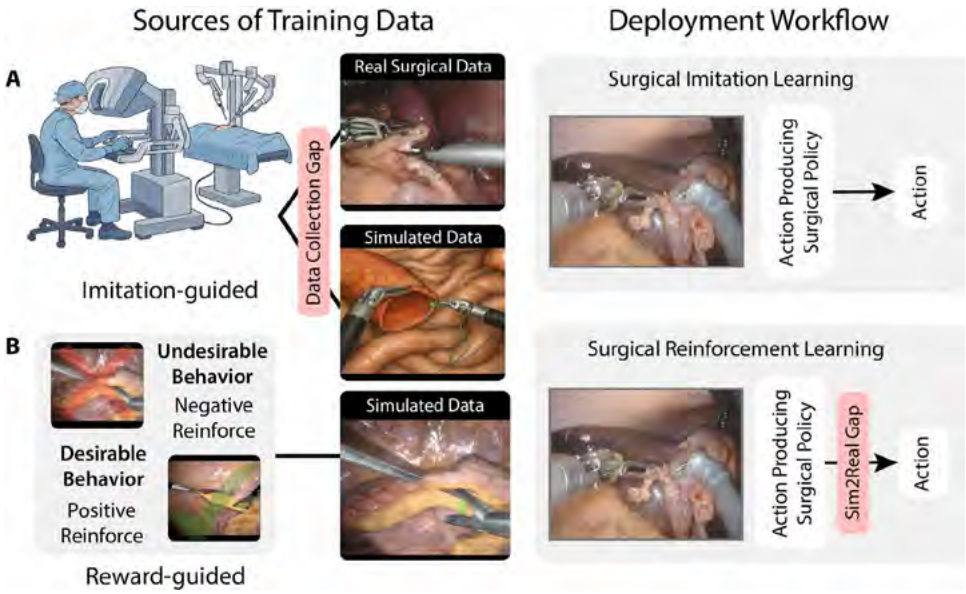


Fig. 4. Overview training data and deployment workflows for RL and IL. (**A**) Imitation relies on expert demonstrations, whereas (**B**) RL relies on randomly sampling behaviors and measuring the quality. IL suffers from the data collection gap, whereas RL suffers from the sim-to-real gap.

performance. Early methods such as (*107*–*109*) used demonstration data to augment standard RL algorithms. For instance, deep Q-learning from demonstrations (*107*) and deep deterministic policy gradient from demonstrations (*108*) bootstrap Q-learning and continuous control agents, respectively, from limited demonstration trajectories. Demonstration-augmented policy gradient (DAPG) (*109*) similarly combines BC with on-policy optimization, showing performance gains on complex robotic tasks.

Offline RL approaches also benefit from combining IL-inspired objectives. Conservative Q-learning (CQL) (*110*) avoids overestimating out-of-distribution actions by penalizing high Q-values on unseen state-action pairs, whereas TD3 + BC (*111*) explicitly adds a BC term to stabilize offline training. Related works like soft-Q imitation learning (SQIL) (*112*) and IQ-Learn (*113*) reframe imitation as sparse-reward Q-learning, effectively prioritizing expert transitions in offline datasets. Beyond purely algorithmic developments, the application of RL and IL in medical robotics has been limited, except one example by Keller *et al.* (*114*). In this work, the authors automated deep anterior lamellar keratoplasty on ex vivo human corneas using RL from demonstration (RLfD). Their findings demonstrated that the final trained policy not only achieved higher precision than that of surgical fellows but also improved consistency and accuracy in needle insertion depth, highlighting the potential of RLfD to improve surgical outcomes.

Case studies

We focus our attention on two representative works that explored promising strategies in IL (*102*) and RL (*80*). SRT (*102*), as shown in Fig. 5, falls in the category of IL that demonstrated complex surgical behavior in tabletop tasks using real-world data. Specifically, SRT demonstrated suture knot tying, needle manipulation, and soft-tissue manipulation using expert demonstration data and the dVRK system. Beyond tabletop settings, SRT showed zero-shot generalization of the trained models in unseen scenarios, such as knot tying on porcine tissues and needle manipulation on chicken tissue, although the models were trained using silicone suture pads only. SRT also addressed a fundamental issue, which involves dealing with the notorious inaccurate forward kinematics of the dVRK system. Specifically, the dVRK joint measurements are prone to errors because of a lack of precise motor encoders, hysteresis, and overall slack and flexibility in its mechanism. Given that the kinematics accuracy is low, the pool of surgical data collected using da Vinci systems is inevitably inconsistent, posing difficult challenges for robot learning. SRT showed that using such inconsistent kinematic data led to high failure rates, and the learned policies were not usable across all tasks. To alleviate this issue, SRT introduced tool-centric and hybrid-relative action representations, which enabled direct use of the inconsistent kinematic data, without performing further calibration or corrective steps to train a feasible policy. This work demonstrated promise for using large-scale approximate kinematic data for learning complex manipulation behaviors, which can be useful for leveraging the large repository of surgeon data to effectively build large-scale models for autonomous surgery.

Another promising work by Scheikl *et al.* (*80*), as shown in Fig. 5, falls in the category of RL while overcoming the sim-to-real gap to transfer policies trained in simulation to the real world. Specifically, (*80*) presented a pipeline that leveraged pixel-level domain adaptation through contrastive generative adversarial networks (GANs) to train visuomotor policies in simulation, which was successfully deployed in the real world on a dVRK without retraining. The core idea was to train a model in simulation using translated images that looked like real-world images so that the sim-to-real gap was reduced when deployed in the real world. The study focused on the tissue retraction task, a critical step during tissue dissection. The proposed approach demonstrated a 50% success rate using raw RGB images, which is a promising result for sim-to-real strategies. This work not only demonstrated the potential for using RL in complex surgical tasks but also set a precedent for efficient transfer of skills from sim-to-real in robotic surgery, making RL a more viable tool for automating surgical tasks.

Comparison of reinforcement learning and imitation learning

RL and IL each present distinct advantages and limitations for increasing robotic autonomy in surgery, as illustrated in Fig. 3. RL relies on a simulator, but because simulating soft tissue deformation is challenging, these environments can lack realism, leading to difficulties in translating robot behaviors to real patients. IL relies on demonstrations by expert surgeons, but if the dataset lacks variety or contains suboptimal demonstrations, the learned policy will fail to generalize or will replicate undesirable behaviors.

Safety considerations also differ. RL leverages random exploration, which is infeasible in live surgical scenarios. Conversely, IL reproduces known expert trajectories, making early deployment safer but potentially overlooking rare complications not present in the original dataset. For adaptability, RL can discover strategies that deviate from standard practice, whereas IL may lack the capacity to adapt to out-of-distribution events.

Despite these differences, it is possible to combine RL and IL into a hybrid framework, merging the safety and stability of IL with the exploratory power of RL. One promising approach is to initialize the robot's policy by imitating expert demonstrations and then refine this policy through reinforcement.

We highlight representative performance metrics from prior IL and RL studies, primarily in laparoscopic settings. Kim *et al.* (*86*) used IL with a few hundred demonstrations per task and reported a 100% success rate in a needle pickup task and 90% in a knot tying task. In (*81*), an RL approach using proximal policy optimization (*115*) achieved 100% success in reaching tasks and 90% in rigid object manipulation. Similarly, Scheikl *et al.* (*93*) reported 78.6% success in rope manipulation and 99% in general object manipulation using diffusion models. These results suggest that, with sufficient data and in relatively simple tabletop environments, learned policies can consistently achieve 80 to 90% success rates in nontrivial manipulation tasks.

Control strategies

Cable-driven surgical robots are susceptible to hysteresis and backlash, which cause non-negligible kinematic errors (*116*). Although hand-eye calibration can compensate, it often requires repeated, impractical recalibration for different tools in time-sensitive settings (*117*–*119*). One strategy is online calibration; Li *et al.* (*120*) used a particle filter fusing kinematic priors with visual features to estimate tool pose in real time, even in deformable environments, while accounting for uncertainty. A simpler alternative avoids absolute kinematics altogether. In (*86*), policies used relative motion commands, enabling robust control where the robot continuously refined its motion with visual feedback instead of seeking a fixed, potentially inaccurate spatial target. For instance, a relative-motion policy can iteratively adjust a needle grasp, whereas an absolute policy might repeatedly fail because of kinematic drift.

Sensors used for autonomous surgical AI

Sensor technology is an important component of autonomous surgical systems. High-resolution imaging sensors—most commonly

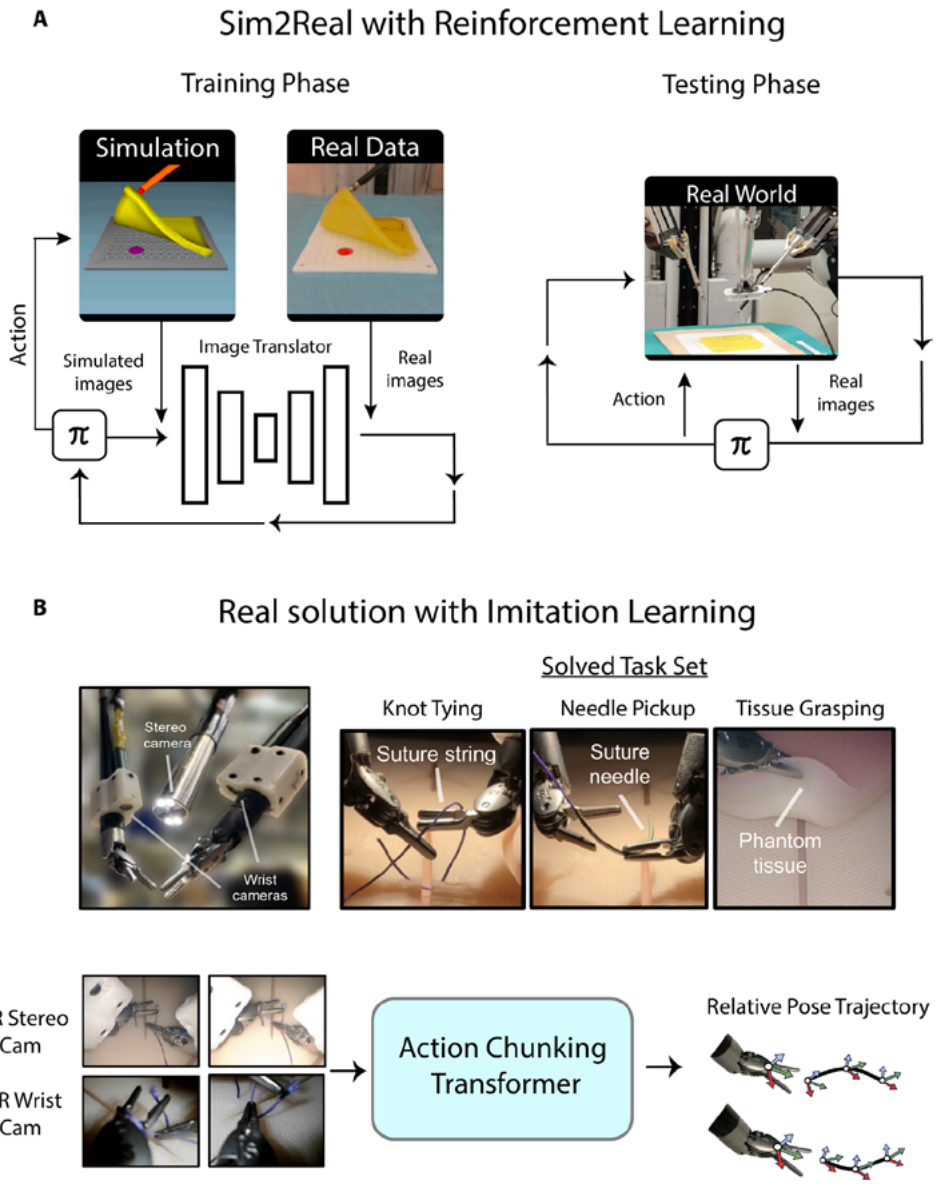


Fig. 5. Examples of two surgical systems that were successfully deployed on hardware. (**A**) Policy that was trained in simulation and used an image translator to synchronize simulation frames with real data. (**B**) Policy that used IL via an action chunking transformer to mimic surgical trajectories.

dual-camera stereoscopic endoscopes (*86*, *90*)—provide the raw RGB data and coarse disparity cues that current learning algorithms rely on. In addition to vision, a variety of other sensors—including force-torque sensors (*121*) and tactile sensors (*122*, *123*)—provide real-time feedback on tool-tissue interactions. The integration of these multimodal sensor streams enables precise depth perception, improved situational awareness, and adaptive control in dynamic surgical environments.

Nonetheless, today's vision and depth-sensing stack remains the rate-limiting step for reliable 3D understanding in vivo. Stereo endoscopes produce noisy disparity maps beyond a few centimeters and fail in specular or blood-covered regions. Time-of-flight and structured-light probes suffer from multipath interference inside fluid-filled cavities, whereas photometric methods are confounded by smoke, glare, and tissue translucency. Existing reconstructions therefore drift, exhibit zipper artifacts, or collapse entirely when the scope pivots or the scene deforms. These shortcomings motivate algorithms that jointly reason over geometry, appearance, and instrument motion—for example, learning-based monocular depth estimation, differentiable simultaneous localization and mapping (SLAM) that fuses frame-to-frame geometry with kinematic priors, and neural implicit surfaces that update in real time. Pioneering systems such as STAR demonstrated submillimeter 3D point-cloud overlays yet required a rigidly mounted scope and precalibrated lighting; next-generation methods must generalize across freehand endoscope motion, variable illumination, and occlusions.

Despite these advances, reliable perception in a live operating room remains notoriously difficult. Blood splatter, tissue debris, and smoke plumes can occlude critical anatomy, whereas irrigation and insufflation introduce dynamic lighting changes and airborne particulates that confuse vision pipelines (*124*–*126*). State-of-the-art systems therefore fuse physics-based data augmentation, domain-randomization techniques that vary lighting and optical properties during training, and multispectral or polarization imaging to increase resilience (*126*–*128*). Learning-based artifact detection modules can identify active bleeding or smoke and trigger adaptive exposure, suction, or complementary sensing to preserve situational awareness (*129*, *130*). Recent demonstrations of autonomous blood-clearing suction and robust sim-to-real transfer with aggressive camera lighting randomization illustrate practical pathways to overcoming these perception challenges (*54*).

THE FUTURE OF AI IN ROBOTIC SURGERY

Now, the primary approach in robot learning has been to try and solve task-specific goals that address individual challenges, such as, in surgery, autonomous suturing or needle pick-up. Recent developments, however, have trended toward high-capacity models that showed major improvements in performance when trained on extensive, diverse, and nonspecific imitation datasets (*131*–*135*). These models excel in adapting to new environments, a capability that grows with increases in data volume and model complexity (*133*).

This approach differs from what has been done with surgical robot systems, which depend on data-driven learning and have progressed more slowly compared with other areas in robotics. This delay can be attributed to a few key issues: the scarcity of large,

CREDIT: (A) ADAPTED WITH PERMISSION FROM *IEEE ROBOTICS AND AUTOMATION LETTERS* (*93*); (B) ADAPTED FROM (*86*)

open-source datasets for training, challenges in modeling the soft-body deformations encountered during surgeries, and the increased risk of patient injury during clinical trials, which calls for stricter safety protocols (*60*). Although small task-level imitation policies with open datasets have been achieved (*136*, *137*), there does not exist a larger repository of data to date. As suggested by new advances in robotics, a path toward improving the autonomy of surgical robots could be through the development of a large, multimodal, VLA model specifically for surgery.

In this section, we outline an approach toward building an "AI robotic surgeon" (Fig. 6) by first building a vision-language model for surgery trained on medical text and image content all the way from premedical studies through surgical residency. This model could then be trained using IL on kinematic data from surgical skills training, animal surgeries, and full-human surgeries. Last, this model could continually improve with experience. Below, we describe these ideas in depth.

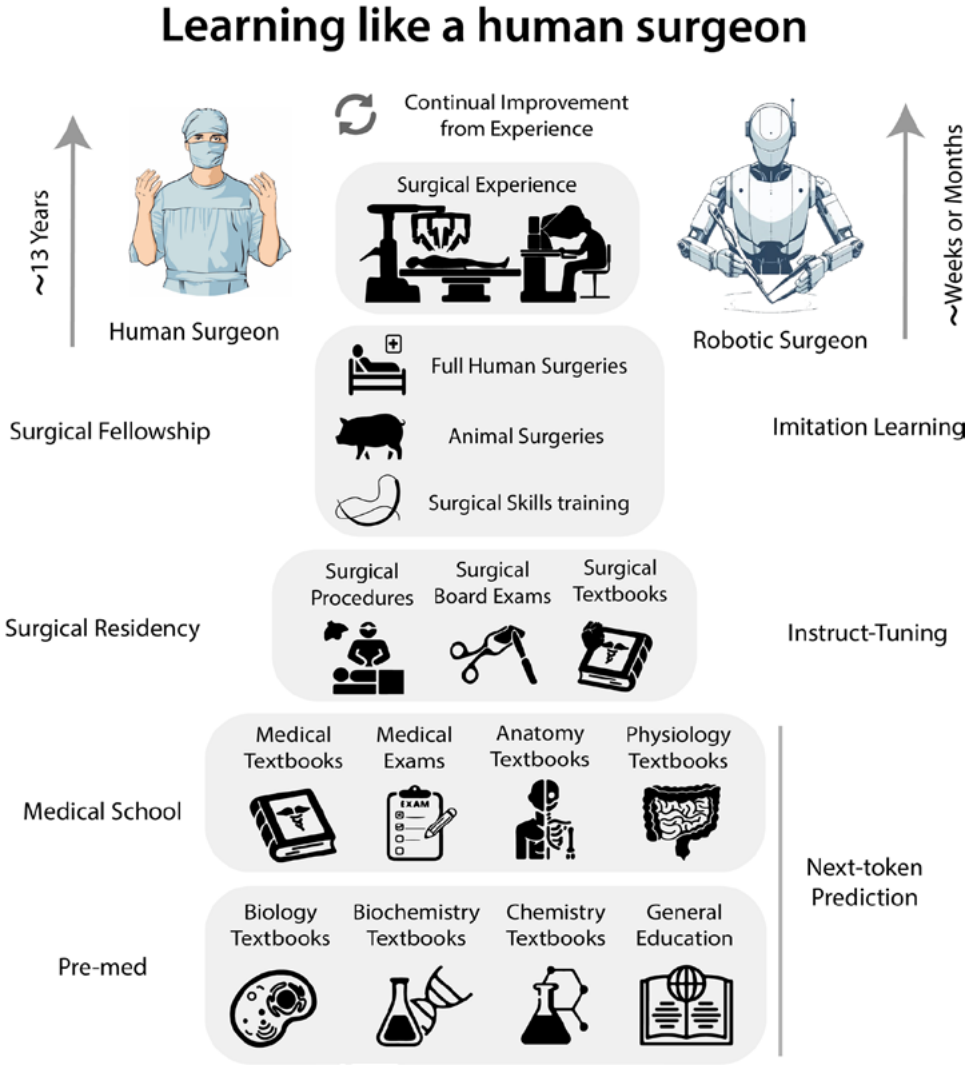


Fig. 6. Outline of the AI robotic surgeon. A language model is trained via next-token prediction and instruct tuning on the same information as surgeons all the way up through surgical residency. A VLA model is then trained via IL on surgical skills (e.g., suture pads), animal surgeries, and full human surgeries. Last, the model continually improves through experience using RL.

Building a foundation for language in surgery

VLAs are typically built using language models as the foundation for further training (*65*, *138*). Common foundations for language models include Llama 2 (*139*), Vicuna (*140*), and PaLM-E (*141*). These models are trained to imitate text from trillions of "tokens" (chunks of words), which represent human-generated natural language through a process called next-token prediction (*142*). This trained model is then often fine-tuned on instruction datasets (e.g., question-answering) in a process called instruct tuning (*143*–*145*). Because most VLAs are just performing household tasks involving common objects, now, the language models serving as the foundation of VLAs do not present issues for learning the task. However, building a VLA that can perform surgical operations introduces unique challenges.

The likely first step toward building an AI-based robotic surgeon would be training a language model on the same information that surgeons must understand before entering practice, such as a comprehensive medical curriculum from premedical training all the way through passing surgical board exams (Fig. 6). Such a model would have an understanding of both medical and surgical terms that would be important for understanding and performing surgery. Currently existing surgical language models have thus far focused on task-specific optimizations and lack a comprehensive understanding of broader concepts in biology, medicine, and even academic surgery (such as textbook question-answering) (*146*, *147*).

VLA models in robotics

VLAs represent a substantial shift from traditional task-specific robotics, integrating visual perception, language understanding, and action generation into a unified framework for more flexible systems (*131*–*135*). Typically based on large-scale transformers, VLAs are trained on diverse datasets of robot demonstrations with language instructions. Their key innovation is processing multimodal inputs—like images and text commands—to directly output low-level control actions in an end-to-end manner.

VLAs show impressive generalization to tasks not seen during training. With a large enough dataset, the model learns general correlations between language, vision, and action spaces, avoiding overfitting. For example, a robot can interpret an instruction like "find me a drink that will energize me" and retrieve an energy drink without explicitly training on that particular command (*133*).

VLAs in robotic surgery

Now, there do not exist any general-purpose VLAs in robotic surgery. However, the potential of VLA models in surgical robotics is particularly promising for a number of reasons (*60*). There are

already thousands of experts collecting teleoperated surgery data that could be used for imitation around the world every single day. In addition, surgical robots do not face the same limitations as mobile robots and readily support intensive onboard computers, which increases the potential capacity of VLA model size. By training on large datasets of surgical demonstrations paired with expert annotations, these models could potentially learn to interpret complex surgical scenarios and generate appropriate actions beyond human capability.

However, challenges remain in applying VLAs to surgical robotics. These include the need for large, high-quality datasets of annotated surgical procedures, ensuring safety and reliability in high-stakes medical environments, and addressing the interpretability of model decisions.

Measurable safety

One of the primary capabilities that would need to be embedded in a surgical robot is the ability to recognize its own limitations during a surgical procedure and appropriately divert control back to a human surgeon. This need arises particularly in scenarios that are not covered by the robot's training, necessitating intervention by a human. Unlike in conventional engineering, where potential problems are exhaustively tested by simulating every conceivable scenario, the unpredictable nature of surgical procedures makes such comprehensive preparation impractical. Consequently, surgical robots must be designed with the capability to handle previously unencountered situations. Robots, particularly in their early developmental stages, cannot be expected to adapt at a level comparable to that of human surgeons.

To mitigate these issues, one potential approach is to use the reinforcement-based technique called CQL (*110*, *148*), a method that trains the robot to steer clear of situations not present in their training data. CQL focuses on developing a conservative estimate of value functions to avoid overestimation, which is common in offline deep RL. This method has been successfully applied in some robot transformer architectures, which was shown in a demonstration where the robot asked for clarification with ambiguous instructions. An alternative method, conformal prediction (*149*, *150*), offers another solution by providing a range of probable outcomes for each decision, thus informing users of the associated uncertainty levels. This method splits incoming data into a training set and a calibration set, with the latter fine-tuning the confidence levels for predictions. Both CQL and conformal prediction, if integrated, could be crucial in enabling surgical robots to avoid risky scenarios and defer to surgeons when necessary.

Adversarial robustness is critical because of vulnerabilities in complex AI models, particularly in perception (*151*). Adversarial attacks, via subtle perturbations to sensor inputs, can cause misclassifications or faulty predictions by the AI model, leading to erroneous robotic actions (*152*, *153*). Mitigating these threats requires robust model training techniques, secure data pipelines, and run-time monitoring (*154*). The necessary hardware and software architectures must work in symbiosis to ensure both high-performance computation and verifiable safety. This involves integrating specialized hardware like AI accelerators and secure processing units with safety-certified software stacks, which may use real-time operating systems, hypervisors for a mixed-criticality approach, and securely containerized environments for deploying AI models (*155*). Formal verification methods are also important for demonstrating system integrity and reliability.

The role of regulation in AI-based surgery

Understanding and developing a framework for regulatory approval for AI in surgery will be essential to ensure the proper testing, evaluation, and compliance of an AI system to prioritize and maintain patient safety and efficacy in the operating room. The FDA has seen an exponential increase in the number of regulatory submissions related to AI medical devices in recent years, rising from only six approved devices in 2015 to more than 200 devices in 2023 alone (*156*). This trend has been driven by more connected devices, more commercial investment in technology, and a growing familiarity of how software can be regulated as a medical device. Today, the FDA has approved more than 950 AI-based products, but most of these products are related to signal and image processing, such as Siemens Healthineers' AI-Rad Companion (*157*), which provides qualitative and quantitative information of clinical images, and Digital Diagnostics' LumineticsCore (*158*), which analyzes images to detect diabetic retinopathy.

As AI becomes a decision-making platform in surgery, new regulatory oversight is needed. In January 2025, the FDA released draft guidance for AI-enabled medical devices, addressing transparency, bias, and product design (*159*). The FDA is adopting a total product life cycle approach, meaning that AI devices must meet regulatory standards throughout their entire life span, not just at initial approval. To manage evolving AI models, the guidance introduces predetermined change control plans, which allow companies to predefine software updates without seeking new clearance for each change, streamlining the regulatory process.

The guidance also mandates transparency, requiring clear explanations of how AI systems make decisions to build trust among clinicians and patients. Furthermore, the FDA emphasizes that models must be tested across diverse populations to avoid bias and ensure equitable outcomes. The adoption of these regulations will require input from the medical device community, and feedback from key opinion leaders is paramount to ensure that AI regulation is managed efficiently by both industry and government to guarantee safe and effective AI-driven systems for the future.

Barriers to entry, clinical deployment, and integration challenges

Limitations in today's regulatory process are a rate-limiting step in adopting AI surgical robots. New FDA guidance mandates a total life cycle policy, requiring perpetual oversight, validation, and postmarket surveillance. To successfully market AI devices, manufacturers must engage the FDA early, develop well-defined predetermined change control plans that outline software validation, and ensure robust postmarket surveillance protocols.

To limit market entry barriers beyond regulatory clearance, companies will likely introduce AI systems in a graduated fashion. For instance, Intuitive Surgical released AI software that operates in the background, analyzing system data to create objective insights for surgeons. Because the AI only analyzes data and leaves decision-making to the surgeon, it is easier to regulate and deploy. This trend is already unfolding with Moon Surgical's Maestro system, an AI-based laparoscope holder that received the first FDA clearance for an intraoperative AI application. Because the task does not interfere with a surgeon's workflow, it reduces barriers to clinical adoption. If successful, other companies will likely follow this approach, slowly building a new infrastructure where AI-based tasks in the operating room become common.

Beyond regulatory and workflow considerations, the clinical translation of autonomous surgical robots faces substantial institutional hurdles, including determining potential liability. Hospitals must invest in infrastructure upgrades, including robust data management systems, high-performance computing, and potentially redesigned operating rooms. Integrating these systems into established workflows requires careful planning; adaptation of team roles; and comprehensive training for surgeons, nurses, and technicians. Furthermore, system maintenance, software updates, specialized sterilization protocols, and overall cost-effectiveness will be critical factors influencing widespread adoption and accessibility.

Acquiring surgical datasets

Surgical video data are extensively available from various procedures, including cataracts (*160*, *161*), neurosurgery (*162*), and cholecystectomy (*163*–*165*). There are also data accessible from general manipulation skills like peg transferring with laparoscopic tools (*137*, *166*, *167*). In addition, public platforms like YouTube host numerous surgical videos, with one curated dataset containing 2000 open-surgery demonstrations (*168*). Despite all of these open data, acquiring data for large-scale surgery projects remains challenging, primarily because of patient privacy concerns and the typically small sample sizes in medical studies.

Recent collaborative efforts in general robotics have shown promise in overcoming these data limitations. A project involving 21 institutions assembled a dataset to train a large-scale RT model capable of controlling multiple robots across diverse tasks (*169*). This work demonstrates the potential for high-capacity models to improve with increased data, even beyond their expected applications. However, replicating such large-scale data collection in surgical settings poses challenges. Successful large-scale collaborations in medicine suggest that realizing a surgical VLA would require extensive data sharing and open collaboration among universities, industry, and hospitals.

CONCLUSION

AI and robotics are enabling surgical autonomy, with the potential to improve outcomes and expand access to care. Surgical robots with built-in autonomy present a promising solution to current challenges, offering the potential to improve efficiency, safety, and consistency in surgical procedures.

We divide current research focuses into two primary AI approaches: simulation-based learning using RL techniques and imitation-based learning that aims to replicate expert surgeon behaviors. Although both approaches show promise, they face considerable challenges. Simulation-based methods struggle with the sim-to-real gap, where models trained in simplified virtual environments may not translate effectively to real-world surgical scenarios. Imitation-based learning, although potentially more adaptable, requires extensive high-quality datasets of expert demonstrations and may inadvertently replicate suboptimal behaviors.

A path toward improving the autonomy of surgical robots could be through the development of large, multimodal VLAs specifically for surgery. These models integrate visual perception, natural language understanding, and action generation into a unified framework. Challenges remain in applying VLAs to surgical robotics, including the need for large, high-quality datasets; ensuring safety and reliability in high-stakes medical environments; and addressing the interpretability of model decisions. A primary capability for a surgical robot is the ability to recognize its own limitations during a procedure and appropriately divert control back to a human surgeon. As the field moves forward, it will be essential to maintain a balance between technological innovation and patient safety, ensuring that the development of autonomous surgical robots ultimately leads to improved patient outcomes and increased access to high-quality surgical care worldwide.

REFERENCES AND NOTES

1. R. L. Siegel, N. S. Wagle, A. Cercek, R. A. Smith, A. Jemal, Colorectal cancer statistics, 2023. *CA Cancer J. Clin.* **73**, 233–254 (2023).
2. H. J. Lujan, G. Plasencia, M. Jacobs, M. Viamonte, R. F. Hartmann, Long-term survival after laparoscopic colon resection for cancer. *Dis. Colon Rectum* **45**, 491–501 (2002).
3. A. Ehrampoosh, B. Shirinzadeh, J. Pinskier, J. Smith, R. Moshinsky, Y. Zhong, A force-feedback methodology for teleoperated suturing task in robotic-assisted minimally invasive surgery. *Sensors* **22**, 7829 (2022).
4. J. E. Knudsen, U. Ghaffar, R. Ma, A. J. Hung, Clinical applications of artificial intelligence in robotic surgery. *J. Robot. Surg.* **18**, 102 (2024).
5. G.-Z. Yang, Digital architecture and robotic construction. *Sci. Robot.* **2**, eaan3673 (2017).
6. P. E. Dupont, B. J. Nelson, M. Goldfarb, B. Hannaford, A. Menciassi, M. K. O'Malley, N. Simaan, P. Valdastri, G. Z. Yang, A decade retrospective of medical robotics research from 2010 to 2020. *Sci. Robot.* **6**, eabi8017 (2021).
7. J. J. Rasouli, J. Shao, S. Neifert, W. N. Gibbs, G. Habboub, M. P. Steinmetz, E. Benzel, T. E. Mroz, Artificial intelligence and robotics in spine surgery. *Global Spine J.* **11**, 556–564 (2021).
8. A. Moglia, K. Georgiou, E. Georgiou, R. M. Satava, A. Cuschieri, A systematic review on artificial intelligence in robot-assisted surgery. *Int. J. Surg.* **95**, 106151 (2021).
9. K. Denecke, C. R. Baudoin, A review of artificial intelligence and robotics in transformed health ecosystems. *Front. Med.* **9**, 795957 (2022).
10. T. Haidegger, Autonomy for surgical robots: Concepts and paradigms. *IEEE Trans. Med. Robot. Bionics* **1**, 65–76 (2019).
11. S. Panesar, Y. Cagle, D. Chander, J. Morey, J. Fernandez-Miranda, M. Kliot, Artificial intelligence and the future of surgical robotics. *Ann. Surg.* **270**, 223–226 (2019).
12. I. Andras, E. Mazzone, F. W. B. van Leeuwen, G. de Naeyer, M. N. van Oosterom, S. Beato, T. Buckle, S. O'Sullivan, P. J. van Leeuwen, A. Beulens, N. Crisan, F. D'Hondt, P. Schatteman, H. van der Poel, P. Dell'Oglio, A. Mottrie, Artificial intelligence and robotics: A combination that is changing the operating room. *World J. Urol.* **38**, 2359–2366 (2020).
13. A. Attanasio, B. Scaglioni, E. De Momi, P. Fiorini, P. Valdastri, Autonomy in surgical robotics. *Annu. Rev. Control Robot. Auton. Syst.* **4**, 651–679 (2021).
14. B. Kuipers, E. A. Feigenbaum, P. E. Hart, N. J. Nilsson, Shakey: From conception to history. *AI Mag.* **38**, 88–103 (2017).
15. R. D. Howe, Y. Matsuoka, Robotics for surgery. *Annu. Rev. Biomed. Eng.* **1**, 211–240 (1999).
16. A. Gawande, Two hundred years of surgery. *N. Engl. J. Med.* **366**, 1716–1723 (2012).
17. A. Buia, F. Stockhausen, E. Hanisch, Laparoscopic surgery: A qualified systematic review. *World J. Methodol.* **5**, 238–254 (2015).
18. Y. Zhou, H. Ren, M. Q.-H. Meng, Z. T. H. Tse, H. Yu, Robotics in natural orifice transluminal endoscopic surgery. *J. Mech. Med. Biol.* **13**, 1350044 (2013).
19. S. Atallah, B. Martin-Perez, D. Keller, J. Burke, L. Hunter, Natural-orifice transluminal endoscopic surgery. *J. Br. Surg.* **102**, e73 (2015).
20. D. Ramirez, M. J. Maurice, J. H. Kaouk, Robotic single-port surgery: Paving the way for the future. *Urology* **95**, 5–10 (2016).
21. R. J. Nelson, J. S. S. Chavali, N. Yerram, P. Babbar, J. H. Kaouk, Current status of robotic single-port surgery. *Urol. Ann.* **9**, 217–222 (2017).
22. C. P. Childers, M. Maggard-Gibbons, Estimation of the acquisition and operating costs for robotic surgery. *JAMA* **320**, 835–836 (2018).
23. Intuitive Surgical, Intuitive Annual Report 2024; http://investor.intuitivesurgical.com/static-files/500ff989-ad91-4b32-a59e-f94a34d75997.
24. M. Kawka, Y. Fong, T. M. H. Gall, Laparoscopic versus robotic abdominal and pelvic surgery: A systematic review of randomised controlled trials. *Surg. Endosc.* **37**, 6672–6681 (2023).
25. A. P. Ng, Y. Sanaiha, S. S. Bakhtiyar, S. Ebrahimian, C. Branche, P. Benharash, National analysis of cost disparities in robotic-assisted versus laparoscopic abdominal operations. *Surgery* **173**, 1340–1345 (2023).
26. I. Kassite, T. Bejan-Angoulvant, H. Lardy, A. Binet, A systematic review of the learning curve in robotic surgery: Range and heterogeneity. *Surg. Endosc.* **33**, 353–365 (2019).
27. S. K. Pasquali, D. Thibault, S. M. O'Brien, J. P. Jacobs, J. W. Gaynor, J. C. Romano, M. Gaies, K. D. Hill, M. L. Jacobs, D. M. Shahian, C. L. Backer, J. E. Mayer, National variation in congenital heart surgery outcomes. *Circulation* **142**, 1351–1360 (2020).
28. T. Xu, M. A. Makary, E. al Kazzi, M. Zhou, T. M. Pawlik, S. M. Hutfless, Surgeon-level variation in postoperative complications. *J. Gastrointest. Surg.* **20**, 1393–1399 (2016).

29. G. F. Sheldon, T. C. Ricketts, A. Charles, J. King, E. P. Fraher, A. Meyer, The global health workforce shortage: Role of surgeons and other providers. *Adv. Surg.* **42**, 63–85 (2008).
30. Y.-Z. Zhou, C.-Q. Wang, M.-H. Zhou, Z.-Y. Li, D. Chen, A.-L. Lian, Y. Ma, Surgical smoke: A hidden killer in the operating room. *Asian J. Surg.* **46**, 3447–3454 (2023).
31. W. Truscott, Impact of microscopic foreign debris on post-surgical complications. *Surg. Technol. Int.* **12**, 34–46 (2004).
32. T. G. Manning, M. Perera, D. Christidis, N. Kinnear, S. McGrath, R. O'Beirne, P. Zotov, D. Bolton, N. Lawrentschuk, Visual occlusion during minimally invasive surgery: A contemporary review of methods to reduce laparoscopic and robotic lens fogging and other sources of optical loss. *J. Endourol.* **31**, 327–333 (2017).
33. Y. Ou, A. Soleymani, X. Li, M. Tavakoli, Autonomous blood suction for robot-assisted surgery: A sim-to-real reinforcement learning approach. *IEEE Robot. Autom. Lett.* **9**, 7246–7253 (2024).
34. D. B. Kaber, M. R. Endsley, The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task. *Theor. Issues Ergon. Sci.* **5**, 113–153 (2004).
35. Z. Liao, X.-D. Duan, L. Xin, L.-M. Bo, X.-H. Wang, G.-H. Xiao, L.-H. Hu, S.-L. Zhuang, Z.-S. Li, Feasibility and safety of magnetic-controlled capsule endoscopy system in examination of human stomach: A pilot study in healthy volunteers. *J. Interv. Gastroenterol.* **2**, 155–160 (2012).
36. S. DiMaio, M. Hanuschik, U. Kreaden, "The da Vinci surgical system" in *Surgical Robotics: Systems Applications and Visions*, J. Rosen, B. Hannaford, R. M. Satava, Eds. (Springer, 2011), pp. 199–217.
37. J. C.-Y. Ngu, C. B.-S. Tsang, D. C.-S. Koh, The da Vinci Xi: A review of its capabilities, versatility, and potential role in robotic colorectal surgery. *Robot. Surg.* **4**, 77–85 (2017).
38. F. Celotto, N. Ramacciotti, A. Mangano, G. Danieli, F. Pinto, P. Lopez, A. Ducas, J. Cassiani, L. Morelli, G. Spolverato, F. M. Bianco, Da Vinci single-port robotic system current application and future perspective in general surgery: A scoping review. *Surg. Endosc.* **38**, 4814–4830 (2024).
39. N. E. Samalavicius, V. Janusonis, R. Siaulys, M. Jasėnas, O. Deduchovas, R. Venckus, V. Ezerskiene, R. Paskeviciute, G. Klimaviciute, Robotic surgery using Senhance® robotic platform: Single center experience with first 100 cases. *J. Robot. Surg.* **14**, 371–376 (2019).
40. G. Alshaikh, R. M. Schols, J. A. Wolfs, R. Cau, T. J. van Mulken, "A dedicated robotic system for open (super-)microsurgery" in *Robotics in Plastic and Reconstructive Surgery*, J. C. Selber, Ed. (Springer, 2021), pp. 139–153.
41. C. Song, X. Ma, X. Xia, P. W. Y. Chiu, C. C. N. Chong, Z. Li, A robotic flexible endoscope with shared autonomy: A study of mockup cholecystectomy. *Surg. Endosc.* **34**, 2730–2741 (2020).
42. J. Gilbert, The EndoAssist™ robotic camera holder as an aid to the introduction of laparoscopic colorectal surgery. *Ann. R. Coll. Surg. Engl.* **91**, 389–393 (2009).
43. T. Bensignor, G. Morel, D. Reversat, D. Fuks, B. Gayet, Evaluation of the effect of a laparoscopic robotized needle holder on ergonomics and skills. *Surg. Endosc.* **30**, 446–454 (2016).
44. Z. Xu, T. D. Khokhlova, C. S. Cho, V. A. Khokhlova, Histotripsy: A method for mechanical tissue ablation with ultrasound. *Annu. Rev. Biomed. Eng.* **26**, 141–167 (2024).
45. L. O. Mair, X. Liu, B. Dandamudi, K. Jain, S. Chowdhury, J. Weed, Y. Diaz-Mercado, I. N. Weinberg, A. Krieger, MagnetoSuture: Tetherless manipulation of suture needles. *IEEE Trans. Med. Robot. Bionics* **2**, 206–215 (2020).
46. S. Yang, R. A. MacLachlan, C. N. Riviere, Manipulator design and operation of a six-degree-of-freedom handheld tremor-canceling microsurgical instrument. *IEEE/ASME Trans. Mechatron.* **20**, 761–772 (2015).
47. P. J. Wijsman, L. Molenaar, F. J. Voskens, C. D. van't Hullenaar, I. A. Broeders, Image-based laparoscopic camera steering versus conventional steering: A comparison study. *J. Robot. Surg.* **16**, 1157–1163 (2022).
48. M. Roche, The MAKO robotic-arm knee arthroplasty system. *Arch. Orthop. Trauma Surg.* **141**, 2043–2047 (2021).
49. K. Nakao, B. Thavara, R. Tanaka, Y. Yamada, G. Joshi, K. Miyatani, T. Kawase, Y. Kato, Surgeon experience of the surgical safety with kinevo 900 in vascular neurosurgery: The initial experience. *Asian J. Neurosurg.* **15**, 464–467 (2020).
50. Y.-F. Cheng, H.-C. Chen, P.-C. Ke, W.-H. Hung, C.-Y. Cheng, C.-H. Lin, B.-Y. Wang, Image-guided video-assisted thoracoscopic surgery with Artis Pheno for pulmonary nodule resection. *J. Thorac. Dis.* **12**, 1342–1349 (2020).
51. J. Reisenauer, M. J. Simoff, M. A. Pritchett, D. E. Ost, A. Majid, C. Keyes, R. F. Casal, M. S. Parikh, J. Diaz-Mendoza, S. Fernandez-Bussy, E. E. Folch, Ion: Technology and techniques for shape-sensing robotic-assisted bronchoscopy. *Ann. Thorac. Surg.* **113**, 308–315 (2022).
52. A. Kuntz, M. Emerson, T. E. Ertop, I. Fried, M. Fu, J. Hoelscher, M. Rox, J. Akulian, E. A. Gillaspie, Y. Z. Lee, F. Maldonado, R. J. Webster III, R. Alterovitz, Autonomous medical needle steering in vivo. *Sci. Robot.* **8**, eadf7614 (2023).
53. A. Murali, S. Sen, B. Kehoe, A. Garg, S. M. Farland, S. Patil, "Learning by observation for surgical subtasks: Multilateral cutting of 3D viscoelastic and 2D orthotropic tissue phantoms" in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2015), pp. 1202–1209.
54. F. Richter, S. Shen, F. Liu, J. Huang, E. K. Funk, R. K. Orosco, M. C. Yip, Autonomous robotic suction to clear the surgical field for hemostasis using image-based blood flow detection. *IEEE Robot. Autom. Lett.* **6**, 1383–1390 (2021).
55. H. Saeidi, J. D. Opfermann, M. Kam, S. Wei, S. Leonard, M. H. Hsieh, J. U. Kang, A. Krieger, Autonomous robotic laparoscopic surgery for intestinal anastomosis. *Sci. Robot.* **7**, eabj2908 (2022).
56. R. H. Taylor, B. D. Mittelstadt, H. A. Paul, W. Hanson, P. Kazanzides, J. F. Zuhars, B. Williamson, B. L. Musits, E. Glassman, W. L. Bargar, An image-directed robotic system for precise orthopaedic surgery. *IEEE Trans. Robot. Autom.* **10**, 261–275 (1994).
57. W. Kilby, J. R. Dooley, G. Kuduvalli, S. Sayeh, C. R. Maurer Jr., The CyberKnife® Robotic Radiosurgery System in 2010. *Technol. Cancer Res. Treat.* **9**, 433–452 (2010).
58. R. J. Harris, J. B. Mygatt, S. I. Harris, Systems and methods for autonomous intravenous needle insertion, US Patent 9,364,171 (2016).
59. R. Montés-Micó, A. Cerviño, T. Ferrer-Blasco, VisuMax®femtosecond laser for corneal refractive surgery. *Expert Rev. Ophthalmol.* **3**, 385–388 (2008).
60. S. Schmidgall, J. W. Kim, A. Kuntz, A. E. Ghazi, A. Krieger, General-purpose foundation models for increased autonomy in robot-assisted surgery. arXiv:2401.00678 [cs.RO] (2024).
61. A. Lee, T. S. Baker, J. B. Bederson, B. I. Rapoport, Levels of autonomy in FDA-cleared surgical robots: A systematic review. *NPJ Digit. Med.* **7**, 103 (2024).
62. A. Shademan, R. S. Decker, J. D. Opfermann, S. Leonard, A. Krieger, P. C. W. Kim, Supervised autonomous robotic soft tissue surgery. *Sci. Transl. Med.* **8**, 337ra64 (2016).
63. G. Fagogenis, M. Mencattelli, Z. Machaidze, B. Rosa, K. Price, F. Wu, V. Weixler, M. Saeed, J. E. Mayer, P. E. Dupont, Autonomous robotic intracardiac catheter navigation using haptic vision. *Sci. Robot.* **4**, eaaw1977 (2019).
64. P. A. York, R. Peña, D. Kent, R. J. Wood, Microrobotic laser steering for minimally invasive surgery. *Sci. Robot.* **6**, eabd5476 (2021).
65. Y. Ma, Z. Song, Y. Zhuang, J. Hao, I. King, A survey on vision-language-action models for embodied AI. arXiv:2405.14093 [cs.RO] (2024).
66. R. S. Sutton, A. G. Barto, Reinforcement learning: An introduction. *Robotica* **17**, 229–235 (1999).
67. R. S. Sutton, A. G. Barto, *Reinforcement Learning: An Introduction* (MIT Press, ed. 2, 2018).
68. C. J. C. H. Watkins, P. Dayan, *Q*-learning. *Mach. Learn.* **8**, 279–292 (1992).
69. S. Schmidgall, A. Krieger, J. Eshraghian, Surgical gym: A high-performance GPU-based platform for reinforcement learning with surgical robots. arXiv:2310.04676 [cs.RO] (2023).
70. Y. Ou, S. Zargarzadeh, P. Sedighi, M. Tavakoli, A realistic surgical simulator for non-rigid and contact-rich manipulation in surgeries with the da Vinci research kit. arXiv:2404.05888 [cs.RO] (2024).
71. Q. Yu, M. Moghani, K. Dharmarajan, V. Schorp, William Chung-Ho Panitch, J. Liu, K. Hari, H. Huang, M. Mittal, K. Goldberg, A. Garg, Orbit-surgical: An open-simulation framework for learning surgical augmented dexterity. arXiv:2404.16027 [cs.RO] (2024).
72. P. Kazanzides, Z. Chen, A. Deguet, G. S. Fischer, R. H. Taylor, S. P. DiMaio, "An open-source research kit for the da Vinci® surgical system" in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2014), pp. 6434–6439.
73. E. Tagliabue, A. Pore, D. Dall'Alba, M. Piccinelli, P. Fiorini, "UnityFlexML: Training reinforcement learning agents in a simulated surgical environment" in *I-RIM 2020 Conference Proceedings* (I-RIM, 2020), pp. 23–24.
74. P. M. Scheikl, B. Gyenes, R. Younis, C. Haas, G. Neumann, M. Wagner, F. Mathis-Ullrich, LapGym—An open source framework for reinforcement learning in robot-assisted laparoscopic surgery. arXiv:2302.09606 [cs.RO] (2023).
75. "PhysX physics engine"; https://developer.nvidia.com/physx-sdk.
76. J. Xu, B. Li, B. Lu, Y.-H. Liu, Q. Dou, P.-A. Heng, "SurRoL: An open-source reinforcement learning centered and dVRK compatible platform for surgical robot learning" in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2021), pp. 1821–1828.
77. W. Zhao, J. P. Queralta, T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: A survey" in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)* (IEEE, 2020), pp. 737–744.
78. S. James, P. Wohlhart, M. Kalakrishnan, D. Kalashnikov, A. Irpan, J. Ibarz, S. Levine, R. Hadsell, K. Bousmalis, "Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks" in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2019), pp. 12619–12629.
79. J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world" in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2017), pp. 23–30.
80. P. M. Scheikl, E. Tagliabue, B. Gyenes, M. Wagner, D. Dall'Alba, P. Fiorini, F. Mathis-Ullrich, Sim-to-real transfer for visual reinforcement learning of deformable object manipulation for robot-assisted surgery. *IEEE Robot. Autom. Lett.* **8**, 560–567 (2023).

81. Q. Yu, M. Moghani, K. Dharmarajan, V. Schorp, W. Panitch, J. Liu, K. Hari, H. Huang, M. Mittal, K. Goldberg, A. Garg, "Orbit-Surgical: An open-simulation framework for learning surgical augmented dexterity" in *2024 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2024), pp. 15509–15516.
82. Y. Ou, M. Tavakoli, Sim-to-real surgical robot learning and autonomous planning for internal tissue points manipulation using reinforcement learning. *IEEE Robot. Autom. Lett.* **8**, 2502–2509 (2023).
83. F. Liu, Z. Li, Y. Han, J. Lu, F. Richter, M. C. Yip, "Real-to-sim registration of deformable soft tissue with position-based dynamics for surgical robot autonomy" in *2021 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2021), pp. 12328–12334.
84. J. Lu, F. Richter, M. C. Yip, Pose estimation for robot manipulators via keypoint optimization and sim-to-real transfer. *IEEE Robot. Autom. Lett.* **7**, 4622–4629 (2022).
85. M. Haiderbhai, R. Gondokaryono, T. Looi, J. M. Drake, L. A. Kahrs, "Robust sim2real transfer with the da Vinci research kit: A study on camera, lighting, and physics domain randomization" in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2022), pp. 3429–3435.
86. J. Kim, T. Zhao, S. Schmidgall, A. Deguet, M. Kobilarov, C. Finn, A. Krieger, "Surgical robot transformer (SRT): Imitation learning for surgical tasks" in *Proceedings of the 8th Conference on Robot Learning* (PMLR, 2024), pp. 130–134.
87. A. Hussein, M. M. Gaber, E. Elyan, C. Jayne, Imitation learning: A survey of learning methods. *ACM Comput. Surv.* **50**, 1–35 (2018).
88. D. A. Pomerleau, "ALVINN: An autonomous land vehicle in a neural network" in *Advances in Neural Information Processing Systems*, D. Touretzky, Ed. (Morgan-Kaufmann, 1988), pp. 305–313.
89. J. Li, Y. Jin, Y. Chen, H. Yip, M. Scheppach, P. Chiu, Y. Yam, H. Meng, Q. Dou, "Imitation learning from expert video data for dissection trajectory prediction in endoscopic surgical procedure" in *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Springer, 2023), pp. 494–504.
90. B. Li, R. Wei, J. Xu, B. Lu, C. Yee, C. Ng, P. Heng, Q. Dou, Y. Liu, "3D perception based imitation learning under limited demonstration for laparoscope control in robotic surgery" in *2022 International Conference on Robotics and Automation (ICRA)* (IEEE, 2022), pp. 7664–7670.
91. S. Schmidgall, J. W. Kim, A. Krieger, Robots learning to imitate surgeons—challenges and possibilities. *Nat. Rev. Urol.* **21**, 451–452 (2024).
92. T. Z. Zhao, J. Tompson, D. Driess, P. Florence, K. Ghasemipour, C. Finn, A. Wahid, ALOHA unleashed: A simple recipe for robot dexterity. arXiv:2410.13126 [cs.RO] (2024).
93. P. M. Scheikl, N. Schreiber, C. Haas, N. Freymuth, G. Neumann, R. Lioutikov, F. Mathis-Ullrich, Movement primitive diffusion: Learning gentle robotic manipulation of deformable objects. *IEEE Robot. Autom. Lett.* **9**, 5338–5345 (2024).
94. S. Ross, G. Gordon, D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning" in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, G. Gordon, D. Dunson, M. Dudík, Eds., vol. 15 of *Proceedings of Machine Learning Research* (PMLR, 2011), pp. 627–635.
95. M. Kelly, C. Sidrane, K. Driggs-Campbell, M. J. Kochenderfer, "HG-DAgger: Interactive imitation learning with human experts" in *2019 International Conference on Robotics and Automation (ICRA)* (IEEE, 2019), pp. 8077–8083.
96. K. Menda, K. Driggs-Campbell, M. J. Kochenderfer, "EnsembleDAgger: A Bayesian approach to safe imitation learning" in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2019), pp. 5041–5048.
97. A. Y. Ng, S. J. Russell, "Algorithms for inverse reinforcement learning" in *ICML '00: Proceedings of the Seventeenth International Conference on Machine Learning* (Morgan Kaufmann Publishers Inc., 2000), pp. 663–670.
98. T. Huang, K. Chen, B. Li, Y.-H. Liu, Q. Dou, "Demonstration-guided reinforcement learning with efficient exploration for task automation of surgical robot" in *2023 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2023), pp. 4640–4647.
99. Y. Long, W. Wei, T. Huang, Y. Wang, Q. Dou, Human-in-the-loop embodied intelligence with interactive simulation environment for surgical robot learning. *IEEE Robot. Autom. Lett.* **8**, 4441–4448 (2023).
100. A. K. Tanwani, P. Sermanet, A. Yan, R. Anand, M. Phielipp, K. Goldberg, "Motion2Vec: Semi-supervised representation learning from surgical videos" in *2020 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2020), pp. 2174–2181.
101. K. Kawaharazuka, K. Okada, M. Inaba, "Robotic constrained imitation learning for the peg transfer task in fundamentals of laparoscopic surgery" in *2024 IEEE International Conference on Robotics and Automation* (IEEE, 2024), pp. 606–612.
102. J. W. Kim, S. Schmidgall, A. Krieger, M. Kobilarov, "Learning a library of surgical manipulation skills for robotic surgery," presented at the workshop Bridging the Gap between Cognitive Science and Robot Learning in the Real World: Progresses and New Directions, Conference on Robot Learning, Atlanta, GA, 6 to 9 November 2023.
103. J. W. Kim, C. He, M. Urias, P. Gehlbach, G. D. Hager, I. Iordachita, M. Kobilarov, "Autonomously navigating a surgical tool inside the eye by learning from demonstration" in *2020 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2020), pp. 7351–7357.
104. J. W. Kim, P. Zhang, P. Gehlbach, I. Iordachita, M. Kobilarov, "Towards autonomous eye surgery by combining deep imitation learning with optimal control" in *Proceedings of the 2020 Conference on Robot Learning*, J. Kober, F. Ramos, C. Tomlin, Eds., vol. 155 of *Proceedings of Machine Learning Research* (PMLR, 2020), pp. 2347–2358.
105. J. Romero, D. Tzionas, M. J. Black, Embodied hands. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* **36**, 1–17 (2017).
106. K. Shaw, S. Bahl, D. Pathak, "VideoDex: Learning dexterity from internet videos" in *Proceedings of the 6th Conference on Robot Learning*, K. Liu, D. Kulic, J. Ichnowski, Eds., vol. 205 of *Proceedings of Machine Learning Research* (PMLR, 2023), pp. 654–665.
107. T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband, G. Dulac-Arnold, J. Agapiou, J. Leibo, A. Gruslys, "Deep Q-learning from demonstrations" in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence* (AAAI, 2018), pp. 3223–3230.
108. M. Vecerik, T. Hester, J. Scholz, F. Wang, O. Pietquin, A. Barto, J. Hwang, B. Piot, N. Heess, T. Rothörl, T. Lampe, M. Riedmiller, Leveraging demonstrations for deep reinforcement learning on robotics problems with sparse rewards. arXiv:1707.08817 [cs.AI] (2018).
109. A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, S. Levine, "Learning complex dexterous manipulation with deep reinforcement learning and demonstrations" in *Robotics: Science and Systems (RSS)* (RSS Foundation, 2018), 10.15607/RSS.2018.XIV.049.
110. A. Kumar, A. Zhou, G. Tucker, S. Levine, "Conservative Q-learning for offline reinforcement learning" in *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, H. Lin, Eds. (Curran Associates, 2020), pp. 1179–1191.
111. S. Fujimoto, S. Gu, "A minimalist approach to offline reinforcement learning" in *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, J. Wortman Vaughan, Eds. (Curran Associates, 2021), pp. 20132–20145.
112. S. Reddy, A. D. Dragan, S. Levine, "SQIL: Imitation learning via reinforcement learning with sparse rewards" in *International Conference on Learning Representations (ICLR)* (ICLR, 2020), pp. 1–14.
113. D. Garg, S. Chakraborty, C. Cundy, J. Song, S. Ermon, "IQ-learn: Inverse soft-Q learning for imitation" in *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, J. Wortman Vaughan, Eds. (Curran Associates, 2021), pp. 4028–4039 (2021).
114. B. Keller, M. Draelos, K. Zhou, R. Qian, A. N. Kuo, G. Konidaris, K. Hauser, J. A. Izatt, Optical coherence tomography-guided robotic ophthalmic microsurgery via reinforcement learning from demonstration. *IEEE Trans. Robot.* **36**, 1207–1218 (2020).
115. J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms. arxiv:1707.06347 [cs.LG] (2017).
116. Z. Cui, J. Cartucho, S. Giannarou, F. Rodriguez y Baena, Caveats on the first-generation da Vinci Research Kit: Latent technical constraints and essential calibrations [Survey]. *IEEE Robot. Autom. Mag.* **32**, 2–17 (2023).
117. O. Özgüner, T. Shkurti, S. Huang, R. Hao, R. C. Jackson, W. S. Newman, M. C. Çavuşoğlu, Camera-robot calibration for the Da Vinci robotic surgery system. *IEEE Trans. Autom. Sci. Eng.* **17**, 2154–2161 (2020).
118. K. Pachtrachai, M. Allan, V. Pawar, S. Hailes, D. Stoyanov, "Hand-eye calibration for robotic assisted minimally invasive surgery without a calibration object" in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2016), pp. 2485–2491.
119. Z. Wang, Z. Liu, Q. Ma, A. Cheng, Y. H. Liu, S. Kim, A. Deguet, A. Reiter, P. Kazanzides, R. H. Taylor, Vision-based calibration of dual RCM-based robot arms in human-robot collaborative minimally invasive surgery. *IEEE Robot. Autom. Lett.* **3**, 672–679 (2018).
120. Y. Li, F. Richter, J. Lu, E. K. Funk, R. K. Orosco, J. Zhu, M. C. Yip, SuPer: A surgical perception framework for endoscopic tissue manipulation with surgical robotics. *IEEE Robot. Autom. Lett.* **5**, 2294–2301 (2020).
121. A. E. Abdelaal, J. Fang, T. N. Reinhart, J. A. Mejia, T. Z. Zhao, J. Bohg, A. M. Okamura, Force-aware autonomous robotic surgery. arXiv:2501.11742 [cs.LG] (2025).
122. Y. Wang, J. Wang, Y. Li, T. Yang, C. Ren, The deep reinforcement learning-based VR training system with haptic guidance for catheterization skill transfer. *IEEE Sens. J.* **22**, 23356–23366 (2022).
123. P. Puangmali, K. Althoefer, L. D. Seneviratne, D. Murphy, P. Dasgupta, State-of-the-art in force and tactile sensing for minimally invasive surgery. *IEEE Sens. J.* **8**, 371–381 (2008).
124. H. Ding, Y. Zhang, H. Shu, Y. Long, C. Gao, T. Lu, R. Liang, L. Seenivasan, Q. Dou, M. Unberath, SegSTRONG-C: Segmenting surgical tools robustly on non-adversarial generated corruptions—An EndoVis'24 challenge. arXiv:2407.11906 [cs.CV] (2024).
125. V. Penza, E. de Momi, N. Enayati, T. Chupin, J. Ortiz, L. S. Mattos, EnViSoRS: Enhanced vision system for robotic surgery. A user-defined safety volume tracking to minimize the risk of intraoperative bleeding. *Front. Robot. AI* **4**, 15 (2017).

126. Y. Ou, M. Tavakoli, CRESSim–MPM: A material point method library for surgical soft body simulation with cutting and suturing. arXiv:2411.14622 [cs.RO] (2024).
127. W. Chang, Y. Li, Z. Zhu, Y. Yang, "LSD3K: A benchmark for smoke removal from laparoscopic surgery images" in *2024 3rd International Conference on Artificial Intelligence, Internet of Things and Cloud Computing Technology (AIoTC)* (IEEE, 2024), pp. 1–5.
128. D. Wang, J. Qi, B. Huang, E. Noble, D. Stoyanov, J. Gao, D. S. Elson, Polarization-based smoke removal method for surgical images. *Biomed. Opt. Express* **13**, 2364–2379 (2022).
129. T. Okamoto, T. Ohnishi, H. Kawahira, O. Dergachyava, P. Jannin, H. Haneishi, Real-time identification of blood regions for hemostasis support in laparoscopic surgery. *Signal Image Video Process.* **13**, 405–412 (2019).
130. W. Reiter, Co-occurrence balanced time series classification for the semi-supervised recognition of surgical smoke. *Int. J. Comput. Assist. Radiol. Surg.* **16**, 2021–2027 (2021).
131. S. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-Maron, M. Gimenez, Y. Sulsky, J. Kay, J. T. Springenberg, T. Eccles, J. Bruce, A. Razavi, A. Edwards, N. Heess, Y. Chen, R. Hadsell, O. Vinyals, M. Bordbar, N. de Freitas, A generalist agent. arXiv:2205.06175 [cs.AI] (2022).
132. A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. J. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K.-H. Lee, S. Levine, Y. Lu, U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. Ryoo, G. Salazar, P. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, B. Zitkovich, RT-1: Robotics transformer for real-world control at scale. arXiv:2212.06817 [cs.RO] (2022).
133. A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, Tsang-Wei Edward Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, B. Zitkovich, RT-2: Vision-language-action models transfer web knowledge to robotic control. arXiv:2307.15818 [cs.RO] (2023).
134. Open X-Embodiment Collaboration, Open X-Embodiment: Robotic learning datasets and RT-X models (2023); https://robotics-transformer-x.github.io.
135. Y. Hu, Q. Xie, V. Jain, J. Francis, J. Patrikar, N. Keetha, S. Kim, Y. Xie, T. Zhang, H. Fang, S. Zhao, S. Omidshafiei, D. Kim, A. Agha-Mohammadi, K. Sycara, M. Johnson-Roberson, D. Batra, X. Wang, S. Scherer, C. Wang, Z. Kira, F. Xia, Y. Bisk, Toward general-purpose robots via foundation models: A survey and meta-analysis. arXiv:2312.08782 [cs.RO] (2023).
136. K.-H. Oh, B. Leonardo, M. Alberto, V. Valentina, P. M. Di, T. Francesco, P. Gioia, A. Luciano, D. Alvaro, Z. Milos, C. Liaohai, P. C. Giulianotti, Comprehensive Robotic Cholecystectomy Dataset (CRCD): Integrating kinematics, pedal signals, and endoscopic videos. arXiv:2312.01183 [cs.RO] (2023).
137. Y. Gao, S. Vedula, C. Reiley, N. Ahmidi, B. Varadarajan, H. Lin, L. Tao, L. Zappella, B. Béjar, D. Yuh, C. Chen, R. Vidal, S. Khudanpur, G. Hager, JHU-ISI gesture and skill assessment working set (JIGSAWS): A surgical activity dataset for human motion modeling, paper presented at Medical Image Computing and Computer-Assisted Intervention - MICCAI 2014 workshop: Modeling and Monitoring of Computer-Assisted Interventions, Boston, MA, 14 September 2014.
138. M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, Q. Vuong, T. Kollar, B. Burchfiel, R. Tedrake, D. Sadigh, S. Levine, P. Liang, C. Finn, OpenVLA: An open-source vision-language-action model. arXiv:2406.09246 [cs.RO] (2024).
139. H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, T. Scialom, Llama 2: Open foundation and fine-tuned chat models. arXiv:2307.09288 [cs.CL] (2023).
140. L. Zheng, W. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing, H. Zhang, J. Gonzalez, I. Stoica, "Judging LLM-as-a-judge with MT-Bench and Chatbot Arena" in *Advances in Neural Information Processing Systems 36*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine, Eds. (Curran Associates, 2024), pp. 46595–46623.
141. D. Driess, F. Xia, M. S. M. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang, Y. Chebotar, P. Sermanet, D. Duckworth, S. Levine, V. Vanhoucke, K. Hausman, M. Toussaint, K. Greff, A. Zeng, I. Mordatch, P. Florence, PaLM-E: An embodied multimodal language model. arXiv:2303.03378 [cs.LG] (2023).
142. Y. Li, Y. Huang, M. E. Ildiz, A. S. Rawat, S. Oymak, "Mechanics of next token prediction with self-attention" in *Proceedings of the International Conference on Artificial Intelligence and Statistics* (PMLR, 2024), pp. 685–693.
143. S. Zhang, L. Dong, X. Li, S. Zhang, X. Sun, S. Wang, J. Li, R. Hu, T. Zhang, F. Wu, G. Wang, Instruction tuning for large language models: A survey. arXiv:2308.10792 [cs.CL] (2023).
144. B. Peng, C. Li, P. He, M. Galley, J. Gao, Instruction tuning with GPT-4. arXiv:2304.03277 [cs.CL] (2023).
145. H. Liu, C. Li, Q. Wu, Y. J. Lee, "Visual instruction tuning" in *Advances in Neural Information Processing Systems 36*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine, Eds. (Curran Associates, 2024), pp. 34892–34916.
146. T. Buckley, J. A. Diao, A. Rodman, A. K. Manrai, Multimodal foundation models exploit text to make medical image predictions. arXiv:2311.0559 [cs.CV] (2023).
147. A. J. Thirunavukarasu, D. S. J. Ting, K. Elangovan, L. Gutierrez, T. F. Tan, D. S. W. Ting, Large language models in medicine. *Nat. Med.* **29**, 1930–1940 (2023).
148. Y. Chebotar, Q. Vuong, K. Hausman, F. Xia, Y. Lu, A. Irpan, A. Kumar, T. Yu, A. Herzog, K. Pertsch, K. Gopalakrishnan, J. Ibarz, O. Nachum, S. A. Sontakke, G. Salazar, H. T. Tran, J. Peralta, C. Tan, D. Manjunath, J. Singh, B. Zitkovich, T. Jackson, K. Rao, C. Finn, S. Levine, "Q-Transformer: Scalable offline reinforcement learning via autoregressive Q-functions" in *Proceedings of the 7th Conference on Robot Learning*, J. Tan, M. Toussaint, K. Darvish, Eds., vol. 229 of *Proceedings of Machine Learning Research* (PMLR, 2023), pp. 3909–3928.
149. M. Fontana, G. Zeni, S. Vantini, Conformal prediction: A unified review of theory and new challenges. *Bernoulli* **29**, 1–23 (2023).
150. J. Sun, Y. Jiang, J. Qiu, P. Nobel, M. Kochenderfer, M. Schwager, "Conformal prediction for uncertainty-aware planning with diffusion dynamics model" in *Advances in Neural Information Processing Systems 36*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine, Eds. (Curran Associates, 2024), pp. 80324–80337.
151. S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, I. S. Kohane, Adversarial attacks on medical machine learning. *Science* **363**, 1287–1289 (2019).
152. X. Ma, Y. Niu, L. Gu, Y. Wang, Y. Zhao, J. Bailey, F. Lu, Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognit.* **110**, 107332 (2021).
153. K. R. Ludvigsen, S. Nagaraja, Dissecting liabilities in adversarial surgical robot failures: A national (Danish) and EU law perspective. *Comput. Law Secur. Rev.* **44**, 105656 (2022).
154. N. Ghaffari Laleh, D. Truhn, G. P. Veldhuizen, T. Han, M. van Treeck, R. D. Buelow, R. Langer, B. Dislich, P. Boor, V. Schulz, J. N. Kather, Adversarial attacks and adversarial robustness in computational pathology. *Nat. Commun.* **13**, 5711 (2022).
155. M. Sadi, B. M. S. B. Talukder, K. Mishty, M. T. Rahman, Attacking deep learning AI hardware with universal adversarial perturbation. *Information* **14**, 516 (2023).
156. V. Muralidharan, B. A. Adewale, C. J. Huang, M. T. Nta, P. O. Ademiju, P. Pathmarajah, M. K. Hang, O. Adesanya, R. O. Abdullateef, A. O. Babatunde, A. Ajibade, S. Onyeka, Z. R. Cai, R. Daneshjou, T. Olatunji, A scoping review of reporting gaps in FDA-approved AI medical devices. *NPJ Digit. Med.* **7**, 273 (2024).
157. B. Yacoub, A. Varga-Szemes, U. J. Schoepf, I. M. Kabakus, D. Baruah, J. R. Burt, G. J. Aquino, A. K. Sullivan, J. O'Doherty, P. Hoelzer, J. Sperl, T. Emrich, Impact of artificial intelligence assistance on chest CT interpretation times: A prospective randomized study. *Am. J. Roentgenol.* **219**, 743–751 (2022).
158. A. A. van der Heijden, M. D. Abramoff, F. Verbraak, M. V. van Hecke, A. Liem, G. Nijpels, Validation of automated screening for referable diabetic retinopathy with the IDx-DR device in the Hoorn Diabetes Care System. *Acta Ophthalmol.* **96**, 63–68 (2018).
159. FDA, "Artificial intelligence-enabled device software functions: Lifecycle management and marketing submission recommendations: Draft guidance for industry and Food and Drug Administration staff" (FDA-2024-D-4488, 2025); https://fda.gov/regulatory-information/search-fda-guidance-documents/artificial-intelligence-enabled-device-software-functions-lifecycle-management-and-marketing.
160. H. Al Hajj, M. Lamard, P.-H. Conze, B. Cochener, G. Quellec, Monitoring tool usage in surgery videos using boosted convolutional and recurrent neural networks. *Med. Image Anal.* **47**, 203–218 (2018).
161. K. Schoeffmann, M. Taschwer, S. Sarny, B. Münzer, M. Jürgen Primus, D. Putzgruber, "Cataract-101: Video dataset of 101 cataract surgeries" in *Proceedings of the 9th ACM Multimedia Systems Conference* (ACM, 2018), pp. 421–425.
162. D. Bouget, R. Benenson, M. Omran, L. Riffaud, B. Schiele, P. Jannin, Detecting surgical tools by modelling local appearance and global shape. *IEEE Trans. Med. Imaging* **34**, 2603–2617 (2015).
163. A. P. Twinanda, S. Shehata, D. Mutter, J. Marescaux, M. de Mathelin, N. Padoy, EndoNet: A deep architecture for recognition tasks on laparoscopic videos. *IEEE Trans. Med. Imaging* **36**, 86–97 (2017).
164. W.-Y. Hong, C.-L. Kao, Y.-H. Kuo, J.-R. Wang, W.-L. Chang and C.-S. Shih, CholecSeg8k: A semantic segmentation dataset for laparoscopic cholecystectomy based on Cholec80. arXiv:2012.12453 [cs.CV] (2020).
165. C. I. Nwoye, T. Yu, C. Gonzalez, B. Seeliger, P. Mascagni, D. Mutter, J. Marescaux, N. Padoy, Rendezvous: Attention mechanisms for the recognition of surgical action triplets in endoscopic videos. *Med. Image Anal.* **78**, 102433 (2022).
166. N. Madapana, M. Rahman, N. Sanchez-Tamayo, M. Balakuntala, G. Gonzalez, J. Bindu, V. Venkatesh, X. Zhang, J. Noguera, T. Low, R. Voyles, Y. Xue, J. Wachs, "DESK: A robotic activity dataset for dexterous surgical skills transfer to medical robots" in *2019 IEEE/RSJ*

International Conference on Intelligent Robots and Systems (IROS) (IEEE, 2019), pp. 6928–6934.
167. A. Huaulmé, K. Harada, Q.-M. Nguyen, B. Park, S. Hong, M.-K. Choi, M. Peven, Y. Li, Y. Long, Q. Dou, S. Kumar, S. Lalithkumar, R. Hongliang, H. Matsuzaki, Y. Ishikawa, Y. Harai, S. Kondo, M. Mitsuishi, P. Jannin, PEg TRAnsfer Workflow recognition challenge report: Does multi-modal data improve recognition? arXiv:2202.05821 [cs.LG] (2022).
168. E. D. Goodman, K. K. Patel, Y. Zhang, W. Locke, C. J. Kennedy, R. Mehrotra, S. Ren, M. Y. Guan, M. Downing, H. W. Chen, J. Z. Clark, G. A. Brat, S. Yeung, A real-time spatiotemporal AI model analyzes skill in open surgical videos. arXiv:2112.07219 [cs.CV] (2021).
169. Open X-Embodiment Collaboration, "Open X-embodiment: Robotic learning datasets and RT-X models," presented at the workshop Towards Generalist Robots: Learning Paradigms for Scalable Skill Acquisition at the 2023 Conference on Robot Learning, Atlanta, GA, 6 to 9 November 2023.

Acknowledgments
Funding: This material is based on work supported by the National Science Foundation under grants DGE 2139757 and NSF/FRR 2144348; NIH grant R56EB033807; and ARPA-H grants 75N91023C00048, AY1AX000023, and D24AC00415. **Author contributions:** S.S., J.O., J.W.K., and A.K. conceptualized this paper; S.S., J.O., and J.W.K. wrote the paper; S.S., J.O., J.W.K., and A.K. edited the paper; A.K. provided funding. **Competing interests:** A.K. is an inventor on a patent application related to autonomous surgery: "Automated Surgical and Interventional Procedures," US Patent application US-2014-0005684-A1.

Submitted 8 September 2024
Accepted 27 June 2025
Published 23 July 2025
10.1126/scirobotics.adt0187

EXOSKELETONS

Deep domain adaptation eliminates costly data required for task-agnostic wearable robotic control

Keaton L. Scherpereel[1,2]*†, Matthew C. Gombolay[2,3], Max K. Shepherd[4], Carlos A. Carrasquillo[1,2], Omer T. Inan[5], Aaron J. Young[1,2]

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

Data-driven methods have transformed our ability to assess and respond to human movement with wearable robots, promising real-world rehabilitation and augmentation benefits. However, the proliferation of data-driven methods, with the associated demand for increased personalization and performance, requires vast quantities of high-quality, device-specific data. Procuring these data is often intractable because of resource and personnel costs. We propose a framework that overcomes data scarcity by leveraging simulated sensors from biomechanical models to form a stepping-stone domain through which easily accessible data can be translated into data-limited domains. We developed and optimized a deep domain adaptation network that replaces costly, device-specific, labeled data with open-source datasets and unlabeled exoskeleton data. Using our network, we trained a hip and knee joint moment estimator with performance comparable to a best-case model trained with a complete, device-specific dataset [incurring only an 11 to 20%, 0.019 to 0.028 newton-meters per kilogram (Nm/kg) increase in error for a semisupervised model and 20 to 44%, 0.033 to 0.062 Nm/kg for an unsupervised model]. Our network significantly outperformed counterpart networks without domain adaptation (which incurred errors of 36 to 45% semisupervised and 50 to 60% unsupervised). Deploying our models in the real-time control loop of a hip/knee exoskeleton ($N = 8$) demonstrated estimator performance similar to offline results while augmenting user performance based on those estimated moments (9.5 to 14.6% metabolic cost reductions compared with no exoskeleton). Our framework enables researchers to train real-time deployable deep learning, task-agnostic models with limited or no access to labeled, device-specific data.

INTRODUCTION

State-of-the-art exoskeleton control architectures increasingly use data-driven techniques to provide more effective assistance during human movement compared with their heuristic counterparts. In these approaches, data-driven models are trained to map data from exoskeleton-mounted sensors to human states [e.g., speed (*1*, *2*), gait phase (*3*–*5*), locomotion mode (*6*, *7*), kinematics (*8*, *9*), and joint moments (*10*–*12*)], environmental conditions [e.g., ground slope (*2*, *13*) or terrain (*14*, *15*)], or human outcomes [e.g., metabolic cost (*16*, *17*) or user preference (*18*)]. When sufficient labeled data (data paired with the correct output values) are available, many of these data-driven models, such as deep learning–based controllers, are capable of user independence (i.e., do not require participant-specific data) and can apply effective assistance across a wide range of tasks without requiring task classification (task agnostic) (*11*, *12*). These advances are pushing exoskeletons closer to handling the human and environmental uncertainty requisite for real-world viability.

Despite these successes, a critical barrier to the development and deployment of these data-driven controllers remains: A substantial amount of labeled training data is required, and these labels are costly to obtain for each device. Researchers must record data from their exoskeleton sensors along with time-synced data from a ground truth source, often necessitating optical motion capture systems and force plates. For instance, these motion and force data are critical to accurately calculate ground truth biological joint moments, which have recently emerged as a powerful human state variable in exoskeleton control (*11*, *12*). This highly specialized equipment is not readily available for many roboticists, and the process of calculating joint moments (or other human state labels) is nontrivial, requiring hundreds of hours of skilled labor for curation of large datasets. The training data are also device specific, prohibiting the reusage of data from previous generations or prototypes; every time researchers add or move sensors or substantially change the control type or magnitude, new labeled data must be collected. Furthermore, the exoskeleton must be actuated during training data collection because of the nonnegligible effect of assistance on users' biomechanics and sensor shifts due to soft tissue deformation and interface compliance. To accomplish this, researchers often must develop hand-tuned, task-dependent "stand-in" controllers that approximate the intended data-driven controller (*11*, *12*). Thus, progress in exoskeleton control research is substantially hindered by its dependence on device-specific, labeled data, which is only compounded by the short relevance time frame and numerous prohibitive barriers to obtaining that data.

Transfer learning is a set of techniques that have emerged to address problems like this by leveraging large, related datasets to more effectively learn to model smaller-scale, labeled data. For lower-limb devices, transfer learning methods have been applied to enable interparticipant and interactivity transfer of deep learning models. However, these techniques are limited in that some participant- or activity-specific data must exist and the specific sensors must remain the same (*19*–*21*). These studies have also been restricted to off-device verification. Researchers have also sought to create larger-scale, labeled, synthetic datasets for training deep learning networks, but this work has focused on dataset augmentation without

[1]George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA. [2]Institute for Robotics and Intelligent Machines, Georgia Institute of Technology, Atlanta, GA 30332, USA. [3]School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA 30332, USA. [4]College of Engineering, Bouvé College of Health Sciences, and Institute for Experiential Robotics, Northeastern University, Boston, MA 02115, USA. [5]School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA.
*Corresponding author. Email: keaton@gatech.edu
†Present address: Skip Innovations, San Francisco, CA, USA.

directly addressing the simulation-to-real gap present when deploying models trained with synthetic data in the real world (*22*, *23*). Thus, the feasibility of integrating large open-source datasets directly into real-time deployable algorithms for exoskeleton control remains an unanswered question.

Domain adaptation, a type of transfer learning, leverages labeled data from a related source domain to improve model performance in a target domain in which few or no labels exist. Domain adaptation techniques strive to uncover common latent features between the target and source distributions, enabling a mapping from unlabeled examples in the target domain to example-label pairs from the source domain. Relevant work from this area has shown that domain adaptation techniques can handle variability in wearable sensor location (*24*–*27*), sensor type (*28*), and user usage (*26*). However, these studies seldom go beyond simple classification tasks toward the important task of translating raw time series data (*26*, *29*), and we are unaware of any prior work that has leveraged domain adaptation in wearable robotics.

To enable scalable deep learning for exoskeletons, we created a domain adaptation framework that overcomes the need for difficult and expensive data collection by leveraging the following two simple-to-obtain data sources: open-source biomechanics datasets, which contain ground truth biological joint moment labels but no exoskeleton (termed the source domain), and data collected from the exoskeleton while participants perform a variety of tasks but without these moment labels (unlabeled data in the target domain). For example, when training end-to-end exoskeleton controllers, such as controllers built off an instantaneous estimate of biological joint moment (*11*, *12*), the most valuable data are also the most expensive to collect: exoskeleton sensor data with time-synced joint moment labels (i.e., labeled data in the target domain). By creating this domain adaptation framework, we can leverage unlabeled exoskeleton data, easily collected outside of the laboratory, to reduce or even eliminate the need for expensive data collection. This would greatly increase the ability to deploy exoskeletons at scale.

The key to our approach is a domain adaptation framework that leverages generative adversarial networks (GANs) formed into a CycleGAN (cycle-consistent generative adversarial network) and built on U-Net backbones to bridge the source and target domains. This bridge is formed by using a stepping-stone domain that simulates sensor data for any sensor location on the basis of samples in open-source biomechanics datasets and thus serves as a common, shared domain across devices and datasets [OpenSim (*30*, *31*)]. This sensor domain is readily developed on the basis of rigid body dynamics and serves as an intermediate representation that allows for the domain adaptation from real, open-source, biomechanics datasets to our target domain. Our framework translates human movement data between this simulated sensor source domain and the real sensor target domain (Fig. 1), enabling the integration of preexisting, large open-source biomechanics datasets in end-to-end exoskeleton controllers. This translation can be performed without any labeled data in the target domain (i.e., unsupervised), or we can supplement the GAN framework with a small-scale, labeled dataset in the target domain to ground the learned representation (i.e., semisupervised). Thus, we can achieve the best of both worlds: accurate predictions of biological moments while wearing an exoskeleton without the need to collect a large-scale dataset across all users and tasks.

In this study, we measured the usefulness of our translation by using the translated data to train downstream deep learning models in the real domain, which can be validated by deploying those models in real time on an exoskeleton. We compared end-to-end joint moment estimation models trained with a minimal number of labeled samples (semisupervised) and no labeled samples (unsupervised) to their respective equivalent baseline models without domain adaptation. These baselines are the best model that could be created from the limited data available using traditional supervised learning techniques and thus represent the current state-of-the-art approach. Comparing with these baselines represents the benefit that our domain adaption approach is providing in comparison with the current standard approach. This work represents three primary contributions: First, we proposed the use of human biomechanical models with simulated sensors as a universal domain for aggregating data with and without a device. Second, we developed a network that translates between this universal source domain and the device-specific, real-sensor, target domain. Last, we validated our translation strategy by deploying real-time models trained from translated

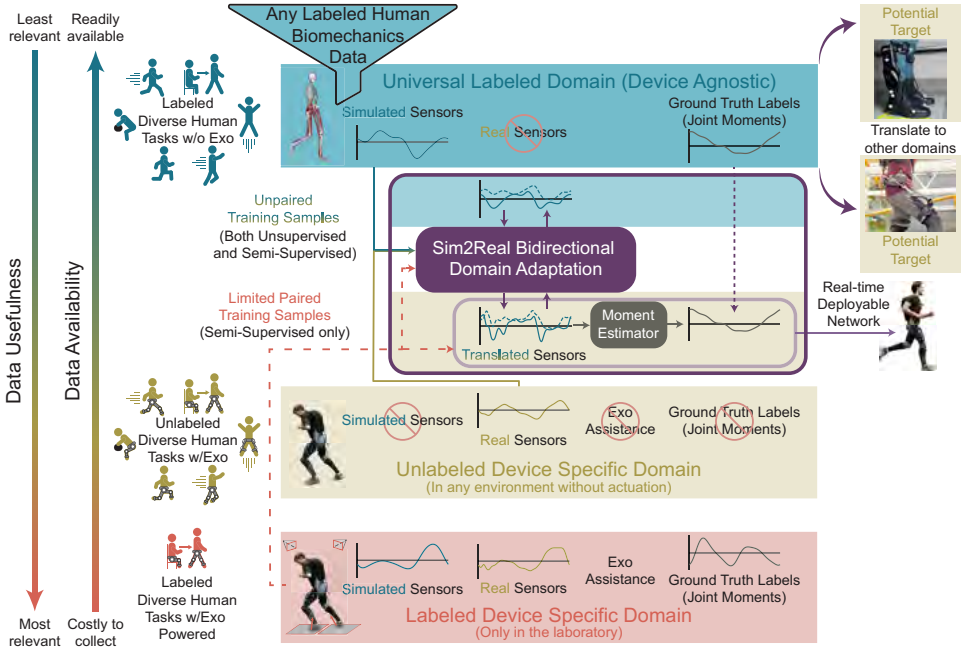


Fig. 1. Our strategy for replacing costly device-specific data with less costly data. Our approach uses the biomechanical modeling domain with simulated sensors as a universal domain for aggregating data. To use these data, we propose a network that performs bidirectional domain adaptation to translate simulated sensors into any specific device domain based on unlabeled data from that device. These translated data can be used to train downstream deep learning models (for our case, a moment estimator) that are deployable in real time on a device. This strategy can be used to create models for new devices and new joints. A fully unlabeled framework (unsupervised) is possible, but the optional labeled device-specific data can be useful (semisupervised).

data on an exoskeleton. These contributions allow our data-limited models to statistically outperform their counterpart models without domain adaptation (reducing error by ~15 to 30%) while also achieving accuracies similar (within ~10 to 30%) to those of models trained with large, costly datasets (best-case models). Accuracies like these allow exoskeleton controllers based on our model estimates to augment human performance, reducing metabolic expenditure by up to 10 to 15% on the basis of the two tasks captured. These are vital and critical steps toward enabling universal access to data-driven approaches in wearable robotics.

RESULTS

Optimization results

To train our semisupervised model, we needed to select a minimal subset of data for the supervised portion. Given that not all datasets are equally valuable, we sought to optimize the number of tasks (e.g., running, jumping, etc.) included in the training set by sequentially adding labeled tasks on the basis of the average generalization score: a score measuring how well each additional task improved moment estimation on the other tasks (Fig. 2A). Figure 2B shows the corresponding root mean square error (RMSE) across all tasks averaged between hip and knee for each task addition. Even after additional tasks no longer improve generalization, overall RMSE can continue to decrease because of improvement on that particular added task. We then compared the performance of our approach with that of a baseline model with the same number of tasks but without domain adaptation. Performance is presented as a percentage error increase compared with a best-case model (i.e., a model with access to all labeled tasks) for both the hip and knee (Fig. 2, C and D). Using the optimized set of tasks for both our approach and the baseline, our approach statistically outperformed the baseline approach regardless of the number of tasks included.

Although additional tasks continue to improve estimation, the goal of this optimization was to select a small, labeled subset of data to use from the target domain. Thus, we chose to limit our semisupervised approach to four tasks based on the plateau in the generalization score (Fig. 2A). The resulting percentage error increase from the best-case model at four tasks was 12.0 ± 3.8% at the hip and 13.6 ± 6.9% at the knee. For all future semisupervised analyses, the baseline and our approach included labeled exoskeleton data from only four optimized tasks. For our approach, these tasks were calisthenics, jump in place, level ground walk, and twister, whereas the baseline tasks were level ground walk, standing poses, calisthenics, and push-and-pull recovery (*12*).

Collecting many tasks is time and labor intensive, as is collecting additional participants; thus, we also sought to limit the number of participants included in our small semisupervised dataset. By folding across left-out participants, we assessed the average performance benefit of including additional participants. The performances of the baseline approach versus our approach for the hip and knee are presented in Fig. 2 (E and F), in terms of percentage increase compared with the same best-case model as above (all participants and tasks included). Only the four optimized tasks from above were included for each participant. Regardless of the number of participants included, our approach statistically outperformed the baseline approach. Given that most of the performance benefits were accrued during the first several participant additions, we limited the number of labeled participants who were included for the semisupervised models. We did this by averaging the hip and knee RMSE for our approach and determining the number of participants needed to reach 5% of the RMSE with all 13 participants. This occurred at four participants with a percentage increase in RMSE compared with the best case of 13.7 ± 5.2% at the hip and 19.7 ± 9.7% at the knee. Thus, on the basis of these optimizations, the following semisupervised models (both our approach and the baseline) included labeled data from only four participants performing four tasks.

Performance on the offline testing set

With the small, labeled dataset chosen for the semisupervised case, we tested the following approaches on a set of eight novel, held-out participants performing 28 tasks: the semisupervised and unsupervised baseline (i.e., without domain adaptation), our semisupervised and unsupervised approach, and the best-case model (i.e., using costly device-specific data from all participants and all tasks). For the semisupervised case, our approach significantly outperformed the baseline approach at the hip (percentage increase in RMSE compared with the best case of 11.3% versus the baseline of 35.5%, $P < 0.01$) and at the knee (percentage increase in RMSE compared with the best case of 20.3% versus the baseline of 45.3%, $P < 0.01$) as seen in Fig. 3 (A and B). For the unsupervised case, our approach significantly outperformed the baseline approach at the hip (percentage increase in RMSE compared with the best case of 19.9% versus the baseline of 49.5%, $P < 0.01$) and at the knee (percentage increase in RMSE compared with the best case of 44.4% versus the baseline of 60.3%, $P < 0.01$) as seen in Fig. 3 (C and D). Figure 3 (E and F) presents a comparison between the baseline and our approach for both the semisupervised and unsupervised cases in terms of the square of the Pearson correlation coefficient [R^2; on the best fit line between the ground truth and the estimate (*11*, *12*)]. Each point on the scatter plot represents one of the 28 tasks where a higher R^2 at the hip and the knee (closer to the top right-hand corner) indicates a better shape match between the estimate and the ground truth joint moment.

To further validate that our translation approach works across different types of estimators, we used our bidirectional translation approach to train a ground reaction force (GRF) estimator model (for both the supervised and unsupervised cases) and an activity classifier (for the unsupervised case). Both models demonstrated improved performance with our approach as compared with the baseline (figs. S4 to S6).

Real-time model performance

We deployed the joint moment estimation models trained with both our approach and the baseline in real time on an autonomous hip/knee exoskeleton with the models providing estimated joint moments to assist users (Fig. 4, A and B). The best-case model estimate was computed post hoc on the collected data for a fair comparison and performed similarly to the current state-of-the-art estimators (fig. S7) (*12*). Across eight novel users performing eight representative tasks, the model trained with our semisupervised approach significantly outperformed the baseline approach at the hip (percentage increase in RMSE compared with the best-case model of 11.0 ± 6.4% versus the baseline of 37.5 ± 10.5%, $P < 0.01$) and at the knee (percentage increase in RMSE compared with the best-case model of 19.5 ± 9.6% versus the baseline of 48.3 ± 19.0%, $P = 0.011$) (Fig. 4, C and D). The final performance of our semisupervised approach was 0.19 ± 0.01 Nm/kg RMSE and 0.71 ± 0.04 R^2 at the hip and 0.16 ± 0.02 Nm/kg RMSE and 0.75 ± 0.04 R^2 at the knee. For the

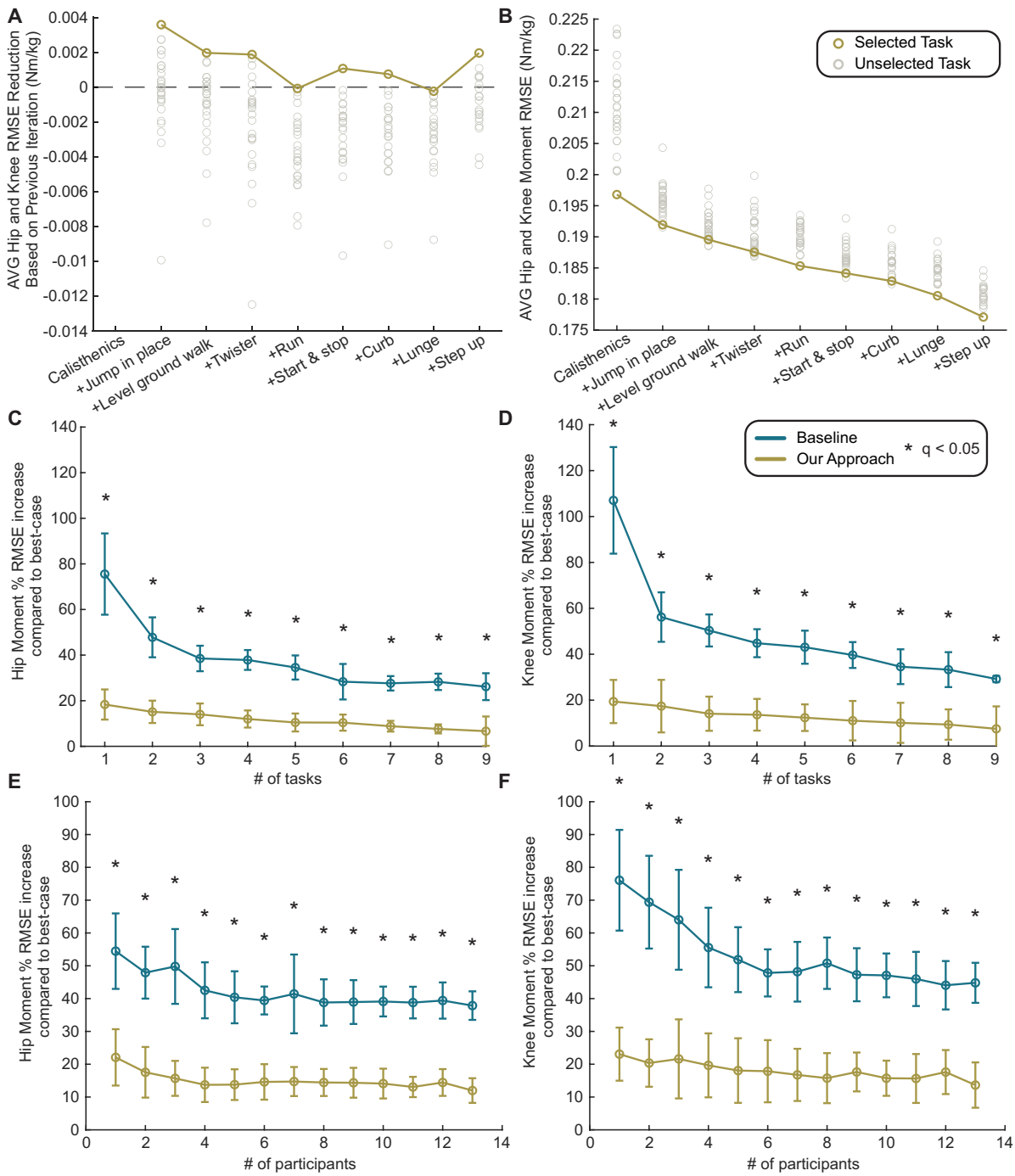


Fig. 2. Task optimization and performance across limited task and participant training sets. After seeding the task optimization with the task that provided the lowest RMSE across all tasks, tasks were sequentially added by maximizing the generalization score to other tasks (**A**). The corresponding RMSE for unselected and selected tasks is shown in (**B**). Optimization was terminated after nine tasks when our approach was not statistically different from the best-case model. Using this optimized task list, we compared the performance of our approach at the hip (**C**) and knee (**D**) with a separately optimized baseline without domain adaptation (but the same limited number of tasks). By cross-folding across participants with four optimized tasks, we assessed the average RMSE for our approach and the baseline for the hip (**E**) and knee (**F**) when sequentially increasing the number of labeled participants used in training. Our approach and the baseline were compared with a best-case model trained with labeled data from all tasks and participants. Error bars represent SD across five cross-folded, left-out participants [omitted from (A) and (B) for clarity]. Statistical significance was determined by controlling the false rate of discovery ($q < 0.05$) across all nine tasks [(C) and (D)] and all 13 participants [(E) and (F)].

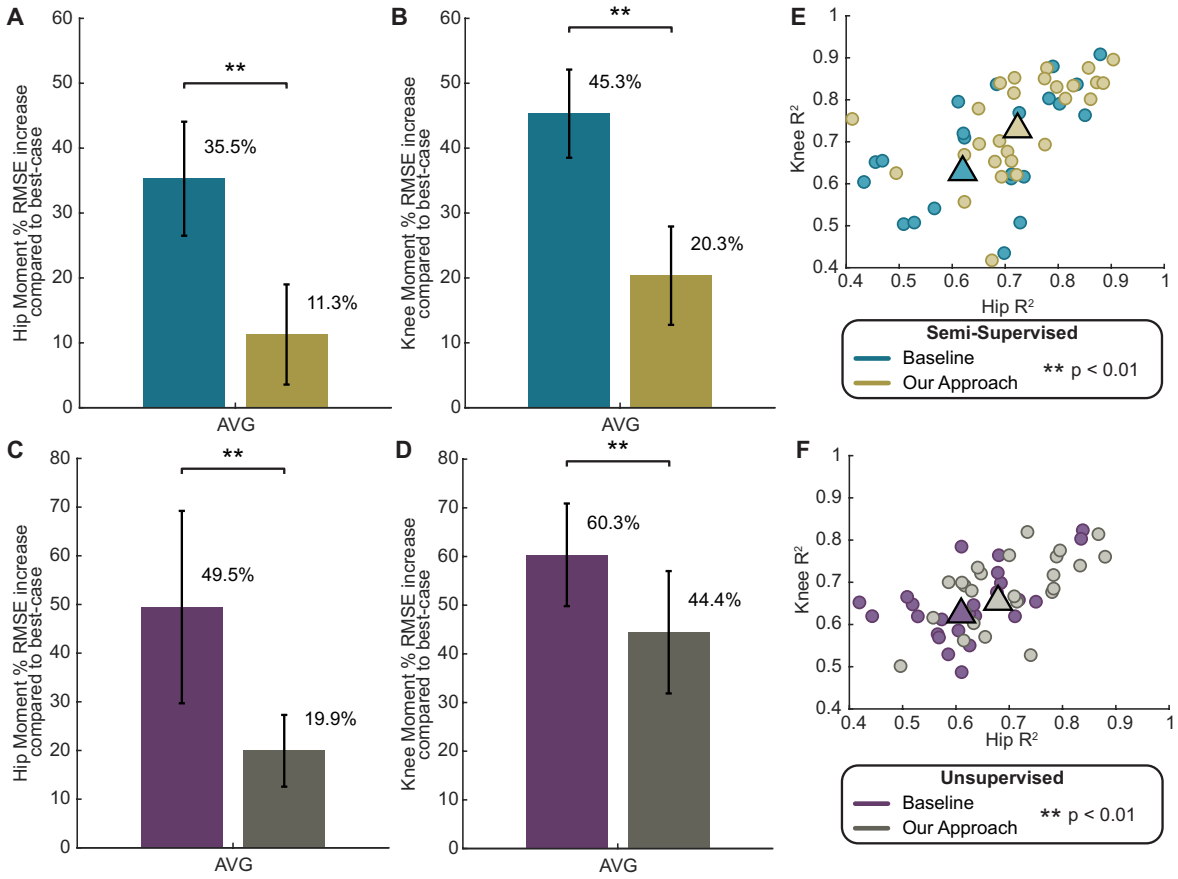


Fig. 3. Offline model performance for moment estimators trained with translated data. For the semisupervised case, the performance of our approach versus the baseline is presented for the hip (**A**) and knee (**B**) in terms of percentage increase in RMSE compared with the best-case model. Parallel results are presented for the unsupervised case at the hip (**C**) and knee (**D**). Averages were first taken across all 28 activities and then across participants. Error bars represent the SD across eight participants, and asterisks represent statistical significance determined by paired *t* tests. The R^2 values between ground truth joint moment and estimate for each of the 28 tasks at the hip and knee are presented for the semisupervised case (**E**) and unsupervised case (**F**). Each point represents the average across eight participants, but error bars were omitted for visual clarity; each triangle represents the average across all tasks.

unsupervised case, our approach significantly outperformed the baseline approach at the hip (percentage increase in RMSE compared with the best-case model of 25.6 ± 13.9% versus the baseline of 63.4 ± 24.8%, $P < 0.01$) and at the knee (percentage increase in RMSE compared with the best-case model of 32.9 ± 6.7% versus the baseline of 54.6 ± 8.1%, $P < 0.01$) (Fig. 4, E and F). The unsupervised baseline approach was computed post hoc because of the poor estimation of the baseline controller causing instability (fig. S3 and movie S1) and deemed unfeasible for real-time performance. The final performance of our unsupervised approach was 0.21 ± 0.01 Nm/kg RMSE and 0.68 ± 0.04 R^2 at the hip and 0.18 ± 0.01 Nm/kg RMSE and 0.71 ± 0.03 R^2 at the knee. The data available to each model and the strategy for using it are depicted in Fig. 4 (G to I).

To understand the effect of these errors on human performance, we deployed the best-case model, our semisupervised model, and our unsupervised model ($N = 8$) during a weightlifting activity and incline walking and compared metabolic cost relative to no exoskeleton (Fig. 5, A and B). There were significant differences between controller types for lifting weight [$F = 8.54$, degrees of freedom (df) = 3, $P < 0.01$] and 5° incline walking ($F = 11.05$, df = 3, $P < 0.01$). Our semisupervised model resulted in statistically significant reductions in metabolic cost compared with no exoskeleton for both lifting a weight and 5° inclines (12.5% for lifting and 14.6% for inclines, both $P < 0.01$). These reductions were similar to the best-case model that also had statistically significant reductions for both tasks (14.4% for lifting and 13.6% for inclines, both $P < 0.01$). In addition, our unsupervised model also reduced metabolic cost compared with no exoskeleton, although it was not statistically significant for eight participants (9.5% for lifting, $P = 0.066$, and 13.8% for inclines, $P = 0.051$). Similar results can be seen between the best-case and our domain adaptation models for tasks that did not require high net-positive work (fig. S8).

To further understand the accuracy of the three models that we deployed in real time, we broke down the results into each specific task. These results are presented in terms of an R^2 shape match between the ground truth joint moment and the estimate (Fig. 6A). The tasks are separated into cyclic, impedance-like, and unstructured activities to demonstrate the performances across all categories (*12*). A one-way analysis of variance (ANOVA) test found statistical differences between the three real-time models for R^2 at the hip ($F = 27.63$, df = 2, $P < 0.01$) and at the knee ($F = 14.47$, df = 2, $P < 0.01$). Follow-up pairwise multiple comparison tests with Bonferroni correction

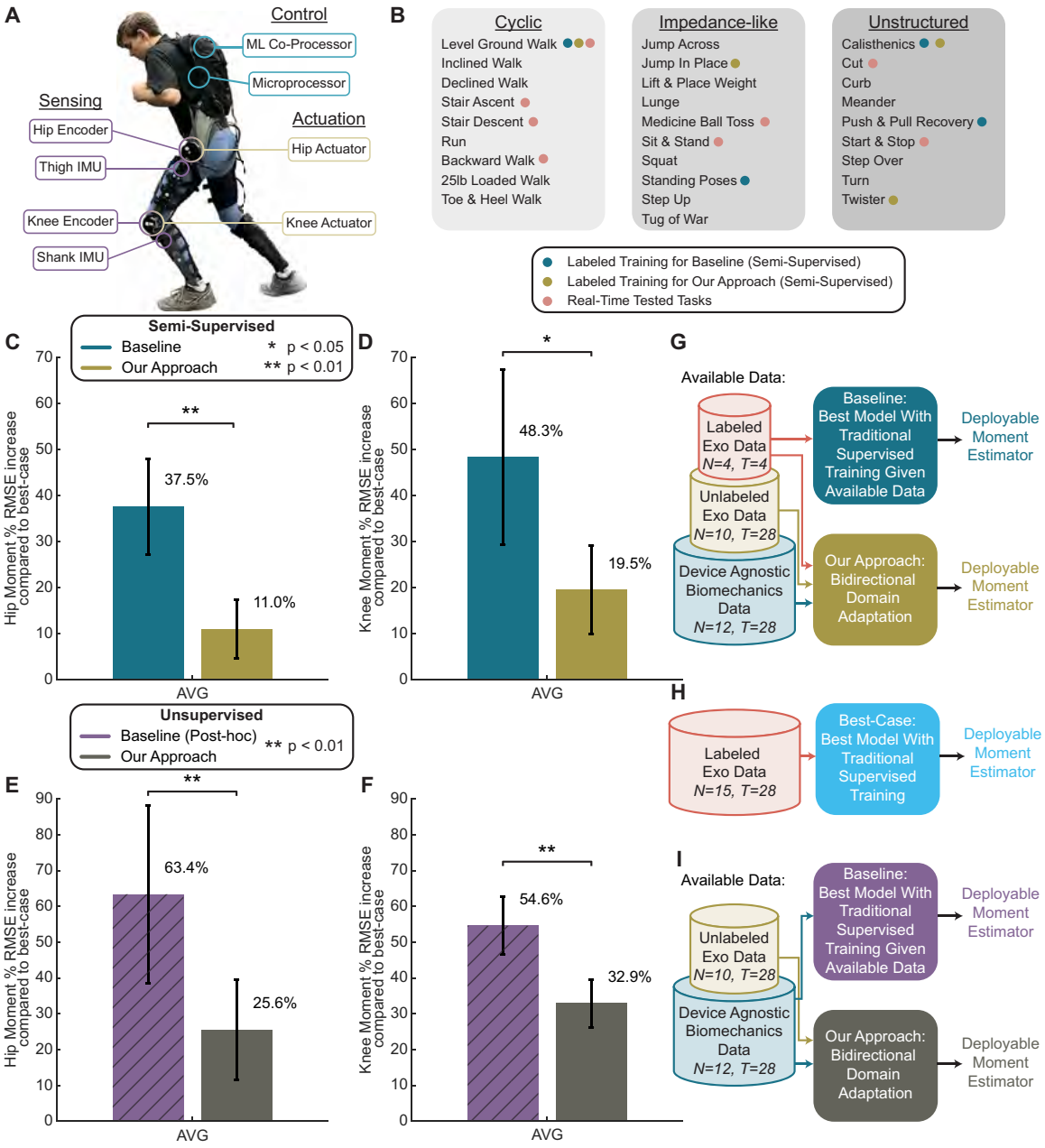


Fig. 4. Real-time model performance for moment estimators trained on translated data. Our autonomous hip/knee exoskeleton (**A**) sends measurements from hip and knee encoders and thigh and shank IMUs through a microprocessor to an onboard machine learning (ML) coprocessor for real-time inference. Inference results converted to actuator commands provide bilateral hip and knee assistance. (**B**) Models with labeled exoskeleton data (semisupervised) included four tasks. Seven real-time tasks were novel to the unsupervised and semisupervised models; level ground walking was in the semisupervised training sets. Unlabeled exoskeleton data and biomechanics datasets included all tasks. The performances of our deployed models versus their baselines are presented for the semisupervised case at the hip (**C**) and knee (**D**) and the unsupervised case at the hip (**E**) and knee (**F**) as a percentage RMSE increase compared with a best-case model (computed post hoc). The unsupervised baseline model was run offline post hoc because of real-time controller instability. Averages were taken across all eight tested tasks and then across eight participants, with error bars representing across-participant SD. Asterisks represent statistical significance determined by paired *t* tests. Pictorial representations of the data for the semisupervised models (**G**), best-case model (**H**), and unsupervised models (**I**) are also presented.

(Fig. 6C) showed a statistically significant increase in R^2 for our semisupervised approach over the semisupervised baseline ($P < 0.01$), our unsupervised approach over the semisupervised baseline ($P = 0.013$), and our semisupervised approach over our unsupervised approach ($P = 0.035$). At the knee, statistically significant improvements in R^2 were found for our semisupervised approach over the semisupervised baseline ($P = 0.010$) and our semisupervised approach over our unsupervised approach ($P = 0.011$).

Figure 7 demonstrates the accurate tracking of joint moments for the semisupervised and unsupervised models through representative

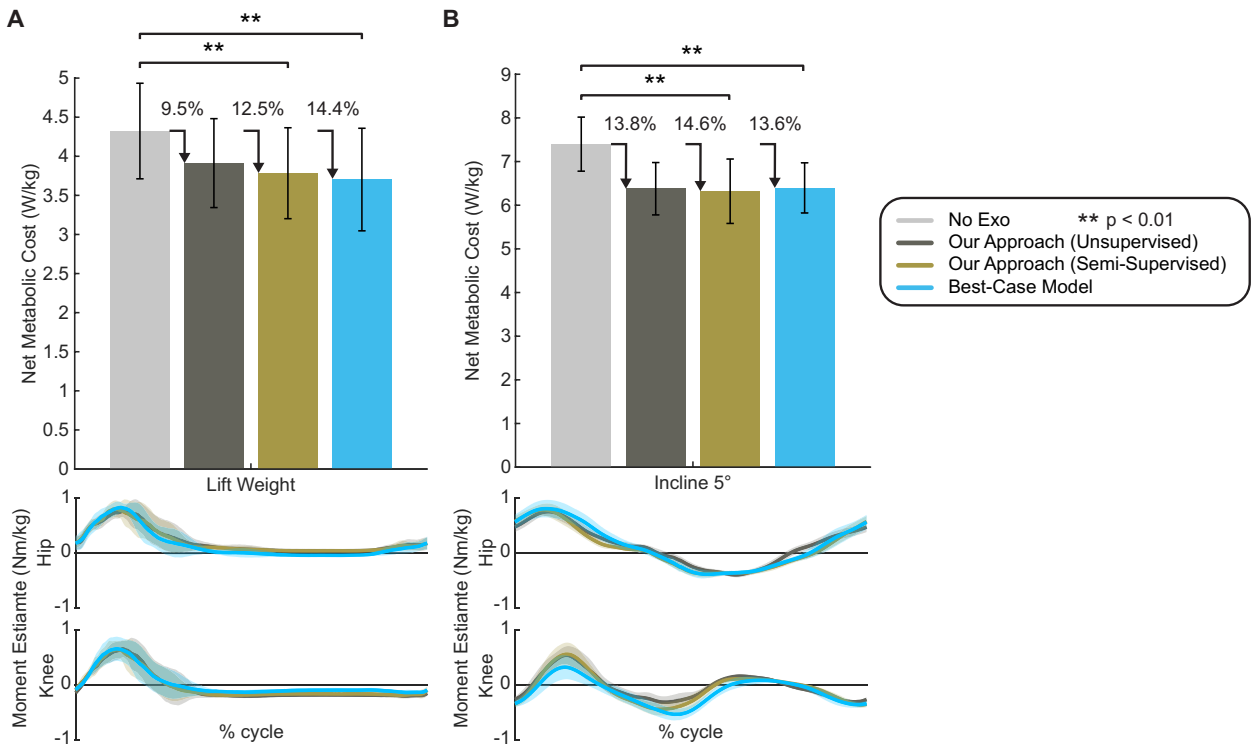


Fig. 5. Net metabolic cost comparisons for moment estimators trained on translated data. Eight participants performed a lift weight task (**A**) and an incline walking task (**B**) with our two controllers and the best-case model as well as without an exoskeleton. Bars represent the average net metabolic cost per body mass, and error bars represent SD across the eight participants. Asterisks indicate statistical significance from follow-up multiple comparison tests with Bonferroni correction ($P < 0.05$). Below the metabolic cost results are cycle-averaged estimates of hip and knee moment for each of the deployed controllers. Real-time estimates are split by cycle and then averaged separately by participant. The haze represents the SD across the eight participants, and the solid line represents the average.

time-series plots from a single cycle of each type of activity. Given that each activity is performed individually with each model, the ground truth moments are not the same for each repetition of the maneuver. The specific strides were selected so that the hip and knee moment RMSE and R^2 are close to the task average and the ground truth moments are also somewhat similar between controllers.

DISCUSSION

These results demonstrate that by leveraging our domain adaptation strategy to exploit data from the biomechanics domain without an exoskeleton, we can create useful data-driven controllers with only limited or even no access to device-specific, labeled data. Although these results are framed in the context of joint moment estimation for task-agnostic control, we showed that the sensor translation can take data without an exoskeleton and replicate characteristics of exoskeleton data, improving downstream models in the real exoskeleton data domain. This is further supported by our results using sensor translation to improve GRF estimation and activity classification. This unified framework for incorporating data from many different sources for real-time deployment has the potential to open previously unrecognized possibilities in device control by creating a large dataset for improving current algorithmic techniques and potentially enabling new, more complex deep learning algorithms.

The strong performance of our approach across a variety of tasks demonstrates its flexibility to improve performance on different size datasets. The task optimization sweep provides a potential look-up table for the number of tasks needed to reach a predefined performance accuracy. There is a tradeoff between the number of labeled tasks and the corresponding estimator performance with diminishing returns as the number of included tasks increases. However, our approach reduces the number of training tasks necessary, so much so that even using nine tasks with the baseline approach has higher error (hip: 26.2%, knee: 29.2%) than our approach with a single task (hip: 18.3%, knee: 19.4%). Thus, our approach makes it feasible to collect far fewer tasks for training. Although the exact optimized tasks differ from those of the baseline, similar trends appear (level ground walk and calisthenics are included in both) emphasizing a diverse set of tasks that covers the extremes of human movement. Similarly, adding labeled participants improves model performance but with decreasing benefit from each additional participant. Including four participants of labeled data achieved moment estimation error within 5% of including all 13 participants; therefore, it is feasible to collect far fewer participants and still achieve comparable results with our approach.

Although these participant and task sweeps involved testing on a left-out participant who was unseen in the training process, aggregate decisions such as hyperparameters were made on the basis of performance on those participants. Thus, the offline performance of our models on the truly novel participant set further confirms that our approach outperforms the baseline approach. When compared with the best-case model with access to all of the labeled data, our semisupervised approach, operating with only 5.1% of the labeled data (525 thousand labels versus 10.3 million labels), only incurred

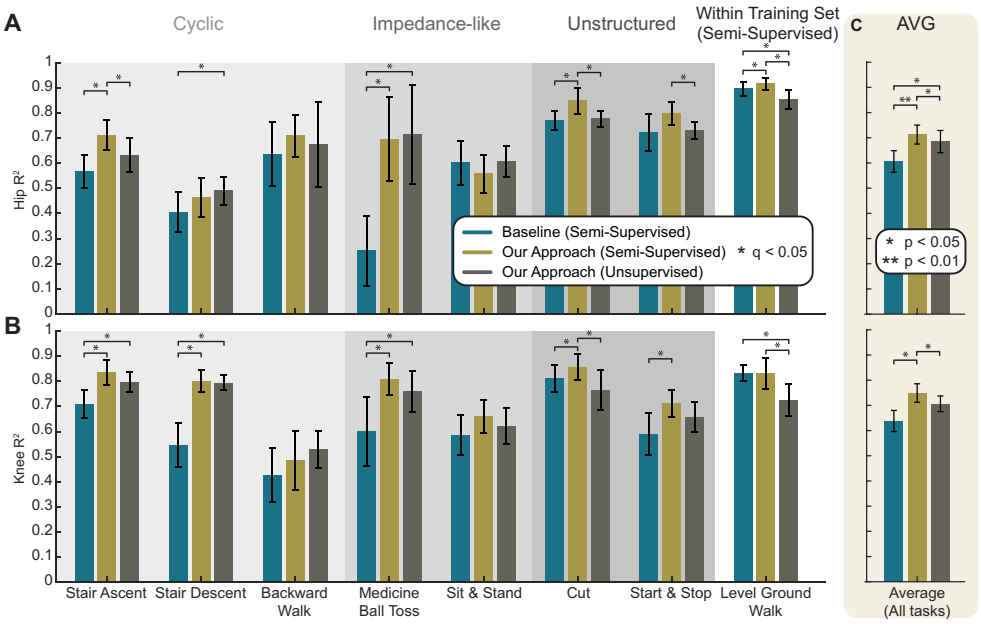


Fig. 6. Real-time deployed model performance broken into specific tasks. The individual R^2 performance of each deployed model on each task is presented for the hip (**A**) and the knee (**B**). Tasks are segmented into four categories. Three are based on the category of activity: cyclic, impedance-like, and unstructured; the fourth is based on the fact that level ground walking was one of the tasks for which the semisupervised models had access to ground truth labels during training, whereas the unsupervised model did not. Statistical significance was assessed using the false rate of discovery ($q < 0.05$) to account for all 24 comparisons. The average R^2 across all tasks (**C**) allows a direct comparison between the three deployed models. Asterisks indicate statistical significance from follow-up multiple comparison tests with Bonferroni correction ($P < 0.05$). Error bars represent the SD across eight participants.

a 10 to 20% increase in moment estimation RMSE at the hip and knee. Our unsupervised model demonstrates the superior performance of our training approach compared with the corresponding baseline using only simulated sensors. This demonstrates the benefits of using unlabeled exoskeleton data to facilitate the translation from simulated to real sensors. Although using no labeled data incurred additional error compared with the best-case model, the performance still remained between 20 and 45% of the best-case moment estimation RMSE at the hip and knee, which led to a stable system in all test scenarios. Although it is difficult to determine which aspects of the unlabeled data the model is learning, our preliminary exploration demonstrated that part of the benefit comes directly from learning to handle real sensor noise and part of the benefit seems to come from understanding the human interacting with the exoskeleton even with the exoskeleton unactuated (fig. S9). Also, by looking at the error reduction as each unlabeled participant was added, we did not see the improvement level off at 10 participants, indicating that more unlabeled data beyond that available in this study could potentially improve these models even further (fig. S10). Overall, these results demonstrate the potential for control engineers without access to a gait laboratory to use control architectures that would otherwise have been impossible.

Our real-time deployment further confirms the offline results and demonstrates that moment estimators trained with our framework maintain offline accuracies while estimating moments in real time (fig. S1). When tested in real time, our semisupervised approach again significantly outperformed the baseline model and incurred only a small penalty relative to the best-case model (between 10 and 20% increase in RMSE across the hip and knee). The same can be seen for the unsupervised model (between 25 and 35% across the hip and the knee). A key factor to note is our inability to run the unsupervised baseline during real time (fig. S3 and movie S1). This demonstrates that our approach takes a previously impossible controller, trained only on simulated data, and makes it possible and useful in the real world. Not only was this viable, but across tasks, our unsupervised approach even outperformed the semisupervised baseline (statistically significant at the hip, but not at the knee). The only case where it performed statistically worse than the semisupervised baseline was during level ground walking, which is reasonable given that this was the only tested task that was part of the labeled training set for the semisupervised baseline. The task breakdown (Fig. 6) demonstrates the additional benefit of including labeled data within our framework, as evidenced by statistically significant differences between our semisupervised approach and our unsupervised approach.

Although previous studies have demonstrated beneficial human outcomes from using moment estimation–based controllers (*11*, *12*), the metabolic cost results presented here demonstrate that models trained with translated data can also provide energetically beneficial assistance to users even on high torque tasks. Our semisupervised models reduced metabolic cost by a similar percentage as the best-case model, indicating that the small increase in estimation error had very little effect on performance for these tasks. Although there is some loss in consistency for metabolic performance without any labeled data, the model still enabled sizable reductions in metabolic cost for users across these tasks. This is unexpected because without our translation approach, the baseline comparison could not even be used in real time. These results demonstrate that our models overcame the metabolic cost penalty of wearing our 7-kg device and provided meaningful assistance to augment users' performances. Reductions greater than 10% are comparable to the current state-of-the-art controllers (*32*).

The unsupervised result demonstrates that it is possible to take labeled data from readily available biomechanics datasets combined with unlabeled data from a novel device and create a fully deployable deep learning network for that new device. Thus, researchers with minimal access to a gait lab can use open-source data, possibly combined from many different sources, to form a large set of labeled simulated exoskeleton data. Then, using that data with some additional data from users moving in a controllerless novel device outside the research laboratory, they can train a fully deployable task-agnostic controller. Given that the framework presented here performs sensor translation first and then downstream moment estimation, this same framework can be applied to many different types of data-driven

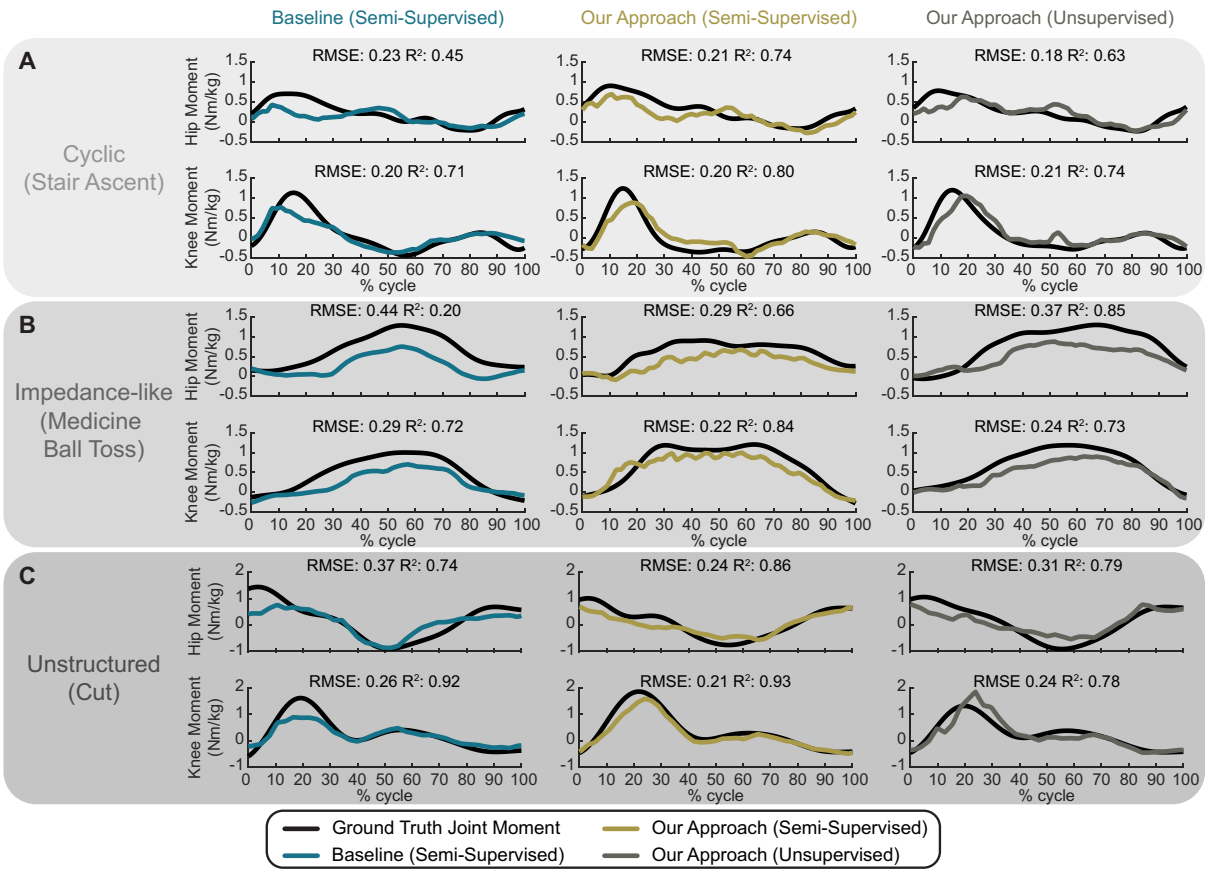


Fig. 7. Time series comparison of deployed models. Representative time-series plots are shown for each category of activity with the ground truth moment in black and the real-time joint moment estimate in color. Stair ascent is representative of cyclic tasks (**A**), medicine ball toss of impedance-like activities (**B**), and the outside leg for a single stride in cutting for unstructured tasks (**C**). To facilitate comparison across controllers, individual cycles were chosen by finding the three strides where the hip and knee RMSE and R^2 most closely matched the task average and then selecting the final cycles on the basis of ground truth moment similarity.

models beyond joint moment estimation, as we showed with GRF estimation and activity classification. This has the potential to enable faster development and broader use of these powerful techniques.

Our semisupervised approach provides a roadmap for continuing to improve these data-driven approaches over time. As more data become available through testing with the initial deployed model, these data can be efficiently exploited to continue to improve model performance. For researchers who already have some labeled data with their device, these results demonstrate a method for incorporating that data into the model. This work shows that with just four tasks from four participants—a tiny fraction of our overall dataset—estimation error with respect to the best-case model can be cut by half compared with having no labeled data.

We propose that our domain adaptation approach is applicable across many areas of exoskeleton research. Our task optimization results and those from Molinaro *et al.* (*12*) demonstrate that having data from a specific activity continues improving data-driven approaches for that specific activity even if the additional data do not promote generalization to other activities. Thus, if researchers and designers want to add additional functionality for a specific task (e.g., a golf swing), at present, they would have to collect exoskeleton data with that specific task. However, with the framework presented here, these data could come from an open-source biomechanics dataset without incurring data collection and labeling costs. Furthermore, this approach could be a valuable tool for creating models for unique populations. It is difficult to collect large participant datasets of movement for elderly individuals or for those with impaired mobility while wearing a device, but some datasets without a device are readily available (*33*–*35*). Hence, our framework could potentially make use of those datasets for use on new devices targeted toward these populations.

Limitations

This study has several limitations. First, our analysis of the unlabeled data demonstrated that we had not saturated performance with 10 participants. More benefit may be found from including additional participants' unlabeled data and selectively adding or removing unlabeled tasks. Second, only IMUs and encoders were used in translation to avoid using sensors outside the device. Exploring other sensing modalities beyond kinematics was beyond the scope of this study. Third, we performed initial optimization of individual network hyperparameters on a network-by-network basis; however, additional gain may be found by optimizing those hyperparameters in the context of the entire network. Fourth, although there is nothing device specific in our translation method, we only tested our models on the hip/knee exoskeleton presented. Last, we did not examine the

ability of our domain adaptation models to generalize to completely novel tasks that are not in the unlabeled dataset or open-source biomechanics datasets because of the relative ease of collecting or obtaining these data.

Conclusion

This paper presents our framework that considers the simulated sensor domain (on the basis of biomechanical models) as a stepping-stone domain for aggregating data. This allowed us to train and optimize a deep domain adaptation network that translates data from large biomechanics datasets into a device-specific, real sensor domain where labels are difficult and costly to collect. These translated data can be used for training downstream deep learning models, such as joint moment estimators, that allow powerful task-agnostic and user-independent exoskeleton control strategies. Using this network, we trained models using both no labeled data (unsupervised) and a small-scale dataset (semisupervised) for a real hip/knee exoskeleton. These models incurred only small additional estimation error compared with a best-case model with a complete labeled dataset from our exoskeleton and significantly outperformed models trained without domain adaptation. We also proved that these models could be deployed in real-time exoskeleton control, achieve comparable performance to the offline models, and augment human performance by reducing metabolic cost. This represents a breakthrough in user accessibility to data-driven approaches for exoskeleton control by cutting the need for costly data collection, a key weakness of our former approach (*11*, *12*), and providing a method to incorporate readily available human biomechanics data to improve model performance.

MATERIALS AND METHODS

Framework overview

The goal of our approach was to decrease the dependence on costly labeled data by replacing those data with easily accessible data. To make these accessible data useful, we translated data (sensor signals) that had associated labels (in our case joint moments) from a domain where the labels were easy to obtain through a stepping-stone domain (simulated sensors) into the data-limited domain (the real exoskeleton domain). We accomplished this using a translator that was trained through a CycleGAN with U-Net backbones. In this work, we considered the following three sources of data, each with a different level of costliness to collect and each allowing access to simulated sensors, real sensors, or both (Fig. 1). Labeled human biomechanics data collected without an exoskeleton are the least costly to obtain (our source domain) and can easily be transformed into our stepping-stone domain by simulating sensors similar to the device (dataset $\mathbb{D}_{ls}$); however, these data are the least valuable for direct use because they do not capture the richness of real sensor inputs. Unlabeled, unpowered exoskeleton data (unlabeled data in our target domain) are collected from a human user wearing a specific target device but without any need for external measuring equipment or a stand-in controller providing assistance (dataset $\mathbb{D}_{ur}$). Thus, these data are relatively easy to obtain in any environment but lie solely in the real domain. The third source, labeled exoskeleton data, is the costliest data to collect because they require a gait laboratory with instrumented floors and motion capture; however, they contain time-synced information from both the real sensor domain and our stepping-stone simulated sensor domain and therefore are easy to use to create a translation between the two domains (dataset $\mathbb{D}_{lr}$). Our framework uses previously underused data from the first two less costly sources to reduce or even remove the need for the third costly data source, which is traditionally the only source used for deployed machine learning algorithms (*3*, *11*).

To achieve real-time deployable models for exoskeleton control, we propose a two-part approach. First, we tackled the domain adaptation problem by training two translator networks through respective GANs, one to convert from the simulated to the real domain ($T_{s\to r}$) and vice versa ($T_{r\to s}$). For translation, we had access to many unpaired samples from the simulated sensor domain (X_s) (drawn from our source human biomechanics dataset domain) and the real sensor domain (X_r) (unlabeled data from the target domain). Although there is no direct time point–by–time point comparison for conventional supervised training, these unpaired samples can be used through domain adaptation to indirectly learn the translation. Our approach, inspired by a CycleGAN, uses adversarial learning to learn to match the data distribution of the translated sensor signals with that of the real sensor signals in the target domain. The unpaired samples from the simulated and real domains are the only data accessible in the unsupervised case.

In the semisupervised case, we had limited access to time-paired data samples from both the real (X_r) and simulated domain (X_s) (labeled exoskeleton data). However, we could not just learn a translation on the basis of these data because the semisupervised time-paired data are only available while wearing the exoskeleton. Thus, although any translation trained on these time-paired data would capture the noise characteristics of real sensors and any orientation differences between the simulated and real sensors, it would fail to capture the user's kinematic changes because of wearing the device (whether from human adaptation to assistance or the exoskeleton mass and range of motion constraints). Those changes are something we wanted the translator to learn. Thus, these data provided a few samples with labels that partially constrain, although imperfectly, the translation ($T_{r\to s}$ and $T_{s\to r}$) but could not perform as well on their own (fig. S2) or allow a direct measure of translator performance.

In the second part of our approach, we used one of the translator networks from domain adaptation ($T_{s\to r}$) to translate data from the generic human biomechanics datasets (the common shared domain X_s) into the real domain. Because these data came from datasets that contain inverse dynamics, the appropriate joint moment labels associated with the data were available (y_s). These label-paired data samples {i.e., translated sensor data $[T_{s\to r}(X_s)]$ and joint moment labels (y_s)} could then be used to train a moment estimator that could operate based on data from the real device. If we had access to some real device data with joint moment labels (semisupervised case X_r, y_r), then these could also be added into the training set for the moment estimator.

To summarize, we used three datasets: Dataset $\mathbb{D}_{ls}$ contained labeled human biomechanics data without an exoskeleton containing simulated sensor signals (X_s) and time-matched joint moment labels (y_s). Dataset $\mathbb{D}_{ur}$ contained unlabeled real sensor data with an exoskeleton containing real sensor signals (X_r) and no labels. Dataset $\mathbb{D}_{lr}$ contained labeled real and simulated sensor data with an exoskeleton containing real sensor signals (X_r), time-matched simulated sensor signals (X_s), and time-matched joint moment labels (y_r).

Domain adaptation network details

The sensor translation portion of the network combined two GANs, one generating data in the real sensor domain conditioned on data

from the simulated domain ($T_{r\to s}$) and the other operating in reverse ($T_{s\to r}$). Each GAN consisted of a U-Net–style translator (T) as the generator and a convolutional neural network (CNN) as the discriminator (D).

The GANs were set up bidirectionally, following a CycleGAN framework (*36*). Results from image translation and human activity recognition have demonstrated that bidirectional translation increases performance by encouraging minimal information loss across the cycle and forcing the translations to be inverses of each other (*29*, *36*, *37*). For moment estimation *ME*, a temporal convolutional network (TCN) architecture was selected on the basis of its success in previous studies (*11*, *12*, *38*).

Given the bidirectional nature of the translation, the moment estimator could be trained to operate based on real data (ME_r) or on simulated data (ME_s). We found that training the moment estimator on the real side (ME_r) outperformed the moment estimator on the simulated side (ME_s) with the added benefit of only requiring one model to be deployed in real time (for real-time implementation, the moment estimator trained on the simulated side would require both the real-to-simulation translator and the moment estimator to operate on incoming data). Thus, all results are based on a moment estimator trained with data that are either converted to or already in the real domain (ME_r). The networks are depicted in Fig. 8 along with the flow of data necessary to compute and combine multiple training losses. Five different types of losses were available to differing degrees based on the specific case, semisupervised or unsupervised. The rationale and equations for each loss function, the optimization and tuning of the hyperparameters for the overall framework, the architecture selection, and further hyperparameter tuning for the U-Net, TCN, and CNN architectures are described in Supplementary Methods and tables S1 and S2. Further details about training procedures are also included there. An in-depth analysis of the value of the GAN loss is provided in the Supplementary Materials and fig. S2.

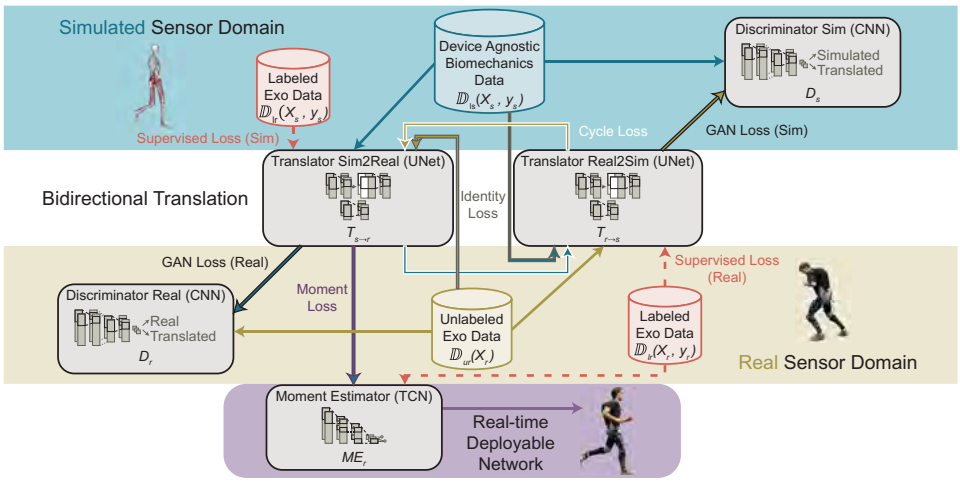


Fig. 8. Detailed network description for training the translator and downstream moment estimator. Bidirectional domain adaptation is accomplished through two pairs of translator and discriminator networks that function as GANs in both simulated and real domains. The forward pass of data is pictured by the arrows with the color indicating the original data source. The outline colors depict the portions of the forward pass that contribute to different components of the loss function including supervised loss, cycle loss, GAN loss, identity loss, and moment loss. The final fifth network, the moment estimator, is trained along with the other components but is the only portion required at run time for deployment on the device.

Training datasets

In training our framework, we chose to estimate hip and knee moments in keeping with Molinaro *et al.* (*12*). For exoskeleton sensors, we chose to use purely kinematic sensors on the thigh and shank. Thus, our sensor suite consisted of a hip and knee encoder (angle and velocity) and a thigh and shank inertial measurement unit (IMU). This mimics the most basic set of mechanical sensors available to a hip/knee exoskeleton without placing sensors outside the device.

Labeled biomechanics data (without an exoskeleton $\mathbb{D}_{ls}$) came from Scherpereel *et al.* (*39*). These data included joint angles, joint moments, and simulated IMUs on each segment for 12 participants performing 28 cyclic and noncyclic groups of tasks. We simulated encoders from the hip and knee inverse kinematics and simulated IMUs for the thigh and shank, matching IMU locations with those on our hip/knee exoskeleton.

Exoskeleton data came from Molinaro *et al.* (*12*), which included several phases of collection with 22 different participants. Our unlabeled real exoskeleton data ($\mathbb{D}_{ur}$) consisted of raw sensor data collected from the exoskeleton, whereas 10 participants performed tasks without exoskeleton assistance (actuators off). We used the real IMU data for the thigh and shank (OpenIMU, Tewksbury, MA) and encoder data from the hip and knee (T-Motor AK80-9s, Nanchang, China). For our semisupervised model, we included a combination of actuated and unactuated data from the first 15 participants across a range of tasks ($\mathbb{D}_{lr}$). Eight participants from "phase 3" of this dataset formed our true test set and were used to evaluate performance. No data from these participants were used to train the models ensuring that these were novel participants to verify that the models were user independent. Additional details on modifications to this dataset are described in the Supplementary Materials.

Assessing the joint moment estimation performance of the model

We assessed the estimator performance using RMSE between the model estimated moments and the actual ground truth joint moments calculated using inverse dynamics from OpenSim. We then averaged the RMSE and R^2 across subtasks in the larger 28 task groups and then across tested participants (*12*).

To provide suitable comparisons to demonstrate the performance of our algorithm, we present the results relative to two models: a best-case model (our goal being to achieve accuracies similar to those of this model) and a baseline model, which was trained without domain adaptation. To construct the best-case comparison, we trained a moment estimator using all of the training tasks and participants, thus replicating the results of Molinaro *et al.* (*12*) but without pressure insoles, a foot IMU, or delayed hip moment estimates and using all 28 tasks for training (Fig. 4H). Although it would be theoretically possible given a substantially larger set of no exoskeleton data to achieve higher accuracies with our method than this best-case model,

our no exoskeleton dataset was of similar size. For the baseline model in the semisupervised case, we trained a moment estimator in the same way as the best-case model but with only the same limited number of tasks and limited number of participants that we gave to our own approach (Fig. 4G). The baseline model in the unsupervised case was trained with only simulated sensor data for participants without an exoskeleton (10 participants and 28 tasks as in Fig. 4I). These baselines were the best models that could be constructed using traditional supervised learning techniques given the data limitations.

To report our results, we computed the percentage increase in error for our algorithm and for the baseline with respect to the best-case model as seen in Eq. 1

$$\%\,\text{increase in RMSE} = \frac{\text{RMSE}_{\text{approach}} - \text{RMSE}_{\text{bestcase}}}{\text{RMSE}_{\text{bestcase}}} \cdot 100\,\% \quad (1)$$

The value of this percentage increase expresses how far the approach is away from the best case, and the percentages can be compared between our approach and the baseline.

Real-time experimental validation

In contrast with offline performance (where previously collected and held-out data are passed through the model), validating real-time performance required us to deploy three models in our real-time controller loop. For the semisupervised case, we chose to deploy both the baseline and our approach in real time. As explained in the Results section, four tasks and four participants were selected for use in the labeled training set. The specific four tasks were based on their respective optimal task ranking. For the unsupervised case, we deployed only a model trained with our approach. Although we attempted to deploy the baseline model for this case, the model accuracy with only simulated data was so poor that the controller was not stable for real-time deployment (see movie S1 and fig. S3). To provide a direct offline comparison, we tested the unsupervised baseline model post hoc with the same data collected with our real-time unsupervised model.

To deploy the models in real time, we used the autonomous hip/knee exoskeleton developed by X, the Moonshot Factory, and detailed in our previous publication (*12*). Because the moment estimators are directly real-time deployable, no additional computational time loss was incurred by running our models compared to our previous work. As in our previous work, a main single board computer (Raspberry Pi, Cambridge, UK) handled the input/output for the device and data logging, whereas a separate coprocessor handled deep learning inference (NVIDIA Jetson Nano, Santa Clara, CA). All experimenter interaction was handled through an offboard laptop connected to a Wi-Fi network hosted by the onboard microprocessor. The device has four actuators (AK809 T-Motor, Nanchang), two at the hips and two at the knees for bilateral sagittal plane assistance at hip and knee. The device and mechatronics are pictured in Fig. 4A. As in our previous work, the estimated biological moment was converted to applied torque by first scaling the moment (by participant mass and then by a scaling factor of 20% for the hip and 15% for the knee), then delaying the torque (by 100 ms for the hip and 50 ms for the knee), and last filtering the torque using a low-pass filter (second-order Butterworth with a 10-Hz cutoff frequency) (*12*).

We performed a protocol similar to that used to collect the training data (*12*, *39*) with a pared-down task list. Eight novel participants not included in the training or offline testing data (four males, four females; age: 23.0 ± 3.8 years; height: 173.4 ± 2.6 cm; body mass: 68.2 ± 10.7 kg) performed eight tasks with three different controllers. Informed consent was obtained from each participant under Georgia Institute of Technology Institutional Review Board protocol H21184. The tasks were split between the three larger task categories used in (*12*) with seven tasks outside of the labeled training set for the semisupervised models as seen in Fig. 4B. These tasks were medicine ball toss [15 lb (6.8 kg) center, right, left], cutting (left and right, fast and slow), sit-to-stand (short and tall chair, with and without armrests), starting and stopping, stair ascent (slow, normal, fast), stair descent (slow, normal, fast), and walking backward (0.6, 0.8, 1.0 m/s). More details on the protocols for each task can be found in our previous dataset publication (*39*). We also included one task that was in the training set for the semisupervised models: level ground walking (0.6 m/s, 1.2 m/s, 1.8 m/s, and shuffle). The order of the models was randomized and blinded for each participant; all tasks were performed with a single model before switching models. During each task, exoskeleton data for the encoders (hip and knee), IMUs (thigh and shank), and the estimated joint moments from the current model were recorded. Motion capture data were recorded at 200 Hz and GFRs at 1000 Hz.

Data processing was performed in a similar manner to our previous work (*12*) where participant-specific musculoskeletal models were fit to each participant standing in a static pose. The exoskeleton mass was then added for the actuators at their respective joints and the additional mass was added at the torso. Inverse kinematics matched the model to the measured motion capture locations, and inverse dynamics were then calculated on the basis of the GFRs using OpenSim (*30*, *31*). An activity flag was computed to determine when the activity was occurring and then used as the sections to assess model performance. Ground truth joint moments were filtered before being compared with the real-time estimates. Further details on biomechanical processing and analysis can be found in our previous work (*12*, *39*).

To assess the real-time model performance, we captured the joint moment estimates and then post hoc aligned those estimates with the ground truth moment labels as in (*12*). We did not deploy the best-case model in real time when collecting data for joint moment accuracy but instead ran the collected sensor signals through the best-case model post hoc, thus allowing a fairer comparison of performance. We did this individually for the data from each model; thus, the best-case model performance was unique to each specific model's data (except in the case of the unsupervised baseline as noted above).

We further assessed human performance outcomes by having eight novel participants collect metabolic cost measurements during a lifting task (six males, two females; age: 25.0 ± 3.9 years; height: 174.6 ± 2.4 cm; body mass: 71.0 ± 9.8 kg) and an incline walking task (five males, three females; age: 25.0 ± 2.1 years; height: 175.3 ± 4.2 cm; body mass: 68.9 ± 7.0 kg). Before any metabolic cost collection, we performed the same acclimation procedure as (*12*). For the lifting task, a metronome was set at 10 beats per minute, and participants were instructed on each tone to use both hands to lift a 25 lb. (11.3 kg) kettle bell weight off a waist-height shelf, touch that weight to the floor between their feet, and then replace the weight on the shelf (*12*). For the incline task, participants walked on a 5° inclined treadmill at 1.25 m/s. Four conditions were tested for each activity: our semisupervised model, our unsupervised model, the best-case model, and no exoskeleton. Participants performed the activity

for 6 min, and conditions were tested using a within-participant counter-balanced design (ABCD-DCBA). Because of an exoskeleton malfunction, one incline trial for one participant only lasted 3 min, so a first-order fit estimate was used for that trial (*40*, *41*) and averaged with the other 6-min trial with the same controller. The order of the three exoskeleton conditions was randomized, and participants were blinded to the condition; however, for a given task, the no exoskeleton condition was placed either at the beginning (A) or the end (D) for don and doff efficiency. Motion capture—and thus ground truth joint moments—was not available during metabolic cost collections. Results for all metabolic cost measurements are reported on the basis of an average of instantaneous metabolic cost (as calculated with the Brockway equation) for the final 3 min of each activity.

Statistical analyses

All statistical analyses were performed in MATLAB (MathWorks, Natick, MA) using a significance level of $\alpha = 0.05$. For the task and participant performance sweeps, significance was assessed by controlling the false rate of discovery ($q < 0.05$) across model conditions (our approach versus the baseline), where percentage RMSE increase compared with the best case was the dependent variable (*42*, *43*). We assessed the significance of our approach versus the baseline with each added task or participant and used the false rate of discovery to control the family-wise error rate across all comparisons (9 comparisons for the task case and 13 for the participant case).

For the final offline testing set, we assessed significance using paired *t* tests between the baseline and our approach at the hip and separately at the knee. Percentage RMSE increase compared with the best case was the dependent variable, which was averaged across task groups. The independent variable was the model type; participants were the fixed effect.

For real-time testing, comparisons between the baseline and our approach were again performed with paired *t* tests separately for hip and knee and for the supervised and unsupervised cases. Tests were performed with percentage RMSE increase compared with the best case as the dependent variable. Again, the independent variable was the model type, and participants were the fixed effect. For metabolic comparisons, we performed repeated one-way analysis of variance (ANOVA) tests for each task with participants as the random effect. This was followed up by planned Bonferroni multiple comparison tests to compare each controller to the no exoskeleton condition but not to each other.

To assess the statistical significance for each individual task with R^2 as the dependent variable, we again controlled the false rate of discovery to account for the number of comparisons between model types (three models were run in real time) and number of tasks (eight performed in the real-time validation), thus controlling for all 24 comparisons. Tests were performed with R^2 as the dependent variable separately for each model type with offline versus real time as the independent variable. We also compared the average performance across all tasks directly between the three models that were run in real time using a one-way analysis of variance test with participants as the random effect. This was followed up by Bonferroni multiple comparison tests between model types. To assess offline versus real-time performance, separate two-sample *t* tests with assumed equal variance were performed for each deployed model at each joint to determine whether running the model in real time had a statistically significant effect on performance.

Supplementary Materials

The PDF file includes:

Methods
Figs. S1 to S10
Tables S1 and S2
References (*44*–*53*)

Other Supplementary Material for this manuscript includes the following:

Movie S1
MDAR Reproducibility Checklist

REFERENCES AND NOTES

1. R. M. Karulkar, P. M. Wensing, Personalized estimation of intended gait speed for lower-limb exoskeleton users via data augmentation using mutual information. *IEEE Robot. Autom. Lett.* **7**, 9723–9730 (2022).
2. I. Kang, P. Kunapuli, H. Hsu, A. J. Young, "Electromyography (EMG) signal contributions in speed and slope estimation using robotic exoskeletons," in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)* (IEEE, 2019), pp. 548–553.
3. M. K. Shepherd, D. D. Molinaro, G. S. Sawicki, A. J. Young, Deep learning enables exoboot control to augment variable-speed walking. *IEEE Robot. Autom. Lett.* **7**, 3571–3577 (2022).
4. I. Kang, P. Kunapuli, A. J. Young, Real-time neural network-based gait phase estimation using a robotic hip exoskeleton. *IEEE Trans. Med. Robot. Bionics* **2**, 28–37 (2020).
5. I. Kang, D. D. Molinaro, S. Duggal, Y. Chen, P. Kunapuli, A. J. Young, Real-time gait phase estimation for robotic hip exoskeleton control during multimodal locomotion. *IEEE Robot. Autom. Lett.* **6**, 3491–3497 (2021).
6. J. Wang, D. Wu, Y. Gao, X. Wang, X. Li, G. Xu, W. Dong, Integral real-time locomotion mode recognition based on GA-CNN for lower limb exoskeleton. *J. Bionic. Eng.* **19**, 1359–1373 (2022).
7. Y. Qian, Y. Wang, C. Chen, J. Xiong, Y. Leng, H. Yu, C. Fu, Predictive locomotion mode recognition and accurate gait phase estimation for hip exoskeleton on various terrains. *IEEE Robot. Autom. Lett.* **7**, 6439–6446 (2022).
8. H. Zhao, Z. Qiu, D. Peng, F. Wang, Z. Wang, S. Qiu, X. Shi, Q. Chu, Prediction of joint angles based on human lower limb surface electromyography. *Sensors (Basel)* **23**, 5404 (2023).
9. T. Lee, I. Kim, S.-H. Lee, Estimation of the continuous walking angle of knee and ankle (talocrural joint, subtalar joint) of a lower-limb exoskeleton robot using a neural network. *Sensors* **21**, 2807 (2021).
10. J. Lin, N. V. Divekar, G. C. Thomas, R. D. Gregg, Optimally biomimetic passivity-based control of a lower-limb exoskeleton over the primary activities of daily life. *IEEE Open J. Control Syst.* **1**, 15–28 (2022).
11. D. D. Molinaro, I. Kang, A. J. Young, Estimating human joint moments unifies exoskeleton control, reducing user effort. *Sci. Robot.* **9**, eadi8852 (2024).
12. D. D. Molinaro, K. L. Scherpereel, E. B. Schonhaut, G. Evangelopoulos, M. K. Shepherd, A. J. Young, Task-agnostic exoskeleton control via biological joint moment estimation. *Nature* **635**, 337–344 (2024).
13. D. Lee, I. Kang, D. D. Molinaro, A. Yu, A. J. Young, Real-time user-independent slope prediction using deep learning for modulation of robotic knee exoskeleton assistance. *IEEE Robot. Autom. Lett.* **6**, 3995–4000 (2021).
14. F. Yang, C. Chen, Z. Wang, H. Chen, Y. Liu, G. Li, X. Wu, ViT-based terrain recognition system for wearable soft exosuit. *Biomim. Intell. Robot.* **3**, 100087 (2023).
15. B. Laschowski, W. McNally, A. Wong, J. McPhee, "Preliminary design of an environment recognition system for controlling robotic lower-limb prostheses and exoskeletons," in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)* (IEEE, 2019), pp. 868–873.
16. P. Slade, M. J. Kochenderfer, S. L. Delp, S. H. Collins, Personalizing exoskeleton assistance while walking in the real world. *Nature* **610**, 277–282 (2022).
17. J. M. Lopes, J. Figueiredo, P. Fonseca, J. J. Cerqueira, J. P. Vilas-Boas, C. P. Santos, Deep learning-based energy expenditure estimation in assisted and non-assisted gait using inertial, EMG, and heart rate wearable sensors. *Sensors* **22**, 7913 (2022).
18. U. H. Lee, V. S. Shetty, P. W. Franks, J. Tan, G. Evangelopoulos, S. Ha, E. J. Rouse, User preference optimization for control of ankle exoskeletons using sample efficient active learning. *Sci. Robot.* **8**, eadg3705 (2023).
19. L. Zhang, D. Soselia, R. Wang, E. M. Gutierrez-Farewik, Lower-limb joint torque prediction using LSTM neural networks and transfer learning. *IEEE Trans. Neural Syst. Rehabil. Eng.* **30**, 600–609 (2022).
20. B. X. W. Liew, D. Rügamer, X. Zhai, Y. Wang, S. Morris, K. Netto, Comparing shallow, deep, and transfer learning in predicting joint moments in running. *J. Biomech.* **129**, 110820 (2021).
21. J. Sloboda, P. Stegall, R. J. McKindles, L. Stirling, H. C. Siu, "Utility of inter-subject transfer learning for wearable-sensor-based joint torque prediction models," in *2021 43rd Annual*

International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) (IEEE, 2021), pp. 4901–4907.
22. Y. Wang, Z. Li, X. Wang, H. Yu, W. Liao, D. Arifoglu, Human gait data augmentation and trajectory prediction for lower-limb rehabilitation robot control using GANs and attention mechanism. *Machines* **9**, 367 (2021).
23. M. Kim, L. J. Hargrove, Generating synthetic gait patterns based on benchmark datasets for controlling prosthetic legs. *J. Neuroeng. Rehabil.* **20**, 115 (2023).
24. Y. Chang, A. Mathur, A. Isopoussu, J. Song, F. Kawsar, A systematic study of unsupervised domain adaptation for robust human-activity recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* **4**, 1–30 (2020).
25. F. Mu, X. Gu, Y. Guo, B. Lo, "Unsupervised domain adaptation for position-independent IMU based gait analysis," in *2020 IEEE SENSORS* (IEEE, 2020), pp. 1–4.
26. A. Z. M. Faridee, A. Chakma, A. Misra, N. Roy, STranGAN: Adversarially-learnt spatial transformer for scalable human activity recognition. *Smart Health* **23**, 100226 (2022).
27. C. Hegde, G. Wen, L. C. Price, Activity classification using unsupervised domain transfer from body worn sensors. *Smart Health* **30**, 100431 (2023).
28. A. Akbari, R. Jafari, "Transferring activity recognition models for new wearable sensors with deep generative domain adaptation," in *Proceedings of the 18th International Conference on Information Processing in Sensor Networks* (Association for Computing Machinery, 2019), pp. 85–96.
29. C. Chen, Y. Miao, C. X. Lu, L. Xie, P. Blunsom, A. Markham, N. Trigoni, MotionTransformer: Transferring neural inertial tracking between domains. *Proc. AAAI Conf. Artif. Intell.* **33**, 8009–8016 (2019).
30. S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, D. G. Thelen, OpenSim: Open-source software to create and analyze dynamic simulations of movement. *IEEE Trans. Biomed. Eng.* **54**, 1940–1950 (2007).
31. A. Seth, J. L. Hicks, T. K. Uchida, A. Habib, C. L. Dembia, J. J. Dunne, C. F. Ong, M. S. DeMers, A. Rajagopal, M. Millard, S. R. Hamner, E. M. Arnold, J. R. Yong, S. K. Lakshmikanth, M. A. Sherman, J. P. Ku, S. L. Delp, OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement. *PLOS Comput. Biol.* **14**, e1006223 (2018).
32. G. S. Sawicki, O. N. Beck, I. Kang, A. J. Young, The exoskeleton expansion: Improving walking and running economy. *J. Neuroeng. Rehabil.* **17**, 25 (2020).
33. M. Serrao, G. Chini, M. Bergantino, D. Sarnari, C. Casali, C. Conte, A. Ranavolo, C. Marcotulli, M. Rinaldi, G. Coppola, F. Bini, F. Pierelli, F. Marinozzi, Dataset on gait patterns in degenerative neurological diseases. *Data Brief* **16**, 806–816 (2018).
34. T. Van Criekinge, W. Saeys, S. Truijen, L. Vereeck, L. H. Sloot, A. Hallemans, A full-body motion capture gait dataset of 138 able-bodied adults across the life span and 50 stroke survivors. *Sci. Data* **10**, 852 (2023).
35. P.-F. David, R.-C. David, J. C. Moreno, T. Diego, Human locomotion databases: A systematic review. *IEEE J. Biomed. Health Inform.* **28**, 1716–1729 (2024).
36. J.-Y. Zhu, T. Park, P. Isola, A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)* (IEEE, 2017), pp. 2242–2251.
37. S. An, A. Medda, M. N. Sawka, C. J. Hutto, M. L. Millard-Stafford, S. Appling, K. L. S. Richardson, O. T. Inan, AdaptNet: Human activity recognition via bilateral domain adaptation using semi-supervised deep translation networks. *IEEE Sens. J.* **21**, 20398–20411 (2021).
38. D. D. Molinaro, I. Kang, J. Camargo, M. C. Gombolay, A. J. Young, Subject-independent, biological hip moment estimation during multimodal overground ambulation using deep learning. *IEEE Trans. Med. Robot. Bionics* **4**, 219–229 (2022).
39. K. Scherpereel, D. Molinaro, O. Inan, M. Shepherd, A. Young, A human lower-limb biomechanics and wearable sensors dataset during cyclic and non-cyclic activities. *Sci. Data* **10**, 924 (2023).
40. J. Zhang, P. Fiers, K. A. Witte, R. W. Jackson, K. L. Poggensee, C. G. Atkeson, S. H. Collins, Human-in-the-loop optimization of exoskeleton assistance during walking. *Science* **356**, 1280–1284 (2017).
41. J. C. Selinger, J. M. Donelan, Estimating instantaneous energetic cost during non-steady-state gait. *J. Appl. Physiol.* **117**, 1406–1415 (2014).
42. Y. Benjamini, Y. Hochberg, Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J. R. Stat. Soc.: B (Methodol.)* **57**, 289–300 (1995).
43. K. L. Scherpereel, D. D. Molinaro, M. K. Shepherd, O. T. Inan, A. J. Young, Improving biological joint moment estimation during real-world tasks with EMG and instrumented insoles. *IEEE Trans. Biomed. Eng.* **71**, 2718–2727 (2024).
44. X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (IEEE, 2017), pp. 2794–2802.
45. Y. Taigman, A. Polyak, L. Wolf, Unsupervised cross-domain image generation. arXiv:1611.02200 [cs.CV] (2016).
46. H. Haresamudram, I. Essa, T. Plötz, Assessing the state of self-supervised human activity recognition using wearables. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* **6**, 116 (2022).
47. F. J. Ordóñez, D. Roggen, Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors* **16**, 115 (2016).
48. O. Ronneberger, P. Fischer, T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, A. F. Frangi, Eds. (Lecture Notes in Computer Science, Springer International Publishing, 2015), pp. 234–241.
49. Y. Zhang, Z. Zhang, Y. Zhang, J. Bao, Y. Zhang, H. Deng, Human activity recognition based on motion sensor using U-Net. *IEEE Access* **7**, 75213–75226 (2019).
50. H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, P.-A. Muller, Deep learning for time series classification: A review. *Data Min. Knowl. Disc.* **33**, 917–963 (2019).
51. S. Bai, J. Z. Kolter, V. Koltun, An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv:1803.01271 [cs.lG] (2018).
52. M. S. B. Hossain, Z. Guo, H. Choi, Estimation of lower extremity joint moments and 3D ground reaction forces using IMU sensors in multiple walking conditions: A deep learning approach. *IEEE J. Biomed. Health Inform.* **27**, 2829–2840 (2023).
53. J. Beil, I. Ehrenberger, C. Scherer, C. Mandery, T. Asfour, "Human motion classification based on multi-modal sensor data for lower limb exoskeletons," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2018), pp. 5431–5436.

Acknowledgments: We thank R. Emadi for his tireless support collecting the model validation data. We also thank J. Leestma, K. Ghonasgi, and C. Nuesslein for reviewing the manuscript and D. Molinaro for being a sounding board for initial ideas. We acknowledge X, the Moonshot Factory, for providing the hardware to test these controllers. We also thank the research cyberinfrastructure resources and services provided by the Partnership for an Advanced Computing Environment (PACE) at the Georgia Institute of Technology, Atlanta, GA, USA. **Funding:** This work was supported by the National Science Foundation Graduate Research Fellowship no. DGE-2039655 (to K.L.S.) and the National Science Foundation Foundational Research in Robotics nos. 2233164 (to A.J.Y. and M.C.G.) and 2328051 and 2328050 (to A.J.Y. and M.K.S.). **Author contributions:** Conceptualization: K.L.S. and A.J.Y. Methodology: K.L.S., M.C.G., A.J.Y., and O.T.I. Software: K.L.S. Validation: K.L.S. Formal analysis: K.L.S. Investigation: K.L.S. and C.A.C. Resources: A.J.Y. Data curation: K.L.S. Visualization: K.L.S. Funding acquisition: K.L.S., M.C.G., M.K.S., and A.J.Y. Project administration: K.L.S. and A.J.Y. Supervision: K.L.S., A.J.Y., M.C.G., M.K.S., and O.T.I. Writing—original draft: K.L.S. and M.K.S. Writing—review and editing: K.L.S., M.K.S., A.J.Y., and M.C.G. **Competing interests:** K.L.S. and A.J.Y. are inventors on a US Patent application, patent pending, titled "Transfer learning of deep learning system and method for task agnostic wearable robot control and human monitoring," through the Georgia Institute of Technology, which describes some of the domain adaptation strategies used in this study. **Data and materials availability:** All data needed to evaluate the conclusions in the paper are present in the paper or the Supplementary Materials. The participant datasets for training are both open source: the data from Scherpereel *et al.* (*39*) can be found at https://doi.org/10.35090/gatech/70296 and the data from Molinaro *et al.* (*12*) can be found at https://doi.org/10.35090/gatech/75759. The new eight-participant dataset of time-synced exoskeleton data and human biomechanics and code for reproducing the main text figures is hosted in the SMARTech repository (https://doi.org/10.35090/gatech/79332).

Submitted 7 September 2024
Accepted 22 October 2025
Published 19 November 2025
10.1126/scirobotics.ads8652

FIELD ROBOTS

Robotic locomotion through active and passive morphological adaptation in extreme outdoor environments

Max Polzin*, Qinghua Guan, Josie Hughes

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

Robotic locomotion has shown substantial advancements, yet robots still lack the versatility and agility shown by animals navigating complex terrains. This limits their applicability in complex environments where they could be highly beneficial. Unlike existing robots that rely on intricate perception systems to construct models of both themselves and their surroundings, a more bioinspired approach leverages reconfiguration to adapt a robot's morphology to its environment. Although interest in such multimodal, terrain-adaptive robots is increasing, their capacity for morphological reconfiguration often remains confined to specific body parts or comes at the expense of increased system complexity and reduced locomotion efficiency. Our study seeks to enhance robotic locomotion by developing robots that can actively reconfigure their morphology, altering their physical properties and leveraging their adaptability to navigate efficiently in diverse environments. We demonstrate how combining a compliant structure with morphological reconfiguration allowed a robot to transition between flat and spherical forms, enabling autonomous, multimodal locomotion—driving, rolling, and swimming—across complex terrains with minimal sensing. By actively reconfiguring its morphology to adapt physical properties for compliant interactions, the robot enhanced locomotion across rough, diverse environments. Leveraging its adaptability in different locomotion modes to navigate a 4.5-kilometer path across mountainous, aquatic, and urban terrains, the robot outperformed traditional and multimodal robots in terms of versatility, energy efficiency, and robustness. Developing versatile, energy-efficient, compliant robots capable of reconfiguring their morphology could substantially enhance autonomous navigation, opening up broader applications in unstructured environments, from environmental monitoring to disaster response and extraterrestrial exploration.

INTRODUCTION

The ability to autonomously traverse and locomote across terrains of varying gradients and to overcome obstacles or extreme surfaces remains a capability firmly associated with animals (*1*). Despite substantial advances in robotic locomotion and exploration, the versatility, efficiency, and autonomy of robots typically limit their application to a small range of preknown terrains or environmental niches. Through advances in robotic hardware, planning methods, and learning algorithms, robots are beginning to be able to operate in geometrically complex and dynamic environments (*2*, *3*). Yet, robots still lack the robustness and adaptability to respond to varied and challenging terrains as demonstrated by animals (*4*). However, the demand for robots of various sizes that can operate reliably under diverse and unpredictable environmental conditions is increasing. Robots with these capabilities can, for example, be used for biodiversity monitoring and surveying of sensitive ecosystems with minimal environmental effects across various habitats, including aquatic, terrestrial, and mountainous regions (*5*). For climate studies in extreme glacial or polar environments, robots with a small environmental footprint can explore confined spaces or cavities to gather essential data in challenging conditions (*6*). Lightweight, versatile, and robust field robots, despite their limited payload capacities, offer substantial benefits in areas like precision agriculture—where they reduce soil compression (*7*)—and space exploration—where their low weight makes them cheaper to transport (*8*). Amid increasingly severe climate events, there is a growing need for robots capable of providing support or rescue in the aftermath of landslides, earthquakes, and volcanic eruptions (*9*). In all of these scenarios, the terrain is complex, varied, and not always known in advance. By developing robots that can operate efficiently across a range of terrains, the use of robots can be extended to such applications, with the potential for substantial scientific, societal, and safety benefits (*10*, *11*).

One approach animals leverage to achieve versatility across terrains is the inclusion of compliance in their bodies (*12*). Compliance can be distributed in different structures and mechanisms (*13*). This has been observed in cockroaches, which can compress their bodies to navigate through tight spaces and absorb shocks (*14*); kangaroos, which store energy in their legs to enable an explosive hopping motion to overcome obstacles (*15*); or soft structured worms or octopuses, which can physically squeeze through narrow gaps or structures (*16*). These animals locally adapt to their environment for better navigation and for physical robustness against harsh or impactful interactions. Compliance can also enable physical reconfiguration, allowing animals to adapt their structural form to respond to a changing environment (*17*). A common example seen across nature is reconfiguration into a spherical form to provide robustness to the environment (Fig. 1A). Hedgehogs curl into a ball by tucking their bodies (*18*), armadillos assume a spherical form to be protected by their armored shell (*19*), and woodlice roll up into a ball for protection (*20*). Tardigrades, known for their ability to survive in the harshest conditions, protect their vital organs by contracting their bodies into a barrel-like shape (*21*). This reconfiguration can aid locomotion; the golden wheel spider (*Carparachne aureoflava*) changes into a spherical shape to facilitate energy-efficient locomotion on sand dunes (*22*). The phenomenon of adopting a spherical shape for mobility is also used by plants for dispersal across new

CREATE Lab, EPFL, Lausanne, Switzerland.
*Corresponding author. Email: max.polzin@epfl.ch

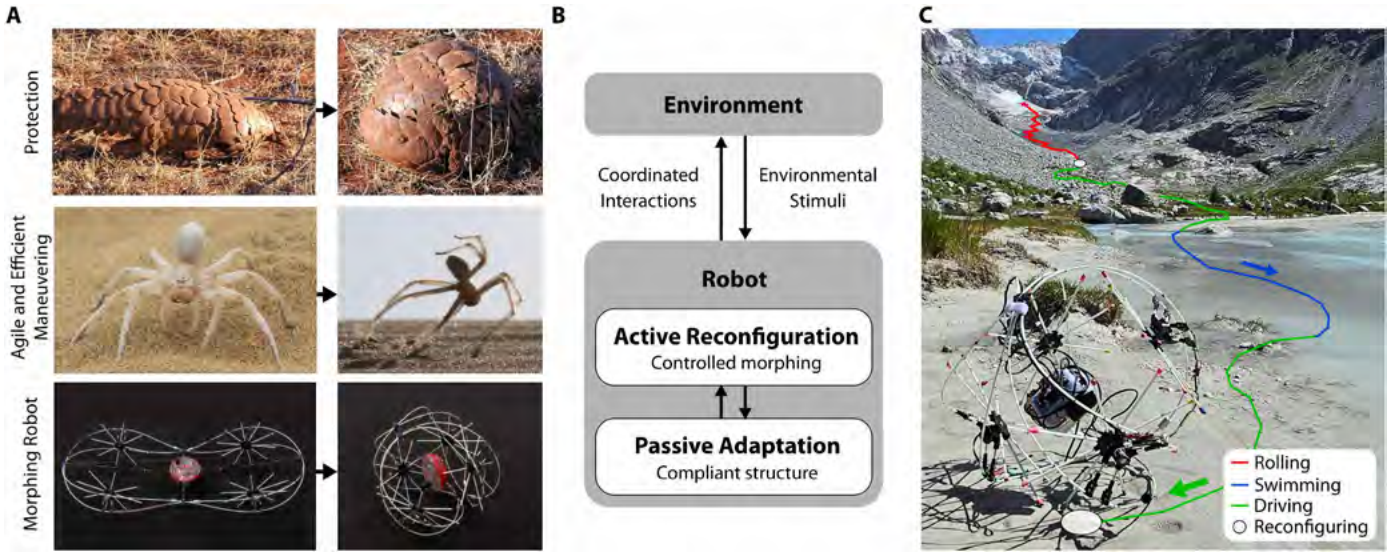


Fig. 1. Morphological reconfiguration and terrain-adaptive locomotion. (**A**) Reconfiguration capabilities of animals from locomotion to spherical shapes and the GOAT robot with similar capabilities. (**B**) The role of active reconfiguration for adapting to the environment and altering the passive properties of the robot. (**C**) Image of the robot in the ball configuration demonstrating the types of terrain that can be covered and the different modalities of locomotion that can achieve this.

habitats (*23*). Tumbleweeds, for instance, of the resurrection plant (*Selaginella lepidophylla*), use this form to be carried by the wind to colonize distant environments (*24*). Active morphological reconfiguration is an evolved response that has occurred across organisms of different sizes and structures and for different environmental niches. In all of these cases, these organisms use the compliance of their bodies to actively reconfigure their shapes in response to environmental stimuli. This reconfiguration leads to a substantial change in the passive properties of these organisms' bodies and provides advantageous interactions with their environment.

Active reconfiguration has been explored as a means of expanding robotic capabilities. Typically demonstrated in rigid robots, reconfiguration can enable switching between locomotion modalities such as flying or locomoting (*25*, *26*), enabling wall climbing (*27*) and navigation in varied terrain (*28*). Reconfiguration can also be localized within the body, for example, in wheels that transform to overcome obstacles (*29*–*33*) or that can be folded to save space (*34*). More modular, origami-inspired approaches rearrange or combine modules to locomote (*35*). However, the modularity can make dynamic adaptation challenging (*36*). Tensegrity robots (*37*) and truss-style morphing enable larger, meter-scale morphing robots (*38*), but the speeds of motion are typically low. Embedding compliance into reconfigurable robots has been shown to be advantageous and can enable locomotion across land and water (*39*, *40*). Examples include reconfigurable soft flippers and wheels on the compliant-legged hexapod robot RHeX (*41*–*43*), compliant augmentation of a rover (*44*), and adaptive morphogenesis for flipper-leg repurposing in a turtle-inspired robot (*45*). Many compliant reconfigurable locomotion designs also take direct inspiration from animals (*46*). Robots inspired by the golden wheel spider move by cartwheeling to traverse sand dunes (*47*, *48*); others autonomously adapt their shape to confined spaces to overcome obstacles (*49*) or release energy stored in their frame to jump and overcome obstacles (*50*). This body of work validates the advantages of compliant reconfigurable robots. However, changes to the underlying passive properties and morphology are often limited such that the range of environments in which these robots can operate robustly remains low.

Drawing inspiration from the concept of reconfiguration observed in animals and plants (Fig. 1A), we propose that active reconfiguration of a robot must lead to a change in its underlying passive properties. We present a robot in which the active reconfiguration of the robot's morphology enables control over the robot's passive properties and the resulting autonomous behavior (Fig. 1B and Movie 1).

This robot was fabricated from elastic fiberglass rods, tethers, and electrically driven winches and wheels to form a meter-scale compliant robot. The robot structure leverages lens-like structures to control torsional stiffness, enabling the structure to reconfigure from a flat rover for driving to a spherical form with its wheels tucked inside. This was achieved by designing the energetics of the structure appropriately and ensuring that the reconfiguration space avoids undesirable twisting or buckling modes. The continuous reconfiguration space of the robot enables rapid online adaptation for different modes of locomotion, including driving, swimming, and rolling. It enables our robot, termed GOAT (Good Over All Terrains), to traverse complex terrains, including mountainous, aquatic, and urban environments. The robot's passive properties vary from highly compliant and adaptive in its flat configuration, enabling locomotion through adaptation to the terrain, to elastic and robust in the spherical configuration, resulting in rolling, falling, and dropping behaviors. By adapting its configuration and passive properties to the environment, the robot shows greater physical robustness and simplifies autonomy, enabling it to navigate autonomously with only positional information.

To demonstrate the potential of active morphological reconfiguration combined with compliance, we present an evaluation of both the active and passive capabilities of our robot. This evaluation highlights the robot's capacity for autonomous behavior with minimal perception systems, relying instead on its passive capabilities to interact with and respond to its surroundings (Fig. 1C).

CREDIT: PANGOLIN (*68*, *69*); SPIDERS (*70*, *71*)



Movie 1. Summary video of the GOAT. Overview of the robot and demonstration of its capabilities for active-passive morphological reconfiguration and terrain-adaptive locomotion in extreme outdoor environments.

Our experiments showcase the robot's proficiency by demonstrating it autonomously traversing more than 4 km of varied and challenging terrains, including snow, rocks, water, and ice. By using low-level decision-making and strategically choosing its morphological configurations, the robot can improve its locomotion efficiency and adeptness in navigating complex landscapes. This approach of incorporating active and passive reconfiguration demonstrates how environmental adaptation can minimize computational demands for path planning and enhance efficiency over challenging and extreme terrains.

RESULTS

Reconfigurable compliant design

For the robot to reconfigure from a wheeled driving configuration into a spherical form, its frame had to deform three-dimensionally, assume a spherical shape, and avoid twisting or buckling. Furthermore, the design of the frame had to support reconfiguration in a way that allowed control of the orientation and tucking inward of the wheels to form a fully spherical shape. To enable this reconfiguration, the robot's frame was fabricated from two elastic fiberglass rings, which were connected to form a circular structure with a woven pattern. Tendons span orthogonally across the frame and were connected to electric winches at the floating payload in the center of the robot. There was a trade-off between frame compliance and payload capacity; higher stiffness supported heavier payloads but reduced compliance and increased the energy needed for morphing. The selected frame stiffness represented a compromise between these factors to support the 1.8-kg payload and enable robust locomotion and efficient morphing. Figure 2A shows the robot's design, including its four rimless wheels, in its open form and its reconfiguration capabilities. By controlling the winch-driven tendons in a given actuation sequence, the robot was able to alter its shape in plane for efficient and robust locomotion and out of plane to compactly reconfigure into a sphere. In its spherical configuration, the payload was centrally suspended within the robot, and the wheels were folded inward toward the center of the robot.

The woven pattern of the elastic rods that form the robot's frame was crucial for achieving stable and unambiguous reconfiguration. Using finite element analysis, the deformation profiles, stability, and tendon forces were predicted for different frame designs. To model the elastic rods that undergo large deformation, we developed a modeling method on the basis of absolute nodal coordinates formulation (ANCF) (*51*) that offered accurate prediction of the three-dimensional (3D) deformation and was computationally efficient. Alongside the 3D deformation, our adapted ANCF methods enabled estimation of flexural rigidity, torsional rigidity, and elastic potential energy. This estimation enabled the nominal energy density distributed throughout the frame to be computed, which indicated the extent of the energy stored in different configurations.

Using this analysis and the estimation of the deformation, we determined the effects of morphing capabilities for different frame designs to demonstrate the importance of the woven lens shape of the frame. For the same winch placement and tendon routing, but with a ring made of a single rod, the frame structure exhibited stable in-plane and 3D, out-of-plane reconfiguration at low tendon forces (Fig. 2B). However, because the rod was free to rotate around its central axis, the wheel orientation was not constrained. Therefore, the wheels were not folded away when reconfiguring into a sphere. To constrain the wheel orientation during reconfiguration, the frame could be designed as a band structure composed of two fiberglass rings rigidly connected with constant separation. This configuration demanded considerably higher tendon forces for reconfiguration. In addition, during out-of-plane deformation, this band structure tended to become unstable, leading to a flipping of the band, which obstructed complete reconfiguration.

The frame of our robot was constructed from two fiberglass rings, interconnected to form a pattern of four symmetric, convex lenses between them. The wheels were positioned at the center of these lenses. During reconfiguration, this structure required similarly low tendon forces as observed in the single ring structure (Fig. 2B, iii). In contrast with the band structure, it allowed the frame to fully and unambiguously reconfigure into a sphere. In addition, the orientation of the wheels was constrained, ensuring that as the frame folded into a spherical shape, the wheels faced toward the robot's center. The frame of the robot allowed for continuous reconfiguration, with the possible reconfiguration space defined by the lengths of the two winch-operated tendons (movie S1). Certain tendon configurations in this reconfiguration space could lead to a twisting of the frame. By examining the torsion stiffness of the pivot point of the reconfiguring frame, we identified the likelihood of unconstrained, unstable, and undesired twisting motions (*52*). This metric was computed systematically across the reconfiguration space (Fig. 2C). When torsion stiffness was below zero, these structures showed unwanted and unstable twisting behaviors.

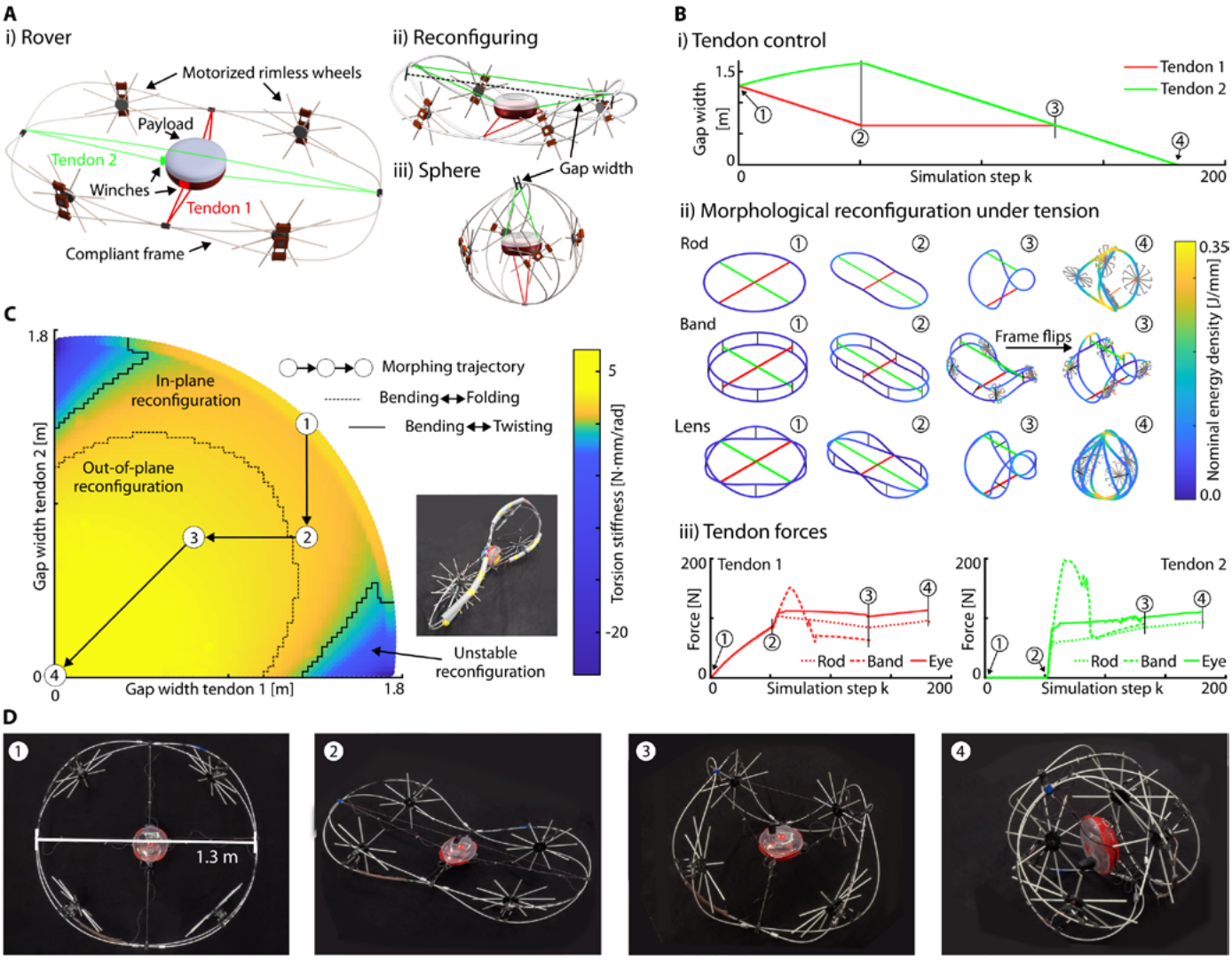


Fig. 2. Robot design and configuration analysis. (**A**) Compliant frame with four rimless wheels and actuated tendons enabling rover configuration (i) and reconfiguring (ii) to a sphere (iii) through tendon length adjustments. (**B**) For the tendon control sequence in (i), frames made from a single rod, a band, and our robot's lens pattern, experience morphological reconfiguration (ii) whereby different tendon forces occur (iii). (**C**) Morphological configuration space showing reconfiguration regions (bending, folding, and twisting), with the *x* and *y* axes representing the tendon gap indicated in (A). The black line illustrates a tendon control sequence to reconfigure from state 1 (circular) to 2 (rover), 3 (reconfiguring), and 4 (sphere). (**D**) Time-lapse sequence of the robot's progression through configuration states 1 (circular), 2 (rover), 3 (reconfiguring), and 4 (sphere). Labels 1 to 4 correspond to the same configurations in (B) (lens), (C), and (D).

Within the "safe" reconfiguration space, there existed three regions. First, in the outer region of the tendon reconfiguration space, in-plane reconfiguration occurred. This reflected a change from a circular shape to a rover shape suitable for driving or one that fits through gaps (Fig. 2D, markers 1 and 2). Second, when the tendon lengths were further reduced, out-of-plane reconfiguration occurred. The wheels of the robot lifted off the ground and folded inward when the frame took on a spherical shape. This continued until the frame was fully reconfigured into a spherical shape. Last, there was the unstable "twisting" region, which had to be avoided. When switching between different configurations, a path of tendon configurations had to be chosen that avoided unstable regions. In Fig. 2C, an exemplary path (1 to 4) for reconfiguring from a circular to rover to spherical shape is indicated. Figure 2D shows an exemplary time-lapse sequence of the robot's reconfiguration, following the control signals detailed in Fig. 2B and the morphological reconfiguration path outlined in Fig. 2C. Last, when keeping the total length of both tendons constant, the payload suspended in the robot's center could be moved independently of the robot's frame. This arrangement enabled the initiation of a rolling motion by moving the center of mass of the robot without altering the overall configuration.

Passive adaptation

The robot's two main configurations, namely, the spherical and rover shapes, allowed different forms of locomotion. The robot leveraged each configuration's specific passive properties to be resilient and robust in different environments (movie S2). In this section, we

illustrate these passive properties and how they are key to enabling autonomous navigation in the robot's two main configurations.

Flat configuration

In its flat configuration, the robot leveraged compliance in its frame to overcome obstacles. The robot overcame obstacles that were 14% higher than its center of mass with a success rate of 38% (Fig. 3A). Further, the compliance in the frame aided in traversing narrow gaps. The robot passively compressed to fit through openings smaller than its width, successfully passing through them if it had adequate traction and forward momentum (Fig. 3B and movie S3). To evaluate this ability, the success rate for the robot passing through gaps of reducing size was calculated. This demonstrated that the robot could successfully squeeze through gaps more than 10% smaller than its starting frame width.

Autonomous driving

In its rover configuration, the robot's compliant frame enabled it to robustly maneuver in diverse environments, overcoming obstacles like steps and driving on varying slopes and surfaces. This meant that the robot did not require detailed terrain analysis or complex path planning but could leverage onboard state estimation to navigate between waypoints. Using measurements from onboard sensors [global navigation satellite system (GNSS)–based position

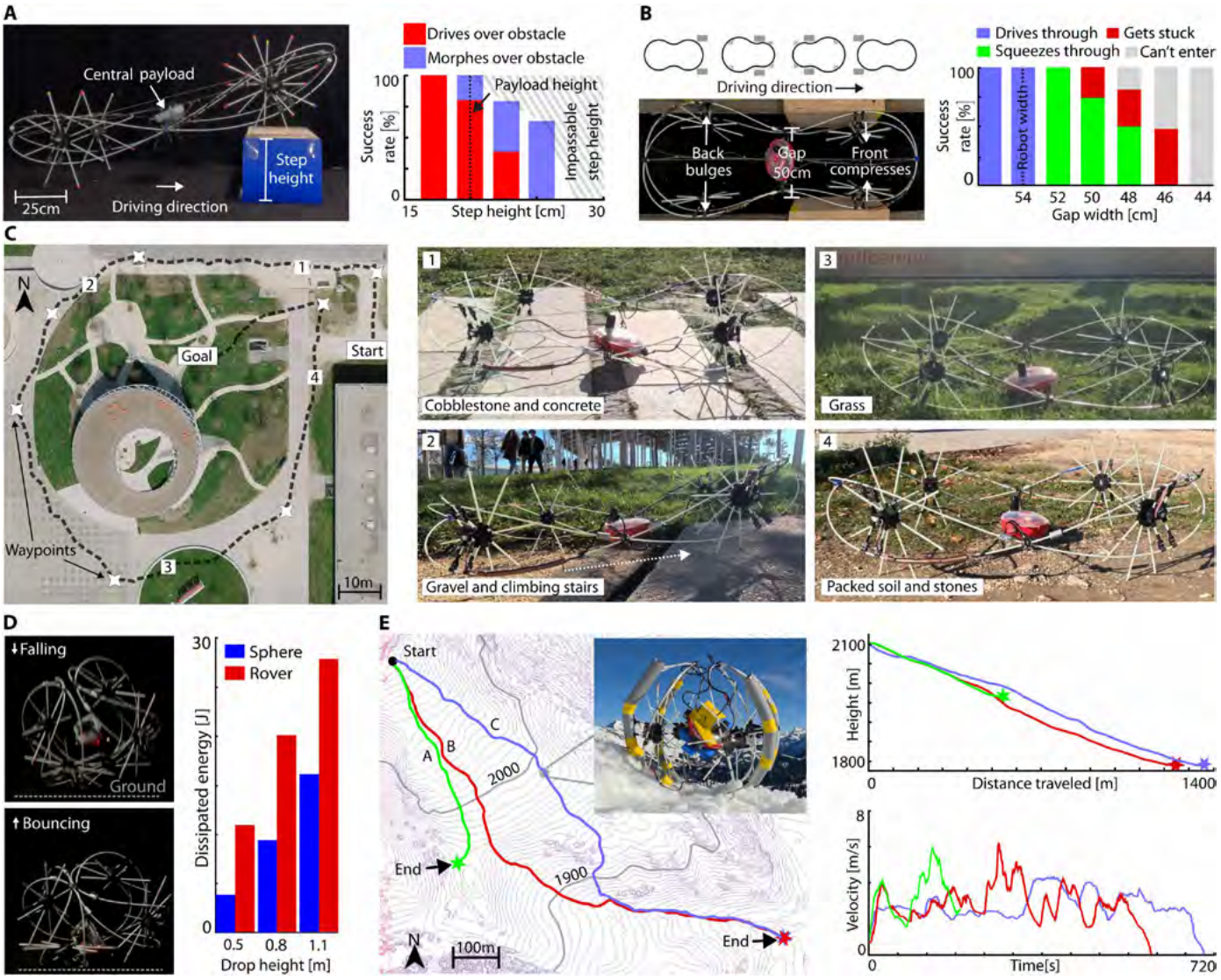


Fig. 3. Locomotion leveraging passive adaptation. (**A**) The robot drove over steps (17 to 26 cm high). When stuck on a step and reconfiguring, the robot can overcome higher obstacles. Its success rate to drive, respectively, morph over steps is shown. (**B**) The robot was commanded through small gaps (56 to 44 cm wide). For gaps narrower than the robot (54 cm), the compliant frame deforms and enables squeezing through. The success rate to get through gaps is shown. (**C**) Autonomously driven path around EPFL, Lausanne, Switzerland, guided by predefined waypoints. The numbered markers and accompanying images showcase the variety of terrains and structures encountered. (**D**) The pictures show the robot in spherical configuration dissipating impact energy through elastic deformation and bouncing. The dissipated energy of two configurations (rover and sphere) is shown for the robot being dropped from three different heights. (**E**) Three rolling trajectories for the robot in its spherical configuration from the same starting point on a peak. Contour lines indicate regions of equal altitude. The inset photo shows the rolling robot, highlighting the snowy mountainous terrain.

sensor, inertial measurement unit (IMU)] to provide a robust estimation of its location and heading, the rover autonomously followed a set of predefined waypoints, where a low-level path-following controller enabled navigation between them. This enabled the robot to autonomously traverse complex and varied terrains, as shown by the 307-m path followed in Fig. 3C, where the robot followed eight predefined waypoints. The low-level autonomy and physical compliance of the robot enabled it to traverse over various challenging terrains with minimal planning. Furthermore, the planning and low-level control algorithms did not need to be specific for different terrains or topographies.

Spherical configuration

The spherical configuration of the robot enabled it to cover diverse and challenging terrains, which could include substantial drops, for example, from cliffs. Because the payload was suspended centrally within the frame, the spherical configuration not only provided physical robustness to the robot frame but also protected the easily damaged payload of the robot. When the robot was dropped from a height, the frame elastically absorbed the impact forces and gradually dissipated them. Initially, it deformed before the elastic deformation in the frame led to the robot bouncing and restoring its initial configuration. This process effectively protected both the payload and the frame from damage. To demonstrate the effectiveness of the spherical configuration, we compared the energy that was dissipated upon impact by the robot in the spherical configuration with that in the flat rover configuration. This energy was measured by observing the rebound height of the robot after drops from varying heights using motion capture. In the rover configuration, impact energy was dissipated by both the robot's frame and the payload upon hitting the ground. In contrast, in the spherical configuration, a portion of the kinetic energy was converted into elastic deformation upon impact, resulting in substantially less energy being absorbed by the frame and none by the payload. This effectively shielded the payload, given that all the energy was dissipated by the frame (see Fig. 3D). Although the impact energy to be dissipated increased linearly with drop height, the spherical configuration's elastic deformation reduced the total energy that had to be dissipated upon impact by the frame by approximately half compared with the rover configuration (Fig. 3D). The robot's capacity for efficient and gradual energy dissipation in the spherical configuration enabled the robot's frame to undergo substantial drops when rolling without damage, including drops of more than 10 m, as visualized in fig. S9. When planning a route for the robot in this configuration, this meant that extreme features such as cliffs or drops did not need to be avoided but could be leveraged to offer an optimally efficient path.

Rolling autonomy

This physical robustness allowed the spherical form to be leveraged for rolling. The robot could self-instigate rolling, as described in the spherical reconfiguration section. Once rolling, it could follow the surrounding environmental gradient autonomously without any control inputs. When leveraging this functionality during navigation, we refer to it as environmental computation. An exemplary visualization is shown in movie S4. Although the path followed depends on the terrain and topology, substantial distances were traveled. The rolling capabilities were tested in a snow-covered mountain environment (Villars-sur-Ollon, Switzerland). Rolling was instigated three times in the same location, and the rolled trajectories were recorded for three repeats (Fig. 3E). When rolling, the robot reached velocities up to 6 m/s. In two instances, the robot covered a distance of 1.4 km. During the top part, when rolling along a mountain ridge, the trajectories diverged but converged to the same end point because of the topographic features of the surrounding terrain. It was not intended for the robot to end up in the same location for the rolled trajectories but to capture the possible variability when rolling in a complex terrain. In another case, the robot rolled to a local topographic minimum and ended in another location. Thus, when rolling, the robot could follow a gradient and terrain-driven path, which could be beneficial for rapidly and efficiently descending slopes.

Active reconfiguration

In addition to the robot leveraging its passive properties in different configurations to simplify autonomy and providing different means of locomotion, the robot could also leverage active reconfiguration to achieve additional modalities of locomotion. Active reconfiguration—such as adjusting its aspect ratio for optimized speed and turning or curling up and redistributing its payload to facilitate rolling—allowed it to adapt to various environments. Integrating these capabilities into autonomous navigation strategies enhanced the robot's adaptability, speed, and energy efficiency on complex terrains.

Flat reconfiguration

When in flat configuration, the wheels of the robot were in contact with the ground, and it could drive. Within this configuration, the lengths of the two orthogonal tendons, l_1 and l_2, were varied to change the aspect ratio of the robot from a circle (aspect ratio, $r = l_1 / l_2 = 1$) to a narrow figure-of-eight form (aspect ratio, $r = 6.5$) as shown in Fig. 4A. The robot's aspect ratio affected the achievable speed and drift when driving and turning. This was evaluated by recording the trajectories for forward driving and turning when the robot was in different aspect ratios. The fastest speed (1.4 m/s) was achieved when the aspect ratio was such that the wheels were aligned with the direction of travel, for an aspect ratio between 2 and 3 (Fig. 4B). The speed then decreased with the two extremes of aspect ratios. The positional error, or percentage drift when moving forward, showed a similar relationship, with the best performance, meaning the lowest positional error, seen when the wheels were aligned with the direction of travel ($r = 2.2$). When turning, the angular velocity was highest when $r = 1$ and the robot was closest to a circle (Fig. 4C). As the ratio increased, it was increasingly difficult for the robot to turn. Eventually, turning became impossible when the wheels were turned inward for $r = 6.4$. The turning ratio similarly increased with aspect ratio. A ratio of 1.4 to 2.2 offered a compromise in terms of forward speed and ease of turning. However, depending on the path of the robot and the turning radius required, morphing within the flat configuration could aid in speed, controllability, and maneuverability of the robot.

Spherical reconfiguration

When in the spherical configuration, the payload of the robot was suspended on the tendons within the robot. By maintaining the same total tendon length, but varying the ratio between tendons 1 and 2, the payload could be moved along the tendons besides maintaining the identical spherical configuration of the robot. Because the payload was substantial with respect to the lightweight frame, this could be used to move the center of mass outside the support polygon of the robot and instigate rolling motion (Fig. 4D and movie S5). Using the encoders on the winches, this operation was automated and took less than 15 s. Using motion capture, the rolling motion of the sphere was recorded for repeated trials on flat ground. First, initial perturbations were observed. After reaching a "tipping"

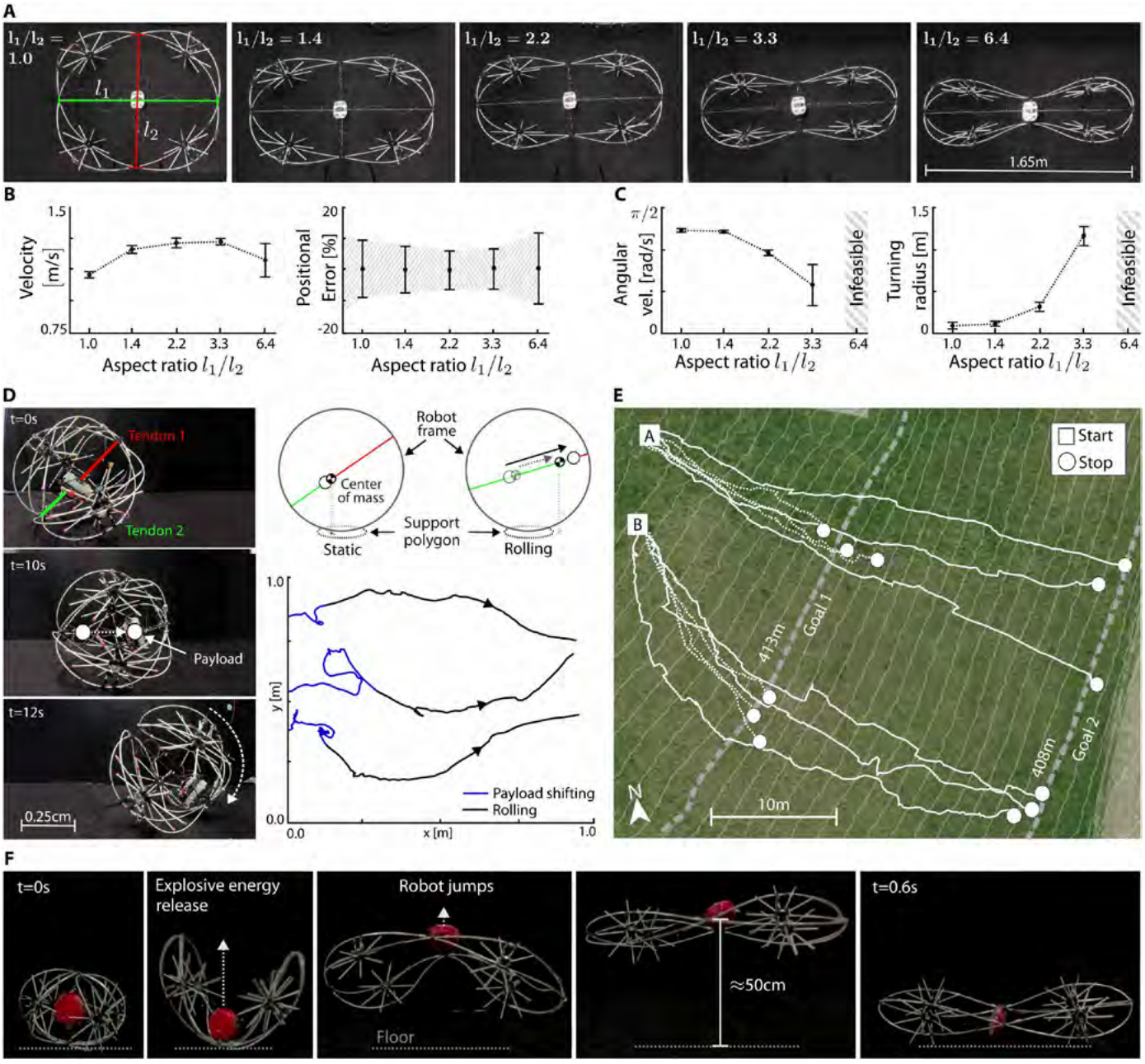


Fig. 4. Active reconfiguration for motion control. (**A**) The ratio of tendons 1 and 2 allows reconfiguration of the robot's aspect ratio. Pictures show a time-lapse sequence of the robot reconfiguring from a circle, $l_1 / l_2 = 1.0$, to a figure eight, $l_1 / l_2 = 6.4$. (**B**) The aspect ratio affects driving velocity and odometry accuracy. The mean and SD for velocity and positional error for five trials per aspect ratio are shown. (**C**) The robot's turning speed and radius are linked to its aspect ratio. The mean and SD for five trials per aspect ratio are shown. (**D**) The winch-driven tendons can be used to move the payload and, hence, the center of mass to initiate rolling on flat ground. The motion of the sphere when rolling is shown for three repeats as measured by motion capture with markers tracking the payload. (**E**) The robot rolled from two starting positions, A and B, and automatically reconfigured from sphere to rover to stop at two different targets, goals 1 and 2. (**F**) Time-lapse images showing the motion of the robot when a tendon is promptly released.

point, the robot rolled smoothly (Fig. 4D). The direction of rolling was determined by the motion of the payload. Rolled distance and path were approximately constant and corresponded to half of the circumference of the robot. The ability to shift its center of mass enabled the robot to automatically initiate rolling on flat or close-to-flat slopes.

Reconfiguration from sphere to rover

After instigating rolling, reconfiguring to a rover enabled control over the distance traveled (movie S6). To illustrate the repeatability of this process, for two different starting points on a grassy slope, reconfiguration from sphere to rover was instigated after a given

time to reach two target goal positions. Figure 4E shows the rolled paths and the stopping locations for the goal distances. Although there were some variations, a repeatability of ±2 m was observed. The robot's ability to actively stop rolling through morphological reconfiguration enabled the integration of controlled rolling as a locomotion modality into autonomous navigation strategies.

Explosive reconfiguration from sphere to rover

Transitioning from spherical to rover shape using the robot's electric winches enabled controlled stopping; however, this process took some time, limiting the robot's response time to a stopping signal. When in the spherical configuration, the fiberglass rods stored elastic potential energy, applied by the winches. Leveraging clutches on the robot's winches enabled a rapid release of the tendons, and the stored energy could be released explosively (movie S7). This rapid release of energy when the robot was in a spherical configuration led to a jumping behavior of the robot, where heights of 0.5 m could be reached in less than 0.3 s (Fig. 4F). This offered an effective maneuver to overcome obstacles, escape challenging situations, or come to a more immediate halt.

Navigation leveraging reconfiguration

The robot's capacity for active morphological reconfiguration enabled the combination of its different locomotion modalities for efficient autonomous navigation. Three navigation strategies, schematically depicted in Fig. 5A, were considered as a means of reaching a target location mid-hill: (i) driving directly to the target; (ii) rolling until passively stopping followed by driving to the target; and (iii) rolling, actively reconfiguring to rover when at the same height as the goal, and driving to the goal. Our path planning algorithm did not have prior information about the surrounding terrain available. A flow chart visualizing the decision-making for autonomous reconfiguration for each of the above navigation strategies is shown in fig. S5. The navigation strategies were evaluated on the basis of the robot's ability to reach a target location on a grassy slope (Fig. 5B). For each strategy, metrics including the path taken, time elapsed, and energy consumption were measured (Fig. 5C). Driving directly to the target resulted in the shortest overall path of 22 m. In comparison, rolling down until naturally stopping and consecutively driving uphill toward the goal resulted in a path 2.5 times longer than driving directly and required more time and energy than the other strategies. The surplus of energy was required because driving uphill consumes more energy than driving downhill. Rolling, actively stopping by reconfiguring to rover, and driving toward the goal took less time than the other strategies despite the distance traveled being slightly larger than driving directly toward the goal. Critically, we observed that the combination of rolling, active reconfiguration, and driving reduced the energy consumption by a factor of four in comparison with driving directly. This demonstrated how the robot's capacity for active reconfiguration allowed it to leverage the advantages of each configuration to increase navigation efficiency.

Real-world deployment

By leveraging the robot's reconfiguration capabilities appropriately, the robot could operate in and transition between diverse natural environments. Two example real-world deployment scenarios are shown, demonstrating how morphological reconfiguration facilitates seamless access and maneuverability in various environmental contexts.

Multiterrain operation: Land, water, and rocks

Transitioning from locomotion on land to surface swimming in water and traversing rocky terrain was challenging because of the different locomotion modes required and the obstacles present. The robot's ability to operate in these environments was tested on the lakefront of Lake Geneva in Lausanne, Switzerland, with a sandy, stony beach cluttered with flotsam and a surrounding wall of boulders. The robot started driving on the beach, heading to an initial waypoint in the water, and then returning to the beach and finally reaching the goal position by traversing a wall of boulders. We combined autonomous and teleoperated operation. Using the same controllers for path following, the robot could automatically transition from the beach to the lake (Fig. 6A and movie S8). Leveraging its wheels to generate thrust, the robot could swim and reach a specified waypoint. In water, the robot showed maneuverability leveraging its four-wheel differential drive. Its velocity in water was constant at 0.5 m/s, a speed comparable to that of locomotion on land. However, the power consumption when swimming was 56% lower and far more consistent. The robot could autonomously transition from water to land, leveraging its open wheel structure. The third terrain to traverse was a wall of boulders. Although the robot could gain some traction to move over and among the boulders, it got stuck. To get unstuck, overcome the boulders, and return to the lake, the robot was commanded to automatically take on its spherical shape. It rolled back into the lake (Fig. 6A). Once again floating, the robot switched back to its rover configuration and swam to the final goal

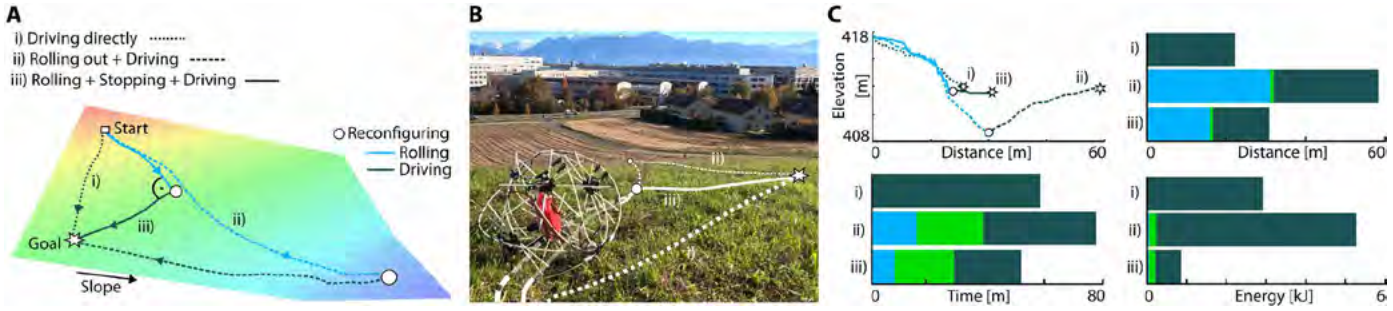


Fig. 5. Autonomous navigation strategies. (**A**) Schematic visualization of three strategies for autonomous navigation: (i) direct driving; (ii) rolling, passively stopping, and driving; and (iii) rolling, actively stopping through reconfiguration, and then driving toward the goal. (**B**) An image of the robot in the experimental environment. The starting point and the trajectories across the sloped terrain for the different navigation strategies (i to iii) are illustrated. (**C**) Comparative data analysis showing the elevation change, distance traveled, time passed, and energy consumed during driving (dark teal), reconfiguring (green), and rolling (blue) for each navigation strategy (i to iii).

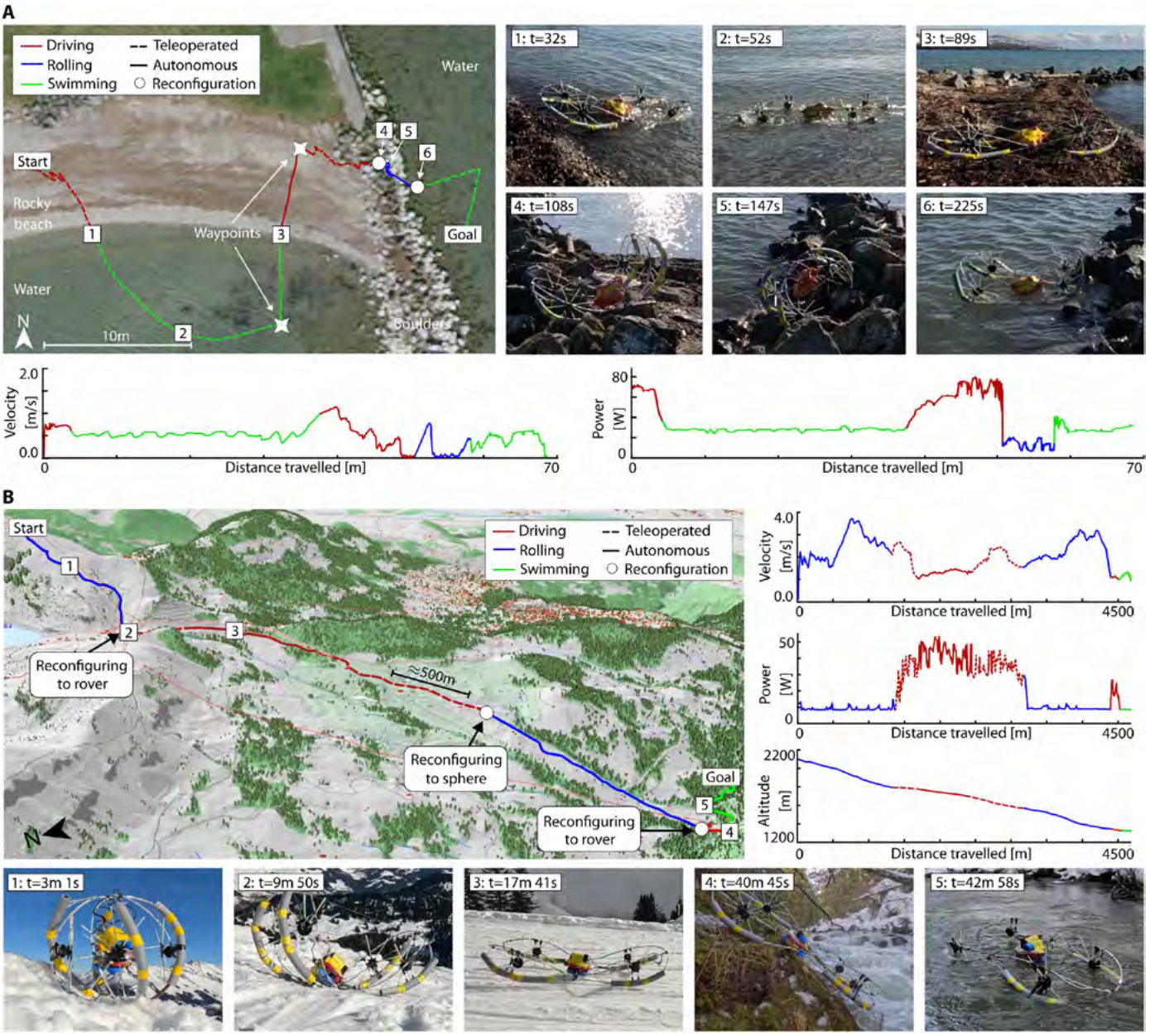


Fig. 6. Multimodal locomotion experiments in extreme outdoor environments. (**A**) Demonstration and evaluation of the robot using active morphological reconfiguration to follow a path over a flotsam-cluttered beach, through water, and across boulders. Pictures show the robot at different points on its path. The velocity profile and power consumption were evaluated in relation to the distance traveled. (**B**) Multimodal navigation in a mountainous environment, with the path taken shown and the velocity, altitude profile, and power consumption evaluated in relation to the distance traveled. Pictures highlight the robot operating in different locomotion modalities (images 1, 3, and 5), reconfiguring from sphere to rover (image 2), and transitioning between environments (image 4).

location in the lake. The total distance traveled was 70 m whereby the robot covered 40 m swimming, 25 m driving, and 5 m rolling. The robot's compliance facilitated seamless navigation across varied terrains and smooth transitions between them because its capacity for active morphological reconfiguration allowed it to effectively overcome obstacles.

Operation in a mountainous environment

The robot was deployed in the varied mountainous landscape around Villars-sur-Ollon (Vaud, Switzerland), tasked with traversing snow-covered slopes, rugged mountains, streams, and rivers to reach a destination 4.5 km away from its starting location (Fig. 6B). This scenario required the robot to engage in driving, rolling, and swimming locomotion. The robot was partially teleoperated and partially autonomous on this path, operated with a single battery charge, and changed its configuration three times. For the first section of the path, a steep slope, the robot rolled down the slope in its spherical configuration, reaching speeds up to 4 m/s; the ability to follow the environment enabled a meaningful path to

be autonomously followed by the robot. During this 10 min of rolling, the only energy consumed was by the robot's onboard computer, which captured data and ran the robot's software stack. This offered a reduction by almost 90% in power consumption compared with its driving configuration. In the subsequent 1.8-km-long flat section, the robot automatically transitioned into its rover configuration and autonomously followed a series of commanded waypoints. The robot drove up to 2.3 m/s and consumed on average 39 W. When reaching a second steeper part, the robot was commanded to reconfigure back into a sphere and leveraged its energy-efficient rolling mode. During this experiment, more than 4.5 km of distance was covered, and around 1000 m of height was descended. Over this 4.5-km traverse, the robot did not experience any failures, nor did it get irrecoverably stuck. Last, a river flow was leveraged to reach the final destination. The robot reconfigured into a rover to enter the stream, after which it leveraged the river flow, requiring minimal power to reach its goal location. Through reconfiguring, the robot leveraged the environment for control and minimized energy consumption.

Comparative metrics

The performance of the robot was compared with that of other robots designated for operating across diverse outdoor environments using several metrics as presented in Table 1. We first focused on key metrics for terrestrial locomotion. These included efficiency metrics, such as the cost of transport (CoT), total weight, and distance per charge, and performance metrics that reflected outdoor behavior, including the maximum slope that could be ascended, maximum drop height, maximum velocity on flat ground, and the payload capacity. Further, we compared performance and efficiency metrics for locomotion in aquatic environments, such as distance per charge, CoT, and maximum velocity where possible. The energy efficiency of GOAT, as represented by its CoT, was higher when compared with that of other robots for terrestrial locomotion. The rolling locomotion mode, when rolling downhill, showcased GOAT's efficiency, being 10 times more efficient than ANYmal (*53*), 26 times more efficient than M4 (*28*), and 100 times more efficient than ART (*45*). In its driving configuration, the GOAT's CoT was approximately 30% lower than that of ANYmal. Weighing only about $^{1}/_{10}$ of ANYmal, the GOAT was designed for operational needs with a reduced payload capacity. Unlike ANYmal, the GOAT could be easily transported in a backpack (see fig. S14). We included ANYmal in the comparative metrics because it is one of the few robots for which comprehensive outdoor test data over long distances in rugged, varied terrain have been reported in the literature (*3*). In aquatic environments, the GOAT robot's CoT was 3.3 times lower than ART's most efficient mode. ANYmal and M4 were infeasible for aquatic locomotion. In terms of distance covered on a single charge, the GOAT traveled 4.5 km, similar to the ANYmal robot covering 2.2 km and M4's estimated maximum range of 2.4 km. The GOAT robot could swim up to 2 km per charge. For ART, no maximum range was reported because it was operated tethered. For terrestrial locomotion, the GOAT robot reached a competitive maximum slope ascending capability of 33° compared with ANYmal's 21° and the thruster-assisted M4's 45°. It achieved a maximum driving velocity of 2.4 m/s, twice that of ANYmal's reported 1.2 m/s, and was robust to drops exceeding 10 m. The GOAT robot's maximum swimming velocity of 1.2 m/s was substantially higher than ART's (0.09 m/s). Diverse environmental adaptation strategies were used by the compared robots: ANYmal relied on precise sensing, planning, and control algorithms; M4 exhibited locomotion plasticity and had the capability to fly for short periods; ART applied adaptive morphogenesis for transitioning between land and water. Our robot optimized navigation and energy usage through coordinated

Table 1. Comparative analysis of robotic locomotion performance in diverse terrains. Side-by-side comparison of nonportable and portable field robots, including ANYmal, M4, ART, and our GOAT, highlighting key performance metrics, such as distance per charge, CoT, weight, and maximum velocity in both terrestrial and aquatic locomotion, as well as capabilities in environmental adaptation.

		Field robots (>10 kg)	Portable field robots (<10 kg)		
		ANYmal (*53*)	**M4 (*28*)**	**ART (*45*)**	**GOAT (ours)**
Terrestrial locomotion	Distance per charge (km)	2.2	0.24–2.5*	Tethered	2.5 (rolling) + 2 (driving) = 4.5
	Cost of transport	1.2	2.6–27.0*	10.6 to 30.0	0.1 (rolling)
					0.8 (driving)
	Weight (kg)	30	5.6	9	2.8
	Payload (kg)	10	1.0*	No data	1.6
	Max. slope climbing (°)	21	45	No data	33
	Max. velocity (m/s)	1.2	0.7*	0.04	3.8 (rolling)
					2.4 (driving)
	Max. drop height (m)	No data	No data	No data	10
Aquatic locomotion	Distance per charge (km)	Infeasible	Infeasible	Tethered	2
	Cost of transport			3.0–4.3	0.9
	Max. velocity (m/s)			0.09	1.2
Environmental adaptation	Strategy	Planning and control	Locomotion plasticity	Adaptive morphogenesis	Active reconfiguration and passive adaptation

*Estimated values, as extracted from raw data.

interactions with its environment by leveraging the passive properties of its compliant structure and actively reconfiguring its morphology under the influence of environmental stimuli.

DISCUSSION

In this work, we present a robot that is capable of active morphological reconfiguration and passively adapting to its environment. The bioinspired morphological reconfiguration enables different locomotion modalities, including driving, swimming, and rolling. Moreover, the robot can dynamically reconfigure its shape while in motion, for example, when rolling, allowing it to adapt instantaneously to changing paths and terrains. The compliance of the robot's structure offers local adaptation to obstacles in highly unstructured and uncertain environments such that complex terrains can be covered autonomously with minimal sensing and path planning. Furthermore, in different morphological configurations, the efficiency, feasible paths, and velocity of the robot are varied, opening up a wider range of environments and terrains for navigation. In doing so, with GOAT, we demonstrated kilometer-long distances of efficient travel across many different challenging terrains. The integration of active reconfiguration with structural compliance, as demonstrated in this work, offers the potential to substantially enhance the ability of future robots to efficiently and autonomously navigate complex and challenging environments.

To increase the robot's autonomy, future research in path planning should focus on creating algorithms that account for the unique capabilities of reconfigurable, compliant robots. Such algorithms should use a priori available information, such as topographic maps, to plan paths that leverage the varied abilities of these robots. Planned paths should include shape reconfigurations during rolling and actively leverage environmental features through actions like falling. The robot's compliant properties and its capacity for dynamic reconfiguration should also be integrated and leveraged by path-following controllers during autonomous navigation for increased performance and robustness. These algorithms could distribute the planning process and offload some computational tasks to the robot's physical form, enhancing the robot's efficiency and ability to follow more complex trajectories (*54*).

The proposed robot design shows promise for physical scalability, both larger and smaller, by adjusting the torsional and bending stiffness of the rods used. The payload capacity can be substantially enhanced by constructing the frame using rods or strips, with customized sections and reinforcements (*55*), and using materials with varying stiffness or structures for controllable stiffness (*56*). Adjusting the stiffness and scale of the frame allows for the exploration of the trade-off between payload capacity and compliance, which in turn influences the robot's morphing process, locomotion performance, and robustness. Optimal compliance not only is linked to the robot's design but also depends on environmental factors such as terrain type, obstacle height, and slope inclination. Computational design could be leveraged to optimize design parameters of the frame, such as scale and stiffness, or of the wheels, such as the number of spokes and their stiffness, for specific environments. The use of elastic fiberglass rods, winch-driven tendons, and electric motors provides a scalable fabrication method for morphing robots. Using these widely available materials allows for rapid design iterations and scalability. This approach could lead to a new generation of elastic, reconfigurable robots that combine structural compliance with active, dynamic, and substantial reconfiguration, capable of robustly navigating complex natural terrains with unprecedented versatility.

Harnessing a combination of passive adaptation and active morphological reconfiguration, the next generation of compliant robots could surpass nature's versatility. By varying their shape while in motion, these robots can dynamically adjust to environmental changes in real time. The potential for these robots to swiftly deploy into uncharted terrains with minimal perception and planning systems substantially enhances their applicability, turning environmental challenges into computational assets in the near future.

MATERIALS AND METHODS

Morphologically reconfigurable frame design

The robot's frame was constructed from four straight fiberglass rods, each 2 m in length and 5 mm in diameter. Fiberglass rods provide a high strength-to-weight ratio and can show substantial elastic deformation before permanent deformation or breakage occurs (*57*). The fiberglass rods were rigidly connected to form two fiberglass rings with a circumference of 4 m each. The two circles were stacked and connected at eight equidistant points on the circumference of each circle. Four of these joints connected both circles with an angle between them. They served as mounting points for the tendons and allowed the structure to bend and fold. Four others were rigid connections and enforced the convex symmetrical lens structure that allowed unambiguous morphological reconfiguration into a sphere and provided mounting points for the wheels. Each rigid connector was composed of two chevrons, the ends of which were attached to both circles using fixed T joints. Each chevron was formed by two 8-cm-long fiberglass rods that were connected in the middle by a fixed 3D-printed connector with an opening angle of 120°. This combination of rigid connections that enforced geometry in the frame enabled morphological reconfiguration.

Four motors were mounted to the fixed middle joints of each rigid connector's chevrons. The motors were waterproof and could supply 45 kg cm^{-1} of torque. Each motor drove a rimless wheel, which was fabricated from a central 3D-printed hub that held 10 spokes. The spokes were 25 cm long and made from 8-mm-thick fiberglass rods. The number and stiffness of the spokes were chosen heuristically, with 10 relatively stiff spokes providing a good balance between smooth motion and traction across various terrains. The spoke length was chosen to be as large as possible without intersecting or touching the robot's frame when rotating.

A waterproof payload in the middle of the frame was connected to the frame of the robot by two tendons from 0.6-mm braided polyethylene fishing line. The tendons spanned orthogonally over the frame, with one tendon running above and the other below the central payload. Each tendon was guided from the central payload through an eyelet on one connector of the robot frame to an eyelet on the diametrically opposite connector and back to the central payload, where they were connected to two electric winches. These two winches could each produce 40 kg cm^{-1} of torque and enabled the lengths of each of the tendons to be independently controlled. Both winches were retrofitted with encoders to measure the absolute angular position of their spools. Under slight tension, the two tendons held the payload of the robot in its center. Under higher tension, the tendons caused the robot's frame to deform. This property was key to enabling the robot to actively reconfigure its morphology.

Modeling morphological reconfiguration

The role of modeling and simulation was twofold: first, to optimize the geometry of the robot for spherical reconfiguration and, second, to analyze the robot's stability in different reconfigured states. Typically, models for simulating compliant morphing structures and robotic bodies are based on deformation parameters, such as piecewise constant curvature or piecewise constant strain. However, these models are suited for the forward simulation of slender structures, such as soft continuum robot arms. Our robot was constructed from elastic rods capable of large deformations and subject to complex loading conditions, so these models were not appropriate. Instead, we developed a modeling approach based on ANCF (*51*, *58*), which accurately predicted 3D deformations besides ensuring computational efficiency. This approach was used to simulate the 3D shape of the deformed robot with different rod geometries undergoing displacement from the two winch-driven tendons. In addition to the 3D shape, three other key metrics were determined: the robot's flexural rigidity, torsional rigidity, and elastic potential energy. The flexural rigidity and the nominal energy density distributed through the frame were important to assess the energy required by the winches to deform the structure and the forces in the tendons. These needed to be kept low so that the reconfiguration remained within the capabilities of the winches. Torsional stiffness was key to understanding the robot's stability during its reconfiguration process. A joint's torsional stiffness dropping below zero could lead to undesirable and unstable twisting movements.

ANCF modeling for elastic rods

We briefly introduce our adapted ANCF modeling method for elastic rods. An elastic rod was modeled as n discrete elements, described by $n + 1$ nodes along its axial center line; this reference frame and parameterization is detailed in Supplementary Materials and Methods. Given this parameterization and the absolute nodal coordinates (which relate to the central axis of the rod), we obtained the displacement field for the entire rod structure. The axial Green strain ε, bending curvature k_b, and torsion k_t were calculated from the displacement field at each node to obtain the energy density distribution along the rod. Integrating along the length of the rod structure allowed us to obtain its elastic potential energy. The generalized nodal force was obtained from the partial differential of the elastic potential energy across the nodal coordinates. From this, we solved a pseudodynamic equation to resolve the quasistatic state that is the configuration of the central axis of the rod. This formed the basis of the simulation. More information can be found in the Quasistatic solution section in Supplementary Materials and Methods.

Robot simulation

The ANCF method was used to build a complete simulation for the robot. This was used to simulate different geometries of robot structures undergoing different tendon displacements as shown in Fig. 2B (ii). The 3D configuration of the robot could, in turn, be used to obtain the tendon forces (Fig. 2B) from the nodal forces in a quasistatic state based on the balanced pseudodynamic equation of the structure (eq. S10 in Supplementary Materials and Methods). The tendon forces provided insight into the energetics and stability of the robot. To simulate different robot geometries undergoing different forms of reconfiguration, such as in-plane or out-of-plane reconfiguration, we had to identify the necessary boundary conditions and connection constraints for each scenario.

Quarter-robot simulation: Dual-symmetric, no-twisting configurations

For in-plane and out-of-plane deformation, when no twisting occurred, there were two planes of symmetry (X-Z and Y-Z) in the robot, so only a quarter had to be simulated; the definition of these axes and planes is given in figs. S2 to S4. For each of the three robot structures evaluated (rod, band, and lens), their coordinate system and constraints had to be designed appropriately.

Single rod. For in-plane deformation of a single-rod structure robot (Fig. 2B, ii), we modeled it as a quarter of a circular rod where one end was fully constrained and the other was moved in the x axis by the given tendon displacement (table S2). For out-of-plane deformation, one end was moved by the given tendon placement in the y axis, and the other end was free to move in the z axis (table S3).

Band. The band structure was simulated as two circular rods connected with orthogonal transversal rods. To represent these physical constraints, we introduced a new node, which was equidistant between the two connected rods and rigidly constrained to lie between them (fig. S3A), as detailed in the "Band structure" section in Supplementary Materials and Methods. The boundary conditions given for the single rod structure were also applied to this new node (fig. S3).

Lens. To represent the structure of the lens with the crossed points at the end of each quarter, we introduced a new node that was rigidly constrained to the two rods but had the average orientation of the two (see the "Lens structure" section in Supplementary Materials and Methods). The standard boundaries as detailed above could then be applied to these new nodes (fig. S4).

Half-robot simulation: Torsional stiffness

Under most tendon displacements, the single rod circle or lens structures had a stable dual-symmetrical configuration. However, when the ratio between the longer (major) axis and the shorter (minor) axis was too high, the structure became unstable and twisted into an antisymmetric complex configuration to reach a lower elastic potential energy state. This twisting was undesirable because it led to a "tent-folding"–like deformation from which it could not easily be recovered. To analyze the occurrence of this twisting, the torsional stiffness of the pivot point along the minor axis was predicted and used as an indication of the stability of the structure.

The simulation of the torsional stiffness was performed using a half-circle structure spanning the minor axis of the robot (fig. S5), where the position and local frame of the two end points of this half circle were coupled on the basis of its antisymmetrical geometry (see table S6). The middle point of the half rod was constrained in the z axis. To evaluate the elastic potential energy with increasing torsional deformation, the x axis of the local frame was moved stepwise in the X-Y plane, simulating changing tendon lengths. More details about boundary conditions are given in table S6 and the "Stability analysis" section in Supplementary Materials and Methods.

Using this half-circle simulation, the potential elastic energy U_E and the twisting angle θ of the structure were determined. The relationship between these was captured with a higher-order polynomial, where the concavity determined the stability of the structure (figs. S6 and S7). Specifically, the second derivative of this function, $\frac{\delta U_E^{\,2}}{\delta \theta^2}|_{\theta=0}$, defined the torsional stiffness along the minor axis of the robot and was used as an indication of the structural stability. This

was calculated by fitting a function between U_E and θ for only a small range of θ between 0° and 2°. This torsional stiffness was computed for different in-plane and out-of-plane configurations and mapped back to the tendon lengths to enable the stability analysis given in Fig. 2C.

As shown in Fig. 2C, this unstable twisting behavior could emerge from different tendon configurations and could lead to in-plane or out-of-plane twisting. The occurrence of these two different twisting behaviors is described in the following paragraphs.

In-plane twisting. In-plane twisting occurred when starting from an in-plane configuration with a high aspect ratio, meaning that the tendon in the *x* axis was short. When this was extreme, the robot structure took on the lowest energy state, which introduced twisting with two antisymmetrical axes (*z* axis and *y* axis), as shown in fig. S6. The shorter the *x* axis tendon, the greater the twisting angle at the midpoint and the lower the stability. As such, its stability decreased with the shortness of the distance along the *x* axis (the minor axis).

Out-of-plane twisting. An out-of-plane configuration emerged when there was an unstable in-plane configuration that had a tendency to twist, followed by a tendon displacement that decreased the distance along the *y* axis (the major axis). Here, its stable twisting angle would decrease until the dual-symmetrical configuration reached the lowest energy state, as shown in fig. S7. This meant that configurations with a shorter distance along the *y* axis (the major axis) showed higher stability in the dual-symmetrical configuration.

Full-robot simulation: Tent folding deformation

Although simulation of the half circle of the robot was used to identify when twisting happened, by simulating the whole robot structure, we could analyze the full 3D-folded unstable out-of-plane configuration (fig. S8). The boundary conditions for this full circle of the structure are defined in table S6. For this simulation, the start and end points of the rod were the same and fully coupled. The distance between the one-quarter point and three-quarter point along the circular rod was fixed to represent the constant length of tendon 1. More details can be found in the "Stability analysis" section in Supplementary Materials and Methods. When simulating the reduction in the length of tendon 2, which was modeled as reduced distance along the *y* axis, the unstable dual-symmetrical configuration resulted in the tent folding configuration as shown in fig. S8 and movie S9.

Electronics

All actuators, sensors, computing, and power electronics, except the motors that drove the robot's four wheels, were housed within the robot's central payload. A diagram of the robot's electronic components is shown in fig. S11. The central payload served as a hub for both power and data communication. The wheel motors received their power through cable conduits that run from this central payload. These conduits extended from the motors, running along the robot's frame and tendons, and converged at the central payload. The robot's actuation system consisted of its four wheel motors and the two winch motors. All motors were compatible with direct power supply from three cell lithium polymer (3S LiPo) batteries. Typically, the robot was powered with a 1600-mAh 3S LiPo battery because this battery provided a good trade-off between operational weight and the robot's designated operation period of approximately 30 min of continuous driving or 2 hours of rolling. Power management, crucial for the robot's functionality, was handled by a Holybro PM02D power module. This board regulated the battery's output, stepping down the voltage from 12 to 5 V, ensuring efficient energy distribution to power the robot's central processing unit and its sensors.

Sensors

The robot's central processing unit, a Raspberry Pi 5 (8 GB), established connectivity with various peripheral electronics essential for operation via the Inter-Integrated Circuit (I2C) protocol. This included an Adafruit PCA9685 16-channel pulse-width modulation (PWM)/servo driver, which generated PWM signals for controlling the robot's winches and motors. Angular measurements of the winches were captured by two AMS AS5048B rotary position sensors. Precise acceleration, angular rate, and magnetic field measurements were obtained by an Adafruit LSM6DSOX + LIS3MDL 9–degrees-of-freedom (DoF) IMU. Last, a Texas Instruments INA220 current and power monitor, which was integrated into the power module, tracked the robot's power consumption metrics. For high-precision location tracking, the system integrated a Septentrio MosaicHat connected to a lightweight ArduSimple Helical GNSS Tripleband + L-band antenna. This setup enabled the acquisition of GNSS measurements at a rate of 100 Hz. Real-time kinematic corrections were streamed over the cellular network from the swipos-GIS/GEO service, enabling localization with sub-centimeter precision. The Raspberry Pi connected to the cellular network via a Wi-Fi connection to a TP-Link M7650 mobile Wi-Fi router, which was likewise housed in the robot's payload.

Software and data recording

The software framework driving the robot was executed on its onboard computer. During operation, the onboard computer and connected sensors consumed approximately 800 mA. The robot's software framework followed a modular approach, with each module developed in compliance with the Robot Operating System (ROS 2) framework (*59*). Despite the availability of many out-of-the-box packages, our project necessitated the development of modules tailored to the robot's functions, such as drivers for the Adafruit PCA9685 PWM/Servo driver. These newly developed modules were released as open-source packages with a permissive license under the ROSbloX project (*60*). The implemented software framework not only drove the robot's core functionalities but also ensured efficient onboard data recording and management during operation and allowed monitoring the state of the robot through a web interface.

State estimation and sensor fusion

The robot's raw sensor measurement streams were recorded for postprocessing and data analysis. Further, these measurements were used for automatic control of the robot and autonomous navigation. However, they were generally subject to noise and not suitable for controlling the robot without using modeling, state estimation, and sensor fusion. The robot's state was divided into two parts: its morphological state, a function of its tendon lengths; and its position and orientation with respect to an inertial frame, that is, its pose (6 DoF) with respect to a fixed north-east-down (NED) coordinate frame and its change over time. Neither its morphological state nor its global pose could be measured directly and had to be estimated. Both winches were retrofitted with rotary position sensors that measured rotations of the spools that hold the tendons. The encoder measurements were integrated to obtain the absolute number of turns for each spool, ψ. Following (*61*), the tendons winding up on the spool were modeled as an Archimedean spiral, and the length of each tendon was estimated as

$$\hat{l}_{1,2} := \pi\left(d_{\max} - 2\delta_{\mathbf{r}}\psi\right) \quad (1)$$

whereby d_{max} and δ_r correspond to the diameter of the spool when the whole tendon was wound up and the radial decay rate. Parameters d_{max} and δ_r were determined in a one-time calibration. The estimated tendon lengths $\hat{l}_{1,2}$ were linked to the robot's morphology as described in the "Modeling morphological reconfiguration" section. The robot's body was fully symmetric. Each side could be defined as its front, back, left, or right side. Equally, the robot top and bottom were the same. However, for autonomous navigation, a coordinate frame was fixed to its central payload, with its x axis following one tendon, its y axis the second tendon, and the z axis completing a right-hand coordinate system. In the following, we refer to the direction of the x axis as forward, the y axis direction as left, and z axis direction as top. An inertial coordinate frame following an NED convention was introduced and served as a global reference. To estimate the robot's absolute orientation, in terms of its roll, pitch, and yaw angles, with respect to the inertial frame, we used a complementary filter (*62*) that fused angular velocities, linear accelerations, and magnetic field direction measurements of the IMU at 20 Hz. The measured angular velocities and linear accelerations, the robot's estimated absolute orientation in the inertial frame, and GNSS measurements were fused using an unscented Kalman filter (*63*) to estimate the robot's 6-DoF pose in the inertial frame at 20 Hz. The applied filter (*64*) estimated a fifteen-dimensional state, with nine states linked to the robot's 3D position, linear velocity, and acceleration and six states representing its 3D orientation and angular velocity. The sensor fusion with GNSS fusion was only applicable in outdoor environments. For laboratory experiments, pose measurements at 120 Hz streamed from an OptiTrack Motion Capture system served as ground truth.

Control, autonomy, and path planning

Automatic control was implemented to control the robot's electric winches and wheel motors to enable teleoperation, autonomous morphological reconfiguration, and navigation.

Morphological reconfiguration

The electric winches controlled the tendon lengths. A cascaded control scheme was applied to vary the tendon lengths and allowed for controlled reconfiguration of the robot's morphology to either take on a spherical shape or a shape suitable for driving, as shown in Fig. 2A. The cascaded control scheme is visualized in fig. S12. The length of each tendon was controlled individually to follow a set reference length. A proportional gain controller was implemented to control the current tendon lengths $l_{est,n}$ to follow the reference length, $l_{ref,n}$,

$$u_{motor,n} = P_{gain}\left(\left(l_{ref,n} - \hat{l}_n\right) \text{ with } n \in [1,2]\right) \quad (2)$$

The controller's gain, P_{gain}, was determined experimentally. When a tuple of reference lengths, $T_{ref} := (l_{ref,1}, l_{ref,2})$, was given to the winch controllers, both winches controlled the tendons to the respective reference lengths independently. For automatic morphological reconfiguration, a trajectory controller was implemented that enabled synchronized control of both tendons. The morphological trajectory controller took as input a morphological reconfiguration trajectory consisting of a series of tuples of K reference lengths, $T_{ref,1}. \ldots T_{ref,K}$. Each tuple was commanded to the low-level tendon controllers sequentially by the trajectory controller, whereby consecutive tuples were executed only if the prior tuple had been successfully reached. The execution of these morphological trajectories allowed precise control of the robot's morphological state and enabled the automatic execution of the morphing trajectories in the robot's morphological reconfiguration space as shown in Fig. 2.

Two morphological states of the robot and the transition between these states were of particular interest because they enabled the robot to navigate efficiently and robustly in outdoor environments. These states were its flat configuration, which enabled the robot to drive using its wheels, and its spherical, curled-up shape, in which the robot's payload and wheels were protected and it was able to roll. These configurations are indicated in the morphological reconfiguration diagram in Fig. 2C by markers 2 (rover configuration) and 4 (spherical configuration).

To transition from a flat to a spherical configuration and vice versa, a morphological trajectory was introduced. This trajectory steered the tendons through an intermediate configuration, labeled as point 3 in the diagram in Fig. 2C. Introducing this trajectory ensured stable morphological reconfiguration and avoided the twisting of the robot indicated by the blue areas in the diagram in Fig. 2C.

The morphological reconfiguration from rover to spherical shape was only one of the robot's features enabled by the morphological trajectory controller. Further morphological reconfigurations included in-plane reconfiguration such as elongation of the robot's frame, which made it more suitable for fast forward motion or to a more circular shape suitable for efficient and stable turning. Last, the morphology controller allowed the robot to change its center of gravity when in a spherical (and almost spherical) shape by altering the position of the payload suspended in the robot's frame. The payload made up nearly two-thirds of the robot's mass. Changing its location inside the robot's frame shifted the robot's center of gravity outside its support polygon and initiated rolling motion even on flat ground, as shown in Fig. 4D.

Low-level driving control

In its flat configuration, the robot resembled a four-wheeled, skid-steered, mobile robot whose movement and direction were controllable by differentially driving the wheels on either side of the robot. The left and right sides of the robot were defined as the robot's frame and wheels, which were located to the left and right sides of the longer tendon. The wheels on each side moved at the same speed. When the wheels on both sides were driven at the same speed and direction, the robot moved forward or backward in a straight line. Steering was achieved by varying the relative speeds of the wheels on either side, causing the robot to skid or slide as it turned. To control the skid-steered robot, input linear velocity, v, and angular velocity, ω, were translated into wheel velocities by a low-level driving controller. The low-level driving controller computed velocities of the left and right wheels v_L and v_R as follows

$$v_L = v - \frac{\omega d}{2} \quad (3)$$

and

$$v_R = v + \frac{\omega d}{2} \quad (4)$$

where d is the robot's track width. This method effectively converted the desired motion into differential wheel speeds, enabling precise maneuvering of the robot.

Autonomous navigation

The robot could be operated remotely and autonomously. Remote control was facilitated through a Sony Dualshock PlayStation 4 game pad,

which wirelessly connected to the robot's Raspberry Pi via Bluetooth. For autonomous navigation, the robot leveraged its morphological reconfiguration capability to alternate between rolling and driving modes. We augmented traditional navigation techniques with innovative strategies to fully use this feature. On slopes, rolling was preferred for its safety and energy efficiency. Hence, it was favorable for the robot to curl up and roll where feasible, only reverting to its rover configuration when rolling became impractical to reach its destination.

Three distinct navigation strategies were implemented using the ROS 2 Navigation2 (Nav2) (*65*) framework to autonomously move toward predefined waypoints. Nav2 is a production-grade navigation framework designed for building highly reliable autonomous systems (*66*). For each navigation strategy, a high-level path planning algorithm continuously computed the shortest path between the robot's current location and a target waypoint. Once a path was planned, a low-level controller computed motor commands to follow the planned path using the dynamic window approach (*67*). The controller used feedback from the robot's sensors to dynamically adjust the robot's speed and direction. The implemented navigation strategies differed in the integration of different high-level behaviors.

We refer to the three implemented navigation strategies as "basic navigation," "reflexive approach," and "trajectory-informed strategy." The basic navigation strategy used the default navigation behavior for skid-steered rovers implemented in Nav2. For simplicity, recovery strategies as navigational behavior were disabled. The reflexive approach combined rolling and driving. Wherever rolling was possible, the robot changed to its spherical configuration and started rolling until it stopped automatically, either because of an obstacle or upon reaching flat ground. It then morphed to its rover configuration and drove toward the goal. The trajectory-informed strategy incorporated information about the robot's prior trajectory. When rolling was feasible, the robot reconfigured into a sphere and rolled toward the target waypoint. It actively stopped rolling and unfolded when closest to its destination.

These strategies are delineated and their performance evaluated in Figs. 5 and 6, illustrating the effectiveness of each approach. A schematic visualization is shown in fig. S13. Currently, our strategies do not incorporate prior topographical data into path planning, indicating a potential area for future refinement.

Statistical analysis

Figure 3 (A and B) visualizes the robot's success rate when driving and morphing over obstacles, respectively, when squeezing through gaps. For the five step heights and seven gap widths, the corresponding bars show the success rate as a percentage of 12 consecutive trials.

Figure 4 (B and C) shows the influence of the aspect ratio on the velocity and positional error, respectively, and the angular velocity and turning radius of the robot. For each of the five evaluated aspect ratios per plot, mean and SD were calculated from six trials.

Supplementary Materials

The PDF file includes:
Materials and Methods
Figs. S1 to S14
Tables S1 to S7
Legends for movies S1 to S9

Other Supplementary Material for this manuscript includes the following:
Movies S1 to S9

REFERENCES AND NOTES

1. P. Ramdya, A. J. Ijspeert, The neuromechanics of animal locomotion: From biology to robotics and back. *Sci. Robot.* **8**, eadg0279 (2023).
2. J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, M. Hutter, Learning quadrupedal locomotion over challenging terrain. *Sci. Robot.* **5**, eabc5986 (2020).
3. T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, M. Hutter, Learning robust perceptive locomotion for quadrupedal robots in the wild. *Sci. Robot.* **7**, eabk2822 (2022).
4. A. Torres-Pardo, D. Pinto-Fernández, M. Garabini, F. Angelini, D. Rodriguez-Cianca, S. Massardi, J. Tornero, J. C. Moreno, D. Torricelli, Legged locomotion over irregular terrains: State of the art of human and robot performance. *Bioinspir. Biomim.* **17**, 061002 (2022).
5. F. Angelini, P. Angelini, C. Angiolini, S. Bagella, F. Bonomo, M. Caccianiga, C. Della Santina, D. Gigante, M. Hutter, T. Nanayakkara, P. Remagnino, D. Torricelli, M. Garabini, Robotic monitoring of habitats: The natural intelligence approach. *IEEE Access* **11**, 72575–72591 (2023).
6. M. Dunbabin, L. Marques, Robots for environmental monitoring: Significant advancements and applications. *IEEE Robot. Autom. Mag.* **19**, 24–39 (2012).
7. A. King, Technology: The future of agriculture. *Nature* **544**, S21–S23 (2017).
8. P. E. Glick, J. B. Balaram, M. R. Davidson, E. Lyons, M. T. Tolley, The role of low-cost robots in the future of spaceflight. *Sci. Robot.* **9**, eadl1995 (2024).
9. J. Delmerico, S. Mintchev, A. Giusti, B. Gromov, K. Melo, T. Horvat, C. Cadena, M. Hutter, A. Ijspeert, D. Floreano, L. M. Gambardella, R. Siegwart, D. Scaramuzza, The current state and future outlook of rescue robotics. *J. Field Rob.* **36**, 1171–1191 (2019).
10. C. Li, K. Lewis, The need for and feasibility of alternative ground robots to traverse sandy and rocky extraterrestrial terrain. *Adv. Intell. Syst.* **5**, 2100195 (2023).
11. H. Kalita, J. Thangavelautham, Exploration of extreme environments with current and emerging robot systems. *Curr. Robot. Rep.* **1**, 97–104 (2020).
12. F. Stella, J. Hughes, The science of soft robot design: A review of motivations, methods and enabling technologies. *Front. Robot. AI* **9**, 1059026 (2023).
13. S. Kim, C. Laschi, B. Trimmer, Soft robotics: A bioinspired evolution in robotics. *Trends Biotechnol.* **31**, 287–294 (2013).
14. K. Jayaram, R. J. Full, Cockroaches traverse crevices, crawl rapidly in confined spaces, and inspire a soft, legged robot. *Proc. Natl. Acad. Sci. U.S.A.* **113**, E950–E957 (2016).
15. D. Morgan, U. Proske, D. Warren, Measurements of muscle stiffness and the mechanism of elastic storage of energy in hopping kangaroos. *J. Physiol.* **282**, 253–261 (1978).
16. W. M. Kier, M. P. Stella, The arrangement and function of octopus arm musculature and connective tissue. *J. Morphol.* **268**, 831–843 (2007).
17. K. Schmidt-Nielsen, *Animal Physiology: Adaptation and Environment* (Cambridge Univ. Press, 1997), vol. 359.
18. A. Berger, L. M. F. Barthel, W. Rast, H. Hofer, P. Gras, Urban hedgehog behavioural responses to temporary habitat disturbance versus permanent fragmentation. *Animals* **10**, 2109 (2020).
19. C. Stapp, "Defensive diversity: Factors influencing the interspecific variation in the armor of armadillos (*Cingulata*)," thesis, California State University, Long Beach (2022).
20. I. H. Tuf, B. Ďurajková, Antipredatory strategies of terrestrial isopods. *ZooKeys* **1101**, 109–129 (2022).
21. N. Møbjerg, K. Halberg, A. Jørgensen, D. Persson, M. Bjørn, H. Ramløv, R. Kristensen, Survival in extreme environments–on the current knowledge of adaptations in tardigrades. *Acta Physiol.* **202**, 409–420 (2011).
22. S. R. Kundu, *Spider Evolution: Genetics, Behavior, and Ecological Influences* (Academic Press, 2021).
23. D. V. Baker, J. R. Withrow, C. S. Brown, K. G. Beck, Tumbling: Use of diffuse knapweed (*Centaurea diffusa*) to examine an understudied dispersal mechanism. *Invasive Plant Sci. Manag.* **3**, 301–309 (2010).
24. A. Rafsanjani, V. Brulé, T. L. Western, D. Pasini, Hydro-responsive curling of the resurrection plant *Selaginella lepidophylla*. *Sci. Rep.* **5**, 8064 (2015).
25. B. T. Wilcox, J. Joyce, M. D. Bartlett, Rapid and reversible morphing to enable multifunctionality in robots. *Adv. Intell. Syst.* **2025**, 2300694 (2025).
26. E. Gefen, D. Zarrouk, Flying STAR2, a hybrid flying driving robot with a clutch mechanism and energy optimization algorithm. *IEEE Access* **10**, 115491 (2022).
27. S. K. Mahmood, S. H. Bakhy, M. A. Tawfik, Propeller-type wall-climbing robots: A review, in *IOP Conference Series: Materials Science and Engineering* (IOP Publishing, 2021), vol. 1094, p. 012106.
28. E. Sihite, A. Kalantari, R. Nemovi, A. Ramezani, M. Gharib, Multi-modal mobility Morphobot (M4) with appendage repurposing for locomotion plasticity enhancement. *Nat. Commun.* **14**, 3323 (2023).
29. Y.-S. Kim, G.-P. Jung, H. Kim, K.-J. Cho, C.-N. Chu, Wheel transformer: A wheel-leg hybrid robot with passive transformable wheels. *IEEE Trans. Robot.* **30**, 1487–1498 (2014).
30. S. Iversen, J. Jouffroy, "Umbrella wheel — A stair-climbing and obstacle-handling wheel design concept," in *2017 International Conference on Advanced Mechatronic Systems (ICAMechS)* (IEEE, 2017), pp. 312–317.

31. T. Sun, X. Xiang, W. Su, H. Wu, Y. Song, A transformable wheel-legged mobile robot: Design, analysis and experiment. *Robot. Auton. Syst.* **98**, 30–41 (2017).
32. K. Shi, Z. Jiang, L. Ma, L. Qi, M. Jin, MTABot: An efficient morphable terrestrial-aerial robot with two transformable wheels. *IEEE Robot. Autom. Lett.* **9**, 1875–1882 (2024).
33. H. Banerjee, S. Kakde, H. Ren, "OrumBot: Origami-based deformable robot inspired by an umbrella structure," in *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (IEEE, 2018), pp. 910–915.
34. J. T. Karras, C. L. Fuller, K. C. Carpenter, A. Buscicchio, D. McKeeby, C. J. Norman, C. E. Parcheta, I. Davydychev, R. S. Fearing, "Pop-up mars rover with textile-enhanced rigid-flex PCB body," in *2017 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017), pp. 5459–5466.
35. C. H. Belke, K. Holdcroft, A. Sigrist, J. Paik, Morphological flexibility in robotic systems through physical polygon meshing. *Nat. Mach. Intell.* **5**, 669 (2023).
36. J. Seo, J. Paik, M. Yim, Modular reconfigurable robotics. *Annu. Rev. Control Robot. Auton. Syst.* **2**, 63 (2019).
37. Y. Liu, Q. Bi, X. Yue, J. Wu, B. Yang, Y. Li, A review on tensegrity structures-based robots. *Mech. Mach. Theory* **168**, 104571 (2022).
38. N. S. Usevitch, Z. M. Hammond, M. Schwager, A. M. Okamura, E. W. Hawkes, S. Follmer, An untethered isoperimetric soft robot. *Sci. Robot.* **5**, eaaz0492 (2020).
39. M. Tolley, R. F. Shepherd, B. Mosadegh, K. C. Galloway, M. Wehner, M. Karpelson, R. J. Wood, G. M. Whitesides, A resilient, untethered soft robot. *Soft Robot.* **1**, 213–223 (2014).
40. D. Shah, B. Yang, S. Kriegman, M. Levin, J. Bongard, R. Kramer-Bottiglio, Shape changing robots: Bioinspiration, simulation, and physical realization. *Adv. Mater.* **33**, 2002882 (2021).
41. G. Dudek, P. Giguere, C. Prahacs, S. Saunderson, J. Sattar, L.-A. Torres-Mendez, M. Jenkin, A. German, A. Hogue, A. Ripsman, J. Zacher, E. Milio, H. Liu, P. Zhang, M. Buehler, C. Georgiades, AQUA: An amphibious autonomous robot. *Computer* **40**, 46–53 (2007).
42. R. Altendorfer, N. Moore, H. Komsuoglu, M. Buehler, H. B. Brown Jr., D. McMordie, U. Saranli, R. Full, D. E. Koditschek, RHex: A biologically inspired hexapod runner. *Auton. Robot.* **11**, 207–213 (2001).
43. B. B. Dey, S. Manjanna, G. Dudek, "Ninja legs: Amphibious one degree of freedom robotic legs," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Syst. (IROS)* (IEEE, 2013), pp. 5622–5628.
44. J. Sun, E. Lerner, B. Tighe, C. Middlemist, J. Zhao, Embedded shape morphing for morphologically adaptive robots. *Nat. Comm.* **14**, 6023 (2023).
45. R. Baines, S. K. Patiballa, J. Booth, L. Ramirez, T. Sipple, A. Garcia, F. Fish, R. Kramer-Bottiglio, Multi-environment robotic transitions through adaptive morphogenesis. *Nature* **610**, 283–289 (2022).
46. N. Savage, Bioinspired robots walk, swim, slither and fly. *Nature Outlook: Robotics and Artificial Intelligence* (2022); 10.1038/d41586-022-03014-x.
47. A. Western, M. Haghshenas-Jaryani, M. Hassanalian, Golden wheel spider-inspired rolling robots for planetary exploration. *Acta Astronaut.* **204**, 34–48 (2023).
48. C. Dunning, R. Cervantes, L. Velasquez, A. Western, M. Sherman, M. Hassanalian, "Design and manufacturing of a bioinspired spider rolling robot for Mars exploration," in *AIAA Scitech 2021 Forum* (AIAA, 2021), p. 0958.
49. B. W. Mulvey, T. D. Lalitharatne, T. Nanayakkara, Deformobot: A bio-inspired deformable mobile robot for navigation among obstacles. *IEEE Robot. Autom. Lett.* **8**, 3828–3835 (2023).
50. R. Armour, K. Paskins, A. Bowyer, J. Vincent, W. Megill, R. Bomphrey, Jumping robots: A biomimetic solution to locomotion across rough terrain. *Bioinspir. Biomim.* **2**, S65–S82 (2007).
51. X. Huang, X. Zhu, G. Gu, Kinematic modeling and characterization of soft parallel robots. *IEEE Trans. Robot.* **38**, 3792–3806 (2022).
52. L. Lu, S. Leanza, J. Dai, X. Sun, R. R. Zhao, Easy snap-folding of hexagonal ring origami by geometric modifications. *J. Mech. Phys. Solids* **171**, 105142 (2023).
53. M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch, R. Diethelm, S. Bachmann, A. Melzer, M. Hoepflinger, "ANYmal - a highly mobile and dynamic quadrupedal robot," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2016), pp. 38–44.
54. Y. I. Sobolev, R. Dong, T. Tlusty, J.-P. Eckmann, S. Granick, B. A. Grzybowski, Solid-body trajectoids shaped to roll along desired pathways. *Nature* **620**, 310–315 (2023).
55. J. Mlýnek, M. Petrů, T. Martinec, S. S. R. Koloor, Fabrication of high-quality polymer composite frame by a new method of fiber winding process. *Polymers* **12**, 1037 (2020).
56. O. Weeger, Y. S. B. Kang, S.-K. Yeung, M. L. Dunn, Optimal design and manufacture of active rod structures with spatially variable materials. *3D Print. Addit. Manuf.* **3**, 204 (2016).
57. M. Esrin, J. Gartner, Test method for evaluation of the resistance of fiberglass rods to combined mechanical and chemical stress. *IEEE Trans. Power Appar. Syst.* **PAS-103**, 2741–2745 (1984).
58. A. A. Shabana, R. Y. Yakoub, Three dimensional absolute nodal coordinate formulation for beam elements: Theory. *J. Mech. Des.* **123**, 606–613 (2001).
59. S. Macenski, T. Foote, B. Gerkey, C. Lalancette, W. Woodall, Robot Operating System 2: Design, architecture, and uses in the wild. *Sci. Robot.* **7**, eabm6074 (2022).
60. M. Polzin, ROSbloX: Accelerating robot software development with modular ROS building blocks (2024); https://rosblox.github.io/.
61. P. McGarey, M. Polzin, T. D. Barfoot, "Falling in line: Visual route following on extreme terrain for a tethered mobile robot," in *2017 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017), pp. 2027–2034.
62. R. G. Valenti, I. Dryanovski, J. Xiao, Keeping a good attitude: A quaternion-based orientation filter for IMUs and MARGs. *Sensors* **15**, 19302 (2015).
63. E. Wan, R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium* (IEEE, 2000), pp. 153–158.
64. T. Moore, D. Stouch, "A generalized extended Kalman filter implementation for the Robot Operating System" in *Intelligent Autonomous Systems 13*, E. Menegatti, N. Michael, K. Berns, H. Yamaguchi, Eds. (Springer International Publishing, 2016), pp. 335–348.
65. S. Macenski, F. Martin, R. White, J. G. Clavero, "The Marathon 2: A navigation system," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2020), pp. 2718–2725.
66. S. Macenski, T. Moore, D. V. Lu, A. Merzlyakov, M. Ferguson, From the desks of ROS maintainers: A survey of modern & capable mobile robotics algorithms in the robot operating system 2. *Robot. Auton. Syst.* **168**, 104493 (2023).
67. D. Fox, W. Burgard, S. Thrun, The dynamic window approach to collision avoidance. *IEEE Robot. Autom. Mag.* **4**, 23–33 (1997).
68. Masteraah, Steppenschuppentier1a.jpg (2004); https://commons.wikimedia.org/wiki/File:Steppenschuppentier1a.jpg.
69. Masteraah, Steppenschuppentier2.jpg (2004); https://commons.wikimedia.org/wiki/File:Steppenschuppentier2.jpg.
70. I. Rechenberg, Cebrennus_rechenbergi_1.jpg (2014); https://commons.wikimedia.org/wiki/File:Cebrennus_rechenbergi_1.jpg.
71. L. Worch, White_lady.jpg (2016); https://commons.wikimedia.org/wiki/File:White_lady.jpg.

Acknowledgments: We thank members of the CREATE Lab, especially K. Junge, for assistance with experiments, practical advice, and support. **Funding:** This work was supported by the Swiss Polar Institute (TEG-2023-001), CROSS, and E4S. **Author contributions:** M.P. conceived the original idea, led the system design and development, and conducted experiments to validate the project. Q.G. was responsible for the mathematical analysis, focusing on simulations and modeling to underpin the project's theoretical framework. J.H. supervised the research, supporting the project's execution and development. All authors contributed to writing the manuscript. **Competing interests:** M.P. and J.H. hold a patent on the design of the robot structure. All other authors declare that they have no competing interests. **Data and materials availability:** All data needed to evaluate the conclusions in the paper are present in the paper or the Supplementary Materials. The software developed in this study has been deposited in a public database and is available at https://doi.org/10.5281/zenodo.13964001 and https://doi.org/10.5281/zenodo.13964003.

Submitted 5 April 2024
Accepted 28 January 2025
Published 26 February 2025
10.1126/scirobotics.adp6419

HUMAN-ROBOT INTERACTION

Social robots as conversational catalysts: Enhancing long-term human-human interaction at home

Huili Chen*, Yubin Kim, Kejia Patterson, Cynthia Breazeal, Hae Won Park

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

The integration of social robots into family environments raises critical questions about their long-term influence on family interactions. This study explores the potential of social robots as conversational catalysts in human-human dyadic interaction, focusing on enhancing high-quality, reciprocal conversations between parents and children during dialogic coreading activities. With the increasing prevalence of social robots in homes and the recognized importance of parent-child exchanges for children's developmental milestones, this work presents a comprehensive empirical investigation involving more than 70 parent-child dyads over a period of 1 to 2 months. We examined the effects of three robot interaction styles—a passive robot listener, an active robot with a fixed behavior strategy, and an active robot with a strategy-switching mechanism—on parent-child conversational dynamics. Our findings reveal that a robot's active participation enhances the quality of parent-child dialogic conversations. The influence of robot facilitation varied on the basis of parental English proficiency. Strategy-switching robots provided greater benefits to non–native English–speaking families, whereas dyads with native English–speaking parents benefited more from fixed-strategy robots. Overall, this study highlights the promise of social robots that empower parents in fostering their children's dialogic development—a contrast with the prevalent design of educational robots that primarily target children. It provides critical insights into the equitable, nuanced design of long-term family-robot interactions at home, especially in supporting diverse family backgrounds.

INTRODUCTION

As social robots enabled by artificial intelligence (AI) become more common in our homes, concerns are growing about their potential harm to human-human connections (*1*). The responsible design of social robots to purposefully enhance, rather than compromise, human-human interactions and relationships becomes unprecedentedly critical. This study explores the potential of social robots as conversational catalysts in human-group interactions, particularly focusing on enhancing long-term human-human reciprocal interaction at home. We focus on parent-child interactions because of their crucial role in child development.

High-quality, sensitive, and reciprocal caregiver-child interactions critically influence a child's overall development, influencing social, emotional, and cognitive growth (*2*–*5*). Such high-quality interactions are also crucial for young children's language development (*6*–*9*). However, children, especially from low socioeconomic status (SES) families, often experience limited exposure to enriched adult-child conversations. Studies highlight that disparities in the frequency, duration, and richness of conversations between low- and higher-SES families contribute to gaps in vocalization and vocabulary development (*9*–*11*).

The lack of access to parental education and guidance critically contributes to the parental "participation gap"—the disparity in caregivers' active participation in their children's education between low- and higher-SES families (*12*, *13*). As noted by Hoover-Dempsey and Sandler (*14*), well-guided parental involvement is a notable factor in children's cognitive and literacy development, prompting the need for interventions to support it.

Among various interventions targeting the participation gap, traditional home-visiting intervention programs have been shown to be effective, such as (*15*). Yet, accessibility to children's educational resources and extracurricular support is limited in under-resourced communities. Providing home-visiting interventions would be even more limited because of the high costs and the need for specialist facilitators. Given these challenges, there is an urgent need and opportunity to develop interactive technologies that can provide personalized, cost-effective interventions at scale. Although recent developments in educational technology, such as educational apps (*16*, *17*), show promise, few can facilitate real-time, affective, and reciprocal adult-child interaction, such as enhancing parent-child conversations about a story (*18*).

As AI-enabled technology interventions in education become increasingly prevalent, social robots present a promising avenue to foster high-quality parent-child interactions. Exploring the integration of robots in these interactions, particularly their influence on conversational dynamics, is essential. Such interventions could not only address educational challenges but also enrich the variety of interactions between parents and children. To examine the potential of social robots designed for enhancing parent-child conversational interactions, we identified three related subareas: parent-child dialogic interaction, multiperson child-robot interaction, and robot behavior design in the multiperson setting. The state of the art in each subarea, alongside its limitations, is presented in the following three subsections. These three subareas together shed light on both the promising potential and importance of our proposed work and illustrate how our proposed design and evaluation are grounded in prior work while advancing the field.

Parent-child dialogic interaction

Parent-child dialogic interaction, particularly during coreading, is crucial for early childhood development, influencing reading skills, language outcomes, and lifelong reading interest (*19*, *20*). However, the quality of these interactions varies across family contexts, with challenges for non–native English–speaking families. Research shows that bilingual children, especially those with limited English input at home, often lag behind monolingual peers in vocabulary development

Personal Robots Group, Media Lab, Massachusetts Institute of Technology, Cambridge, MA, USA.
*Corresponding author. Email: hchen25@media.mit.edu

(*21*, *22*). This gap persists from ages 3 to 10, highlighting the long-term influence of the home language environment (*22*). Both home language use and maternal vocabulary knowledge substantially influence children's language development (*23*, *24*).

To enhance parent-child dialogic coreading, educational specialists have introduced protocols to foster meaningful dialogue (*25*–*27*). Coaching techniques, such as the model strategy (*28*, *29*), provide exemplary dialogic reading sessions. Although prerecorded videos offer scalability, they lack personalization. Real-time facilitator coaching provides personalized guidance but is less scalable (*30*, *31*).

To address this trade-off, various technological tools have been developed, including e-book apps (*32*), augmented reality picture books (*33*), and robot companions (*34*). Social robots offer a promising solution, potentially providing real-time, personalized coaching during dialogic coreading sessions. By considering factors such as parental English proficiency, these robotic interventions could address the needs of linguistically diverse families, promoting more equitable outcomes in early childhood language development.

Multiperson child-robot interaction

Although social robots have predominantly been deployed for one-on-one educational interactions with children (*35*), the domain of multiperson child-robot interactions remains relatively underexplored. Existing research in multiperson settings primarily concentrates on interactions among groups of children (*36*–*40*). Such studies have explored roles for robots in mediating conflict resolution among children (*37*) and recognizing social dynamics within groups (*39*).

However, investigations into interactions involving both adults and children are notably more limited (*34*, *36*, *41*). Rudovic *et al.* (*41*) focused on robot-assisted therapy for children with autism, emphasizing the child-robot interaction with less adult participation. Gvirsman *et al.*'s (*36*) Patricc platform showed how robotic facilitation could surpass traditional tablets in triadic interactions but fell short of thoroughly exploring parent-toddler dynamics. The intervention described by Chan *et al.* (*42*), although suggesting benefits such as reduced parental frustration and increased child independence, did not incorporate socially intelligent robots. Chen *et al.* (*34*) designed a teleoperation system to support robot-augmented parent-child story reading, yet the influence on conversation quality needs more investigation.

To our knowledge, the specific influence of robotic facilitation on the conversational dynamics of parent-child interactions has yet to be thoroughly investigated. Our study seeks to fill this gap by elucidating the conversational dynamics within robot-assisted parent-child interactions.

Robot behavior design in multiperson contexts

Social robots have been designed to enhance the dynamics and processes of human groups, such as conflict resolution and participation enhancement (*43*–*45*). Among these prior studies, the robot's support role emerges as a critical design factor influencing the robot's influence on human groups. In multiperson interactions, social robots can assume various roles, ranging from followers to leaders, depending on the level of activity-related support provided. Leader or expert roles involve explicit support, such as coaching or hosting activities (*43*, *46*–*48*). Follower roles support interactions implicitly, displaying vulnerability or emotional expressions (*49*, *50*). Peer-like robots, positioned similarly to humans, can mediate conflicts and promote positive group dynamics (*37*, *51*, *52*). These various roles enable robots to either facilitate effective activity completion or shape group dynamics (*46*, *50*, *52*), potentially providing distinct benefits to individual users and the group as a whole. The robot role design has been conceptualized as target behavior dimensions that a robot could adapt to enhance group experience (*53*).

Another crucial design factor for robot social behavior in multiperson interactions is the communication style. Both verbal and nonverbal behaviors of a robot can substantially influence group dynamics. Verbally active robots can influence trust building (*54*, *55*), group engagement (*56*), psychological safety (*57*), and conversational equality and positivity (*49*). On the nonverbal side, behaviors such as gestures (*58*), gaze (*59*, *60*), navigation (*61*, *62*), and physical orientation (*63*, *64*) also influence group interactions and perceptions. An alternative approach, inspired by Ju and Leifer's theory (*65*) of implicit interactions, involves robots exhibiting implicit nonverbal behaviors as peripheral entities, effectively promoting active participation and collaboration (*44*, *66*).

Recent research has highlighted the potential benefits of enabling robots to dynamically switch between different roles or strategies during interactions. This dynamic approach has proven advantageous in single-person contexts, both in physical cooperation (*67*–*69*) and social interactions (*70*–*72*). In educational settings, role-adaptive robots have enhanced students' motivation in creative dance (*71*), aided learning of computer science concepts (*72*), and promoted learning and affective experiences by leveraging multiple roles (*70*). Although less explored in multiperson scenarios, the concept of dynamic role switching in human-robot interactions shows potential for promoting users' cognitive and affective experiences, such as flow state (*53*).

Overview of our work

Our study aims to integrate robot roles and communication styles into a unified behavior design framework. Motivated by the promising results of emerging behavior-switching paradigms, we examine whether a robot that alternates among six role-style strategies can produce synergistic positive effects on parent-child interactions compared with one that maintains a single strategy. To explore this, we developed social robots capable of engaging in dialogic interactions with parent-child pairs in real home environments (see Fig. 1). Over a period of 1 to 2 months, these robots were deployed in the households of more than 70 parent-child pairs across the United States, with children aged 3 to 7. Our large-scale, real-world empirical investigation focuses on the long-term influence of integrating a robot into home-based parent-child interactions on user conversational behaviors.

RESULTS

Experiment overview

We conducted a between-participant experiment with 71 families, randomly assigned to three conditions: control, fixed-strategy robot, and strategy-switching robot, as illustrated in Fig. 2. Regardless of the experimental condition, each family completed a dialogic reading curriculum consisting of six dialogic interaction sessions and six 4-min instructional videos introducing dialogic concepts. The decision to provide instructional videos to all parents was made to eliminate parental knowledge of dialogic reading as a potential confounding factor. The curriculum was supplemented by prestudy and poststudy parent-child coreading sessions conducted without

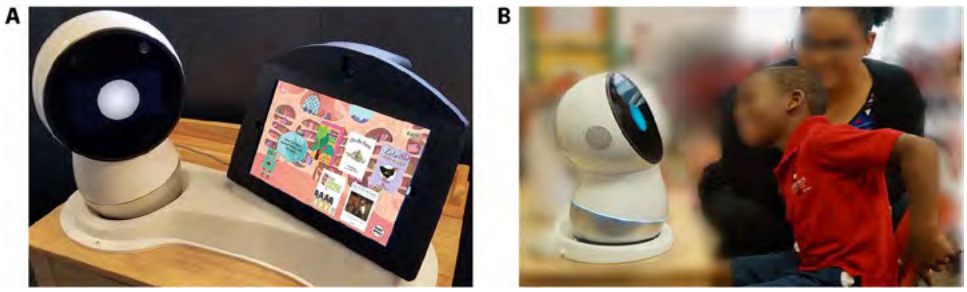


Fig. 1. Home deployment of an integrated robot system. The integrated system consists of two main components: (**A**) the hardware setup including Jibo, Android tablet, and web camera and Intel NUC machine that forms the core interactive system. (**B**) A representative parent-child dyad engaging with the integrated robot system during the 1- to 2-month deployment period, demonstrating the real-world application of the system in home environments.

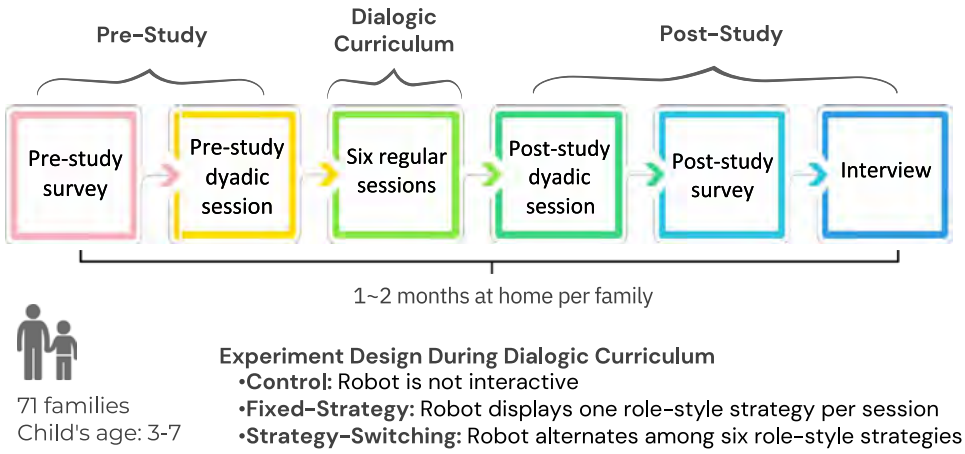


Fig. 2. Experimental design. Study timeline showing pre-/postassessments, six-session dialogic curriculum with robot interaction, and final interview. Three experimental conditions were used for parent-child pairs during the curriculum phase.

the robot's engagement to establish baselines and measure behavioral changes. In the two experimental conditions, families interacted with a robot that actively participated in the dialogic coreading activities. This contrasted with the control group, which included a noninteractive robot serving merely as a listener in the interaction. The difference between the two experimental conditions was that the fixed-strategy robot displayed only one role-style behavior strategy in each session, whereas the strategy-switching robot alternated among six strategies within each session.

Design for active robot participation in parent-child dialogic interaction

In the fixed-strategy and strategy-switching robot conditions, active robot participation incorporated two dimensions of robot behavior: the robot's support role and that of communication style (see Fig. 3 and the "Robot behavior design in multiperson contexts" section). Within the context of parent-child dialogic reading, the three identified support roles for the robot were demonstrator, moderator, and playmate, and the two communication styles were verbal and nonverbal robot intents, as further elaborated in the "Robot interaction design" section and Movie 1.

The demonstrator robot acted as a leading figure who asked high-quality dialogic questions. The moderator robot reminded the parent-child pair to creatively converse about the story. The playmate robot acted as the child's playful peer, mirroring their curiosity and interest in books. An exemplar interaction for each robot role is illustrated in Fig. 4.

When using the nonverbal communication style, the robot relied on nonverbal cues, like eye blinking, to indicate its speaking intent and waited for the parent and child's approval before joining their conversation. In contrast, the verbal communication style allowed the robot to actively and directly join parent-child conversations without indicating its speaking intent and waiting for their approval.

This resulted in six unique robot behavior strategies, such as a verbal demonstrator or nonverbal playmate. Our design aimed to use these role-style strategies to promote parent-child dialogic conversations. The fixed-strategy robot consistently displayed one behavior strategy per session, ensuring that all six strategies were experienced during the six-session dialogic curriculum. In contrast, the strategy-switching robot switched among the six strategies within each session. This strategy-switching mechanism was expected to provide more tailored and engaging interactions, potentially leading to more positive effects on parent-child conversational behaviors compared with a fixed-strategy approach.

Research questions

Our study addresses three main research questions. Our first research question (RQ1) examines how a robot's active participation affects in-the-moment parent-child conversational behaviors during robot-facilitated interaction. This question addresses a common concern that incorporating a robot into parent-child interactions might detract from direct conversational opportunities between the parent and child by dividing the conversational engagement among three parties instead of two. Through a direct comparison of interactions with and without the robot's active participation, we seek to develop a comprehensive understanding of the in-the-moment benefits and drawbacks of integrating a robot into parent-child interactions.

The second research question (RQ2) examines how engaging in a robot-facilitated dialogic curriculum influences parent-child conversational behaviors in their postcurriculum dyadic interaction without the robot's participation. This question evaluates the effectiveness of the robot-facilitated dialogic curriculum in fostering positive changes in parent-child conversational behaviors through long-term at-home robot interactions spanning 1 to 2 months.

The third research question (RQ3) examines how the influence of the robot-facilitated dialogic curriculum on parent-child conversational behaviors varies between dyads with native and non–native speaker parents. This question aims to illuminate the equitable design of long-term robot interactions in parent-child dialogic activities. Considering the pivotal role of parental English proficiency in parent-child dialogic reading, RQ3 examines how social robot

design may bridge or enlarge the learning and participation gap among families with diverse backgrounds.

Grounded in the potential of robot interaction in children's conversational and social learning, we hypothesized generally more positive benefits from the two interactive robot conditions compared with the non-interactive robot condition across all three main RQs. Regarding the comparison between the two interactive robot conditions, we hypothesized more positive effects in the strategy-switching robot condition than in the fixed-strategy robot condition, supported by prior empirical findings on the benefits of robot behavior-switching paradigms.

Overall, the findings in our study revealed three key insights. First, active robot participation enhanced parent-child conversational quality (RQ1). Second, the influence of robot-facilitated interventions varied on the basis of parental language proficiency (RQ2). Third, different robot interaction styles (fixed versus switching) benefited different family profiles (RQ3).

To assess the quality of conversational behaviors between parent and child, six conversational behaviors were operationalized as conversational quality indicators, where higher scores reflected more desirable conversing behavior in parent-child dialogic interaction. These six quality indicators were extracted from the speech of parents and children during real-time coreading sessions, encapsulating the duration and dynamics of conversational patterns for the parent (conv-time-parent and conv-ratio-parent), for the child (conv-time-child and conv-ratio-child), and for the dyad as a group (conv-time-dyad and conv-turns-dyad). To capture the intrapersonal dynamics of a person's conversational interaction, a speaker's conversing ratio was defined as the percentage of a speaker's utterances in conversing relative to their utterances in reading, excluding robot utterances. In addition, three reading-oriented features (read-time-parent, read-time-child, and read-time-dyad) were extracted as supplementary features to provide a more comprehensive view of the parent-child or parent-child-robot dialogic interactions.

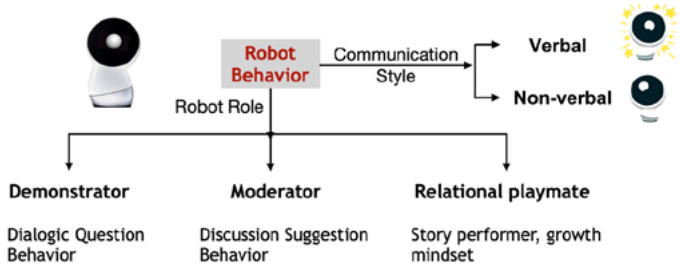
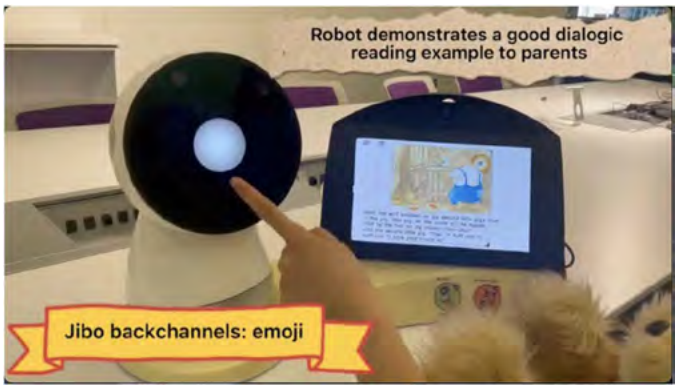


Fig. 3. Robot role-style strategies. Six strategies combining three roles and two communication styles for multiperson interaction.



Movie 1. Robot role-style behavior design in dialogic reading. The integrated robot system dynamically alternates among the roles of demonstrator, moderator, and playmate, using both verbal and nonverbal communication styles during real-time interactive reading sessions with a parent and child.

RQ1: Robot influence on in-the-moment parent-child conversational behaviors

We first analyzed the conversational dynamics across six sessions of a standard dialogic curriculum under three experimental conditions.

Conversational quality indicators

The Kruskal-Wallis analysis revealed significant differences in two of six conversational quality indicators across the conditions, as shown in Fig. 5. The indicators analyzed were the dyad's total conversing time (conv-time-dyad) and the parent's conversing ratio (conv-ratio-parent).

For conv-time-dyad, the analysis revealed a significant effect of the experimental condition on the total conversing time [$H(2) = 23.90$, $P = 3.88 \times 10^{-5}$], with an effect size of $\varepsilon^2 = 0.32$ and a power of

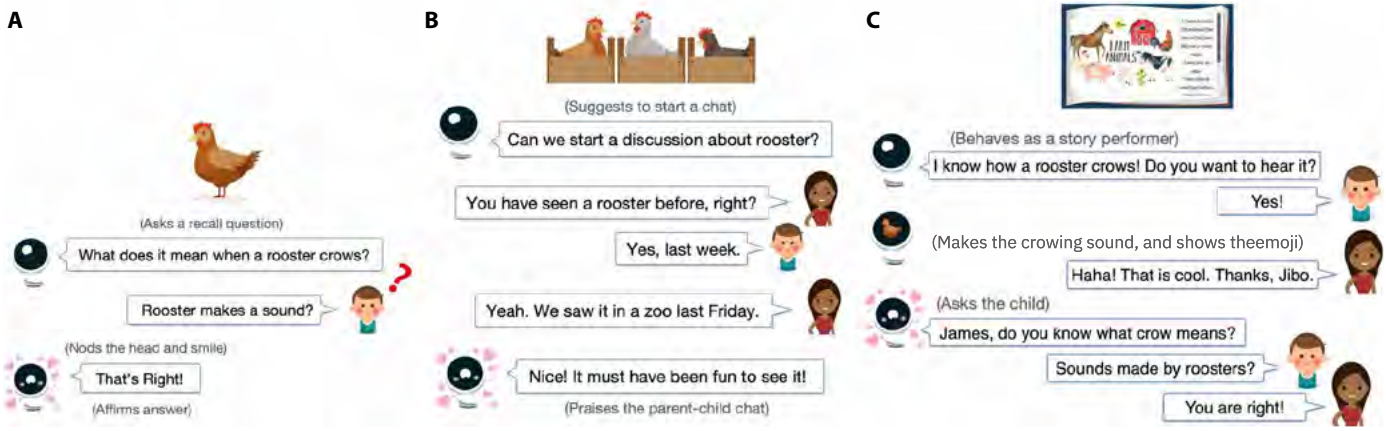


Fig. 4. Robot roles in parent-child interactions. Robot acting in the (**A**) demonstrator role, (**B**) moderator role, and (**C**) playmate role. Each role represents a distinct interaction strategy designed to facilitate different aspects of parent-child dialogic reading.

$1 - \beta > 0.99$. According to Dunn's post hoc test, the control group's total conversing time [mean (M) ± standard deviation (SD) = 34.02 ± 9.08] was significantly lower than those of the fixed-strategy group (M ± SD = 44.11 ± 5.66, $P = 5.31 \times 10^{-4}$) and the strategy-switching group (M ± SD = 45.22 ± 6.72, $P = 1.62 \times 10^{-5}$), with no marked difference between the latter two.

Regarding conv-ratio-parent, the analysis indicated a significant condition effect [$H(2) = 8.67$, $P = 0.039$], an effect size of $\varepsilon^2 = 0.10$, and a power of $1 - \beta = 0.68$. The control condition's mean conversing ratio (M ± SD = 0.48 ± 0.17) was significantly lower than that of the strategy-switching condition (M ± SD = 0.63 ± 0.19, $P = 0.02$), whereas the fixed-strategy condition (M ± SD = 0.62 ± 0.17) showed no significant difference compared to the other two conditions. Overall, a robot's active participation in the parent-child interaction improved the parent-child conversational duration and parent's conversing ratio.

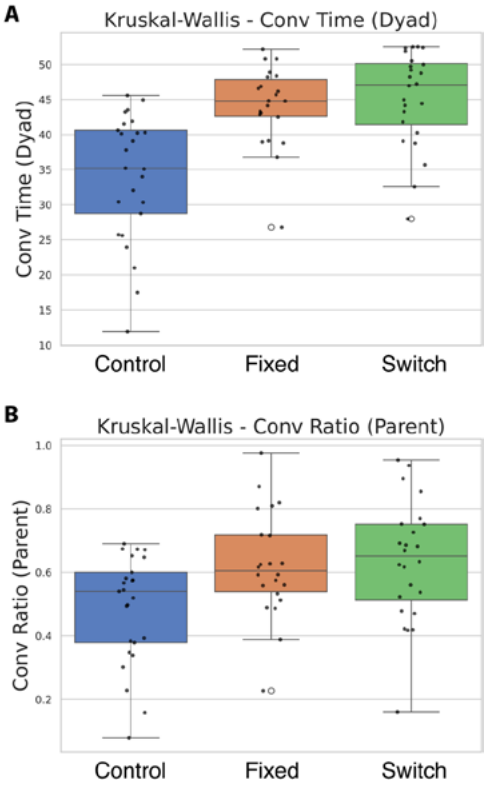


Fig. 5. Influence of robot interaction styles on parent-child conversational patterns. Statistical analysis using Kruskal-Wallis tests revealed significant differences across three robot interaction conditions (control: $N = 25$; fixed-strategy: $N = 22$; strategy-switching: $N = 24$). Box plots show the median, interquartile range (IQR), and data range within 1.5 × IQR, with outliers. (**A**) The dyad's conversational time showed significant variation [$H(2) = 23.90$, $P = 3.88 \times 10^{-5}$], with mean durations of 34.02 ± 9.08 min for the control group, 44.11 ± 5.66 min for the fixed-strategy group, and 45.22 ± 6.72 min for the strategy-switching group. (**B**) The parent's conversational ratio also demonstrated significant differences [$H(2) = 8.67$, $P = 0.039$], with mean ratios of 0.48 ± 0.17 for the control group, 0.61 ± 0.17 for the fixed-strategy condition, and 0.63 ± 0.19 for the strategy-switching condition ($P = 0.02$).

Reading features

The observed increases in conversing time and ratio prompted an examination of three additional reading features to identify which aspects of parent-child dialogic reading were affected by the robot's active engagement. Kruskal-Wallis tests revealed significant differences in both read-time-dyad [$H(2) = 24.05$, $P < 1.10 \times 10^{-5}$, $\varepsilon^2 = 0.32$, and a power of $1 - \beta > 0.99$] and read-time-parent [$H(2) = 23.65$, $P < 1.10 \times 10^{-5}$, $\varepsilon^2 = 0.32$, and a power of $1 - \beta > 0.99$]. Post hoc analyses showed that the control condition had significantly higher reading times for both dyads (M ± SD = 27.07 ± 9.33) and parents (M ± SD = 24.77 ± 9.64) compared with the fixed condition (dyads: M ± SD = 16.46 ± 5.86, $P < 3.29 \times 10^{-4}$; parents: M ± SD = 13.22 ± 5.87, $P < 1.08 \times 10^{-4}$) and the strategy-switching condition (dyads: M ± SD = 15.30 ± 7.21, $P < 2.05 \times 10^{-5}$; parents: M ± SD = 13.40 ± 7.77, $P < 7.00 \times 10^{-5}$). In summary, a robot's active engagement led to a decrease in dyad's reading duration and parent's reading duration in the first 15 min of their dialogic reading interaction.

RQ2: Robot participation and behavioral changes in parent-child conversations

We assessed the influence of robot participation on the changes of parent-child conversational quality in their postcurriculum dialogic interaction without the robot's participation. For each of the six conversational quality indicators, we computed the score differences between the pre- and postcurriculum dyadic dialogic sessions for each parent-child pair.

Table 1 presents the changes in average scores for all six conversational quality indicators from pre- to postcurriculum sessions across all conditions. There was an overall increase in scores for all indicators, except for a −0.95% marginal decrease in the child's conversing ratio (conv-ratio-child) within the control group. Notable

Table 1. Dyad's behavioral change in conversational quality after the dialogic curriculum's completion per condition.

	Condition—score change in mean ± SD (%)		
Quality indicator	**Control**	**Fixed-strategy**	**Strategy-switching**
Conv time (dyad)	8.3 ± 9.8 (38.8%)	5.1 ± 15 (19.2%)	8.0 ± 15 (28.9%)
Conv turn (dyad)	3.3 ± 3.0 (76.6%)	2.4 ± 5.0 (37.4%)	2.1 ± 4.1 (31.1%)
Conv time (parent)	5.4 ± 5.9 (36.8%)	3.6 ± 10.5 (21.7%)	6.5 ± 10.3 (39.3%)
Conv ratio (parent)	0.2 ± 0.2 (56.7%)	0.1 ± 0.3 (17.4%)	0.2 ± 0.3 (47.5%)
Conv time (child)	3.5 ± 4.8 (58.2%)	1.6 ± 6.8 (15.7%)	0.5 ± 10 (4.3%)
Conv ratio (child)	0.0 ± 0.3 (−1.0%)	0.0 ± 0.3 (3.1%)	0.2 ± 0.3 (22.2%)

improvements included a 76.58% increase in the average number of turn-takes per conversation (conv-turns-dyad) in the control condition, 37.40% in the fixed-strategy condition, and 31.12% in the strategy-switching condition. The parent's conversing time (conv-time-parent) also increased significantly, by 36.77% in the control condition, 21.65% in the fixed-strategy condition, and 39.28% in the strategy-switching condition, highlighting a universal enhancement in dyadic conversational quality postcurriculum across all conditions. Kruskal-Wallis tests revealed no significant differences in score changes across conditions for any of the six quality indicators, suggesting consistent improvements regardless of robot participation.

RQ3: Role of parent's English proficiency in robot-facilitated parent-child conversations

This analysis examined the influence of the parent's English proficiency on the effects of robot participation on the parent-child conversational quality, using linear mixed-effects model (LMM) tests for each of the six conversational quality indicators. The results, depicted in Fig. 6, highlighted statistically significant findings for the average number of turns per conversation (conv-turns-dyad) and the parent's conversing ratio (conv-ratio-parent).

For conv-turns-dyad, a significant moderation effect of parent's English proficiency was observed [LR(1) = 8.28, $P = 0.024$] with an effect size of Cohen's $f^2 = 0.13$, indicating that the relationship between robot participation and dyadic conversational turns depended on the parent's English proficiency. Specifically, in the strategy-switching condition, dyads with non–native-speaking parents (M ± SD = 4.34 ± 4.40) exhibited more substantial improvements compared with those with native-speaking parents (M ± SD = 0.62 ± 3.29). Conversely, in the fixed-strategy condition, native-speaking parents (M ± SD = 3.55 ± 5.04) showed more pronounced improvements than non–native speakers (M ± SD = 0.58 ± 4.68). The control condition showed no significant difference, maintaining a consistent gap between dyads with native (M ± SD = 3.51 ± 3.21) and non-native (M ± SD = 2.74 ± 2.28) parents, when compared to the other two active robot participation conditions.

For conv-ratio-parent, significant moderation effects were also detected [LR(1) = 6.11, $P = 0.040$] with an effect size of Cohen's $f^2 = 0.09$, demonstrating that the influence of the robot's participation on the parent's conversing ratio varied on the basis of English proficiency. Non–native-speaking parents (M ± SD = 0.33 ± 0.34) in the strategy-switching condition saw greater enhancements in conversing behavior than native speakers (M ± SD = 0.07 ± 0.29). However, this trend was reversed in the fixed-strategy condition, where native speakers (M ± SD = 0.11 ± 0.27) outperformed non–native speakers (M ± SD = −0.01 ± 0.26). The control condition indicated a slight positive trend for both native (M ± SD = 0.13 ± 0.16) and non–native-speaking (M ± SD = 0.22 ± 0.17) parents.

These significant moderation effects highlighted a consistent pattern: The parent's English proficiency systematically affects the influence of robot participation on parent-child conversational dynamics. When engaged with a fixed-strategy robot, dyads with native-speaking parents benefited more regarding conversational turns and conversing ratios than those with non–native-speaking parents. In contrast, with a strategy-switching robot, the advantage shifts toward dyads with non–native-speaking parents. In the control condition, improvements remained relatively parallel between the two groups.

DISCUSSION

This study explores the influence of an in-home learning companion robot on parent-child conversational dynamics during dialogic interactions. Our findings reveal that active robot involvement increases, rather than reduces, conversational opportunities in parent-child interactions. The dialogic interaction curriculum effectively promotes positive behavioral changes across all conditions. Strategy-switching robots benefit dyads with non–native speaker parents more, whereas fixed-strategy robots show greater benefits for native-speaking dyads.

Overall, we demonstrate that social robots can positively influence parental behaviors, enhancing both child behavior and overall parent-child dynamics. These results emphasize the importance of flexible and diverse robot interaction strategies in ensuring equitable benefits across different family profiles. These findings highlight the importance of parent-focused interventions and equitable, long-term interaction design.

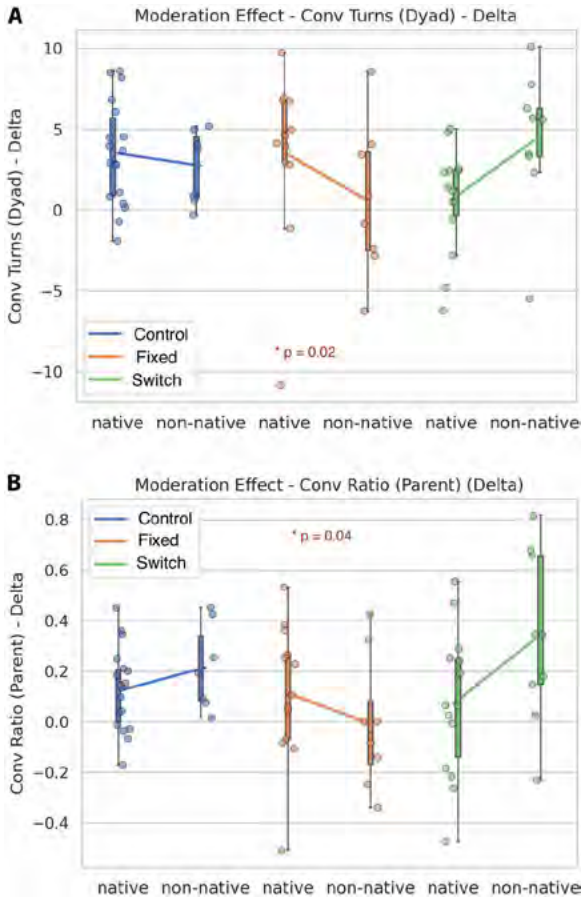


Fig. 6. Influence of parental English proficiency on conversational dynamics. LMM analysis revealed significant moderation effects across conditions (control: $N = 25$, native $N = 18$; fixed-strategy: $N = 22$, native $N = 18$; strategy-switching: $N = 24$, native $N = 18$]. Box plots show the median, interquartile range (IQR), and data range within 1.5 × IQR, with outliers. (**A**) Dyad's average turns per conversation showed significant variation [LR(1) = 8.28, $P = 0.024$], with fixed-strategy robots favoring native speakers and strategy-switching robots benefiting non–native speakers. (**B**) Parents' conversing-to-reading ratio demonstrated similar patterns [LR(1) = 6.11, $P = 0.040$]. The control group showed consistent changes across both language profiles.

RQ1: Enhancing parent-child conversational dynamics through robot interaction

Both active robot types increased dyad's conversing time, with parents showing a notable increase in conversing relative to reading time. This enhancement in conversational behavior was more pronounced among parents than children, with no notable increase observed in the children's conversing behavior when an active robot was present.

Further analysis revealed that dyads engaged less in passive reading during sessions where a robot actively participated in the first 15 min of the activity. The parents, rather than the children, were primarily responsible for this reduction in reading time. It is crucial to note that a decrease in reading time during the initial 15 min does not imply less reading overall; rather, it indicates that reading occurs at a slower pace, allowing for the completion of stories at a rate comparable to that in sessions without robot involvement. These findings address common concerns about the potential for robots to diminish parent-child conversational opportunities. Instead, they demonstrate that robots can enhance the quality of parent-child conversations, suggesting that social robots can be effectively designed to support and enrich parent-child interactions by fostering more active parental involvement in conversation rather than passive reading.

RQ2: Evaluating the long-term influence of robot interaction on parent-child conversational quality

All dyads showed improved conversational dynamics postcurriculum, with increased scores across all quality indicators. This positive influence was observed across all conditions, regardless of the presence or type of robot facilitation during the coreading sessions. Such findings confirms the overall effectiveness of our dialogic reading curriculum consisting of instructional videos and reading sessions in enhancing parent-child conversational quality in their own dyadic interaction even with the absence of the robot's active participation.

There were no notable differences in the magnitude of behavioral improvements among dyads interacting with an active robot versus those without or between dyads engaging with fixed-strategy robots versus strategy-switching robots. This finding can be attributed to the complex interplay of factors unique to individual dyads. These factors, such as language barriers and learning challenges, may have a more notable influence on behavioral improvements than the type of robot participation alone. When analyzing all dyads collectively, without accounting for specific family profiles critical to parent-child conversational interaction, such as parent's language proficiency, these individual-based factors may effectively cancel out and absorb any potential differences in improvements or benefits that dyads might derive from interacting with different robot styles.

RQ3: Improvement variability in parent-child conversational quality based on parental English proficiency

Parental English proficiency notably influenced improvements in conversational quality across conditions, particularly in turn-taking and parental conversing ratios. A consistent pattern emerged from the data, indicating that dyads with non–native-speaking parents derive more benefit from interactions with a strategy-switching robot than those with native-speaking parents. In contrast, native-speaking dyads experienced greater advantages from interactions with a fixed-strategy robot compared with their non–native-speaking counterparts. This observation suggests a reversal in the direction of English proficiency's moderating effect on positive behavioral changes between the strategy-switching and fixed-strategy robot interactions. When dyads did not interact with a robot, the influence on behavioral improvements remained consistent across both native-speaking and non–native-speaking groups, highlighting no notable differences in outcomes. This uniformity further supports the idea that the divergent moderation relationships observed between fixed- and strategy-switching robot conditions may be due to the robots switching their behavior strategies during interactions. When interacting with a fixed-strategy robot, dyads need to adjust to the robot's predetermined patterns, which may be particularly challenging for non–native English–speaking parents. The cognitive load of conversing in a non-native language while adapting to the robot's fixed interactions may amplify these difficulties. Conversely, a strategy-switching robot dynamically adapts to the dyad's real-time behaviors. This accommodation may particularly benefit non–native English–speaking parent-child pairs, potentially explaining the outcome differences between the two interactive robot conditions across user groups.

These findings emphasize the importance of flexibility of the interaction design in robotic systems, especially for non-native parents, who may face challenges with English-centric dialogic reading strategies. For instance, a strategy-switching robot could pose high-quality dialogic questions initially and then shift to a moderating role to encourage more creative conversations. Such strategic flexibility is crucial for supporting learning in dyads facing substantial language barriers, unlike fixed-strategy robots that demand that dyads adapt to a static set of behaviors.

This investigation highlights that although robotic interventions have the potential to enhance conversational dynamics across all dyads, the design and implementation of robot behaviors are critical in determining the extent of behavioral improvements and identifying which population groups benefit the most. Furthermore, it is crucial to recognize that the flexibility of robot interaction styles—such as the ability to switch among diverse behavioral strategies in real time—can substantially influence which user groups and populations derive the most benefit from these interactions. These insights underscore the urgent need for tailored robotic interventions that account for individual differences and demographic profiles in early childhood education and parent-child conversational enhancement.

Empowering parents in parent-child conversations through robot interaction

Traditionally, social agents have primarily targeted one-on-one interactions with children or multiperson interactions focusing on children. However, our research introduces an underexplored paradigm: leveraging the potential of social robots to enhance parent-child interactions. Upon completing the dialogic curriculum, we observed notable improvements in conversational behaviors at both individual (parent and child) and dyadic levels, as shown in RQ2. These enhancements were predominantly driven by changes in the parents' behaviors, underscoring the critical role of parental involvement. Our findings from RQ1 further highlight that a robot's active participation substantially influences parents' conversational behaviors more than children's. In addition, results from RQ3 indicate that a parent's English proficiency critically moderates the extent of positive behavioral changes induced by the robot, affecting both the parent's and the dyad's conversational dynamics.

These analysis results show that the robot's active participation had a more pronounced influence on parents than on children. This

outcome is consistent with the dynamics of dialogic coreading activities, where parents, especially those of children aged 3 to 7, typically lead the activity. Given that these children are usually in the early stages of learning to read and are not yet fluent, the parent's role in guiding the interaction is pivotal.

Our study illustrates the potential of designing social robots as effective interventions that enhance parental dialogic behaviors, thereby fostering positive behavioral changes in parent-child interactions. This insight marks a notable shift in the design paradigm for early childhood education robots. Previous research has primarily focused on targeting children directly in the home environment (*73*, *74*), often overlooking the role of parents or relegating them to a secondary position in child-robot interactions. In contrast, our research highlights the importance of focusing on parents in the development of future robotic interventions within parent-child interactions, recognizing their crucial role in a child's development. This perspective paves the way for further research in social robotics within early childhood education, advocating for innovative robot interaction designs and systems that empower parents to actively engage in and enhance their children's at-home learning experiences.

Equitable design considerations in long-term robot interaction

The contrasting relationships observed between a dyad's parental language proficiency and their improved conversational behaviors in strategy-switching versus fixed-strategy robot conditions, as noted in RQ3, highlight the importance of equitable design in long-term social robotic interventions. Our findings with the fixed-strategy robot emphasize that although a robot with a consistent behavior strategy may foster positive behavioral changes across multiple sessions as noted in RQ2, it might disproportionately benefit dyads with fewer learning barriers as noted in RQ3, potentially widening the learning gap between dyads of varying backgrounds.

It is important to clarify that the fixed-strategy robot in our study was not confined to repetitive behaviors. Instead, it demonstrated a diverse array of behaviors, each tailored and contextually relevant to the specific narrative of the story page it interacted with. The term "fixed strategy" refers to the consistent robot role and communication style maintained throughout a session, such as adopting a demonstrator role with a verbal communication style.

In designing robot-facilitated learning interventions, it is crucial to recognize the importance of offering diverse interaction modes and strategies during extended interactions. A robot capable of switching its interaction strategies provides a more flexible and diverse approach that can notably benefit dyads facing greater challenges, such as those with limited English proficiency. By offering flexible, needs-based support, the strategy-switching robots have the potential to bridge, rather than widen, the learning gap among diverse dyads. This flexibility is particularly crucial for ensuring equitable benefits across parent-child dyads, potentially more so in prolonged engagements than in brief encounters.

These insights also stress the need for nuanced robot interaction design and underscore the necessity to evaluate relative benefits across diverse user groups. As we design robot interactions, it is essential to delve into the nuances of individual differences among dyads, uncovering any hidden differential influences beneath the overall positive effects observed on various populations and individuals. Without thoughtful design and user-profile–specific analysis, social robotics technology, although generally beneficial, could inadvertently exacerbate social and learning disparities.

Limitations

Our study has several limitations that should be considered when interpreting the results. First, the pre- and postcurriculum assessments were conducted as single sessions because of resource constraints, which may not fully capture long-term behavioral changes. Future studies should include multiple assessment points over an extended period to provide a more comprehensive understanding of the intervention's influence. Second, our study focused solely on parent-child interactions during structured dialogic reading activities. We did not examine the potential spillover effects on daily conversations outside these sessions. Future research should investigate how robot-facilitated interventions might influence broader family communication patterns. Last, our study was limited to dyadic parent-child interactions and did not consider the broader family ecosystem, including siblings or other caregivers. To fully understand the potential of social robots as conversational catalysts in home environments, future studies should explore their influence on parental interactions with multiple children and their integration into diverse family structures.

Although social robots are now less accessible than other digital tools, our findings on interaction design, strategies, and conversational styles are applicable to diverse platforms, including emerging low-cost options. As the market for affordable social robots grows, our insights can guide the development of more accessible technologies that support family interactions across diverse backgrounds.

MATERIALS AND METHODS

Experiment objective and design

Study protocol

Our between-participant experiment was designed to assess the research questions under three distinct conditions: control, fixed-strategy robot, and strategy-switching robot. The control condition served as a baseline to compare against the two active robot participation conditions. The fixed-strategy robot exemplified a typical design for robot interaction in short-term deployments with brief encounters, where a single behavior strategy guided the robot's actions (such as adopting a specific role like tutor or tutee during the activity). This fixed-strategy condition provided a direct comparison with the strategy-switching robot condition to evaluate whether real-time switching among diverse robot roles and communication styles influenced parent-child conversational dynamics and long-term behavioral changes.

As depicted in Fig. 2, we placed robots in the homes of participant families for a period ranging from 4 to 8 weeks. Each dyad followed a structured schedule that included a precurriculum parent-child dyadic coreading session, six regular sessions as part of the dialogic curriculum, and a concluding postcurriculum dyadic session. Each session typically lasted between 15 and 30 min. All participants received the same robot system and used a tablet mounted on the robot station for reading stories to ensure consistency across the study. During the six designated coreading sessions of the dialogic reading curriculum, the robot interactions varied on the basis of the assigned condition. In the control condition, participants read stories without active robot involvement. In both robot conditions, the robot actively participated in six coreading sessions with parents and children. However, the pre- and postcurriculum dyadic coreading sessions involved only the dyads, with no robot participation.

The study was approved by the Institutional Review Board (IRB) of the Massachusetts Institute of Technology (reference number

1811583321). Each participant's parent signed a form providing consent for both themselves and their child to participate in the study.

Dialogic reading curriculum

Before each dialogic curriculum session, parents in all three conditions were provided access to a 4-min instructional video on effective dialogic reading techniques (see Supplementary Methods). These videos, produced by education specialists from our lab, were designed to introduce parents to foundational dialogic reading practices grounded in the dialogic reading protocols introduced by Whitehurst and Lonigan (*26*), which also guided the robot's interactive design. Each video featured a researcher explaining a specific dialogic reading strategy with practical examples. These videos were emailed to participants before each of the six intervention sessions.

The inclusion of these instructional videos aimed to standardize knowledge about dialogic reading techniques across all groups, thereby isolating the effect of the robot's active participation from potential knowledge gains in the study's outcomes. By ensuring that all participants received the same instructional content, we could more accurately determine whether observed changes in parent-child conversational behaviors were due to the robot's active involvement rather than merely the informational influence similar to that provided by a tablet or phone.

Participants

We recruited 86 US families with children aged 3 to 7 for an at-home study during the 2022–2023 academic year. We took comprehensive steps to ensure a diverse participant pool through targeted recruitment. Each dyad consisted of one parent and one child throughout the study. Out of the initial recruitment, 71 families successfully completed the study, and 15 withdrew for reasons unrelated to robot interaction, as detailed in Supplementary Methods. The final analysis included 25, 22, and 24 families in the control, fixed-strategy, and strategy-switching robot conditions, respectively. Of these, 18, 14, and 14 parents were native English speakers in the control, fixed-strategy, and strategy-switching robot conditions, respectively. Kruskal-Wallis tests confirmed no significant differences in baseline characteristics across conditions.

Integrated robot system

A social robot named Jibo, an Intel Next Unit of Computing machine, an Android tablet, and a web camera set above the tablet made up the integrated robot station (Fig. 1) (*75*). The Android tablet provided an e-book app loaded with more than 20 storybooks recommended by early childhood education experts and teachers. The same app with the same story corpus had been used in prior parent-child dyadic coreading work (*34*, *76*). Dyads had the freedom to select any book from the corpus to read with the robot during a coreading session. This robot system had been tested and evaluated in a prior real-world deployment study that we conducted (*34*).

Our robot system used a balanced approach to autonomy and flexibility in the dialogic interaction activity, designed to minimize confounding effects while ensuring reliable interactions in early childhood education settings. The robot operated with low-level autonomy, responding to easily detectable cues like page turns, and used prewritten, educationally vetted speech scripts for content reliability. This design choice prioritized precision, safety, and responsible deployment over full, flexible autonomy, ensuring that the robot's interactions remained predictable, child-friendly, and free from potential confounds that could affect the study's outcomes.

Robot interaction design

The robot's behavior strategies for both fixed-strategy and strategy-switching conditions were developed along two primary design dimensions, support role and communication style, which were grounded in prior multiperson human-robot interaction research.

Robot roles

We designed three distinct robot support roles to facilitate parent-child dialogic interaction. Overall, the demonstrator focused on the pedagogical aspect, coaching parents on dialogic questioning. The playmate emphasized the social and affective elements. The moderator served as a neutral facilitator, encouraging dialogue without dictating its direction. This placed the moderator role between the pedagogical demonstrator and the social playmate on a continuum. The specific design rationales and the anticipated benefits of the three roles are outlined below.

Drawing from techniques where a human expert provides real-time coaching (*30*, *31*), the demonstrator role involved the robot offering direct instructions and feedback, akin to a tutor tailored to each dyad. This method, rooted in demonstration coaching techniques (*28*, *29*), aimed to enhance parents' question-asking behaviors by modeling positive reading behaviors, such as asking dialogic questions, thereby encouraging parents to adopt these practices in their coreading interactions. It also provided opportunities for dyads to answer high-quality dialogic questions.

When taking on a moderator role, the robot acted as a facilitator that encouraged parent-child pairs to initiate dialogic questions and discussions. By subtly suggesting opportunities for interaction without directly demonstrating, in this role the robot aimed to foster more frequent and diverse parent-child exchanges, enhancing the overall quality of the coreading experience.

A playmate role enabled the robot to adopt a child-like persona and engage in curious and playful interactions, supporting the "learning-by-teaching" approach (*77*–*82*). This role was designed to boost the enjoyment and engagement of both parent and child, promoting deeper story comprehension and vocabulary learning through interactive storytelling and question-asking.

Robot communication styles

We designed two robot communication styles by differentiating between verbal and nonverbal robot intents. The verbal style involved the robot directly participating in the conversation by asking questions or making comments without seeking prior approval, aiming for immediate and proactive engagement with the dyad. The nonverbal style was characterized by the robot's use of nonverbal cues, such as eye contact, body rotation, and leaning forward, to signal its intent to speak without actually verbalizing unless explicitly permitted by the dyad. This style respected the dyad's autonomy, allowing them to choose whether to engage with the robot's contributions.

The fundamental difference between these two styles lay in who initiated the robot-facilitated parent-child conversation: In the verbal style, the robot took the lead, whereas in the nonverbal style, the initiative rested with the dyad. Combining these support roles and communication styles, we designed six unique role-style strategies: verbal-demonstrator, nonverbal-demonstrator, verbal-moderator, nonverbal-moderator, verbal-playmate, and nonverbal-playmate, all enriched with animated speech and nonverbal cues.

Robot's strategy-switching policy

The robot's strategy-switching policy was grounded in the reinforcement learning (RL) framework and built on our prior theoretical work on the conceptual framework for dynamic role switching (*53*)

and our empirical work on strategy-switching for children's language learning (*70*). This computational implementation enabled our robot to dynamically alternate among six role-style behavior strategies during real-time interaction. The use of RL to incorporate users' affective and/or cognitive data for adapting robot behaviors had been widely explored in prior work (*70*, *83*, *84*). Following this established approach, we adapted and customized the RL framework to suit the specific needs and contexts of our work.

We implemented our strategy-switching policy using reward machine-based Q-learning (*85*), a particular implementation of the RL framework demonstrated to be effective for agent interaction scenarios consisting of multiple subpolicies. In our case, the Markov decision process encoded short-term interactions, and the reward machine captured long-term cognitive skills and affective moods.

Our robotic system's affective-cognitive perception module continuously monitored the dyad's behavior-based affects, such as facial expressions, and conversational patterns, such as speaker's utterances. Both the affective and dialogic interaction perception submodules were developed or adapted from our previous work (*34*, *86*, *87*). The user affective and dialogic behavior data detected in real time through the perception model were fed into the RL-based strategy-switching policy.

The policy was initially trained in simulation using data from our pilot study (*34*). In real-world deployment, it switched the robot's behavior strategy in response to each group in real time. More details on the strategy-switching policy are provided in Supplementary Methods.

Data collection and feature extraction

Coreading sessions were video and audio recorded using the robot station's integrated camera. Recordings were limited to coreading activities and professionally transcribed. The transcripts included detailed speaker identifications and time stamps for each turn of speech, facilitating a nuanced analysis of each interlocutor's conversational behaviors. The conversational quality indicators were extracted from the transcripts for the parent, the child, and the dyad as a group in each coreading session.

Following the framework suggested by Whitehurst and Lonigan (*26*), we identified three key dimensions of effective parent-child shared reading: dialogic reading, which involves discussions between parent and child alongside story reading; high child participation, which involves substantial portions of child reading and speaking; and active parent-child turn-taking, which involves frequent switching of reading or speaking turns between child and parent. Accordingly, we extracted six conversational quality indicators—two each for the parent, child, and dyad—focusing on both the conversational duration and the dynamics of reading-conversing interactions within each stakeholder. Reading duration for each of the three stakeholders was also extracted as the supplementary reading feature.

Statistical analysis

We assessed distribution normality and variance homogeneity using Shapiro and Levene tests. Kruskal-Wallis tests evaluated differences across experimental conditions for RQ1 and RQ2. In exploring RQ3, we used an LMM to examine the moderation effect of the parent's English proficiency on dyadic behavioral changes. Adjustments for multiple comparisons in our analyses were made using the Benjamini-Hochberg procedure, ensuring the control of the false discovery rate. Results yielding an adjusted P value below 0.05 were deemed statistically significant and are reported accordingly.

Supplementary Materials

The PDF file includes:
Materials and Methods
Tables S1 to S12
Figs. S1 to S4
References (*88*, *89*)

Other Supplementary Material for this manuscript includes the following:
Data files S1 and S2
MDAR Reproducibility Checklist

REFERENCES AND NOTES

1. S. Turkle, "Why these friendly robots can't be good friends to our kids," *The Washington Post* (7 December 2017); https://www.washingtonpost.com/outlook/why-these-friendly-robots-cant-be-good-friends-to-our-kids/2017/12/07/bce1eaea-d54f-11e7-b62d-d9345ced896d_story.html.
2. M. H. Bornstein, C. S. Tamis-LeMonda, Maternal responsiveness and cognitive development in children. *New Dir. Child Adolesc. Dev.* **1989**, 49–61 (1989).
3. S. A. Denham, J. Mitchell-Copeland, K. Strandberg, S. Auerbach, K. Blair, Parental contributions to preschoolers' emotional competence: Direct and indirect effects. *Motiv. Emot.* **21**, 65–86 (1997).
4. S. A. Denham, J. Mitchell-Copeland, K. Strandberg, S. Auerbach, K. Blair, "Value socialization in a bidirectional context," in *Parenting and Children's Internalization of Values: A Handbook of Contemporary Theory* (Wiley & Sons, 1997), pp. 23–50.
5. S. H. Landry, K. E. Smith, P. R. Swank, C. L. Miller-Loncar, Early maternal and child influences on children's later independent cognitive and social functioning. *Child Dev.* **71**, 358–375 (2000).
6. S. Roseberry, K. Hirsh-Pasek, R. M. Golinkoff, Skype me! Socially contingent interactions help toddlers learn language *Child Dev.* **85**, 956–970 (2014).
7. C. S. Tamis-LeMonda, M. H. Bornstein, L. Baumwell, Maternal responsiveness and children's achievement of language milestones. *Child Dev.* **72**, 748–767 (2001).
8. C. S. Tamis-Lemonda, M. H. Bornstein, L. Baumwell, A. M. Damast, Responsive parenting in the second year: Specific influences on children's language and play. *Early Hum. Dev.* **5**, 173–183 (1996).
9. B. Hart, T. R. Risley, *Meaningful Differences in the Everyday Experience of Young American Children* (Paul H. Brookes Publishing, 1995).
10. M. Rowe, Child-directed speech: Relation to socioeconomic status, knowledge of child development and child vocabulary skill. *J. Child Lang.* **35**, 185–205 (2008).
11. J. Gilkerson, J. A. Richards, S. F. Warren, J. K. Montgomery, C. R. Greenwood, D. Kimbrough Oller, J. H. L. Hansen, T. D. Paul, Mapping the early language environment using all-day recordings and automated analysis. *Am. J. Speech Lang. Pathol.* **26**, 248–265 (2017).
12. S. B. Neuman, D. C. Celano, Worlds apart: One city, two libraries, and ten years of watching inequality grow. *Am. Educ.* **36**, 13–23 (2012).
13. S. B. Neuman, D. C. Celano, *Giving Our Children a Fighting Chance: Poverty, Literacy, and the Development of Information Capital*. (Teachers College Press, 2015).
14. K. V. Hoover-Dempsey, H. M. Sandler, Parental involvement in children's education: Why does it make a difference? *Teach. Coll. Rec.* **97**, 310–331 (1995).
15. C. A. Peterson, G. J. Luze, E. M. Eshbaugh, H.-J. Jeon, K. R. Kantz, Enhancing parent-child interactions through home visiting: Promising practice or unfulfilled promise? *J. Early Interv.* **29**, 119–140 (2007).
16. K. McNab, R. Fielding-Barnsley, Digital texts, iPads, and families: An examination of families' shared reading behaviours. *Intern. J. Learn. Annu. Rev.* **20**, 53–62 (2013).
17. L. Takeuchi, R. Stevens, "The new coviewing: Designing for learning through joint media engagement," report from the Joan Ganz Cooney Center at Sesame Workshop (Joan Ganz Cooney Center, 2011); https://joanganzcooneycenter.org/wp-content/uploads/2014/03/JGC_CoViewing_iPadPrint.pdf.
18. A. Chang, C. Breazeal, "TinkRBook: Shared reading interfaces for storytelling," in *IDC '11: Proceedings of the 10th International Conference on Interaction Design and Children* (ACM, 2011), pp. 145–148.
19. M. Hood, E. Conlon, G. Andrews, Preschool home literacy practices and childrens literacy development: A longitudinal analysis. *J. Educ. Psychol.* **100**, 252–271 (2008).
20. L. M. Baker, K. Mackler, S. Sonnenschein, R. Serpell, Parents interactions with their first-grade children during storybook reading and relations with subsequent home reading activity and reading achievement. *J. Sch. Psychol.* **39**, 415–438 (2001).
21. E. Hoff, C. Core, S. Place, R. Rumiche, M. Señor, M. Parra, Dual language exposure and early bilingual development. *J. Child Lang.* **39**, 1–27 (2012).
22. E. Bialystok, G. Luk, K. F. Peets, Y. Sujin, Receptive vocabulary differences in monolingual and bilingual children. *Biling. Lang. Congn.* **13**, 525–531 (2010).
23. B. A. Collins, Dual language development of Latino children: Effect of instructional program type and the home and school language environment. *Early Child. Res. Q.* **29**, 389–397 (2014).

24. C. S. Hammer, E. Hoff, Y. Uchikoshi, C. Gillanders, D. C. Castro, L. E. Sandilos, The language and literacy development of young dual language learners: A critical review. *Early Child. Res. Q.* **29**, 715–733 (2014).
25. S. E. Davis, "An enhanced dialogic reading approach to facilitate typically developing pre-school childrens emergent literacy skills," thesis, East Tennessee State University (2004); https://dc.etsu.edu/etd/867/.
26. G. J. Whitehurst, C. J. Lonigan, Child development and emergent literacy. *Child Dev.* **69**, 848–872 (1998).
27. A. A. Zevenbergen, G. J. Whitehurst, "Dialogic reading: A shared picture book reading intervention for preschoolers," in *On Reading Books to Children* (Lawrence Erlbaum Associates, 2003).
28. J. Blom-Hoffman, T. O'Neil-Pirozzi, R. J. Volpe, J. Cutting, Instructing parents to use dialogic reading strategies with preschool children. *J. Appl. Sch. Psychol.* **23**, 117–131 (2007).
29. C. E. Huebner, A. N. Meltzoff, Intervention to change parent-child reading style: A comparison of instructional methods. *J. Appl. Dev. Psychol.* **26**, 296–313 (2005).
30. J. A. McCollum, Dyad as focus, triad as means: A family-centered approach to supporting parent-child interactions. *Infants Young Child.* **6**, 54–63 (1994).
31. J. Han, S. Neuharth-Pritchett, Parents interactions with preschoolers during shared book reading: Three strategies for promoting quality interactions. *Childh. Educ.* **90**, 54–60 (2014).
32. J. Parish-Morris, N. Mahajan, K. Hirsh-Pasek, R. Golinkoff, M. Collins, Once upon a time: Parent-child dialogue and storybook reading in the electronic era. *Mind Brain Educ.* **7**, 200–211 (2013).
33. K.-H. Cheng, C.-C. Tsai, The interaction of child-parent shared reading with an augmented reality (AR) picture book and parents conceptions of AR learning. *Br. J. Educ. Technol.* **47**, 203–222 (2016).
34. H. Chen, A. K. Ostrowski, S. J. Jang, C. Breazeal, H. W. Park, "Designing long-term parent-child-robot triadic interaction at home through lived technology experiences and interviews," in *31st IEEE International Conference on Robot & Human Interactive Communication (RO-MAN2022)* (IEEE, 2021).
35. T. Belpaeme, J. Kennedy, A. Ramachandran, B. Scassellati, F. Tanaka, Social robots for education: A review. *Sci. Robot.* **3**, eaat5954 (2018).
36. O. Gvirsman, Y. Koren, T. Norman, G. Gordon, "Patricc: A platform for triadic interaction with changeable characters," in *ACM/IEEE International Conference on Human-Robot Interaction* (ACM, 2020).
37. S. Shen, P. Slovak, M. F. Jung, "'Stop. I see a conflict happening': A robot mediator for young children's interpersonal conflict resolution," in *ACM/IEEE International Conference on Human-Robot Interaction* (ACM, 2018), pp. 69–77.
38. E. S. Short, University of Southern California, L. Angeles, CA, USA, K. Swift-Spong, H. Shim, K. M. Wisniewski, D. K. Zak, S. Wu, "Understanding social interactions with socially assistive robotics in intergenerational family groups," in *IEEE International Symposium on Robot and Human Interactive Communication* (IEEE, 2017), pp. 236–241.
39. S. Strohkorb, I. Leite, N. Warren, B. Scassellati, "Classification of children's social dominance in group interactions with robots," in *ICMI '15: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction* (ACM, 2015), pp. 227–234.
40. I. Leite, M. McCoy, D. Ullman, N. Salomons, B. Scassellati, "Comparing models of disengagement in individual and group interactions," in *ACM/IEEE International Conference on Human-Robot Interaction* (IEEE, 2015).
41. O. Rudovic, J. Lee, L. Mascarell-Maricic, B. W. Schuller, R. W. Picard, Measuring engagement in robot-assisted autism therapy: A cross-cultural study. *Front. Robot. AI.* **4**, 36 (2017).
42. M.-Y. Chan, Y.-H. Lin, L.-F. Lin, T.-W. Lin, W.-C. Hsu, C.-y. Chang, R. Liu, K.-Y. Chang, M.-h. Lin, J. Y.-j. Hsu, "WAKEY: Assisting parent-child communication for better morning routines," in *CSCW '17: Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing* (ACM, 2017), pp. 2287–2299.
43. D. Utami, T. Bickmore, "Collaborative user responses in multiparty interaction with a couples counselor robot," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (IEEE, 2019), pp. 294–303.
44. H. Tennent, S. Shen, M. Jung, "micbot: A peripheral robotic object to shape conversational dynamics and team performance," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (IEEE, 2019), pp. 133–142.
45. M. F. Jung, N. Martelaro, P. J. Hinds, "Using robots to moderate team conflict: The case of repairing violations," in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction* (ACM, 2015), pp. 229–236.
46. M. R. Fraune, S. Šabanović, T. Kanda, Human group presence, group characteristics, and group norms affect human-robot interaction in naturalistic settings. *Front. Robot. AI.* **6**, 48 (2019).
47. M. Žarkowski, Multi-party turn-taking in repeated human-robot interactions: An interdisciplinary evaluation. *Int. J. Soc. Robot.* **11**, 693–707 (2019).
48. H. Claure, Y. Chen, J. Modi, M. Jung, S. Nikolaidis, "Multi-armed bandits with fairness constraints for distributing resources to human teammates," in *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction* (ACM, 2020), pp. 299–308.
49. M. L. Traeger, S. S. Sebo, M. Jung, B. Scassellati, N. A. Christakis, Vulnerable robots positively shape human conversational dynamics in a human-robot team. *Proc. Natl. Acad. Sci. U.S.A.* **117**, 6370–6375 (2020).
50. G. Hoffman, O. Zuckerman, G. Hirschberger, M. Luria, T. Shani Sherman, "Design and evaluation of a peripheral robotic conversation companion," in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction* (IEEE, 2015).
51. D. Druckman, L. Adrian, M. F. Damholdt, M. Filzmoser, S. T. Koszegi, J. Seibt, C. Vestergaard, Who is best at mediating a social conflict? Comparing robots, screens and humans. *Group Decis. Negot.* **30**, 395–426 (2020).
52. C. Birmingham, Z. Hu, K. Mahajan, E. Reber, M. Mataric, "Can I trust you? A user study of robot mediation of a support group," in *2020 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2020), pp. 8019–8026.
53. H. Chen, S. Alghowinem, C. Breazeal, H. W. Park, "Integrating flow theory and adaptive robot roles: A conceptual model of dynamic robot role adaptation for the enhanced flow experience in long-term multi-person human-robot interactions," in *2024 19th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* [Association for Computing Machinery (ACM), 2024], pp. 389–397.
54. F. Correia, S. Mascarenhas, R. Prada, F. S. Melo, A. Paiva, "Group-based emotions in teams of humans and robots," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction* (ACM, 2018), pp. 261–269.
55. S. Strohkorb Sebo, M. Traeger, M. Jung, B. Scassellati, "The ripple effects of vulnerability: The effects of a robot's vulnerable behavior on trust in human-robot teams," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction* (ACM, 2018), pp. 178–186.
56. Y. Matsuyama, I. Akiba, S. Fujie, T. Kobayashi, Four-participant group conversation: A facilitation robot controlling engagement density as the fourth participant. *Comput. Speech Lang.* **33**, 1–24 (2015).
57. S. Sebo, L. L. Dong, N. Chang, M. Lewkowicz, M. Schutzman, B. Scassellati, The influence of robot verbal support on human team members: Encouraging outgroup contributions and suppressing ingroup supportive behavior. *Front. Psychol.* **11**, 3584 (2020).
58. P. Liu, D. F. Glas, T. Kanda, H. Ishiguro, N. Hagita, "It's not polite to point: Generating socially-appropriate deictic behaviors towards people," in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (ACM, 2013), pp. 267–274.
59. B. Mutlu, T. Shiwa, T. Kanda, H. Ishiguro, N. Hagita, "Footing in human-robot conversations: How robots might shape participant roles using gaze cues," in *Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction* (IEEE, 2009), pp. 61–68.
60. G. Skantze, "Predicting and regulating participation equality in human-robot conversations: Effects of age and gender," in *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction* (IEEE, 2017), pp. 196–204.
61. H. Kidokoro, T. Kanda, D. Brščic, M. Shiomi, "Will I bother here? - A robot anticipating its influence on pedestrian walking comfort," in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (IEEE, 2013), pp. 259–266.
62. C. Mavrogiannis, A. M. Hutchinson, J. Macdonald, P. Alves-Oliveira, R. A. Knepper, "Effects of distinct robot navigation strategies on human behavior in a crowded environment," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (IEEE, 2019), pp. 421–430.
63. M. Vázquez, E. J. Carter, B. M. Dorman, J. Forlizzi, A. Steinfeld, S. E. Hudson, "Towards robot autonomy in group conversations: Understanding the effects of body orientation and gaze," in *2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (IEEE, 2017), pp. 42–52.
64. M. Shiomi, T. Kanda, H. Ishiguro, N. Hagita, "A larger audience please encouraging people to listen to a guide robot," in *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (ACM, 2010), pp. 31–38.
65. W. Ju, L. Leifer, The design of implicit interactions: Making interactive systems less obnoxious. *Des. Issues.* **24**, 72–84 (2008).
66. J. Fink, S. Lemaignan, P. Dillenbourg, P. Rétornaz, F. Vaussard, A. Berthoud, "Which robot behavior can motivate children to tidy up their toys? Design and evaluation of ranger," in *Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction* (ACM, 2014), pp. 439–446.
67. N. Jarrassé, V. Sanguineti, E. Burdet, Slaves no longer: Review on role assignment for human–robot joint motor action. *Adapt. Behav.* **22**, 70–82 (2014).
68. Y. Li, K. P. Tee, W. L. Chan, R. Yan, Y. Chua, D. K. Limbu, Continuous role adaptation for human–robot shared control. *IEEE Trans. Robot.* **31**, 672–681 (2015).
69. A. Mörtl, M. Lawitzky, A. Kucukyilmaz, M. Sezgin, C. Basdogan, S. Hirche, The role of roles: Physical cooperation between humans and robots. *Int. J. Rob. Res.* **31**, 1656–1674 (2012).
70. H. Chen, H. W. Park, C. Breazeal, Teaching and learning with children: Impact of reciprocal peer learning with a social robot on children's learning and emotive engagement. *Comput. Educ.* **150**, 103836 (2020).
71. R. Ros, E. Oleari, C. Pozzi, F. Sacchitelli, D. Baranzini, A. Bagherzadhalimi, A. Sanna, Y. Demiris, A motivational approach to support healthy habits in long-term child-robot interaction. *Int. J. Soc. Robot.* **8**, 599–617 (2016).

72. C. Howard, P. Jordan, B. Di Eugenio, S. Katz, Shifting the load: A peer dialogue agent that encourages its human collaborator to contribute more to problem solving. *Int. J. Artif. Intell. Educ.* **27**, 101–129 (2017).
73. B. Scassellati, L. Boccanfuso, C.-M. Huang, M. Mademtzi, M. Qin, N. Salomons, P. Ventola, F. Shic, Improving social skills in children with ASD using a long-term, in-home social robot. *Sci. Robot.* **3**, eaat7544 (2018).
74. J. E. Michaelis, B. Mutlu, Reading socially: Transforming the in-home reading experience with a learning-companion robot. *Sci. Robot.* **3**, eaat5999 (2018).
75. J. Van Camp, "Review: Jibo social robot," *Wired Magazine* (7 November 2017); https://www.wired.com/2017/11/review-jibo-social-robot/.
76. H. Chen, A. K. Ostrowski, S. J. Jang, C. Breazeal, H. W. Park, Dyadic affect in parent-child multimodal interaction: Introducing the dami-p2c dataset and its preliminary analysis. *IEEE Trans. Affect. Comput.* **14**, 3345–3361 (2023).
77. R. D. Roscoe, M. T. H. Chi, Understanding tutor learning: Knowledge-building and knowledge-telling in peer tutors explanations and questions. *Rev. Educ. Res.* **77**, 534–574 (2007).
78. F. Tanaka, S. Matsuzoe, Children teach a care-receiving robot to promote their learning: Field experiments in a classroom for vocabulary learning. *J. Hum. Robot Interact.* **3**, 74–99 (2012).
79. H. W. Park, A. M. Howard, "Robot learners: Interactive instance-based learning and its application to therapeutic tasks," in *AI for Human-Robot Interaction, 2014 AAAI Fall Symposium* (AAAI, 2014).
80. D. Chin, I. Dohmen, D. Schwartz, Young children can learn scientific reasoning with teachable agents. *IEEE Trans. Learn. Technol.* **6**, 248–257 (2013).
81. G. Biswas, K. Leelawong, D. Schwartz, N. Vye, The Teachable Agents Group at Vanderbilt, Learning by teaching: A new agent paradigm for educational software. *Appl. Artif. Intell.* **19**, 363–392 (2005).
82. S. Brophy, G. Biswas, T. Katzlberger, J. Bransford, D. Schwartz, "Teachable agents: Combining insights from learning theory and computer science," in *Artificial Intelligence in Education*, S. P. Lajoie, M. Vivet, Eds. (IOS Press, 1999), pp. 21–28.
83. G. Gordon, S. Spaulding, J. Kory Westlund, J. Lee, L. Plummer, M. Martinez, M. Das, C. Breazeal, "Affective personalization of a social robot tutor for children's second language skills," in *Proceedings of the AAAI Conference on Artificial Intelligence* (AAAI, 2016), vol. 30.
84. H. W. Park, I. Grover, S. Spaulding, L. Gomez, C. Breazeal, "A model-free affective reinforcement learning approach to personalization of an autonomous social robot companion for early literacy education," in *Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33* (AAAI, 2019).
85. R. T. Icarte, T. Klassen, R. Valenzano, S. McIlraith, "Using reward machines for high-level task specification and decomposition in reinforcement learning," in *Proceedings of the 35th International Conference on Machine Learning (ICML)* (MLResearchPress, 2018), pp. 2107–2116.
86. Y. Kim, H. Chen, S. Algohwinem, C. Breazeal, H. W. Park, "Joint engagement classification using video augmentation techniques for multi-person HRI in the wild," in *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems* (International Foundation for Autonomous Agents and Multiagent Systems, 2023), pp. 698–707.
87. Y. Kim, D. W. Lee, P. P. Liang, S. Alghowinem, C. Breazeal, H. W. Park, "HIINT: Historical, intra-and inter-personal dynamics modeling with cross-person memory transformer," in *Proceedings of the 25th International Conference on Multimodal Interaction* (ACM, 2023), pp. 314–325.
88. R. Toro Icarte, T. Q. Klassen, R. Valenzano, S. A. McIlraith, Reward machines: Exploiting reward function structure in reinforcement learning. *J. Artif. Intell. Res.* **73**, 173–208 (2022).
89. R. S. Sutton, A. G. Barto, *Reinforcement Learning: An Introduction* (MIT Press, ed. 2, 2018).

Acknowledgments: We thank S. Alghowinem, M. Xi, F. Hu, X. Zhang, and others who shared feedback on our project. **Funding:** This work was supported by the Inclusive AI Literacy and Learning Research Grant from DP World (award no. 021898-00342) and the IITP grant funded by the Korea government (MSIT) (no. 2020-0-00842). **Author contributions:** H.C. conceptualized the project, designed and developed the robot interaction system, designed and executed the human study, acquired IRB approval, performed data analysis, and led the authoring of the paper; Y.K. designed and developed the robot interaction system and revised the paper; K.P. performed the experiments and revised the paper; C.B. conceptualized the project; provided funding, feedback, and guidance; and revised the paper; H.W.P. conceptualized the project; designed and validated the experiments and results; provided funding, project management, feedback, and guidance; and revised the paper. **Competing interests:** The authors declare that they have no competing interests. **Data and materials availability:** All data needed to evaluate the conclusions in the paper are present in the paper, the Supplementary Materials, or Dryad (https://doi.org/10.5061/dryad.hdr7sqvsh).

Submitted 6 July 2024
Accepted 18 February 2025
Published 12 March 2025
10.1126/scirobotics.adk3307

COLLECTIVE BEHAVIOR

Reverse engineering the control law for schooling in zebrafish using virtual reality

Liang Li[1,2,3,4]*, Máté Nagy[1,2,3,5,6]*, Guy Amichay[1,2,3,7,8,9], Ruiheng Wu[1,2,4], Wei Wang[10], Oliver Deussen[2,4], Daniela Rus[11], Iain D. Couzin[1,2,3]*

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

Revealing the evolved mechanisms that give rise to collective behavior is a central objective in the study of cellular and organismal systems. In addition, understanding the algorithmic basis of social interactions in a causal and quantitative way offers an important foundation for subsequently quantifying social deficits. Here, with virtual reality technology, we used virtual robot fish to reverse engineer the sensory-motor control of social response during schooling in a vertebrate model: juvenile zebrafish (*Danio rerio*). In addition to providing a highly controlled means to understand how zebrafish translate visual input into movement decisions, networking our systems allowed real fish to swim and interact together in the same virtual world. Thus, we were able to directly test models of social interactions in situ. A key feature of social response is shown to be single- and multitarget-oriented pursuit. This is based on an egocentric representation of the positional information of conspecifics and is highly robust to incomplete sensory input. We demonstrated, including with a Turing test and a scalability test for pursuit behavior, that all key features of this behavior are accounted for by individuals following a simple experimentally derived proportional derivative control law, which we termed "BioPD." Because target pursuit is key to effective control of autonomous vehicles, we evaluated—as a proof of principle—the potential use of this simple evolved control law for human-engineered systems. In doing so, we found close-to-optimal pursuit performance in autonomous vehicle (terrestrial, airborne, and watercraft) pursuit while requiring limited system-specific tuning or optimization.

INTRODUCTION

Collective behavior arises from positive and/or negative local feedback loops, which enable repeated local interactions to scale up into highly robust coordinated activities without the need for regulation by global supervision or via a preestablished template (*1*, *2*). Given their ubiquity and importance across scales of biological organization, the mechanisms that give rise to coordinated motion among cells (*3*) and organisms such as swarming insects (*4*, *5*), schooling fish (*6*–*8*), flocking birds (*9*, *10*), and humans in crowds (*11*, *12*) have been of particular interest across multiple disciplines. In addition to providing deeper insights into biology, an understanding of the evolved strategies animals use to coordinate collective behavior can offer new opportunities for the development of engineered solutions (*13*), such as for the coordination of autonomous vehicles (*14*, *15*).

To date, however, it has been extremely difficult to infer the nature and causal structure of biological interactions that give rise to collective behavior using conventional experimental approaches (*16*–*18*). Consequently, the sensory-motor feedback mechanisms that have evolved to regulate collective behavior are often poorly understood (*19*), with our inability to identify, or test among, alternative hypotheses being a major bottleneck. Recent advances in immersive volumetric virtual reality (VR) technology (*20*–*22*) provide a new means to control, and thus interrogate, the causal structure of social relationships among individuals. In addition, they allow the direct testing of experimentally derived hypothetical models of social interactions in situ by allowing reciprocal coupling between real organisms and virtual robot fish as counterparts (*23*). Thus, analogous to how the dynamic patch clamp method has revolutionized neuroscience, creating a real-time interface between living cells and experimentally derived models (*24*), VR opens up a dynamic social clamp approach (*23*) to the study of animal behavior.

In this study, we applied a VR platform (*20*) to control virtual conspecifics that can interact with real fish (RF), enabling us to reverse engineer the fundamental sensory-motor control (SMC) mechanisms governing leader-follower dynamics under various conditions (Fig. 1A, fig. S1, and Movie 1). We found that a simple, yet powerful, proportional derivative–like controller, which we term "BioPD," effectively captures the social pursuit behavior of fish (Fig. 1, B to D). To validate the model, we explored its assumptions, assessed its predictive capabilities, and conducted a Turing test with RF—all of which confirmed the model's effectiveness (Fig. 1, D to F). Last, we applied BioPD to different robotic systems, tested the scalability of the model in large-scale fish schools across different species, and integrated it into various robotic platforms for diverse group sizes and pursuit tasks (Fig. 1, G to I). Our findings highlight the potential of robotic platforms, such as VR, as valuable tools for understanding collective behavior while, simultaneously, in return, inspiring advancements in robotic control design.

[1]Department of Collective Behaviour, Max Planck Institute of Animal Behavior, 78464 Konstanz, Germany. [2]Centre for the Advanced Study of Collective Behaviour, University of Konstanz, 78464 Konstanz, Germany. [3]Department of Biology, University of Konstanz, 78464 Konstanz, Germany. [4]Department of Computer and Information Science, University of Konstanz, 78464, Konstanz, Germany. [5]MTA-ELTE "Lendület" Collective Behaviour Research Group, Hungarian Academy of Sciences, 1117 Budapest, Hungary. [6]Department of Biological Physics, Eötvös Loránd University, 1117 Budapest, Hungary. [7]Department of Engineering Sciences and Applied Mathematics, Northwestern University, Evanston, IL 60208, USA. [8]Northwestern Institute on Complex Systems, Northwestern University, Evanston, IL 60208, USA. [9]National Institute for Theory and Mathematics in Biology, Northwestern University, Evanston, IL 60208, USA. [10]Department of Mechanical Engineering, University of Wisconsin–Madison, Madison, WI 53706, USA. [11]Computer Science and Artificial Intelligence Lab (CSAIL), Massachusetts Institute of Technology, Cambridge, MA 02139, USA.
*Corresponding author. Email: lli@ab.mpg.de (L.L.); mate.nagy@ttk.elte.hu (M.N.); icouzin@ab.mpg.de (I.D.C.)

Science Robotics

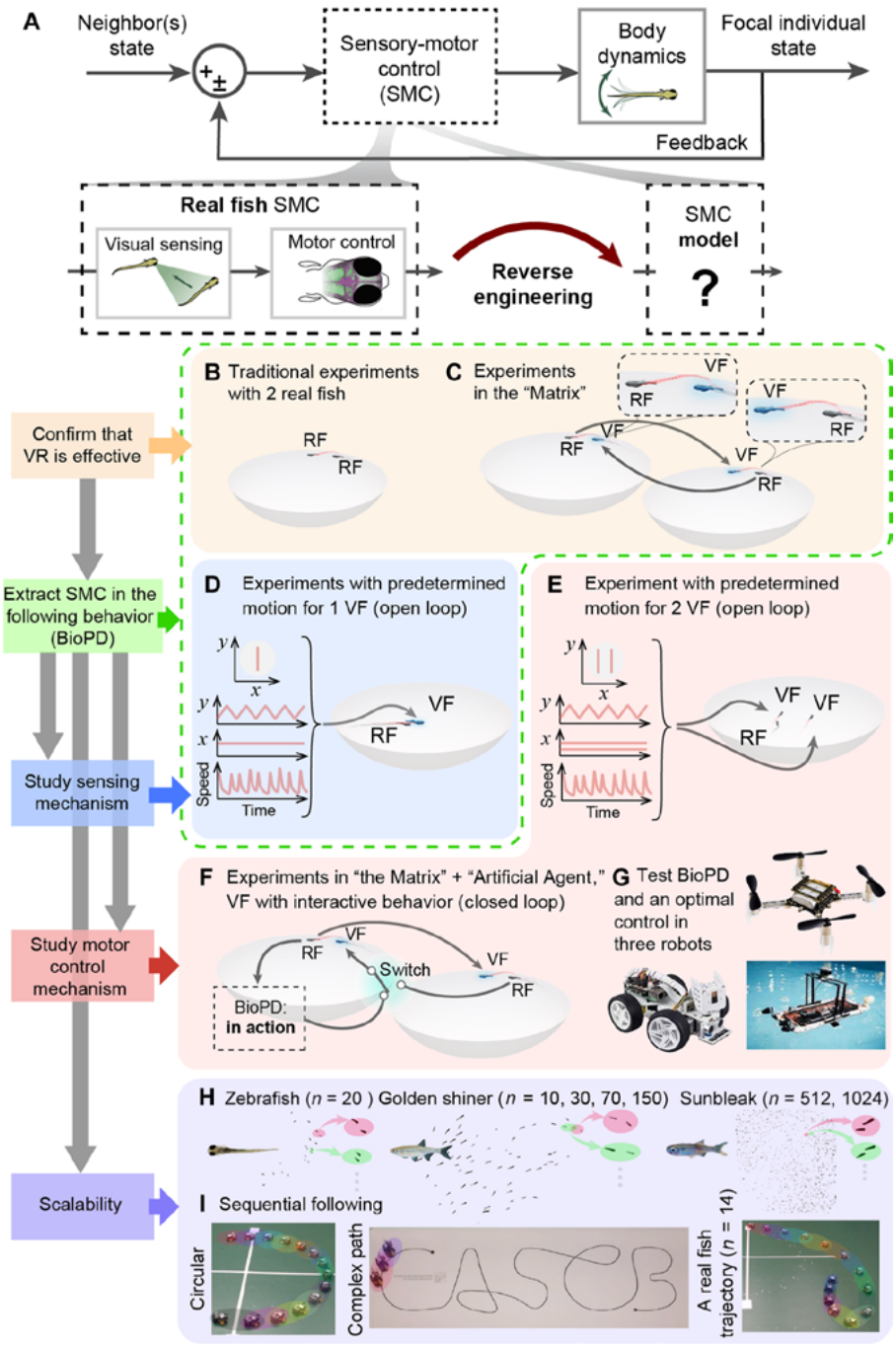


Fig. 1. Schematic of the study of SMC of schooling behavior. (**A**) The flow diagram of the SMC of the social response to neighbors. We reverse engineered the sensory-motor control (SMC) of following behavior to a model, which we term BioPD. (**B**) Traditional experiment in which two RF swim together in one bowl-shaped arena. Note that the lines trailing each fish indicate their movement trajectories, which are shown at a 1:1 scale relative to the dimensions of the bowl. The fish themselves are enlarged by a factor of 3 to ensure better visibility. (**C**) The Matrix system, where each arena contains a single individual, each of which can interact with a real-time volumetric projection of the other. (**D**) Open-loop experiments with one VF as a leader swimming back and forth at a recorded swimming speed (0.04 m/s in average). (**E**) Experiments with two VF swimming side by side as two leaders to verify the SMC. (**F**) Experiments with two RF interacting within the Matrix to verify the SMC of the following behavior. A VF becomes a follower controlled by the BioPD when the RF becomes a leader (when the RF swims in front). Otherwise, the VF copies the position and direction of the RF in the other arena. (**G**) Evaluating the performance of the BioPD model by comparing it with an MPC in three robotic systems (terrestrial, airborne, and watercraft). (**H** and **I**) Evaluating BioPD's scalability across three species in groups up to 1024 individuals and with up to 14 robots.

RESULTS

Development and validation of the immersive VR system

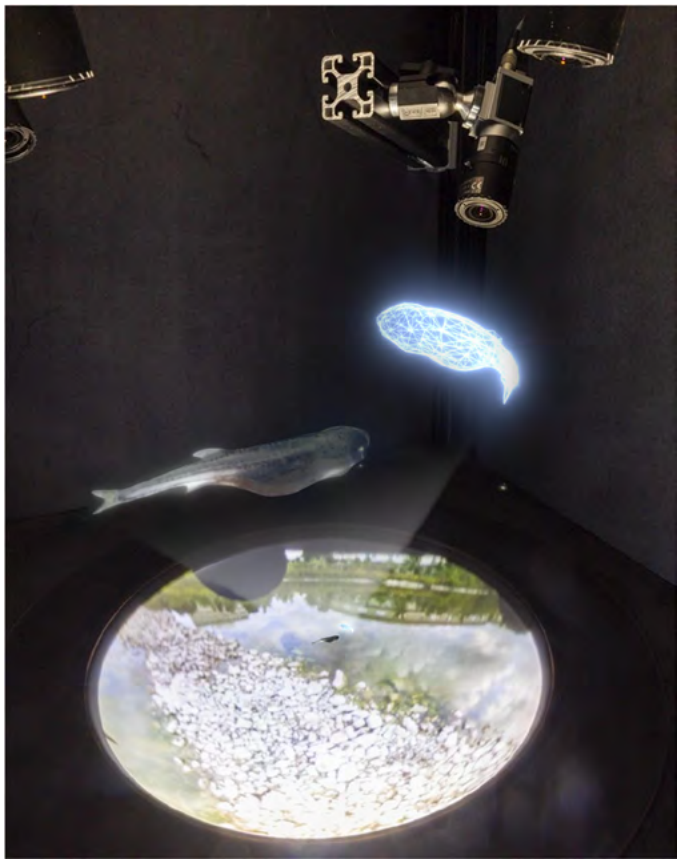
We applied an immersive VR for fish (*20*) to investigate the SMC used in regulating schooling behavior in a model vertebrate, the juvenile zebrafish (*Danio rerio*) (1 ± 0.1 cm in length, 24 to 26 days postfertilization; Fig. 1 and Movie 2). At this age, zebrafish predominantly use vision to coordinate response to conspecifics when schooling [the lateral line being dominated by self-generated motion because of viscous adhesion forming a boundary layer around such small fish (*25*)].

Given that leading others is known to be driven by different internal processes, such as indifference to others (*26*) and motion toward external goals (*27*), as a valuable starting point—and because of their general importance—we focus here only on socially mediated interactions. In addition to uncovering a key algorithm used in regulating schooling behavior (Fig. 1, B to F and H, Supplementary Methods, fig. S1, and table S1), we demonstrated its application to motion control in engineered systems (Fig. 1, G and I).

Because our immersive VR, for freely swimming animals, relies on correct volumetric rendering from the perspective of a single individual (via the anamorphic illusion), it is not possible to put more than one individual in each VR arena. We can, however, connect systems (*28*) such that individuals can see, and thus interact with, a real-time "holographic" projection of the other (Fig. 1C shows this principle for a pair of individuals), which we term "the Matrix" (Movie 2). We found that as in the real world (figs. S2, A to D; S3, A to D; and S4, A and C), individuals in "the Matrix" (figs. S2, E to H; S3, E to H; and S4, B and D) interact only when they occupy the same x-y plane, with even small movements out of that plane (in the z dimension), either toward the surface or to deeper water, being associated with rapid decoupling of social interactions (fig. S5). The structure, and strength, of the interactions within this plane [as quantified by decomposing motion to lateral speed v_x, which is perpendicular to the leader's head direction, and forward speed v_y, which is along the leader's head direction (Fig. 2A)] is found to be near identical when they interact within the physical world (fig. S4, A and C) as when they interact in the same holographic world (figs. S4, B and D, and S6; Kolmogorov-Smirnov test, $P = 0.26$ for v_x and $P = 0.9$ for v_y). This suggests that our VR system is ideal for dissecting sensory-motor feedback control.



Movie 1. Summary video. Video abstract of the study.



Movie 2. Overview of the experiments with VF interacting with RF. Variety of experiments using our VR platform for freely swimming zebrafish, including platform validation in the Matrix; one VF with varying average speeds, movement patterns, update rates, and visibilities; a Turing test in the Matrix; and two VF with different average speeds.

Reverse engineering the sensory-motor feedback control

To do so, we first conducted open-loop experiments, which allow us to control the causal flow of information from a leading (virtual) fish to a following (real) fish (Fig. 2B). Our platform also enables us to isolate and analyze how social responses are influenced by both spatial factors and average swim speed. Here, we evaluated average swim speed across its natural range while also accounting for its inherently bursty nature—characterized by rapid tail undulations followed by a friction-dominated glide (Fig. 2C and fig. S7).

Fish tend to follow/pursue the virtual leader at a relatively stable distance, with this distance increasing approximately linearly as a function of the leader's speed (Fig. 2, D to J, and fig. S8), but with different "times to collision" (with respect to the current position of the leader, if the leader were to suddenly stop; fig. S9). The decomposed lateral (v_x) and forward (v_y) components of the follower's speed, as a function of the spatial position of the follower relative to a leader positioned at $x = 0$ and $y = 0$, are shown in Fig. 2 (E and F, respectively). As shown in Fig. 2G, the average lateral speed increases as a function of lateral distance (x axis) up to a specific distance, $r_x = 0.07$ m (determined by finding the maximum lateral speeds following a bootstrapping procedure; see Supplementary Methods for details), indicated by the dotted line, after which it starts to decrease. Lateral speed is minimally influenced by swim speed (as seen by the similarity of panels in Fig. 2E and the average plots in Fig. 2G). The magnitude of the forward speed component as a function of the front-back distance, shown in Fig. 2H, also increases up to a similar distance, $r_y = 0.07$ m (see Supplementary Methods for details), but unlike lateral speed, it increases in absolute magnitude as a function of average swim speed of the leader.

According to the above properties (speed control being proportional to the distance lag and to the average swimming speed of the leader and a lateral point of speed reduction at a specific distance) and following reverse engineering methods in biological studies (*29*–*35*), we proposed a parsimonious bioinspired proportional-derivative (PD) controller, BioPD

$$\begin{cases} x_e &= x_\mathrm{F} - x_\mathrm{L} \\ v_x &= -\left(K_\mathrm{p} x_e + K_\mathrm{d} \dot{x}_e\right) e^{-\frac{x_e^2}{2r_x^2}} \\ y_e &= y_\mathrm{F} - y_\mathrm{L} \\ v_y &= -\left(K_\mathrm{p} y_e + K_\mathrm{d} \dot{y}_e\right) e^{-\frac{y_e^2}{2r_y^2}} \end{cases} \quad (1)$$

where x_F and y_F (and, respectively, x_L and y_L) are the positions of the follower (leader) in a global coordinate system resolved in the x and y axes according to the leader, respectively (Fig. 2A). r_x and r_y describe the critical distances at which the strength of social interactions is the largest (Supplementary Methods). K_d and K_p are the derivative gain and proportional gain parameters, respectively, which are the two main parameters in the model. We first determined the derivative parameter on the basis of the relationship between the average

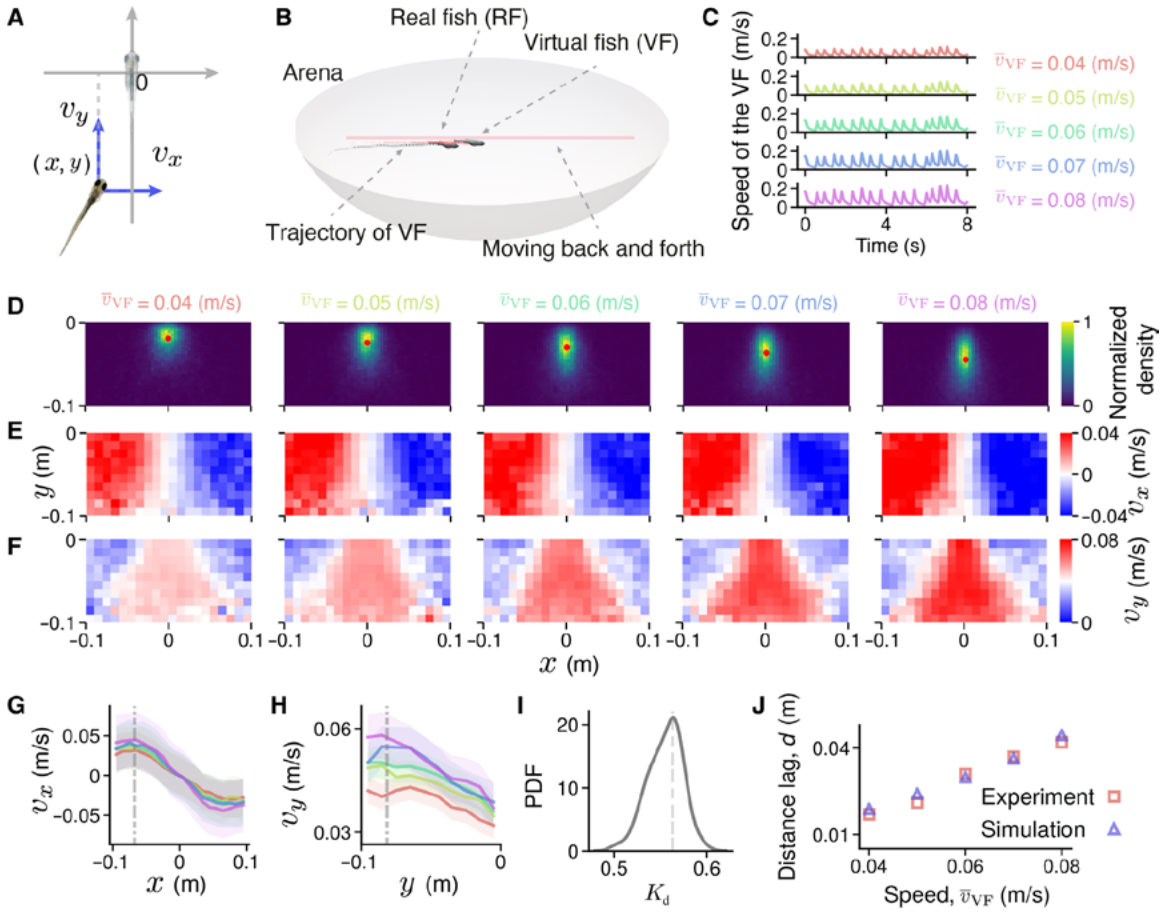


Fig. 2. Reverse engineering SMC of fish to a bioinspired proportional-derivative controller, BioPD. (**A**) The local coordinate system is based on the position and direction of the leader. The RF's swimming speed is resolved into lateral speed v_x and forward speed v_y. (**B**) Schematic to show the experimental setup, where an RF follows one VF that is swimming back and forth in a straight line. (**C**) The VF exhibits a realistic burst-and-glide swimming pattern for five different average swimming speeds $\bar{v}_{VF}$ (0.04 to 0.08 m/s with an interval of 0.01 m/s). (**D**) The higher the average swimming speed of the leader, the greater the distance maintained by the follower to the leader. (**E** and **F**) Lateral (E) and forward (F) speed control as a function of the position of the follower in the local coordinate of the leader with different average swimming speeds. (**G** and **H**) Average lateral (G) and forward (H) speeds as a function of the follower's position in the x (G) and y (H) axes in the local coordinate of the leader. The shaded areas denote the standard deviation after 100 bootstraps. In general, the lateral swimming speed, v_x, is less sensitive to the leader's average swimming speed, because there is no difference in both the density (E) and average plots (G). However, the forward speed, v_y, shows sensitivity in both the density (F) and average plots (H). (**I**) The distribution of the derivative parameter K_d is based on the maximum forward swimming speed at each average swimming speed of the leader. PDF, probability density function. (**J**) The comparison between the experiments and simulations shows that the model describes the experimental data.

forward swimming speed of RF and virtual fish (VF) (see Supplementary Methods for details), finding $K_d = 0.56$ (Fig. 2I). The proportional parameter K_p was estimated on the basis of the stable distance lag under different average swimming speeds of the leader, which for our zebrafish was found to be a constant with value $K_p = 2.3$ (Fig. 2J).

Despite its simplicity, we found that the BioPD model can account for all of the main features observed in our experiments, including the stable swim speed of the follower, which is matched to that of the leader (fig. S10, A and B), a similar spatial probability density with respect to the leader, and similar lateral and forward swimming speeds as a function of x_e and y_e (fig. S10, C to F). This indicates that a simple PD-like controller, with identical parameters, can effectively regulate schooling fish, irrespective of the average swim speed of the leader.

Validation of the assumptions of the BioPD model

Speed input in a PD controller can be either the instantaneous speed or the average speed over some period of time, as perceived directly through the fish's vision. Therefore, we further used our virtual robotic conspecifics to evaluate, directly, which features of the speed of conspecifics are used in regulating social response. Previously, it has been suggested that the motion characteristics associated with burst-and-glide locomotion of juvenile zebrafish may provide an important social cue (*36*). However, using VF swimming with burst and glide as the control [Fig. 3A(i)] and constant-speed swimming at the same average speed as the treatment [Fig. 3A(ii) and Movie 2], we observed that the following distance lag [Fig. 3A(iii) versus (iv)], lateral speed [Fig. 3A(v) versus (vi)], and forward speed [Fig. 3A(vii) versus (viii)] are the same. This indicates that zebrafish respond similarly to continuous motion as they do to biological (bursty) motion (see fig. S11 for a detailed comparison).

This suggests that fine-scale instantaneous speed is not used in the regulation of schooling. To evaluate this further, we investigated how the temporal resolution of visual input influences social response. By systematically changing the temporal update frequency,

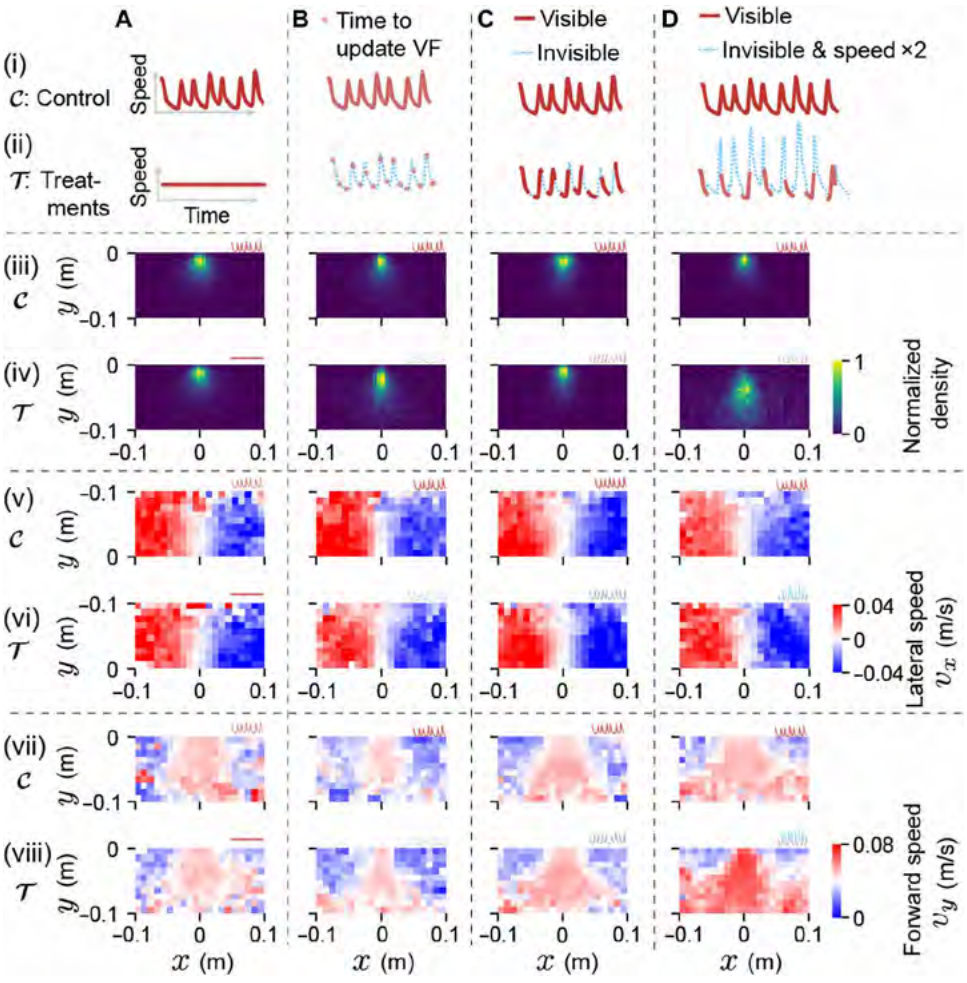


Fig. 3. Evaluating the perceptual information used in the regulation of social response to a leader. (**A**) VF swims with the same average but different instantaneous speeds: burst and glide as the control (i) or constant speed as the treatment (ii). (**B**) VF swims with different update frequencies at 100 Hz as the control (i) or at 5 Hz as the selected treatment (ii) while keeping a fixed location between updates. (**C**) VF swims with different visibilities [always visible as the control (i) or periodically become invisible (times being visible and invisible are both set to 0.2 s) as the treatment (ii)] to decouple the presented position and speed information. (**D**) Same as (C), except the VF jumps to a location further away by increasing its speed during the period of invisibility by a factor of 2 as compared with speed during being visible. Selected swimming performances, including relative position (iii and iv), lateral speed (v and vi), and forward speed (vii and viii), are presented.

such that the VF is always visible but its position is only updated at a certain rate (for instance, if the frequency is 10 Hz, the position of the VF will be updated 10 times per second, the frames between which it does not change position; fig. S12 and Movie 2), we found that only if the update rate falls below ~5 Hz does the distance lag increase [Fig. 3B(iii) to (iv)], indicating a diminishing of the effectiveness of the social response below this frequency (Fig. 3B). This indicates that zebrafish integrate information over ~0.2 s, a timescale close to the typical period of their burst-and-glide gait (fig. S7), which may imply the use of spatial working memory (*37*) and is captured in the model by averaging speed with a similar time window.

To establish how robust the schooling response is in the face of incomplete information, we decoupled speed and position by manipulating the visibility of the VF. As may be expected of animals that need to deal with regular occlusions of others, such as by vegetation or in patches of high turbidity, they do not respond to the sudden disappearance or appearance of a conspecific (Fig. 3, C and D, figs. S13 and S14, and Movie 2). By adjusting both the duration of the windows of time during which information is available (figs. S12 and S13) and whether the perceived speed in these windows is [Fig. 3C(ii)] or is not [Fig. 3D(ii)] congruent with the displacement (such as the average speed) between these windows (fig. S14), we observed no difference in distance lag, lateral speed, or forward speed when the perceived speed matches the invisible case [Fig. 3C(iii) to (viii)]. However, there was a substantial difference in the nonmatching case [Fig. 3D(iii) to (viii)]. This suggests that the algorithm used by zebrafish uses positional information as the input for speed control and not estimates of instantaneous speed (Fig. 3, C and D, and figs. S13 and S14).

Predictive power of BioPD

With the core assumptions of BioPD validated, we then asked whether it can account for further dynamical features of natural schooling. To establish the above control law, we used virtual robot fish as conspecifics that move in a constant direction and at a constant average swim speed. In reality, however, fish dynamically modulate both properties. By presenting exactly the same trajectories, obtained from RF leaders, both to RF and to agents using BioPD (Fig. 4, A and B), we compared directly the response of real followers with agents using BioPD. We found that BioPD provides a robust and effective response to the dynamic changes in speed and direction exhibited in the natural system and results in pursuit behavior highly comparable to that exhibited by RF (Fig. 4, E and F).

Our VR systems allow us to take an even further step in establishing SMC; we could also ask whether leaders react differently to real followers versus followers using BioPD. This can be thought of as a Turing test for the leader: Is an agent using BioPD sufficiently convincing to allow natural bidirectional interactions? To do so, we then allow two RF, A and B, to interact in the Matrix, but each time fish A becomes a leader (such as when it occupies a frontal position), we can immediately replace the natural control of B with our BioPD control (Fig. 4, C and D, fig. S15, and Movie 2). Thus, we compared what we predicted fish would do with what they actually did for every pursuit event. We found that despite its simplicity, BioPD facilitates the maintenance of qualitatively similar, and effective, reciprocal social relationships among hybrid simulated-real individuals (Fig. 4, G and H).

Having established the response to a single conspecific, we then asked whether BioPD can also predict the response of RF to two virtual conspecifics (*38*). To do so, we considered its response to two

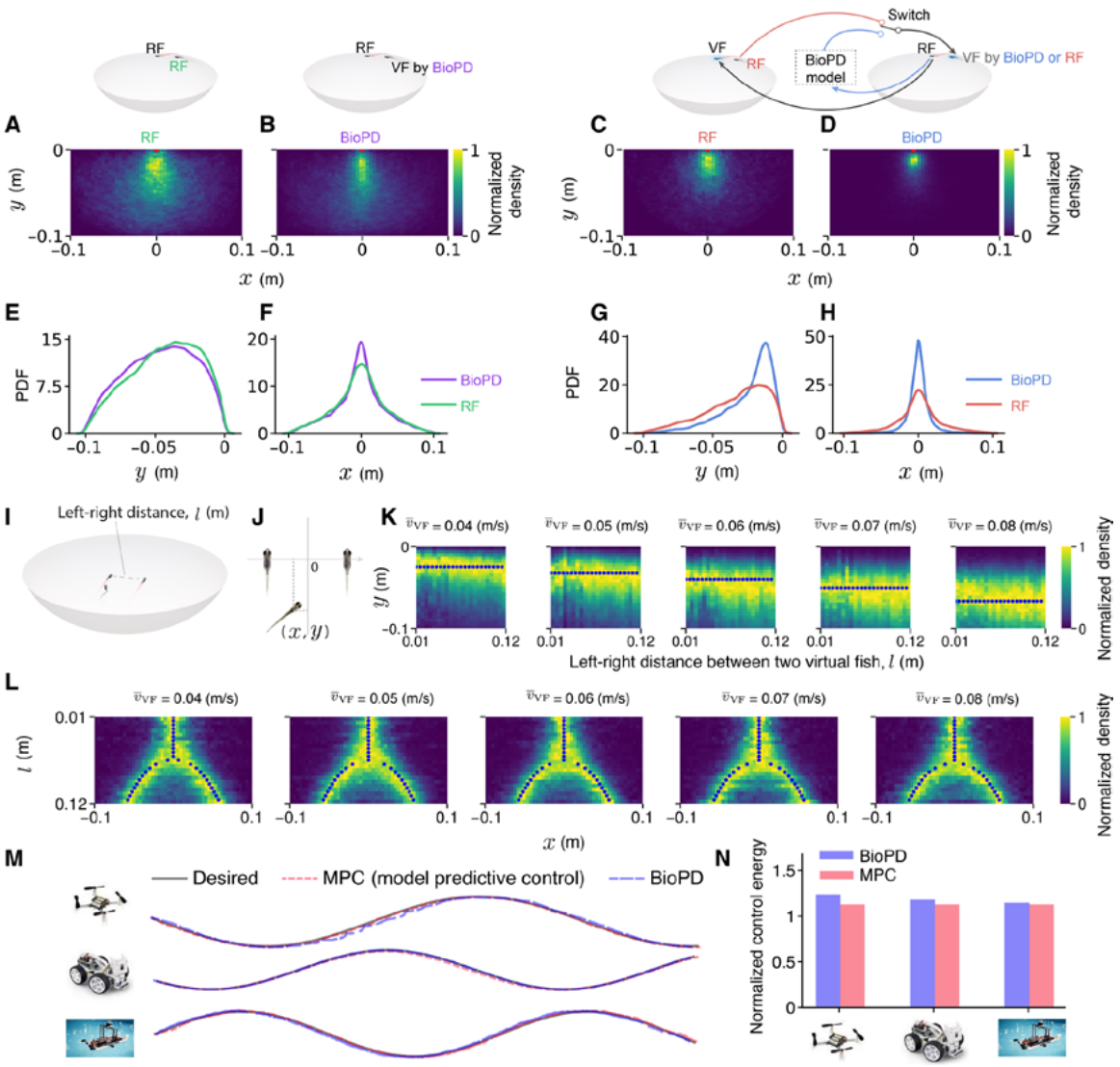


Fig. 4. Evaluating the BioPD: simulations, experiments in the Matrix system and in two virtual leader scenarios, and using robotics. (**A** and **B**) Positions of RF (A) as followers and simulated followers controlled by the BioPD [VF (B)] relative to a real leading fish (positioned at the origin). (**C** and **D**) Real (C)/virtual (D) follower's positions relative to the virtual (C)/real (D) leader's position in two arenas in the Matrix system. The VF in (D) represents an avatar of the RF from (C), unless the RF in (D) is swimming in front. In that case, the VF in (D) is controlled by BioPD. The VF in (C) is always controlled by the RF from (D). The RF in (C) follows the VF (C), which represents the RF shown in (D). (**E** to **H**) Distributions of relative distance in the *y* [(E) and (G)] and *x* [(F) and (H)] axes of the simulations [(E) and (F)] and experiments [(G) and (H)]. JSD = 0.03, 0.0, 0.08, and 0.0 for (E) to (H), respectively. (**I**) Setup for two VF leaders swimming side by side at different left-right distances and average swimming speeds. (**J**) Definition of the coordinate system. The origin is the center of the two VF. Positive *y* points to the head direction of the VF. (**K**) The model predicts the relative distance between the RF and the VF swimming at different average swimming speeds. (**L**) The model also predicts the bifurcations in the following behavior of the RF when they follow two virtual leaders. (**M**) Three robots are controlled by BioPD and MPC to follow a virtual leader moving in a sinusoidal wave. (**N**) Comparison of the control energy of BioPD and MPC.

leaders swimming side by side at a range of interindividual lateral distances and swim speeds (Fig. 4, I and J, and Movie 2). We simply applied the BioPD controller for an agent receiving sensory input from the two leaders but taking into account the linear perspective in the fish eye (see Supplementary Methods for details). We found that BioPD accounts, quantitatively, for a key experimental finding, given that the blue points predicted by the model match the high-density regions of RF behavior data in Fig. 4 (K and L); that RF will both change their distance lag (Fig. 4K) and suddenly switch from adopting a position in between the targets (here, the leaders) to deciding among them (swimming predominantly with one of the VF) as a function of increasing the lateral distance (l) between the virtual leaders (Fig. 4L and fig. S16). Furthermore, it also accounts for the observed increase in the critical distance (l_c) at which this transition occurs as a function of increasing swim speed (Fig. 4L and figs. S17 to S19).

Scalability of BioPD

To investigate the scalability and generalizability of our BioPD and its potential in elucidating collective behavior in large groups, we analyzed the social response of followers in schools of juvenile

zebrafish (*D. rerio*, $n = 20$) tracked with custom code (*39*), juvenile golden shiners [*Notemigonus crysoleucas*; $n = 10$, 30, 70, and 150; data from previous publication (*40*)], and juvenile sunbleak (*Leucaspius delineatus*, $n = 512$ and 1024) tracked with custom code (*39*). Across all three species, we observed pervasive leader-following behavior, even in larger group sizes. Analogous to the two-fish system, the distance lag correlates with the leading fish's average swimming velocity. The follower's turning and forward speeds align with characteristics previously observed in our VR system (Fig. 5, A to C, and fig. S20). Such findings suggest that the BioPD derived from smaller groups can be extrapolated to larger collectives.

Robotic applications of BioPD

Reverse engineering natural control laws—which have been subject to evolution by natural selection for millennia—could, in principle, provide simpler and/or more robust solutions for human-engineered problems (*13*). The effective pursuit of mobile targets, along with the maintenance of appropriate spacing with respect to a target (which can include interception or pursuit while also avoiding collisions, for example), is a central challenge in the effective control of autonomous vehicles, such as self-driving cars and guided aircraft and spacecraft. Human-made controllers, such as the widely used model predictive controller (MPC) (*41*), have been shown to be optimal for certain tasks but typically are computationally expensive because MPCs are solving an optimization problem over a predicted horizon (predicting future system behavior over a defined time frame) at each control step. Moreover, MPCs need to be individually optimized—a very time-consuming process—for each specific application (because they depend on an accurate underlying model of the dynamical systems in which they are to be embedded). Natural systems, by contrast, are under selection to evolve highly robust and cheap strategies that approximate optimal solutions under a wide range of conditions. Seldom, however, are such evolved solutions evaluated in situ in real physical systems.

To gain insight into such potential application domains, we implemented and compared the pursuit performance of a state-of-the-art optimal MPC controller (see Supplementary Methods for details) and BioPD in three very different robotic platforms: terrestrial vehicles, airborne drones (*42*), and watercraft (Fig. 1H); the task was to follow a virtual leader on a predefined sine-shaped trajectory (Fig. 4M and Movie 3). Furthermore, unlike the MPC controller, which required a complex and time-consuming optimization procedure for each robotic system, we used BioPD with exactly the same parameters as estimated from zebrafish in all scenarios. We found that with suboptimal control parameters extracted from the biological system (figs. S21 and S22), and despite its simplicity, BioPD exhibits highly robust and effective performance, providing very close to optimal control energy (*43*) in the vehicle, drone, and roboboat control tasks (Fig. 4N and figs. S23 to S25). To further assess the scalability of BioPD (Movie 3), we used the BioPD to direct 14 robots in a sequential circular motion (Fig. 5D), three robots in tracing an intricate path of the letters associated with our research center, "CASCB" (Fig. 5E), and 14 robots in emulating RF trajectories (18× larger scale) at both constant (Fig. 5F) and varying speeds (Fig. 5G). The corresponding deviation of each robot's trajectory to the desired path, distance lag between successive pairs of robots in the sequence, moving speeds, and control energy are illustrated in Fig. 5 (H to K) and figs. S26 to S29.

DISCUSSION

In this study, we constructed a virtual robotic "conspecific" for juvenile zebrafish and used it to reverse engineer the algorithm used to regulate social response. After confirming that RF accept the virtual robotic fish as a conspecific at the behavioral level, we demonstrated

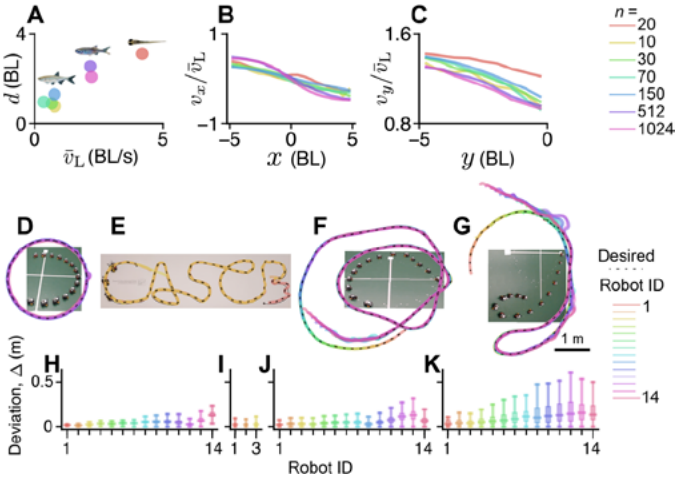
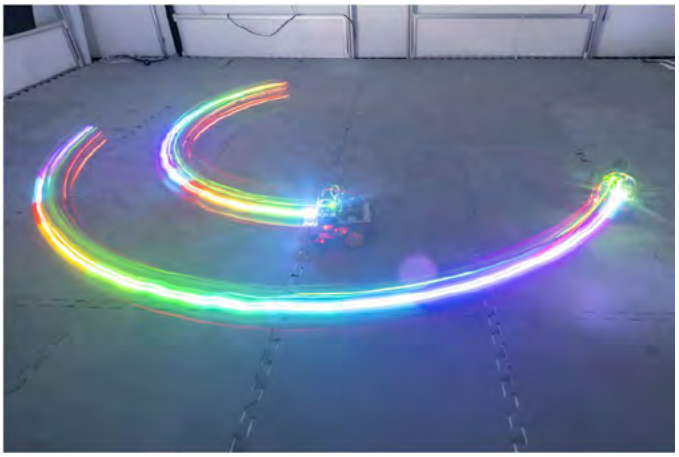


Fig. 5. Assessing the scalability of BioPD across three fish species (up to 1024 individuals) and robots (up to 14). (**A**) Distance lag plotted against the leader's average swimming speed. (**B** and **C**) Normalized turning speed, $u_x / \bar{u}_L$, and forward speed, $u_y / \bar{u}_L$, mapped to the left-right distance, x [body length (BL)], and front-back distance, y (BL), respectively. Data span three species: zebrafish (*D. rerio*, $n = 20$), golden shiner (*N. crysoleucas*, $n = 10$, 30, 70, and 150), and sunbleak (*L. delineatus*, $n = 512$ and 1024). (**D**) Group of robots exhibiting sequential circular motion. (**E**) Three robots tracing the intricate "CASCB" logo in sequence. (**F**) Fourteen robots sequentially emulating a constant-speed fish trajectory. (**G**) Fourteen robots mimicking a burst-and-coast fish trajectory, similar to natural fish movement patterns. Scale bar, 1 m [applies to (D) to (G)]. (**H** to **K**) Box plots featuring the median, interquartile range, and potential outliers are applied to illustrate the deviations (Δ) of each robot's trajectory from the desired path for conditions (D) to (G).



Movie 3. Overview of the experiments with robots controlled by BioPD for different pursuit behaviors. Series of experiments demonstrating the application of BioPD to robotic pursuit behaviors, including drone, terrestrial, and watercraft pursuits of a leader moving along a sinusoidal path; 14 robots engaged in sequential circular pursuit; three robots sequentially navigating complex letter-shaped trajectories; and 14 robots simulating RF movements at both constant and varying speeds.

that a simple PD-like controller (BioPD) can account for the social response exhibited toward it by RF (Fig. 2). Taking advantage of the virtual robot platform, we systematically checked the assumptions of the model (Fig. 3), verified the effectiveness of the model (Fig. 4, A to H), tested its power of prediction (Fig. 4, I to L), explored its potential applications (Fig. 4, M and N), and demonstrated its scalability (Fig. 5). These tests illustrate that the platform is a powerful tool for quantitatively and causally exploring hidden internal controls, offering capabilities that have been largely inaccessible in previous systems.

By controlling the visibility of the virtual robot to decouple position and speed perception [something not possible in traditional biological experiments or even with those physical replicas (*28*)], we found that fish primarily rely on position perception rather than instantaneous speed perception (Fig. 3). In addition, the input for derivative control on long-term average speed is derived from positional information (fig. S13). This mechanism is similar to a low-pass filter and could facilitate noise reduction in the velocity control. However, it remains unclear how fish obtain the neighbor's positional information from their visual scene, and this will be the focus of our future studies. Despite being derived from the response to a single leader, we found that BioPD can also account for the fish behavior when presented with two leaders: notably an abrupt transition (bifurcation) from averaging to deciding to follow one or the other of the potential leaders as a function of the distance between them (Fig. 4, I to L). This finding is consistent with our previous work on geometric principles that emerge from individual decision-making (*38*), where the linear distortion of perspective provides a comparable "non-Euclidean" distortion of space (*38*, *44*).

Our study suggests that fish may have evolved a sensorimotor control like BioPD given that it provides a cognitively minimal yet highly effective means to regulate schooling. Compared with traditional PD-like controllers in engineering systems (*45*, *46*), our controller differs in two key ways: First, it introduces a biologically meaningful cutoff in the interaction, embedded through a nonlinear first-order Gaussian function (Fig. 2, G and H), and second, it has been validated in biological systems through assumption verification (Fig. 3, A to D), predictive power testing (Fig. 4, I to L), and even an embodied Turing test (Fig. 4, A to H). This uniquely bridges engineering and biological systems. Unlike traditional PD controllers in biological systems that either explore PD functions within the system (*33*, *47*) or aim to find the best controllers to fit biological data (*30*, *48*, *49*), our approach focuses on reverse engineering the complete dynamics of pursuit behavior. We offer a detailed controller along with measured parameters, rather than merely fitting the data.

The reason BioPD generalizes well to the three robotic platforms may be the same reason it has evolved: Because the control is based on a kinematic model, without involving specific dynamics, it is highly generalized. Therefore, there is room for optimization for each robot based on the dynamic models of their respective systems (Fig. 4N and figs. S21 and S22), which will be our future focus. In the future, we will also explore in further detail the limitations and potential applications of this controller. In our preliminary analysis of scalability, we found that fish adopt the same interaction rule in the pursuit behavior as when in large groups (Fig. 5). Overall, our work suggests that reverse engineering the control law in schooling fish using virtual conspecifics offers a complementary approach to traditional methods in both biology and engineering. It is valuable not only for systematically exploring potential internal SMCs in biological systems but also for designing controllers that are efficient and robust, requiring minimal sensing and computational resources.

MATERIALS AND METHODS

Methodological overview

We developed and used a VR platform to reverse engineer the SMC algorithm for schooling behavior (fig. S1). To confirm its effectiveness, we compared the speed control of an RF to another RF in a traditional platform with that to a VF in our system. Subsequently, we conducted open-loop experiments in the virtual environment with a single VF, where we controlled the leader's swimming properties, such as average speed, patterns, and visibility. By doing so, we developed a biologically inspired proportional-derivative control model for the following behavior and verified its assumptions. Furthermore, we estimated the model through various methods, including simulations, experiments in the VR system, experiments with two virtual leaders, and tests with three types of robots (terrestrial, airborne, and watercraft).

The VR platform

The test arena was bowl shaped, holding 4.5 liters of water, with a depth in the center of 9.1 cm and a diameter of 33.8 cm. The bowl was made from a material that is opaque to visible light but transparent to infrared light, allowing us to illuminate it from below with infrared light (wavelength of 850 nm) for tracking purposes. Positioned above the bowl were four Basler acA640 cameras, capturing images at 100 frames per second in real time. The three-dimensional (3D) position of the RF was reconstructed from its detected position in each view using blob detection. With these reconstructed 3D positions, we tracked the fish's movements on the basis of an extended Kalman filter and the nearest neighbor standard filter for data association. An LED (light-emitting diode) DLP (digital light processing) projector (Optoma ML500) illuminated the entire surface of the bowl to allow us to present the virtual scene in the visible light spectrum. To ensure accurate projection, the projected image was mapped onto the 3D curvature of the bowl using a calibration model of the display geometry. A grid was rendered during calibration, mapping each pixel to its corresponding geometric coordinates. The anamorphic illusion was applied to render the image as a 3D scene from the correct perspective of the RF. A VF, modeled in Blender (Blender Foundation) on the basis of juvenile zebrafish [26 days postfertilization (dpf), 1 cm], was rendered within the system to interact with freely swimming RF (24 to 27 dpf, 1 ± 0.1 cm) in 3D (*20*, *50*). In the Matrix setup, similar to (*28*), the RF's position and orientation were tracked on one machine and transferred in real time to another machine via a socket connection, allowing for seamless bidirectional communication between the two systems. For further details, please refer to our previous work (*20*, *50*).

VR experiments

We conducted experiments with zebrafish (*D. rerio*) of age 24 to 26 dpf raised in a room at 28°C on a 16-hour light, 8-hour dark cycle. The variation in age was to allow us to always use fish of a similar body length (1 ± 0.1 cm). Four hundred and ninety-eight zebrafish were used (see table S1). Experiments were conducted in a fish VR setup procured from Loopbio GmbH [refer to ref. (*20*) for details]. After a fish was introduced into the arena (a bowl-shaped container with a diameter of 34 cm and a depth of 9 cm at the water level),

we allowed the fish to acclimate to the environment for 20 min. This was followed by a 10-min control, during which the fish was presented with a single virtual conspecific (1 cm in body length) swimming in a circle with a diameter of 16 cm. After this, the RF was exposed to the VF, initialized with various swimming conditions (Supplementary Methods). Each experiment lasted 90 min. We analyzed the data using a custom Python 3.7 code. All experiments were conducted in accordance with the animal ethics permit approved by Regierungspräsidium Freiburg, G-16/116, G-17/88, and G-17/170.

Simulations

We used the BioPD algorithm to simulate the behavior of following a virtual leader in the VR experiment, as well as a real leader in a pair of fish performing leader-follower behavior extracted from RF data. For following the virtual leader, we set the follower's initial position to a range of −0.05 to 0.05 m on the x axis and −0.05 to 0 m on the y axis. We introduced variability by adding white noise to the follower's speed control, with a standard variance of 0.016 for the x axis and 0.45 times the average speed of the follower for the y axis. The maximum swimming speed was limited to 0.1 m/s. In contrast, for following an RF leader, we initialized the geometry position and swimming probabilities of the follower to match the starting point of leader-follower behavior in the pair of RF swimming in the same arena. The only difference in this model from the previous one is that the leader is extracted from RF leader data, which dynamically changes both average swimming speeds and directions. No noise was added in the second simulation.

Parameter estimation for the model

The BioPD model has four primary parameters (r_x, r_y, K_p, and K_d), which we determined by measuring RF data collected in the VR experiments. In the following behavior, when the distance between the leader and the follower was larger than the threshold distance r_x (r_y), the follower reduced its speed to follow the leader. Therefore, we modeled the turning point of speed as a first-order Gaussian derivative function, with the threshold distance corresponding to the peak of the function. We determined the threshold distance by bootstrapping (Supplementary Methods). Given that the follower must be within the distance threshold to catch up with the leader, we simplified the model to a traditional PD controller. By analyzing the PD controller, we found that the average swimming speed of the leader and follower is determined by a first-order linear function, where the slope is determined by K_d only and the intercept is determined by both K_p and K_d. We obtained these two parameters through a similar bootstrap analysis, and detailed derivations are given in Supplementary Methods.

Robotic experiments

We tested both BioPD and an optimal controller on four different types of robots: the Crazyflie drone (*42*), the SunFounder Robot PiCar-X, the Osoyoo Robot Pi Car, and a robot boat. To create a virtual leader for the robots to follow, we programmed it to move in a sinusoidal curve. We applied BioPD with the same mathematical model and parameters, which were scaled by the body size of the robots. In addition, we considered an MPC based on our previous study (*41*). To achieve the best performance with MPC, parameters were optimized for each type of robot. Further details can be found in Supplementary Methods.

Statistical analysis

We applied bootstrap resampling to estimate the variability in the relationships between lateral speed (v_x) and position in x, as well as forward speed (v_y) and position in y. Specifically, we resampled the collected data 100 times and plotted the average trends along with their variability, as shown in Fig. 2 (G and H). To evaluate the similarity between RF interactions in the physical world and those in the Matrix, we computed the root mean square deviation of lateral speed (v_x) and forward speed (v_y) as functions of the neighbor's position. Data from 22 real interaction pairs in the physical world were pooled to serve as a reference. We then compared this reference with the root mean square deviation of the remaining 24 real pairs and 24 virtual (the Matrix) pairs. Kolmogorov-Smirnov tests revealed no significant differences between the real and virtual conditions (lateral speed: $P = 0.26$; forward speed: $P = 0.9$). To further check the similarity among the simulations, the BioPD model, and the Turing test, we applied the Jensen-Shannon divergence (JSD). The results showed minimal divergence: JSD = 0.03 for forward swimming between simulations and the model (Fig. 4E), JSD = 0.0 for turning speed between simulations and the model (Fig. 4F), JSD = 0.08 for forward swimming between RF and the model in the Turing test (Fig. 4G), and JSD = 0.0 for turning speed between RF and the model in the Turing test (Fig. 4H). A standard box plot—featuring the median, interquartile range, and potential outliers—was used to illustrate the deviation between the robots' trajectories and the desired path, as shown in Fig. 5 (H to K).

Supplementary Materials

This PDF file includes:

Supplementary Methods
Figs. S1 to S29
Table S1

REFERENCES AND NOTES

1. S. Camazine, J.-L. Deneubourg, N. R. Frank, J. Sneyd, G. Theraulaz, E. Bonabeau, *Self-Organization in Biological Systems* (Princeton Univ. Press, 2003).
2. D. J. T. Sumpter, *Collective Animal Behavior* (Princeton Univ. Press, 2010).
3. T. S. Deisboeck, I. D. Couzin, Collective behavior in cancer cell populations. *Bioessays* **31**, 190–197 (2009).
4. J. Buhl, D. J. T. Sumpter, I. D. Couzin, J. J. Hale, E. Despland, E. R. Miller, S. J. Simpson, From disorder to order in marching locusts. *Science* **312**, 1402–1406 (2006).
5. P. Ramdya, P. Lichocki, S. Cruchet, L. Frisch, W. Tse, D. Floreano, R. Benton, Mechanosensory interactions drive collective behaviour in *Drosophila*. *Nature* **519**, 233–236 (2015).
6. A. J. W. Ward, J. E. Herbert-Read, D. J. T. Sumpter, J. Krause, Fast and accurate decisions through collective vigilance in fish shoals. *Proc. Natl. Acad. Sci. U.S.A.* **108**, 2312–2315 (2011).
7. A. Berdahl, C. J. Torney, C. C. Ioannou, J. J. Faria, I. D. Couzin, Emergent sensing of complex environments by mobile animal groups. *Science* **339**, 574–576 (2013).
8. R. C. Hinz, G. G. de Polavieja, Ontogeny of collective behavior reveals a simple attraction rule. *Proc. Natl. Acad. Sci. U.S.A.* **114**, 2295–2300 (2017).
9. M. Ballerini, N. Cabibbo, R. Candelier, A. Cavagna, E. Cisbani, I. Giardina, V. Lecomte, A. Orlandi, G. Parisi, A. Procaccini, M. Viale, V. Zdravkovic, Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study. *Proc. Natl. Acad. Sci. U.S.A.* **105**, 1232–1237 (2008).
10. A. Flack, M. Nagy, W. Fiedler, I. D. Couzin, M. Wikelski, From local collective behavior to global migratory patterns in white storks. *Science* **360**, 911–914 (2018).
11. M. Moussaid, D. Helbing, G. Theraulaz, How simple rules determine pedestrian behavior and crowd disasters. *Proc. Natl. Acad. Sci. U.S.A.* **108**, 6884–6888 (2011).
12. A. C. Gallup, J. J. Hale, D. J. T. Sumpter, S. Garnier, A. Kacelnik, J. R. Krebs, I. D. Couzin, Visual attention and the acquisition of information in human crowds. *Proc. Natl. Acad. Sci. U.S.A.* **109**, 7245–7250 (2012).
13. G.-Z. Yang, J. Bellingham, P. E. Dupont, P. Fischer, L. Floridi, R. Full, N. Jacobstein, V. Kumar, M. McNutt, R. Merrifield, B. J. Nelson, B. Scassellati, M. Taddeo, R. Taylor, M. Veloso,

Z. L. Wang, R. Wood, The grand challenges of *Science Robotics*. *Sci. Robot.* **3**, eaar7650 (2018).
14. J. Werfel, K. Petersen, R. Nagpal, Designing collective behavior in a termite-inspired robot construction team. *Science* **343**, 754–758 (2014).
15. F. Berlinger, M. Gauci, R. Nagpal, Implicit coordination for 3D underwater collective behaviors in a fish-inspired robot swarm. *Sci. Robot.* **6**, eabd8668 (2021).
16. M. Nagy, Z. Akos, D. Biro, T. Vicsek, Hierarchical group dynamics in pigeon flocks. *Nature* **464**, 890–893 (2010).
17. S. Marras, R. S. Batty, P. Domenici, Information transfer and antipredator maneuvers in schooling herring. *Adapt. Behav.* **20**, 44–56 (2012).
18. K. R. Pilkiewicz, B. H. Lemasson, M. A. Rowland, A. Hein, J. Sun, A. Berdahl, M. L. Mayo, J. Moehlis, M. Porfiri, E. Fernández-Juricic, S. Garnier, E. M. Bollt, J. M. Carlson, M. R. Tarampi, K. L. Macuga, L. Rossi, C. C. Shen, Decoding collective communications using information theory tools. *J. R. Soc. Interface* **17**, 20190563 (2020).
19. A. Laan, R. G. de Sagredo, G. G. de Polavieja, Signatures of optimal control in pairs of schooling zebrafish. *Proc. R. Soc. B* **284**, 20170224 (2017).
20. J. R. Stowers, M. Hofbauer, R. Bastien, J. Griessner, P. Higgins, S. Farooqui, R. M. Fischer, K. Nowikovsky, W. Haubensak, I. D. Couzin, K. Tessmar-Raible, A. D. Straw, Virtual reality for freely moving animals. *Nat. Methods* **14**, 995–1002 (2017).
21. H. Naik, R. Bastien, N. Navab, I. D. Couzin, Animals in virtual environments. *IEEE Trans. Vis. Comput. Graph.* **26**, 2073–2083 (2020).
22. S. Sayin, E. Couzin-Fuchs, I. Petelski, Y. Günzel, M. Salahshour, C.-Y. Lee, J. M. Graving, L. Li, O. Deussen, G. A. Sword, I. D. Couzin, The behavioral mechanisms governing collective motion in swarming locusts. *Science* **387**, 995–1000 (2025).
23. G. Dumas, G. C. de Guzman, E. Tognoli, J. A. S. Kelso, The human dynamic clamp as a paradigm for social interaction. *Proc. Natl. Acad. Sci. U.S.A.* **111**, E3726–E3734 (2014).
24. A. A. Sharp, M. B. Oneil, L. F. Abbott, E. Marder, The dynamic clamp: Artificial conductances in biological neurons. *Trends Neurosci.* **16**, 389–394 (1993).
25. M. J. McHenry, G. V. Lauder, The mechanical scaling of coasting in zebrafish (*Danio rerio*). *J. Exp. Biol.* **208**, 2289–2301 (2005).
26. A. Strandburg-Peshkin, D. Papageorgiou, M. C. Crofoot, D. R. Farine, Inferring influence and leadership in moving animal groups. *Philos. Trans. R. Soc. B* **373**, 20170006 (2018).
27. M. Mischiati, H. T. Lin, P. Herold, E. Imler, R. Olberg, A. Leonardo, Internal models direct dragonfly interception steering. *Nature* **517**, 333–338 (2015).
28. M. Karakaya, S. Macrì, M. Porfiri, Behavioral teleporting of individual ethograms onto inanimate robots: Experiments on social interactions in live zebrafish. *iScience* **23**, 101418 (2020).
29. L. Ristroph, A. J. Bergou, G. Ristroph, K. Coumes, G. J. Berman, J. Guckenheimer, Z. J. Wang, I. Cohen, Discovering the flight autostabilizer of fruit flies by inducing aerial stumbles. *Proc. Natl. Acad. Sci. U.S.A.* **107**, 4820–4824 (2010).
30. M. J. McHenry, J. L. Johansen, A. P. Soto, B. A. Free, D. A. Paley, J. C. Liao, The pursuit strategy of predatory bluefish (*Pomatomus saltatrix*). *Proc. R. Soc. B Biol. Sci.* **286**, 20182934 (2019).
31. A. M. Hein, D. L. Altshuler, D. E. Cade, J. C. Liao, B. T. Martin, G. K. Taylor, An algorithmic approach to natural behavior. *Curr. Biol.* **30**, R663–R675 (2020).
32. M. S. Madhav, N. J. Cowan, The synergy between neuroscience and control theory: The nervous system as inspiration for hard control challenges. *Annu. Rev. Control Robot. Auton. Syst.* **3**, 243–267 (2020).
33. S. C. Whitehead, S. Leone, T. Lindsay, M. R. Meiselman, N. J. Cowan, M. H. Dickinson, N. Yapici, D. L. Stern, T. Shirangi, I. Cohen, Neuromuscular embodiment of feedback control elements in *Drosophila* flight. *Sci. Adv.* **8**, eabo7461 (2022).
34. P. Ramdya, A. J. Ijspeert, The neuromechanics of animal locomotion: From biology to robotics and back. *Sci. Robot.* **8**, eadg0279 (2023).
35. T. J. Prescott, S. P. Wilson, Understanding brain functional architecture through robotics. *Sci. Robot.* **8**, eadg6014 (2023).
36. J. Larsch, H. Baier, Biological motion as an innate perceptual mechanism driving social affiliation. *Curr. Biol.* **28**, 3523–3532.e4 (2018).
37. T. Pasternak, M. W. Greenlee, Working memory in primate sensory systems. *Nat. Rev. Neurosci.* **6**, 97–107 (2005).
38. V. H. Sridhar, L. Li, D. Gorbonos, M. Nagy, B. R. Schell, T. Sorochkin, N. S. Gov, I. D. Couzin, The geometry of decision-making in individuals and collectives. *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2102157118 (2021).
39. T. Walter, I. D. Couzin, TRex, a fast multi-animal tracking system with markerless identification, and 2D estimation of posture and visual fields. *eLife* **10**, e64000 (2021).
40. J. D. Davidson, M. M. G. Sosna, C. R. Twomey, V. H. Sridhar, S. P. Leblanc, I. D. Couzin, Collective detection based on visual information in animal groups. *J. R. Soc. Interface* **18**, 20210142 (2021).
41. W. Wang, L. A. Mateos, S. Park, P. Leoni, B. Gheneti, F. Duarte, C. Ratti, D. Rus, "Design, modeling, and nonlinear model predictive tracking control of a novel autonomous surface vehicle," in *Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2018), pp. 6189–6196.
42. W. Giernacki, M. Skwierczynski, W. Witwicki, P. Wronski, P. Kozierski, "Crazyflie 2.0 quadrotor as a platform for research and education in robotics and control engineering," in *Proceedings of the 22nd International Conference on Methods and Models in Automation and Robotics (MMAR)* (IEEE, 2017), pp. 37–42.
43. K. Ogata, *Modern Control Engineering* (Prentice Hall, 2010).
44. D. Gorbonos, N. S. Gov, I. D. Couzin, Geometrical structure of bifurcations during spatial decision-making. *PRX Life* **2**, 013008 (2024).
45. J. Thomas, J. Welde, G. Loianno, K. Daniilidis, V. Kumar, Autonomous flight for detection, localization, and tracking of moving targets with a small quadrotor. *IEEE Robot. Autom. Lett.* **2**, 1762–1769 (2017).
46. D. Panagou, V. Kumar, Cooperative visibility maintenance for leader–follower formations in obstacle environments. *IEEE Trans. Robot.* **30**, 831–844 (2014).
47. J.-M. Mongeau, S. N. Sponberg, J. P. Miller, R. J. Full, Sensory processing within cockroach antenna enables rapid implementation of feedback control for high-speed running maneuvers. *J. Exp. Biol.* **218**, 2344–2354 (2015).
48. D. Natesan, N. Saxena, Ö. Ekeberg, S. P. Sane, Tuneable reflexes control antennal positioning in flying hawkmoths. *Nat. Commun.* **10**, 5593 (2019).
49. C. H. Brighton, A. L. R. Thomas, G. K. Taylor, Terminal attack trajectories of peregrine falcons are described by the proportional navigation guidance law of missiles. *Proc. Natl. Acad. Sci. U.S.A.* **114**, 13495–13500 (2017).
50. G. Amichay, L. Li, M. Nagy, I. D. Couzin, Revealing the mechanism and function underlying pairwise temporal coupling in collective motion. *Nat. Commun.* **15**, 4356 (2024).

Acknowledgments: We thank all members of the Department of Collective Behavior who assisted with the project: R. Bastien for discussions at the beginning of the project, S. Zhao for fruitful discussions regarding the feedback control, J. Davidson for data analysis suggestions, A. Gonzalez-Garcia for helping conduct experiments on watercraft, and N. Troje for insightful comments. We thank A. Poehlmann, J. Stowers, and M. Hofbauer from Loopbio GmbH for technical support with the VR systems and the animal care staff at the University of Konstanz, including A. Bruttel, C. Bauer, J. Weglarski, and D. Leo, for help in taking care of and preparing fish for experiments. We thank M. Günther, J. Peters, S. Wang, G. Wang, K. Kumari, and I. Hachen for help in running robot experiments. **Funding:** This work was supported by the following: Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy–EXC 2117-422037984; to I.D.C.: the Max Planck Society, the European Union's Horizon 2020 Research and Innovation Programme under Marie Skłodowska-Curie Grant 860949, the DFG Gottfried Wilhelm Leibniz Prize 2022 584/22, the Struktur-und Innovationsfonds für die Forschung of the State of Baden-Württemberg, the PathFinder European Innovation Council Work Programme 101098722, and the Office of Naval Research grant N0001419-1-2556; to M.N.: Hungarian Academy of Sciences Grant 95152 (to the MTA-ELTE "Lendület" Collective Behaviour Research Group) and Eötvös Loránd University; to L.L.: Messer Foundation Award and Sino-German grant M-0541; to W.W. and D.R.: a grant from the Amsterdam Institute for Advanced Metropolitan Solutions (AMS) in the Netherlands; to W.W.: a startup grant provided by the Department of Mechanical Engineering and the College of Engineering at the University of Wisconsin-Madison, and the Office of the Vice Chancellor for Research and Graduate Education with funding from the Wisconsin Alumni Research Foundation. **Author contributions:** L.L., M.N., and I.D.C. conceived the idea and designed the project. L.L. conducted the fish VR experiments and collected data. L.L. and G.A. conducted experiments and collected the 3D tracking and matrix data. L.L. and M.N. analyzed data with contributions from I.D.C. and O.D. L.L., R.W., and W.W. conducted experiments on robots and collected data with contributions from O.D. and D.R. L.L., M.N., and I.D.C. wrote the initial draft, and all authors contributed to the revision of the text. **Competing interests:** The authors declare that they have no competing interests. **Data and materials availability:** All data needed to support the conclusions of this manuscript are included in the main text or Supplementary Materials. Source data and code are available via Dryad (DOI: 10.5061/dryad.np5hqc02k).

Submitted 26 May 2024
Accepted 2 April 2025
Published 30 April 2025
10.1126/scirobotics.adq6784

ACTUATORS

Explosion-powered eversible tactile displays

Ronald H. Heisser[1]†‡, Khoi D. Ly[1]†, Ofek Peretz[1], Young S. Kim[1], Carlos A. Diaz-Ruiz[1], Rachel M. Miller[1], Cameron A. Aubin[1]§, Sadaf Sobhani[1], Nikolaos Bouklas[1,2], Robert F. Shepherd[1]*

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

High-resolution electronic tactile displays stand to transform haptics for remote machine operation, virtual reality, and digital information access for people who are blind or visually impaired. Yet, increasing the resolution of these displays requires increasing the number of individually addressable actuators while simultaneously reducing their total surface area, power consumption, and weight, challenges most evidently reflected in the dearth of affordable multiline braille displays. Blending principles from soft robotics, microfluidics, and nonlinear mechanics, we introduce a 10-dot–by–10-dot array of 2-millimeter-diameter, combustion-powered, eversible soft actuators that individually rise in 0.24 milliseconds to repeatably produce display patterns. Our rubber architecture is hermetically sealed and demonstrates resistance to liquid and dirt ingress. We demonstrate complete actuation cycles in an untethered tactile display prototype. Our platform technology extends the capabilities of tactile displays to environments that are inaccessible to traditional actuation modalities.

INTRODUCTION

The dream of communicating electronic information through the sense of touch is nearly as old as modern computing. Among the first tactile display attempts were Robert Gault's teletactor in 1933 (*1*), a device that attempted to spread the various frequencies in speech as vibrations across the palm for the deaf, and a "hearing glove" prototype that Norbert Wiener designed and demonstrated for Helen Keller in 1949 (*2*, *3*). The Optacon, a "seeing finger" for blind readers, directly converted images from a scanning stylus to vibration patterns in a 144-pin array connected to piezoelectric bimorph actuators, providing high-resolution tactile images of any printed text to one index finger (*4*, *5*). Although simple vibrotactile feedback mechanisms are now ubiquitous in mobile phones, game controllers, and laptop track pads, knowledge of the millions of touch receptors in our skin motivates modern platforms that poke and prod with the resolution of skin's spatial acuity over large surface areas. Design requirements for wearable, untethered haptic feedback devices add constraints to this endeavor; for example, an actuation surface must also conform to the body, resist dirt and moisture, and tolerate user wear and tear.

Here, we focus on the development of millimeter-scale tactile displays needed to modernize precision computer access for people who are blind or deafblind or have limited vision—a litmus test for high–degree-of-freedom (DoF) actuation arrays (we use "blind" generally here). The "Holy Braille" challenge outlines design requirements for a currently unrealized full-page refreshable braille e-reader (*6*, *7*). A full page of braille has 25 lines, each line having 40 braille cells that use combinations of six possible dots to form characters, totaling 6000 actuators in a refreshable display. Given the size of a page of braille, portable multiline braille displays will likely have fewer than 6000 dots, although no single design has been decided. Dots should rise ~1 mm in a sequence fast enough to refresh the entire display in 0.1 to 1 s, depending on the reading application (*6*).

[1]Department of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14850, USA. [2]Pasteur Labs, Brooklyn, NY 11205, USA.
*Corresponding author. Email: rfs247@cornell.edu
†These authors contributed equally to this work.
‡Present address: Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA.
§Present address: Robotics Department, University of Michigan, Ann Arbor, MI 48109, USA.

Same as visual literacy, the ability to read braille words constructed from individual characters is crucial for cognitive fluency and strongly correlates with upward social mobility (*8*–*10*). Unlike computer screens, most commercially available braille displays have a single line of 40 characters or fewer; users who are blind can only read small snapshots of digital text at a time (see fig. S7A). The lack of progress in braille display technology has allowed for an increasing reliance on text-to-speech screen reading software that circumvents learning braille. Although convenient, the greater use of screen readers and the disincentivization of teaching braille in public school systems continue to lower global braille literacy. Therefore, multiline displays are essential for two reasons. First, they enable computer graphical unit interface interactions, such as editing spreadsheets, reading sheet music, and practicing mathematics and computer science. Second, these technologies promote the mastery of braille in an increasingly digital age. We provide additional contextual discussion in Supplementary Methods.

In the past 5 years, new commercial ventures have proposed untethered, portable multiline display solutions (fig. S7). For example, Dot Inc. has engineered a braille cell (a six-dot unit) of electromagnetic flip-latch actuators for use in their own multiline e-reader (Dot Pad, 2400 actuators) and kiosk (*11*, *12*). Tactile Engineering Inc. recently announced a handheld, four-line–by–12-cell device (Cadence, 384 actuators) that can wirelessly mate with additional devices to extend the reading range to eight lines of 24 cells (*13*). These actuation systems have fast refresh rates and scalable architectures yet suffer the drawbacks of tight manufacturing tolerances, high part counts, and high costs (see table S3). These systems are also sensitive to dirt accumulation, liquid ingress, and drop impact, requiring frequent maintenance because a single malfunctioning dot can render a cell or line useless.

The interdisciplinary field of soft robotics embraces diverse and unconventional modes of actuation, aiming to improve device size, weight, power consumption, and cost (SWaP-C) (*14*, *15*). Hallmark soft robotic devices rout pressurized air to inflate elastomeric chambers, producing mechanical work without sliding parts, and are naturally resistant to damage from impact, having potential to solve some elusive tactile display design challenges. However, valve-based control strategies have prohibitively low SWaP-C performance when valves are individually connected to actuators, given their bulk, power consumption,

and cost (*16*). The benefits of haptic pneumatic actuation continue to motivate advances in microfluidic logic systems (*17*), including efforts from NewHaptics to build a pneumatic braille display (*18*). Recent electrofluidic soft actuation methods have been proposed for haptic arrays, including submillimeter-thick hydraulically amplified electrostatic soft actuators that can produce 1 mm of deformation in a 5-mm^2–by–5-mm^2 footprint and electro-osmotic actuators that can raise 2-mm-diameter dots to heights >1 mm in 1 s (*19*, *20*).

Here, we extend the operational capabilities of multiline tactile displays with an array of domed membranes that evert (turn inside out) upward by explosive pressure differentials and downward by vacuum suction. Previous work demonstrates that by spark-igniting microliter volumes of oxyfuel gas mixtures inside elastomeric chambers, arrays of fluidic actuators can be individually controlled without valves (*16*, *21*, *22*). By leveraging elastomeric bistability, we have produced a latching display platform that is hermetically sealed, has no sliding parts, and reduces the total moving part count from thousands to a single-molded sheet. Applied to tactile displays, our device architecture can bring affordable, large-format systems to environments (i.e., transit stations, the kitchen, or outdoor spaces) in which current displays cannot feasibly operate (*23*). We compare current technologies in Supplementary Methods and table S3.

RESULTS

Array design

Figure 1A shows our eversible tactile display, consisting of 100 equidistant dots arranged in a 10-dot by 10-dot pattern. Each dot has a 2-mm base diameter, is ~1-mm high, and is spaced 2.5-mm center-to-center from neighboring dots. Compression molding formed the arrayed actuation membrane from one-part high-consistency silicone rubber (Fig. 1A). The full array comprised three layers of molded polydimethylsiloxane (PDMS) stacked on a rigid printed circuit board (PCB) that we encapsulated in PDMS to form a compatible adhesion surface.

Each of the four nonactuating layers had microfluidic conduits that carried fuel and exhaust gas mixtures throughout the device (Fig. 1A, gas route). We organized these channels perfusively; they

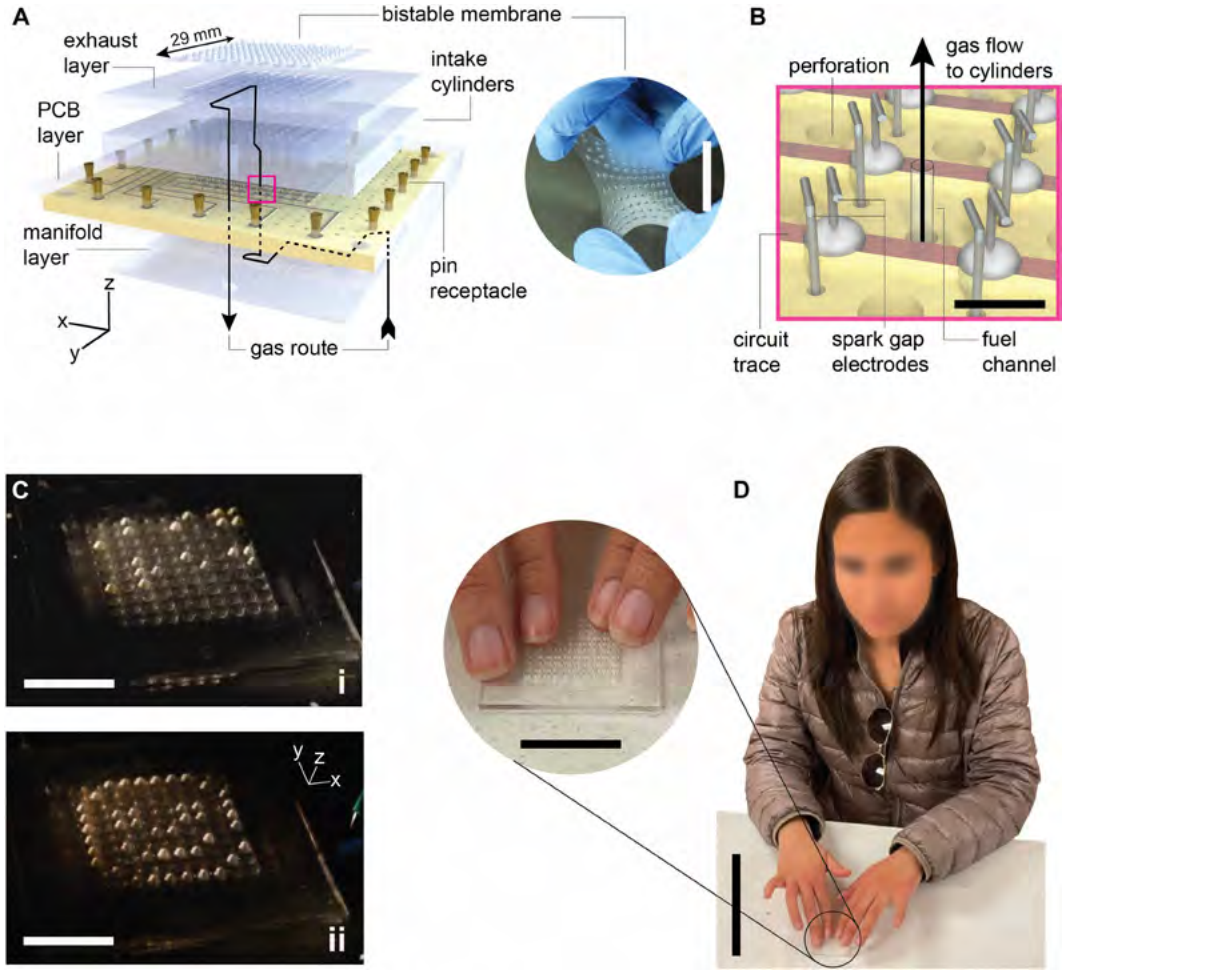


Fig. 1. Microcombustion generates legible tactile display patterns. (**A**) Stacked, molded PDMS layers efficiently distribute fuel and exhaust through actuator cylinders. The gas route goes through a single cylinder (three down, six across). Photo of membrane layer visualizing its hyperelasticity. Magenta square corresponds to design detail in (B). Scale bar, 25 mm. (**B**) Magnified rendering of PCB layer showing arrangement of electrodes bent and resting on top of the PDMS surface. Perforations mechanically engage cured PDMS with both sides of the PCB. Scale bar, 1.5 mm. (**C**) Display shows "Big Red" in uncontracted or grade 1 braille (i). Display shows smiling emoji with an included nose (ii). Scale bars, 15 mm. (**D**) A participant tests feel and legibility of the display surface. Scale bars, 125 mm (main image) and 25 mm (zoomed-in image).

three-dimensionally penetrated through the structure of the actuators, mimicking arterioles and venules that connect to form capillary networks (*24*). A single intake gas channel is connected to the bottom of the device, where oxygenated reactant gas entered and successively branched until it flowed upward through the PCB into one of the 100 possible combustion cylinders (Fig. 1B). After combustion, deoxygenated exhaust gas coalesced into one exhaust channel and exited the device, refreshing with new reactant gas.

Spark-gap ignition triggered combustion in each cylinder. Pin receptacles along the PCB border transmitted high-voltage (HV) signals to the cylinder through crisscrossing circuit traces. New silicon carbide opto-MOSFET (metal–oxide–semiconductor field–effect transistor) relays provided a compact means to switch HV signals as high as 3.3 kV from a single dc-dc converter (*16*). To generate a spark, one HV relay switched a top, widthwise (*x* direction Fig. 1A) trace to +V, and another relay switched a bottom, lengthwise trace to an HV-return or low-voltage (LV) line, generating a spark above from where the two traces intersected. Spark gaps were formed by soldering copper wires (40 American Wire Gauge, AWG) to pads on widthwise and lengthwise traces (Fig. 1B), given that they have simpler construction and greater wear resistance than microfabricated spark gaps (*25*). We left a portion of the wires protruding from the encapsulated PDMS surface to evenly space and laid them flat next to the cylinder intake port. Each igniter had an average gap distance of 87 ± 18 μm. We continuously switched sparks at 10 Hz for more than 200 hours and found no appreciable change in spark intensity or consistency. See Supplementary Methods and movies S1 to S3 for further manufacturing details.

We conducted initial pattern generation tests by connecting the device to an external HV control system and gas supply to sequentially ignite methane-oxygen mixtures with an equivalence ratio $\Phi = 0.34$. The equivalence ratio is a nondimensional relationship comparing the actual fuel-air (*F*/*A*) ratio with the stoichiometric *F*/*A*

$$\Phi = \frac{\left(\frac{F}{A}\right)_{\text{actual}}}{\left(\frac{F}{A}\right)_{\text{stoich}}} = \frac{\left(\frac{n_{\text{fuel}}}{n_{\text{air}}}\right)_{\text{actual}}}{\left(\frac{n_{\text{fuel}}}{n_{\text{air}}}\right)_{\text{stoich}}} \quad (1)$$

where *n* is the moles of the mixture species. Combining eversion with the kinetics of gas-phase combustion resulted in fast actuation speeds. High-speed videography showed that each dome flips upward in ~0.24 ms, having an average tip velocity of ~8.3 m s^{-1} (movie S4). Considering an average braille reading speed, devices need enough output power to facilitate ~1000 dot actuations per minute (*6*). During pattern generation, our system drew only 2.2 W from the HV amplifier and a pair of relays. We maintained consistent ignition performance at or below 30 Hz or 1800 actuations per minute, limited by switching times in the HV relays. Beyond this frequency, we found that misfiring occurs. Given that the relays we used have a specified total switching cycle time of ~1 ms, we believe that pattern generation rates can be improved to reduce the display refresh rate. Tactile displays have highly variable refresh rates that depend on the rising speed of each dot, aggregate dot power draw, and control sequences of onboard driving electronics. We defined refresh rate by the time taken to go from one static complete pattern to the next. Refresh rates in our system are limited by gas mixing and refilling times, discussed further in the "Untethered system integration" section.

As seen in Fig. 1C and movie S5, our array repeatably produced prescribed display states. Figure 1Ci shows a braille inscription of "Big Red." Figure 1Cii shows the ability to form tactile graphics, making a smile emoji. Once raised, the dots resisted downward eversion from finger pressure (movie S4). Pushing the dot to the surface of the tactile display with the finger did not displace the membrane to its critical buckling point, and the dot rose back to its raised position when the finger was lifted. We tested pressing forces of up to 35 N with a 9-mm-diameter post, and dots remained raised throughout (fig. S8). Further, we demonstrated some of the environmental advantages of our device's soft robotic design by pouring water on the display as a hexagonal pattern formed. The dots even splashed water out of the way as they rose (movie S6).

In our search of prior art, we could not find previous displays made of rubber. Therefore, we confirmed that our rubber displays tactile comfort and legibility by evaluating prescribed text patterns with nine braille readers. Participants had variable braille-reading experiences from years to 3 weeks. With our equidistant dot spacing (different from standard braille spacing), eight of the nine participants successfully read "Big Red." Advanced braille readers could quickly recognize the smile emoji.

Actuator mechanics

Core to our design is a thin, elastomeric membrane that everts from a concave pit to a convex dome. Our mold produced the convex geometry, so the initial configuration is the convex (upward) state. The dome is thimble shaped, having a variable thickness from 125 μm at its flexural hinge to 180 μm at its top, and is ~1 mm tall (Fig. 2A). The thickness variability is a unique control that can allow desirable actuation performance through design geometry optimization. Our design balances tactile legibility, manufacturing ease, and stability in the two everted states. Decades of research have explored the buckling, wrinkling, and snap-through instabilities of elastic shell structures, often focusing on spherical and ellipsoidal geometries. For a thin spherical cap, bistability depends on a relationship between the bending and stretching stresses along the membrane, being independent of material elasticity. Mechanicians typically represent this relationship with λ (*26*, *27*)

$$\lambda = \sqrt[4]{12(1-\nu^2)}\sqrt{\frac{R}{t}}\alpha \quad (2)$$

where ν, R, t, and α, respectively, represent Poisson's ratio, radius of curvature, thickness, and planar angle. Although this analysis is limited to spherical, constant thickness geometries, the formula provides intuition for our design effort. Previous experimental work demonstrated that free boundary condition (BC) "deep" shells with a curvature approaching a half sphere or $\alpha \lesssim 75°$ reliably exhibit bistable behavior for $\lambda > 5.75$. A 100-μm-thick spherical cap with a 2-mm base diameter and a 1-mm radius of curvature has $\lambda = 8.6$, comfortably bistable for the free case yet less so for our clamped BCs because of the added moment at the base (*28*).

Given a fixed thickness and base radius, taller ellipsoidal shells have higher axial stiffnesses than spherical shells (*29*, *30*). Therefore, our thimble-shaped braille dot could maintain a pronounced tactile feel without increasing the thickness, which would decrease eversion stability. It may also be possible to induce snap-through inversion at lower pressures by thoughtfully introducing membrane thickness variations or defects (*31*).

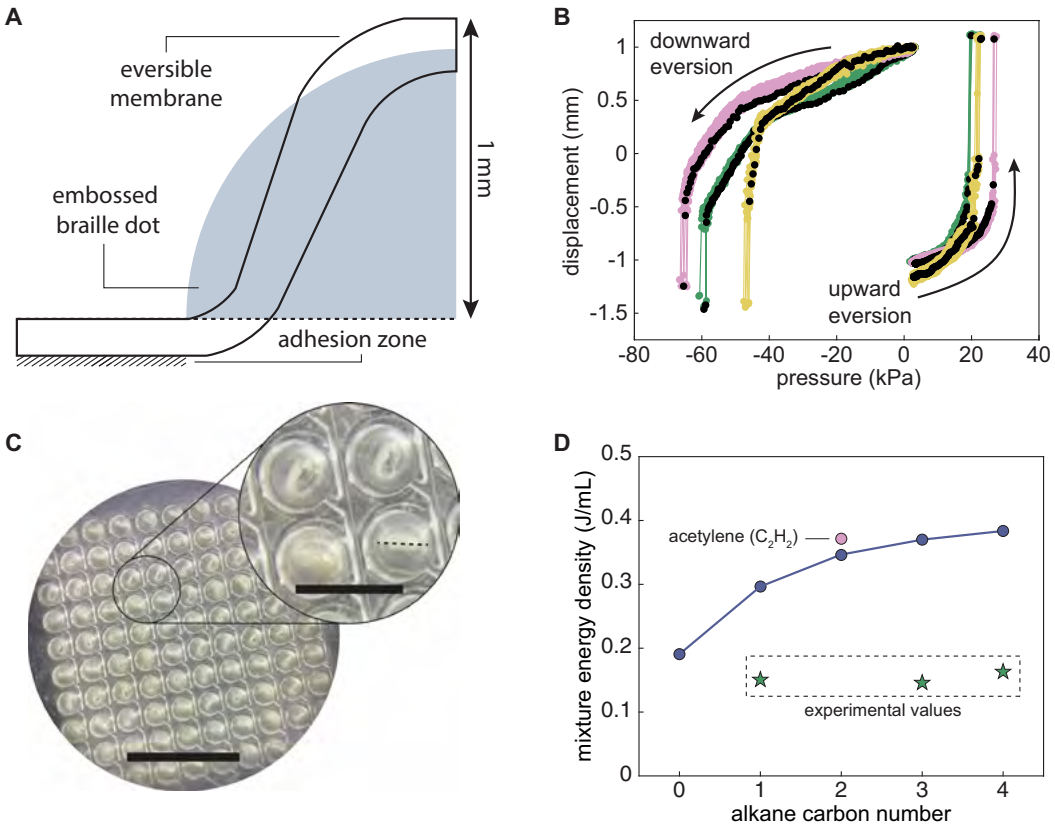


Fig. 2. Mechanical characteristics of a bistable braille dot. (**A**) Diagram compares membrane geometry cross section with maximum standard dimensions of embossed braille dot (in gray). The outer edge adheres to the lower PDMS layer on the bottom surface. (**B**) Experiments determine pressure-displacement relationships for individual membranes. The colored shaded areas are overlays of all points obtained during the eight measurements taken for each membrane sample, directly demonstrating low variability between individual actuation cycles. Each color represents a different membrane sample. The black dots show a single actuation cycle measurement for each membrane. (**C**) Image taken under the exhaust layer with downward-everted actuators. Inset shows exhaust channels. Dotted line shows leading edge of pinched duckbill shape. Scale bars, 10 mm and 2.5 mm (inset). (**D**) Points connected by the line show stoichiometric volumetric energy densities for oxyfuel mixtures of alkanes (C_nH_{2n+2}) that are gases at room temperature and pressure. We added acetylene to compare alkane energy densities with another commonly used premixed fuel. Green stars show energy densities for mixtures empirically determined to be compatible with our array system.

We experimentally characterized quasistatic downward and upward eversion by simultaneously measuring pressure and tip displacement while actuating the membrane with a syringe connected to a single chamber (Fig. 2B). Results showed that upward eversion requires, on average, ~20 kPa and downward eversion requires ~60 kPa of vacuum. If we assume a conservative estimated downward eversion pressure of 40 kPa (Fig. 2B) and a full 2-mm contact area, then the downward eversion pressure corresponds to a force of 125 mN, within previously described holding force limits of 50 mN (*6*). Critical membrane positions occurred near the dot surface (0 mm) for upward eversion and 0.5 to 0.75 mm under the dot surface for downward eversion. Thickness variability in membrane manufacturing influenced the snap-through critical load, especially when flipping downward. Finite element simulations suggested that our molding methods maintained membrane thickness deviations within 50 μm of the intended design (fig. S2). Downward-everted actuators tended to pinch into a duckbill shape, losing their upwardly axisymmetric form (Fig. 2C). Dots must resist snap-through from downward presses to be useful as a tactile reading surface. Fingers are generally not slender enough to push dots past the critical position, guaranteeing that a pattern will remain under normal operation until vacuum reset. Movie S6 shows that our actuators remain upright against different degrees of finger pressure despite being momentarily collapsible.

During actuation, pressure rise from combustion and membrane elasticity contribute to braille dot dynamics. Fingers that remain on a display as it refreshes will obstruct the motion of the moving dots. The microliter combustion volume and submilligram mass of each actuator's membrane amount to a brief, barely perceptible sensation on a fingertip. From video data, we estimated a membrane kinetic energy of 8 μJ compared with that of a stinging airsoft ball bearing, which can be more than 0.8 J (*32*). When slowly pressurized, rising dots critically buckle when the membrane center is within 0.1 mm of the display surface. Combustion introduces additional nonlinearities into the membrane's displacement. High-speed video reveals that when combustion-actuated membranes collide with an obstruction up to 0.9 mm above the display surface, they briefly remain in place and then suddenly flip back downward (Supplementary Methods). Given that an actuator stably crosses the critical buckling point under quasistatic conditions, further investigation will determine how combustion can be tuned to produce stable actuations under obstruction.

Although the actuator itself does no external work, we can still define efficiency with a minimum actuation energy (MAE) or the minimum energy required to induce upward eversion. To the first order, we estimated an average MAE of ~100 μJ from our experiments by calculating the pressure-volume work done on the membrane until it everted upward. Testing methane, propane, and butane, we determined oxyfuel stoichiometries that provided sufficient pressures to meet the MAE for our 2-mm-diameter, 7-mm-height actuator cylinder geometry (Fig. 2D). In our microliter combustion system, flames became unstable outside a range of intermediate fuel concentrations. Low fuel concentrations resulted in quenched flames, and high fuel concentrations caused flames energetic enough to propagate backward (flashback) through the microfluidic channels and ignite other chambers. We also established flashback and flame-quenching limits for each fuel in our actuator geometry (table S2). Oxymethane-powered eversions at $\Phi = 0.34$ had a chemomechanical efficiency of 0.086%. Operational propane ($\Phi = 0.35$) and butane ($\Phi = 0.33$) mixtures had efficiencies of 0.076 and 0.078%, respectively. In our system, combustion was naturally less efficient; open channels, low wall temperature, and high surface-to-area ratio all contributed to energetic losses at microliter scales (*16*, *33*). Considering mixture volumetric energy density and the required equipment for our fuel candidates, we believe that butane best suits this application.

Untethered system integration

We untethered our actuation system by integrating all functional components required for repeated display pattern generation onto an acrylic board. Figure 3A presents a system diagram of the main functional components (see Supplementary Methods and fig. S3 for system component details). An Arduino Mega controlled all relays, valves, and the vacuum pump. We programmed three buttons to execute the fuel intake, combustion, and vacuum steps of a display cycle (Fig. 3, B to D, respectively).

We connected butane and oxygen canisters to miniature gas regulators to make them compatible with the inline solenoid valves. The pressure regulator also reduced inline butane pressure below the room-temperature boiling point to ensure that only gaseous butane proceeded, given that small changes in butane stoichiometry can easily cross flashback and flammability limits at these length scales (*16*). The size and number of combustion chambers in our device require that gases be premixed before fuel intake. We accomplished this task simply by introducing a 3D printed elastomeric mixing chamber manifold connecting butane, oxygen, and fuel intake lines with integrated solenoid valves (Fig. 3B). The oxygen valve opened first, inflating a flat circular membrane until it reached a $\Delta p_{mix} = 5.3$ kPa set point (movie S7), measured by an onboard pressure sensor connected to the chamber. Then, the butane valve opened for 16 ms to mix in the fuel. Opening the third valve caused the membrane to deflate from its own hyperelasticity, filling the actuator array with new reactant gas and expelling the remaining exhaust gas through a check valve. Further, deflating the membrane ensured that no combustible amount of gas remained in the mixing chamber.

After combustion and tactile pattern generation, dots reset via a separate three-way valve that connects the vacuum chamber directly to the cylinder array. A portable diaphragm pump first pulled air from a 50-ml conical tube until reaching a gauge pressure of $\Delta p_{vac} = -62$ kPag (Fig. 3D). Powering the valve connected the array to the vacuum chamber and flipped all dots downward, causing the soft array to slightly "pucker" inward (movie S7). Excluding membrane and fluid dynamics, the pressure required to pull the dots back down can be expressed by the following relationship

$$p_{vac} = \frac{V_{ch} + V_{ar}}{V_{ch}} \left(p_{st} - p_{amb} \frac{V_{ar}}{V_{ch} + V_{ar}} \right) \quad (3)$$

where V_{ch} is the vacuum chamber volume, V_{ar} is the actuator array volume, p_{st} is the membrane downward snap-through pressure, and p_{amb} is the ambient pressure. We calculated $p_{vac} = -63.5$ kPag when assuming $p_{st} = -60$ kPag, close to our measured vacuum set point. Given that the system's exhaust port check valve prevents backflow of ambient air, the actuator array remained at a negative gauge pressure, and new gas re-established ambient pressure; the pressure differential helped pump fresh reactant into the array faster than just with the mixing chamber membrane's own elasticity. Once gas had replenished the array, the next actuation sequence could occur.

Compared with conventional electromechanical strategies, our combustion-powered display refresh cycles had additional steps to process fuels. Our current untethered display prototype is oxygen-limited in its capacity. Accounting for additional reactant gas in the external fuel channels, we conservatively assumed that a refresh volume twice the volume of the actuator can refill the display. Then, our 18-ml butane canister had enough fuel for almost 20,000 refreshes, whereas our 5-liter oxygen canister carried 1200 refreshes worth of gas.

The combined time for mixing and filling (20 s), pattern generation (3 s), and vacuum reset (1.5 s) was 24.5 s (movie S7). The mixing and refilling step is slow at present because of our simple elastomeric mixing chamber. We can reduce this time by directly injecting needed quantities of butane in line with a pressurized oxygen stream. By matching oxygen pressure to the 243-kPa vapor pressure of liquid butane at room temperature, we can achieve flow rates greater than 10 ml s^{-1} in our microfluidic channel geometries. Given that the total volume of all 100 actuators in our system is 2.2 ml, we believe that these flow rates can refill our current system in under 1 s. Our tethered experimental setup produced reactant flow rates of 2 ml s^{-1} and enabled >1-Hz combustion in single actuators. Figure S9 contains additional discussion of how refreshing processes can run in parallel to be compatible with braille reading times, especially for large-format braille displays.

DISCUSSION

We have developed a soft, scalable actuation array system that repeatably produces latched display states at the braille scale. Our simple architecture of spark gap electrodes, microfluidic channels, and eversible elastomeric membranes enables massive moving part reduction and thereby lowers material cost and manufacturing complexity, scaling to an arbitrary number of actuators. For $2n$ row-column pairs, the total HV relay area scales as $A_{relay} \sim n$ and the dot area scales as $A_{dot} \sim n^2$. For our unit relay area, the two footprints reach parity after 18 rows. For our design, a full page of braille needs 155 relays; relays occupy a fraction of the page area (Fig. 3E and fig. S3). Therefore, the components required to control actuation can fit neatly under the actuation surface for larger reading formats.

Our prototype system mechanically resets dots through vacuum and replenishes gas in each chamber to refresh the display with a new pattern. The added fuel steps currently result in a lower refresh rate for our device than other systems on the market; given longer instances of reading for scaled display systems, fast refresh rates are a less critical device metric. We believe that this drawback is worth

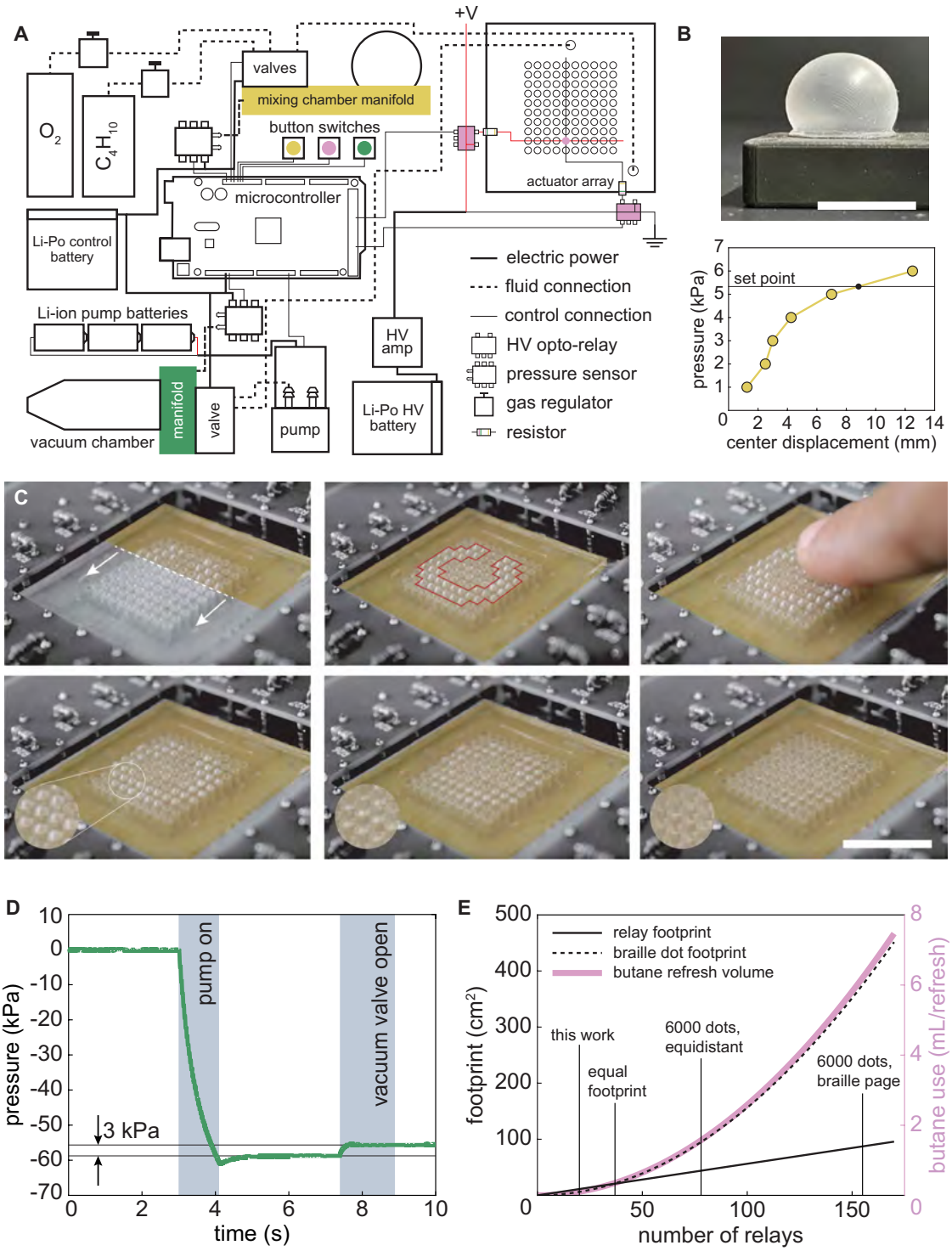


Fig. 3. Untethered tactile display operation. (**A**) Simple electrical and fluidic components connect to perform fuel intake, pattern generation, and vacuum reset actions. (**B**) The elastomeric mixing chamber inflates with oxygen until reaching a pressure set point and then receives a metered volume of butane gas, pumping the mixture when the intake valve is opened. Scale bar, 15 mm. (**C**) A large, bold "C" forms on the tactile display and resets after reading. The actuator array plugs into the system from under the PCB (gold pins). Scale bar, 20 mm. (**D**) Plot shows vacuum chamber pressure over a single-array reset. After initial vacuum establishment, the pump only needs to restore ~3 kPa per display cycle. (**E**) Many actuation strategies have repeated control components that scale incompatibly with the size of the overall device. Although the total area of 20 relays is larger than our 10-dot–by–10-dot display, the HV control relays would have a smaller footprint than the actuators after only 38 relays, or 361 dots, are used. A full page of braille is 816 cm^2, including margins, and would require 155 relays, which occupies 87 cm^2.

exploring further, given the system advantages of our soft, fluidic framework. The display surface is sealed from the external environment and can operate in wet, dirty environments that risk failure in current displays. For example, we envision a desktop implementation of our device useful for reading recipes while cooking, where oil and ingredients would not cause dot jamming in our membranes. The surface can be wiped clean or washed with soap and water. As a kiosk, our display could provide valuable information in unpredictable public spaces such as museums, libraries, and convention centers. Outside, our device can operate through rain, dirt, and pollution at parks and bus stops.

Tactile displays require maintenance from dot failure during use. Most display systems are highly integrated and must be shipped for professional repairs, historically leaving users without digital braille access for weeks at a time. Because our display connects simply to the other system components from press-fit gas connections and electrical contacts, no formal training is required to install a new display. We take less than 30 s to replace the 10-dot–by–10-dot array in our prototype.

Our untethered system contains all necessary components to repeatedly produce tactile patterns. Given the nature of components needed to supply fuel and pull vacuum, this system is unlikely to have the slim profile of a typical e-reader. The need for a stored source of oxygen is the core limitation of this technology because it has a lower density than the liquid butane fuel and will need more frequent replacement. Oxygen concentrators that normally supply >5 liters min^{-1} for breathing could be miniaturized to supply the milliliter min^{-1} gas consumption requirements for this system. The recent severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) pandemic has inspired research into compact electrochemical oxygen generators that could be modified to fit our system's oxygen requirements (*34*). The scale and brevity of combustion needed in our system warrant an investigation into other recyclable chemical actuation strategies such as water electrolysis–generated hydrogen combustion, ultraviolet photocatalysis of electrolyzed conjugate base pairs, and the controlled decomposition of hydrogen peroxide (*35*, *36*).

The most expensive component in this display is the HV relay. At our low-volume cost of $40 per relay, a full page of braille would require $6200 of these components (see fig. S5 for the materials cost of this prototype). We generate sparks at 2 kV, although we observed sparks at voltages as low as 1.2 kV. With more precise alignment of electrodes, we believe that we can use 1.5-kV relays, which have unit costs of under $5 at scale (see Panasonic AQV258). Figure S5 contains a material cost breakdown of our prototype.

Our system has potential to scale to thousands of actuators and serve applications beyond tactile displays. Designers can integrate bistable domes into other microfluidic systems to improve mechanical capabilities of droplet production and other manufacturing processes (*37*, *38*). High-throughput cell processing equipment could benefit from large arrays of pumps, valves, or microwell agitators (*39*). With our general actuation framework, high-density arrays of combustion cylinders could couple to cell substrates such that finely tuned pulses apply small strains to the cells, enhancing cell proliferation, alignment, and biochemical production (*40*–*42*).

MATERIALS AND METHODS

Device fabrication

We cured individual layers of PDMS using 3D printed molds to form the intake and exhaust channels and combustion cylinders. The electrode layer with out-of-plane channels was a perforated PCB encapsulated in PDMS to provide a compatible bonding surface with the other actuator layers. Each PDMS layer used 50 μm of prepolymer as adhesive. Eversible actuator sheets were hot-press molded out of one-part silicone (high-consistency rubber, Elastosil 4001/40 MH, Wacker) using a custom-machined aluminum mold and were bonded to the top of the device with a 25-μm layer of silicone epoxy (Sil-Poxy, Smooth-On).

The system prototype combined off-the-shelf and custom components onto a single laser-cut acrylic sheet to visualize all system parts with their fluidic and electric connections. The female pins lining the edge of the actuator array were connected to opposing pins on a system PCB. We fastened the corners of the actuator to the acrylic board to ensure connection. Separate onboard batteries powered the electronic control system, HV electronic components, and vacuum pump. The 3D printed mixing chamber and vacuum chamber had integrated channels that are connected to the solenoid valves. The 3D printed scaffolds held the oxygen and butane canisters in the depressed position for downstream valve flow control.

Pressure-displacement test

We obtained the pressure-displacement relationship of our bistable actuator by simultaneously measuring the membrane position and chamber pressure through eversion and inversion cycles, manually controlling pressure for the experiments. We bonded a single membrane to an isolated chamber, connected the intake port to a syringe, and connected the exhaust port to a piezoresistive pressure sensor (SSCDRRN015PDAA5, Honeywell). Above the membrane is a laser displacement sensor aligned with the membrane's center. We took measurements by slowly pushing the syringe depressor until the membrane flipped up and pulling the depressor until the membrane flipped down.

Fuel test

Using the actuator array, we empirically determined operational oxyfuel stoichiometries by varying the reactant flow rate and igniting a single dot until successful latching occurred. Mass flow controllers (MFCs) maintained flow rates with a precision of 0.1 ml min^{-1}. We used a laboratory compressed gas cylinder for methane experiments. We connected a butane canister to a water column to filter out butane steam that caused stoichiometry errors. We connected a miniature propane tank to a gas grill regulator to match operating pressures with the MFCs.

Pattern formation test

We conducted initial actuator array tests with our tethered experimental setup using methane for simplicity. Each HV trace is connected to an HV relay through PCB wires connected to a breadboard. Arduino protocols controlled the ignition sequences.

Untethered system actuation cycle test

The untethered system components were arranged in an exploded assembly format on an acrylic sheet to make all parts visible. Three separate batteries isolated the HV electronics, control electronics, and vacuum pump. A PCB was custom-designed to combine the microcontroller and HV components into one component that connected to the actuator array. To test the entire system, a display pattern was loaded onto the Arduino, and three input buttons were programmed to initiate fuel mixing and intake, ignition sequences, and

vacuum reset. Onboard oxygen (10 liters of Natural, Boost Oxygen) and butane (75 ml of butane fuel, Ronson) connected via miniature pressure regulators (3834 T51, McMaster-Carr) to two-way solenoid valves (X-8-05-S-F, Parker-Hannifin) provided the necessary oxyfuel mixture species, with volume control given by the pressure sensor. After filling, the combustion sequence initiated and produced the desired tactile pattern. Then, the vacuum chamber was set to the appropriate negative gauge pressure, and the dots were reset when pressing the final input button.

Supplementary Materials

The PDF file includes:
Methods
Figs. S1 to S9
Tables S1 to S3
Legends for movies S1 to S7
References (*43*)

Other Supplementary Material for this manuscript includes the following:
Movies S1 to S7
MDAR Reproducibility Checklist

REFERENCES AND NOTES

1. R. H. Gault, L. D. Goodfellow, Eliminating hearing in experiments on tactual reception of speech. *J. Gen. Psychol.* **9**, 223–228 (1933).
2. N. Wiener, Problems of sensory prosthesis. *Bull. Am. Math. Soc.* **57**, 27–35 (1951).
3. M. Mills, On disability and cybernetics: Helen Keller, Norbert Wiener, and the hearing glove. *differences* **22**, 74–111 (2011).
4. N. Efron, Optacon—a replacement for braille? *Aust. J. Optom.* **60**, 118–129 (1977).
5. L. H. Goldfish, H. E. Taylor, The optacon: A valuable device for blind persons. *J. Vis. Impair. Blind.* **68,** 49–56 (1974).
6. N. Runyan, D. Blazie, "EAP actuators aid the quest for the 'Holy Braille' of tactile displays" in *Electroactive Polymer Actuators and Devices* (*EAPAD*) *2010* (SPIE, 2010), vol. 7642, p. 764207; https://www.spiedigitallibrary.org/conference-proceedings-of-spie/7642/764207/EAP-actuators-aid-the-quest-for-the-Holy-Braille-of/10.1117/12.847764.short.
7. A. Russomanno, S. O'Modhrain, R. B. Gillespie, M. W. M. Rodger, Refreshing refreshable braille displays. *IEEE Trans. Haptics* **8**, 287–297 (2015).
8. A. M. Silverman, E. C. Bell, The association between braille reading history and well-being for blind adults. *J. Blind. Innov. Res.* **8**, 10.5241/8-141 (2018).
9. E. C. Bell, A. M. Silverman, Rehabilitation and employment outcomes for adults who are blind or visually impaired: An updated report. *J. Blind. Innov. Res.* **7**, 10.5241/8-148 (2018).
10. R. Ryles, The impact of braille reading skills on employment, income, education, and reading habits. *J. Vis. Impair. Blind.* **90**, 219–226 (1996).
11. Dot Inc.; http://www.dotincorp.com/.
12. J. Kim, B.-K. Han, D. Pyo, S. Ryu, H. Kim, D.-S. Kwon, Braille display for portable device using flip-latch structured electromagnetic actuator. *IEEE Trans. Haptics* **13**, 59–65 (2020).
13. Tactile Engineering; https://tactile-engineering.com.
14. Y. Jung, K. Kwon, J. Lee, S. H. Ko, Untethered soft actuators for soft standalone robotics. *Nat. Commun.* **15**, 3510 (2024).
15. C. A. Aubin, B. Gorissen, E. Milana, P. R. Buskohl, N. Lazarus, G. A. Slipher, C. Keplinger, J. Bongard, F. Iida, J. A. Lewis, R. F. Shepherd, Towards enduring autonomous robots via embodied energy. *Nature* **602**, 393–402 (2022).
16. R. H. Heisser, C. A. Aubin, O. Peretz, N. Kincaid, H. S. An, E. M. Fisher, S. Sobhani, P. Pepiot, A. D. Gat, R. F. Shepherd, Valveless microliter combustion for densely packed arrays of powerful soft actuators. *Proc. Natl. Acad. Sci. U.S.A.* **118**, e2106553118 (2021).
17. A. A. Stanley, E. S. Roby, S. J. Keller, High-speed fluidic processing circuits for dynamic control of haptic and robotic systems. *Sci. Adv.* **10**, eadl3014 (2024).
18. NewHaptics; https://www.newhaptics.com/.
19. A. Firouzeh, A. Mizutani, J. Groten, M. Zirkl, H. Shea, PopTouch: A submillimeter thick dynamically reconfigured haptic interface with pressable buttons. *Adv. Mater.* **36**, e2307636 (2024).
20. V. Shen, T. Rae-Grant, J. Mullenbach, C. Harrison, C. Shultz, "Fluid reality: High-resolution, untethered haptic gloves using electroosmotic pump arrays" in *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (Association for Computing Machinery, 2023), pp. 1–20.
21. C. A. Aubin, R. H. Heisser, O. Peretz, J. Timko, J. Lo, E. F. Helbling, S. Sobhani, A. D. Gat, R. F. Shepherd, Powerful, soft combustion actuators for insect-scale robots. *Science* **381**, 1212–1217 (2023).
22. D. P. Holmes, A. J. Crosby, Snapping surfaces. *Adv. Mater.* **19**, 3589–3593 (2007).
23. P. M. Reis, A perspective on the revival of structural (in)stability with novel opportunities for function: From buckliphobia to buckliphilia. *J. Appl. Mech.* **82**, 111001 (2015).
24. J. A. E. Spaan, "Structure and perfusion of the capillary bed" in *Coronary Blood Flow: Mechanics, Distribution, and Control*, J. A. E. Spaan, Ed. (Springer, 1991), pp. 69–86.
25. P. Seriburi, D. Kercher, M. G. Allen, An experimental study of microfabricated spark gaps: Wear and erosion characteristics. *J. Micromech. Microeng.* **11**, 165–174 (2001).
26. M. Taffetani, X. Jiang, D. P. Holmes, D. Vella, Static bistability of spherical caps. *Proc. R. Soc. A Math. Phys. Eng. Sci.* **474**, 20170910 (2018).
27. J. R. Fitch, The buckling and post-buckling behavior of spherical caps under concentrated load. *Int. J. Solids Struct.* **4**, 421–446 (1968).
28. P. M. Sobota, K. A. Seffen, Effects of boundary conditions on bistable behaviour in axisymmetrical shallow shells. *Proc. R. Soc. A.* **473**, 20170230.
29. A. Lazarus, H. C. B. Florijn, P. M. Reis, Geometry-induced rigidity in nonspherical pressurized elastic shells. *Phys. Rev. Lett.* **109**, 144301 (2012).
30. D. Vella, A. Ajdari, A. Vaziri, A. Boudaoud, Indentation of ellipsoidal and cylindrical elastic shells. *Phys. Rev. Lett.* **109**, 144302 (2012).
31. J. W. Hutchinson, Buckling of spherical shells revisited. *Proc. R. Soc. A Math. Phys. Eng. Sci.* **472**, 20160577 (2016).
32. C. Grocock, R. McCarthy, D. Williams, Ball bearing (BB) guns, ease of purchase and potential for significant injury. *Ann. R. Coll. Surg. Engl.* **88**, 402–404 (2006).
33. E. Sher, I. Sher, Theoretical limits of scal announced a handheld ing-down internal combustion engines. *Chem. Eng. Sci.* **66**, 260–267 (2011).
34. Y. Zhang, K. Xie, F. Zhou, F. Wang, Q. Xu, J. Hu, H. Ding, P. Li, Y. Tan, D. Li, J. Zhu, H. Zhou, C. Zhao, S. Lin, Y. Wu, Electrochemical oxygen generator with 99.9% oxygen purity and high energy efficiency. *Adv. Energy Mater.* **12**, 2201027 (2022).
35. L. M. Egolf, J. T. Keiser, Photon-initiated hydrogen-chlorine reaction: A student experiment at the microscale level. *J. Chem. Educ.* **70**, A208 (1993).
36. M. Adami, A. Seibel, On-board pneumatic pressure generation methods for soft robotics applications. *Actuators* **8**, 2 (2018).
37. S. Battat, D. A. Weitz, G. M. Whitesides, An outlook on microfluidics: The promise and the challenge. *Lab Chip* **22**, 530–536 (2022).
38. M. Duran, A. Serrano, A. Nikulin, J.-L. Dauvergne, L. Derzsi, E. Palomo del Barrio, Microcapsule production by droplet microfluidics: A review from the material science approach. *Mater. Des.* **223**, 111230 (2022).
39. P.-Y. Gires, M. Thampi, M. Weiss, Miniaturized magnetic stir bars for controlled agitation of aqueous microdroplets. *Sci. Rep.* **10**, 10911 (2020).
40. C. L. Thompson, S. Fu, H. K. Heywood, M. M. Knight, S. D. Thorpe, Mechanical stimulation: A crucial element of organ-on-chip models. *Front. Bioeng. Biotechnol.* **8**, 602646 (2020).
41. A. Wahlsten, D. Rütsche, M. Nanni, C. Giampietro, T. Biedermann, E. Reichmann, E. Mazza, Mechanical stimulation induces rapid fibroblast proliferation and accelerates the early maturation of human skin substitutes. *Biomaterials* **273**, 120779 (2021).
42. B. Rios, A. Bu, T. Sheehan, H. Kobeissi, S. Kohli, K. Shah, E. Lejeune, R. Raman, Mechanically programming anisotropy in engineered muscle with actuating extracellular matrices. *Device* **1**, 100097 (2023).
43. T. Premkumar, "Braille display by rotating multi-octagonal segment" in *2017 Innovations in Power and Advanced Computing Technologies* (*i-PACT*) (IEEE, 2017), pp. 1–4.

Acknowledgments: We thank the Laboratory of Atomic and Solid State Physics (LASSP) Professional Machine Shop for experimental fabrication services. We thank the Carroll Center for the Blind and the Colorado Center of the Blind for display evaluation and discussions. We thank interviewees of the I-Corps program for myriad and fruitful discussions about braille and life experience. We thank N. Runyan for a lifetime of devotion to tactile display innovation and equity for the blind. **Funding:** This work was supported by the DOE Advanced Research Project Agency-Energy (ARPA-E) DE-AR0001855 (to R.F.S.), Office of Naval Research (ONR) N00014-20-1-2438 (to R.F.S.), and National Science Foundation (NSF) TI-2231592 (to R.F.S.). **Author contributions:** Conceptualization: R.H.H. and R.F.S. Methodology: R.H.H., K.D.L, O.P., C.A.D-R., C.A.A, S.S., N.B., Y.S.K., and R.F.S. Investigation: R.H.H., K.D.L, O.P., C.A.D.-R., R.M.M., C.A.A, S.S., N.B., Y.S.K., and R.F.S. Visualization: R.H.H., K.D.L., and R.M.M. Funding acquisition: R.F.S. Supervision: S.S., N.B., and R.F.S. Writing—original draft: R.H.H., K.D.L., O.P., and R.F.S. Writing—review and editing: R.H.H., K.D.L, O.P., C.A.A., S.S., N.B., and R.F.S. **Competing interests:** R.H.H., C.A.A., and R.F.S. are listed authors on US11353046B2. All other authors declare that they have no competing interests. **Data and materials availability:** Pressure-displacement data and control code for the untethered display are available for download on Dryad (https://doi.org/10.5061/dryad.70rxwdc9s).

Submitted 29 October 2024
Resubmitted 2 June 2025
Accepted 31 July 2025
Published 27 August 2025
10.1126/scirobotics.adu2381

MOTION PLANNING AND CONTROL

RoboBallet: Planning for multirobot reaching with graph neural networks and reinforcement learning

Matthew Lai[1,2]*, Keegan Go[3], Zhibin Li[2], Torsten Kröger[3], Stefan Schaal[3], Kelsey Allen[1]†, Jonathan Scholz[1]†

Copyright © 2025 The Authors, some rights reserved; exclusive licensee American Association for the Advancement of Science. No claim to original U.S. Government Works

Modern robotic manufacturing requires collision-free coordination of multiple robots to complete numerous tasks in shared, obstacle-rich workspaces. Although individual tasks may be simple in isolation, automated joint task allocation, scheduling, and motion planning under spatiotemporal constraints remain computationally intractable for classical methods at real-world scales. Existing multiarm systems deployed in industry rely on human intuition and experience to design feasible trajectories manually in a labor-intensive process. To address this challenge, we propose a reinforcement learning (RL) framework to achieve automated task and motion planning, tested in an obstacle-rich environment with eight robots performing 40 reaching tasks in a shared workspace, where any robot can perform any task in any order. Our approach builds on a graph neural network (GNN) policy trained via RL on procedurally generated environments with diverse obstacle layouts, robot configurations, and task distributions. It uses a graph representation of scenes and a graph policy neural network trained through RL to generate trajectories of multiple robots, jointly solving the subproblems of task allocation, scheduling, and motion planning. Trained on large randomly generated task sets in simulation, our policy generalizes zero-shot to unseen settings with varying robot placements, obstacle geometries, and task poses. We further demonstrate that the high-speed capability of our solution enables its use in workcell layout optimization, improving solution times. The speed and scalability of our planner also open the door to capabilities such as fault-tolerant planning and online perception-based replanning, where rapid adaptation to dynamic task sets is required.

INTRODUCTION

In robotics applications such as welding (*1*), assembly (*2*), painting (*3*), construction (*4*), and object manipulation (*5*), robotic arms are programmed to perform many tasks in a shared workspace. To reduce the overall execution time and increase throughput, more robots can be placed in the same cell to perform tasks concurrently. Such high-density setups have the potential for reduced execution time with minimal increases in cost and space compared with duplicating entire workcells.

However, coordinating multiple robots efficiently in confined spaces is both theoretically and technically difficult because the planner must solve the task assignment, scheduling, and collision-free motion planning problems jointly, all of which have high combinatorial complexity, often in environments with complex obstacle geometries. This work addresses this challenge in a setting where multiple robots can perform all tasks in any order, which represents the most challenging scenario in terms of the size of the search space.

Finding a collision-free path in joint space that moves a robot end effector to a pose target is the motion planning aspect of the problem and has been conventionally addressed using sampling-based approaches such as rapidly exploring random trees (RRT) and its variants (*6*, *7*). However, although these sampling-based algorithms are designed to tackle high-dimensional configuration spaces and offer probabilistic completeness, their runtime scales exponentially in the dimensionality of the configuration space and the complexity of the obstacles (*8*). In practice, these techniques are suitable only for relatively small configuration spaces [two or fewer robots (*4*)] when attempting to plan with all robotics at the same time. One potential solution is to plan for each robot sequentially; however, this formulation introduces its own challenges because other robots would have to be treated as moving obstacles, which sampling-based motion planners also struggle with, and would require techniques such as converting paths of other robots into swept volumes, which then unduly limits the search space of the planner and might lead to suboptimal solutions or planning failures.

Similarly to the path planning problem, the task scheduling problem (to decide the order in which the tasks should be done) is also polynomial space (PSPACE) complete (*9*, *10*), formalized as the traveling salesman problem (TSP). Although exact solutions to the TSP are not computationally feasible for real-world problem sizes, many approximate solutions exist with varying optimality/efficiency trade-offs. The main challenge in adapting these solutions to task scheduling in our context is that, instead of having a fixed cost to move between each pair of tasks, as is the case in the classically formulated TSP, the joint-space trajectories required to move from task to task can be modified depending on the trajectories of other robots, and each potential trajectory has a different cost. Furthermore, there is often more than one inverse kinematics (IK) solution per task, either because the task is not fully constrained or the robot has more degrees of freedom (DoFs) than the task. In the case of seven-DoF robot arms, each six-DoF task pose has up to an infinite number of IK solutions. A good path planning algorithm must take the entire sequence of tasks into account when selecting the best IK solutions to use to solve each task.

The task allocation problem adds yet another layer of complexity. Finding an allocation of tasks to robots that minimizes the total execution time is conceptually similar to a multicontainer variant of the classical knapsack problem (*11*), except that we do not know the marginal cost of allocating an additional task to a robot without

[1]Google DeepMind, London, UK. [2]University College London, London, UK. [3]Intrinsic, Mountain View, CA, USA.
*Corresponding author. Email: matthewlai@google.com
†These authors contributed equally to this work.

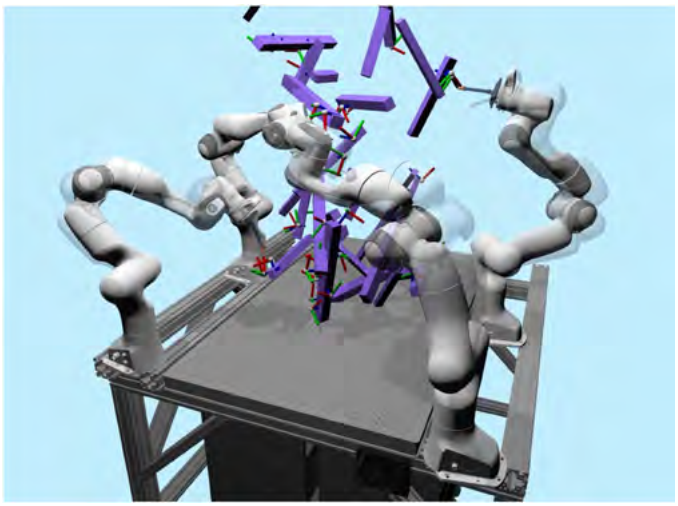
solving the entire planning problem, nor are the costs independent of other tasks being solved by the robots. Therefore, algorithmic solutions to the knapsack problem are not practically helpful in this setting.

As a result of the high asymptotic complexity of each of the subproblems, it has not been computationally feasible to combine existing algorithms with completeness guarantees to jointly solve high-density multirobot planning on real-world problem sizes because of their exponential scaling in the number of robots, number of tasks, and complexity of obstacles (*12*). Instead, existing approaches focus on breaking the problem down into manageable subproblems, solving them one at a time, often iteratively, to generate solutions with realistic compute requirements (*4*). However, this approach sacrifices potential solution optimality for computational feasibility and still does not scale to environmental complexities of real-world applications, given that the current state of the art reported scalability to only five robots and 10 tasks by iterating between a high-level task planner and a low-level motion planner, with some heuristics based on reachability and availability (*13*). By necessity, industry currently relies instead on manual trajectory planning with complex schedules of interlocks to ensure collision-free operation. These trajectories often take hundreds or thousands of hours to design by hand (*1*), and any change to the tasks or the environment requires time-consuming manual replanning.

To tackle this challenge, we present RoboBallet, a learned heuristics-based approach for a task and motion planning on randomly generated environments, capable of planning multiarm reaching trajectories in environments different from those seen during training, with arbitrary obstacle geometries, task poses, and robot placements. Task and motion planning is a large class of problems that require both high-level discrete scheduling and low-level continuous planning to solve. In this work, we focus on the reaching subclass for simplicity. Specifically, we address problems where scalability and efficient task scheduling is the main challenge, with no implicit or explicit task interdependencies and where each individual task is simple to accomplish. For extensions to manipulation tasks or problems with ordering/compatibility constraints, see the Limitations and future work section.

Our approach tackles the task allocation, scheduling, and motion planning problems jointly with no hand-designed simplifications common in existing approaches, such as presampling a fixed number of IK solutions, preallocating tasks to robots on the basis of heuristics, or planning subtrajectories for one robot at a time. Our key insight is to adopt a deep reinforcement learning (deep RL) approach that trains an agent to control the arms at each time step. This shifts expensive online computations to an offline phase, amortizing and shifting the compute requirement from the planning phase to the training phase, benefiting applications that require planning solutions on many similar setups or a setup with many variations. By training on configuration variations that we wish to support, for example, robot positioning and morphology, task types, or obstacle types, RoboBallet can quickly generate plans for a wide range of environments and novel configurations at test time without further training (Movie 1). After training, we can either simply execute these controllers in the real world, or, for safety, we can generate the solution trajectory by unrolling the policy step-by-step with a forward kinematic simulator so it can be inspected and potentially processed further.

However, naive applications of deep RL to multiarm task and motion planning would also fail to scale beyond trivial scenes, given



Movie 1. RoboBallet: Planning for multirobot reaching with GNNs and RL. We trained the model on environments with randomized robot placement, randomly generated obstacles, and randomly generated tasks. These randomized environments are generated per episode in training. Throughout the training process, the model interacts with about 1 million of these randomized environments. Each RGB (red, green, blue) coordinate frame on the obstacles represents an end-effector pose the robots need to reach. For testing, the evaluation setup used a fixed robot placement, real-world obstacles, and presampled tasks. Despite having never seen the evaluation robot placements, obstacles, or tasks during training, the model, trained on the randomized training environment, is able to transfer without any additional training to the evaluation cell and generates time-efficient and collision-free trajectories.

that the scene complexity still scales combinatorially and this leads to exponential scaling in the required size of the underlying deep neural network representing the agent's policy. The main contribution of this work is in the use of graphs to represent the state and the use of graph neural networks (GNNs) for RL policy and state-action value estimation. GNNs use weight sharing to allow scaling of the graph size without scaling the model complexity, similarly to how convolutional neural networks (CNNs) (*14*) have fixed kernel sizes independent of image sizes, and inference time complexity is quadratic in the number of robots—refer to the Discussion section for more details. The state graphs are constructed such that graph nodes represent robots, obstacles, and tasks and graph edges connect each task and each obstacle to each robot, and each robot to each other, supporting coordination among the robots and allowing each node to receive sufficient information about the environment to plan a collision-free path to tasks (Fig. 1).

With this formulation, RoboBallet directly and simultaneously controls multiple (here up to eight) seven-DoF robots, coordinates an up to 56-dimensional configuration space, and handles up to 40 shared tasks, with completely unconstrained robot allocation and task scheduling. The model decides at each 100-ms time step the target joint velocities of all joints in all of the robots, and our environment setup simply applies velocity and acceleration clamping, stops robots from colliding, and integrates the resulting velocities to generate joint configuration waypoints.

RoboBallet adopts an approach that prioritizes scalability and trades off theoretical guarantees to enable real-world deployment on a much larger scale of problems (test with up to eight robots in dense environments) while demonstrating empirically high-quality solutions. RoboBallet opens up exciting possibilities for many industrial

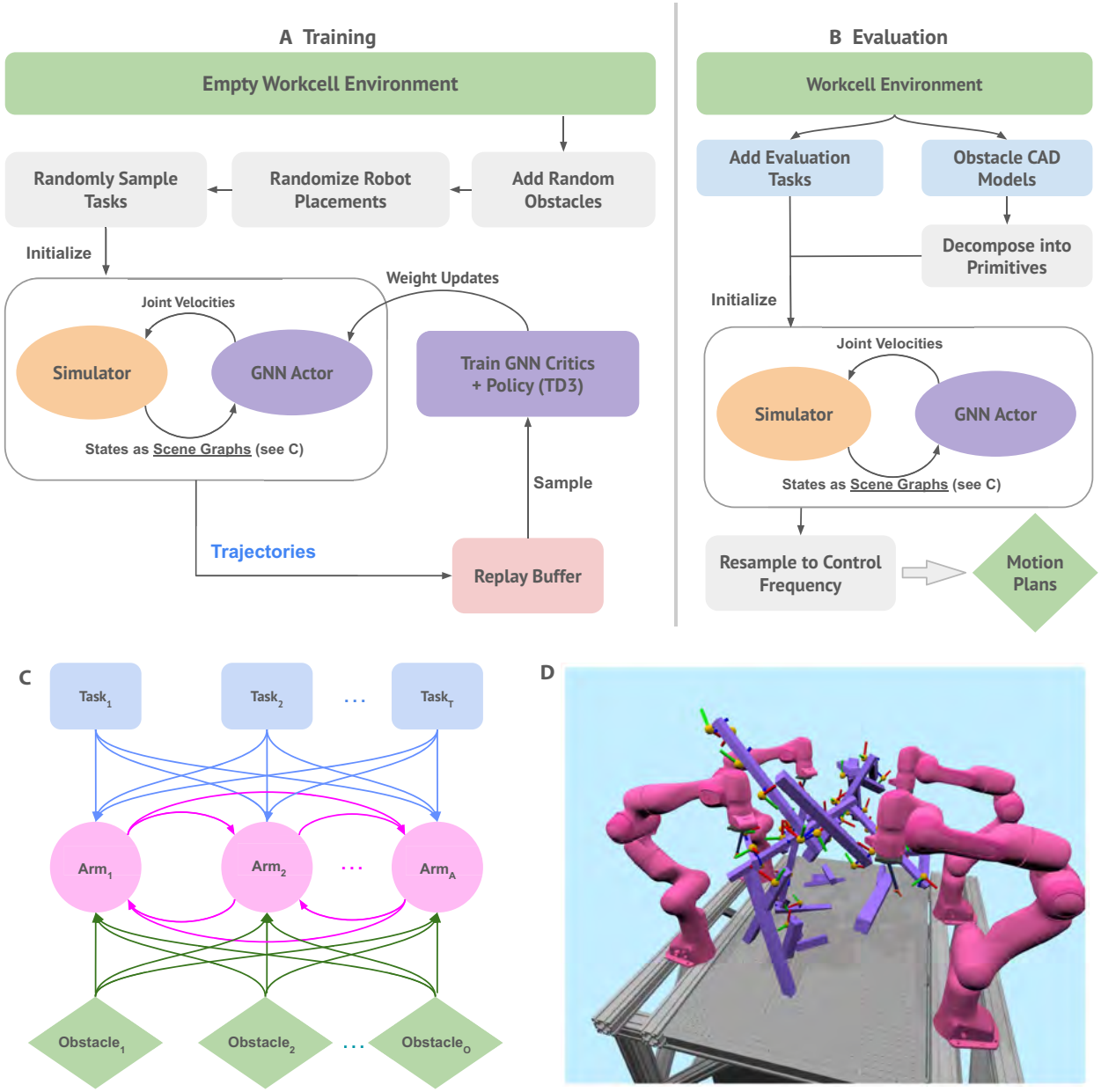


Fig. 1. Training and evaluation pipeline. (**A**) Training pipeline. Each episode starts with an empty table, upon which we add random cuboid obstacles, randomize robot placements, and randomly sample end-effector tasks. The GNN actor interacts with the kinematic simulator to try to complete the tasks. The trajectories are stored in a replay buffer and used to train a critic and policy function using TD3. (**B**) The evaluation pipeline, where externally defined tasks, together with obstacles, decomposed into collision primitives, are used to construct the simulated environment, with which the GNN interacts in the same way as in training. (**C**) Topographical view of the scene graph, containing robot, task, and obstacle nodes, with bidirectional edges between robots and unidirectional edges from each task and obstacle primitive to each robot. (**D**) Simplified training cell with a reduced number of obstacles and tasks. Each task is specified and visualized as a coordinate frame that the end effector of the robots must match, where the blue/*z* axis is in line with the radial axis of the end effector.

applications, such as workcell layout optimization, fast replanning in case one robot fails, and handling run-to-run variations in workpiece pose using image-based pose estimation to capture the workpiece pose for the planner, as well as environment changes using three-dimensional (3D) environment reconstruction with cameras or LIDAR (light detection and ranging) systems. RoboBallet is an enabling technology that opens the door to these important capabilities that have thus far not been possible because of the high cost and latency of manual task and motion planning in the multirobot manufacturing setting.

RESULTS

To demonstrate the capability of RoboBallet, we trained the model over several days on randomly generated workcells of different sizes in simulation and validated the performance of trained models in a

simulated workcell with eight robot arms, a workpiece constructed out of aluminum struts, and 40 tasks (Movie 1). To demonstrate the feasibility of real-world execution of our generated trajectories, we trained a model with four robot arms using exactly the same framework, and we validated the generated trajectories on a real workcell with minimal postprocessing. We observed that our models generated high-quality trajectories in seconds and can effectively use many robots to shorten execution time even on the largest settings tested (eight robots, 40 tasks, 30 obstacle primitives), on a single GPU (graphics processing unit), and in both a real and a simulated multiarm workcell.

To showcase one of the most important capabilities enabled by a fast automated planner, we used RoboBallet, combined with a black box optimizer, to perform workcell layout optimization (optimizing placement of robots given fixed sets of tasks and obstacles). We saw that this optimization setup improved task execution times by up to 33% compared with nonoptimized robot placements in our test cell on our fixed test set of tasks.

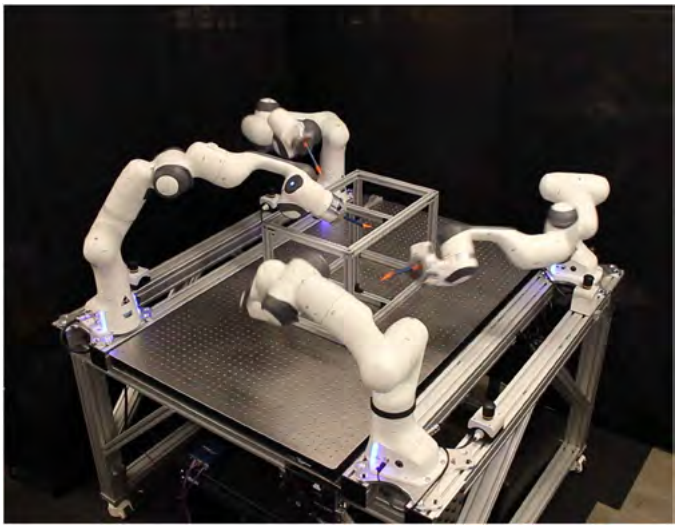
The training environments had either four or eight Franka Panda seven-DoF robots: four mounted on a table and an additional four mounted on the ceiling mirroring the four on the table in the case of eight robots (Movie 1). At the beginning of each episode, we first randomized the location of the robots along virtual 1.6-m-long rails parallel to the side of the table, only rejecting configurations in collision at the start. Then, we sampled and placed 30 cuboid obstacles with randomly generated positions and orientations, only rejecting obstacles that would be in collision with the robots in their start configurations. Next, we uniformly sampled tasks on the surfaces of the obstacles with a small offset away from the surfaces, where each task is a position and an orientation that a robot end effector must reach. We then tried to generate IK solutions from each robot to the task pose and rejected the task if it did not have at least one collision-free IK solution from at least one robot. Information about the IK solutions and reachability was not given to the model.

In the motion planning literature, the execution time (the durations of trajectories) is the primary metric to be optimized for. However, because no published approach exists that can scale to our problem sizes in the multiarm task and motion planning setting with realistic execution time, we provide a comparison against a baseline method for trajectory optimality in a simplified setup. This is given in the "Optimality of generated trajectories" section below. The rest of the Results section reports on the scalability and generalizability of the algorithm on the full problem. For a qualitative evaluation, please refer to Movie 2 and movies S1 to S4 showing generated trajectories.

Optimality of generated trajectories

In both real-world applications and academic benchmarks, trajectories are most commonly evaluated by their time cost (duration). In this section, we tested our approach on a set of smaller and simplified problems where we could generate near-optimal trajectory solutions through exhaustive search and preassignments of tasks to robots. The preassignment is necessary because exhaustively testing all possible allocations in addition to schedules is not computationally feasible for all but the smallest problems.

Specifically, we generated 10 sets of 20 tasks, where each set contains five tasks for each of the four robots. We filtered the sampled tasks to ensure that each can be reached by only one robot. To generate solutions that are as close as possible to optimal to compare against, we needed to tackle the same four challenges that our model needs to tackle simultaneously—task allocation, task scheduling, IK solution selection, and motion planning. Optimal task allocation is ensured by problem design (only one allocation is possible). Optimal task scheduling is done by exhaustively testing all possible task schedules. In the case of seven-DoF robot arms, each six-DoF task pose has up to an infinite number of IK solutions when poses are within the feasible reachability workspace, and choosing good IK solutions in the context of each trajectory is also important in minimizing total trajectory cost. We present multiple minimum costs based on different numbers of IK solutions sampled (and exhaustively tested) to compare against. Last, because a computationally feasible optimal motion planning algorithm for seven DoFs does not exist, we used RRT-Connect (*15*) with shortcutting, which is a widely used motion planning algorithm known for generating low-cost trajectories in practice. We first planned for each robot separately and then, for each task set, took the cost of the longest single-robot subtrajectory as the cost of the multirobot trajectory.



Movie 2. Four Franka Panda robots executing a plan generated by RoboBallet.

In Fig. 2, we first showed that, with a random task schedule for each robot, the trajectory costs were much higher than with optimal task scheduling, demonstrating the importance of good task scheduling even in this simplified setting. Then, we showed that, by increasing the number of IK solutions sampled (and selecting the best IK solution for each task in each trajectory), the trajectory costs also decreased substantially, demonstrating the importance of choosing good IK solutions. Last, we showed that RoboBallet generated trajectories with costs that are competitive with the baseline approach at eight IK samples per task. These results showed that, although an RL-based approach cannot offer the probabilistic completeness that sampling-based methods do, the costs of trajectories generated in our problem settings are comparable to our baseline results in practice. The baseline also uses orders of magnitude more computation and relies on several simplifying assumptions, unlike RoboBallet.

There are two notable differences between the baseline and RoboBallet setups that need to be taken into account when interpreting these results. First, the RRT-Connect algorithm builds a tree from both the start and the finish poses, generating a valid path that puts the end effector at exactly the goal pose, whereas our approach uses

a tolerance of 2.5 cm and 15°. In a real-world application, our approach would integrate a linear driver to achieve final docking. This would be implemented as part of the environment, where the RL agent only receives a reward if the final docking is successful, ensuring that the agent chooses solution poses where the final docking is feasible. This integration is excluded from the current study for simplicity, and its inclusion would result in a marginal increase in trajectory costs for RoboBallet. Second, the RRT-Connect results are generated with each robot in isolation (other robots removed), whereas RoboBallet moves all robots at the same time. This means that the RRT-Connect results here ignore potential collisions between robots during task execution. If collisions did occur in practice when all robot trajectories are executed simultaneously, this would require adjustment of task sequences to create a collision-free multirobot trajectory.

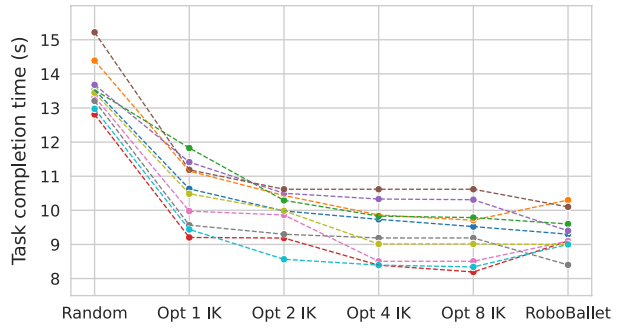


Fig. 2. Comparison against RRT-Connect plus exhaustive task scheduling baseline in a simplified setting. Each color represents one task set with 20 tasks (five per robot). All columns except "RoboBallet" use RRT-Connect and shortcutting for motion planning. The "Random" column averages over all possible task schedules. The "Opt [*N*] IK" columns use *N* IK solutions per task and the optimal task schedule and IK solution selection by exhaustive search.

Training time scaling in the number of robots and tasks

As discussed in the Introduction, classical approaches scale exponentially in the number of robots and number of tasks, and this prevents them from being applicable to large settings. In Fig. 3, we demonstrated that our approach scales much more favorably at training time. Quadrupling the number of tasks only required marginally more training steps to converge, despite the 4× difference in the number of tasks and therefore in the difficulty of the scheduling problem. This does not correspond to equal amounts of computation. The number of edges in our scene graph is linear in the number of tasks, and therefore, each training step takes up to four times more computation in the 40-task case compared with the 10-task case.

The second important aspect of computational scalability pertains to the number of robots. Similarly to the scaling with number of tasks, we saw that the training curves converged at a similar number of training steps. The theoretical asymptotic time complexity is $O(N_{robots}^2)$ for N_{robots} number of robots, given that the robot nodes are fully connected with each other for coordination. In practice, quadratic scaling here is not a problem because there is an inherent limit in the density of placement of robots, beyond which there would be no benefit in execution time given that it becomes more

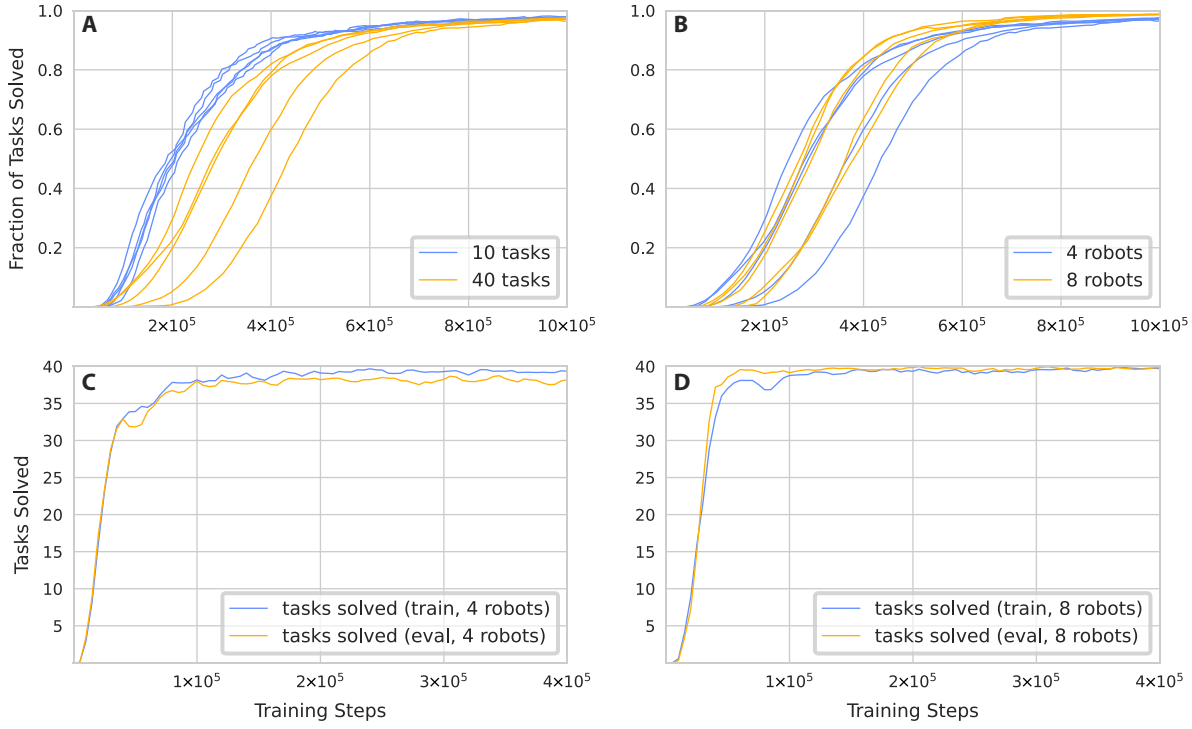


Fig. 3. Training curves for scalability and generalizability. (**A**) Training curves for 10 versus 40 tasks, four robots. (**B**) Training curves for four versus eight robots, 40 tasks. The number of training steps required to reach convergence does not scale exponentially. However, in terms of asymptotic time complexity, the amount of computation required is linear in the number of tasks and quadratic in the number of robots. (**C**) Results on randomly generated training cells versus hand-designed evaluation cell over training, four robots. (**D**) Results with eight robots.

and more difficult for the robots to avoid one another. For very large workcells with more than a dozen robots, the robots would be positioned such that they would not all have overlapping work volumes. If two robots do not have overlapping work volumes, then they do not need to coordinate with each other because they cannot complete the same tasks or collide with each other. Therefore, in the asymptotic case, assuming that we have a limiting density in placement of robots, and add more robots by expanding the workcell without connecting robots that do not have overlapping work volumes, the algorithm's asymptotic time complexity falls back to $O(N_{robots})$.

In practice, because of the performance characteristics of GPUs, the runtime differences are even smaller than suggested by the asymptotic complexity. In our case, each training step took 33 ms for the four-robot–and–10-task case, 52 ms for the four-robot–and–40-task case, and 80-ms for the eight-robot–and–40-task case, with a batch size of 128 state transitions per training step. Figure 3 shows the training curves with three different configurations—four robots with 10 versus 40 tasks and 40 tasks with eight robots.

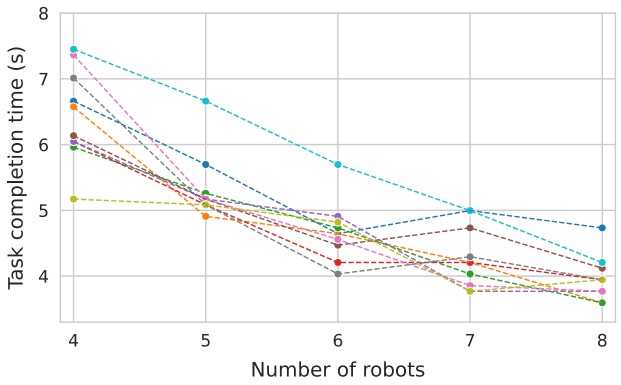


Fig. 4. Task completion time scaling with the number of robots (optimized placements). Shortest trajectory execution times achieved with the 10 evaluation configurations using optimized placements of four to eight robots. Each color represents one evaluation configuration.

Inference time scaling

At inference time, even the largest setup with eight robots and 40 tasks only took about 0.3 ms per planning step on an NVIDIA A100 GPU, allowing for greater than 300× real-time planning at 10-Hz time steps. On a single Intel Cascade Lake CPU core, each step takes about 30 ms, which is still three times faster than real time with 10-Hz time steps. The planning process consists of one inference and one collision check for the entire scene at each time step.

Efficient use of multiple robots

Another important aspect of scalability is the reduction in task completion time as the number of robots increases. The ability to control more robots is only meaningful if the controller is able to solve tasks more quickly when given more robots to use. For this experiment, we first randomly sampled 10 sets of robot placements, each with four robots—two on either side of the table and two on either rail on the ceiling. For each set of placements, we sampled a set of 20 tasks that can be solved using those four robots in those randomly sampled placements. These sets of tasks made up the evaluation configurations, each of which is a task set that is guaranteed to be solvable using four robots. Then, we performed experiments to find the shortest trajectory lengths we can generate for each evaluation configuration using four, five, six, seven, and eight robots. For each number of robots and evaluation task set, we performed placement optimization (see the "Workcell layout optimization" section) to find the best placements for the given number of robots and set of tasks and recorded the minimum trajectory length achieved. Figure 4 shows the best execution time achieved using four to eight robots with optimized placements for each evaluation configuration. The plot shows strong execution time scaling from four to eight robots, with the average execution time reducing from ~7.5 s using four robots to 4.5 s using eight robots (a reduction of 60%). This is approximately optimal scaling, given that in the four-robot case, with uniform task distribution, each robot needed to perform five tasks, whereas in the eight-robot case, four of the robots needed to perform two tasks and the other four needed to perform three tasks.

Generalizability to hand-designed environments

Our models were trained with randomized robot placements, randomized obstacles, and randomized tasks to enable maximum flexibility at evaluation time. Movie 1 shows the evaluation setup, where the obstacle is a hand-designed shape made out of aluminum struts, robot placements are fixed to the locations in our real test workcell, and the tasks are randomly sampled on the faces of the obstacle. We generated 10 sets of 40 tasks for evaluation, which were not used in training. Figure 3 (C and D) shows performance on these evaluation configurations compared with that in the training environment over the training process. Our results show that the model, trained on randomly generated environments, can generalize very well to an unseen hand-designed evaluation cell.

Workcell layout optimization

A fast task and motion planner enables use cases that are not currently possible with manual planning. One such use case is workcell layout optimization. In this experiment, we trained the model on fully randomized environments as described above and used the Google Vizier (*16*) black box optimizer with a modified Gaussian process bandit algorithm (*17*) to find the best placement of robots to achieve the shortest execution time on each of the 10 sets of fixed evaluation tasks. Figure 5 shows the execution time improvements from the optimized placements compared with the common default placements with one robot at each corner of the table, as well as the breakdown of time spent on each task by each robot, both with motion plans generated using RoboBallet. We saw that there is universal improvement in trajectory execution time over all evaluation configurations and that the improvement comes primarily from the more even redistribution of tasks to robots and secondarily from reduced motion time to reach the task poses. In setups where the task completion time is long relative to the motion time, and therefore the even distribution of tasks is more crucial, the benefit of placement optimization would be even more pronounced. We chose to use a dwell time of 0.5 s (simulating task completion time) at each task here to show the result on a more balanced setup where both motion time and task completion time contribute substantially to the total solution length.

DISCUSSION

Our results demonstrate that GNNs combined with RL enable efficient solutions to the joint problem of task allocation, scheduling,

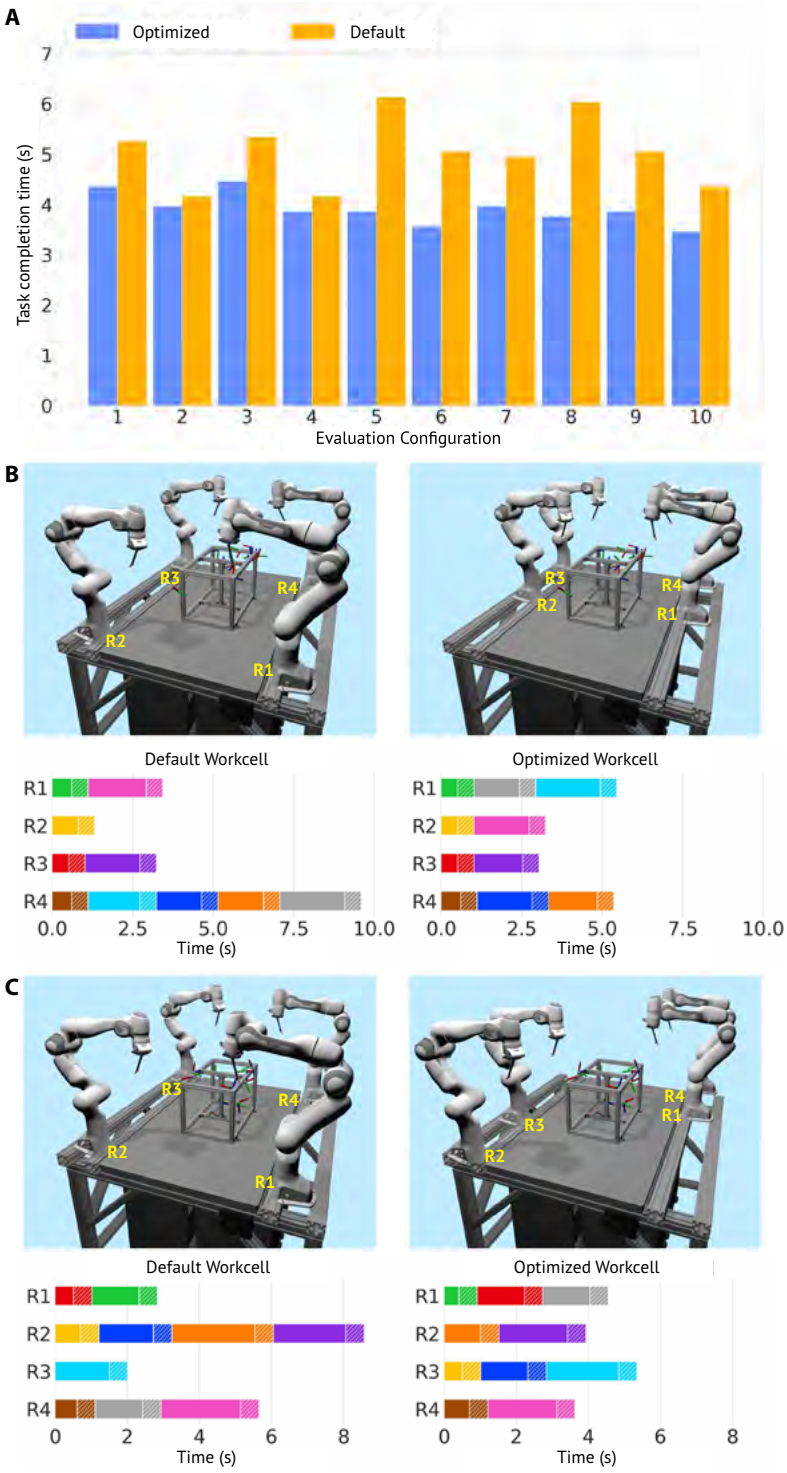


Fig. 5. Layout optimization of the workcell. (**A**) Execution time comparison between optimized and default layouts on the 10 evaluation configurations (each with 10 tasks). (**B**) Execution time breakdown of configuration #5, the configuration with the biggest improvements. Each bar represents the tasks done by a robot in the trajectory, and each block represents the motion time and dwell time (simulating time needed to perform work at the site, e.g., a spot welding task). Each task has a unique color so that the redistribution can be observed directly. (**C**) Same for configuration #8.

and motion planning. Although the conventional sampling-based approaches offer probabilistic completeness, a valuable property in certain contexts, our approach is able to scale to much larger real-world problem sizes, delivering high-quality solutions with minimal computation requirements, especially at evaluation time. This trade-off between theoretical completeness and scalability is critical for practical applications in large-scale, dynamic environments.

The high scalability of our method comes from the fact that in terms of model complexity of our learned functions, the network only has to learn the interactions between different types of nodes (tasks to robots, obstacles to robots, and robots to robots) once. This results in O(1) scaling in model complexity (model size) with respect to the number of robots, number of tasks, and complexity of obstacles. This means that, theoretically, the amount of training data needed does not scale with the size of the environment, and we observed that to be broadly true in our experiments. In terms of asymptotic time complexity, computation is dominated by edge processing, which is proportional to the number of edges. We have $N_{\text{robot}} \times N_{\text{robot}}$ robot-to-robot edges, $N_{\text{robot}} \times N_{\text{robot}}$ task-to-robot edges, and $N_{\text{robot}} \times N_{\text{obstacle}}$ robot-to-obstacle edges, for a total time complexity of $O(N_{\text{robot}}^2 + N_{\text{robot}}N_{\text{task}} + N_{\text{robot}}N_{\text{obstacle}})$. This complexity is linear in both number of tasks and number of obstacles and quadratic in the number of robots, which is manageable because of the inherent limit in how many robots can be usefully placed in the same space.

A fast and automated task and motion planner enables many use cases that are previously not known to be practical because of the speed of existing planners. As an example that we explored in this work, we used the planner to design the layout of new robot workcells by running a black box optimizer on top of the planner to find the best robot placement for each set of tasks. The cost of robots and floor space is one of the primary limiting factors in many manufacturing setups, and our results show that we are able to substantially reduce the execution time of each of our evaluation configurations by performing layout optimization. With our RL setup, the reward function can easily be modified to incorporate different optimization objectives with different levels of importance. As an example, for our demonstrations on real robots, we trained the model with an additional reward term for returning to the start position at the end of the episode using the Euclidean (L2) norm in configuration space from the start configuration, only activated once all tasks are solved. We also used a weak penalty term based on squared joint space acceleration to discourage superfluous movements and unnecessarily extreme accelerations. Robotic applications often have unique requirements, and with an RL formulation, we only need to design a cost function to reflect them, which is often much easier than designing behaviors, making RL-based systems more easily adaptable to different requirements compared with both hand-coded systems and supervised learning–based systems learning from either an algorithmic planner or human demonstrations.

As another example use case, in situations where assembly line halts due to robot failures are not acceptable, the only current strategy in the industry is to create fail-safe motion plans with each robot removed, which is extremely costly in the manual labor required to design these plans and not scalable to more than one robot failure per cell. With our approach, the trained model can either generate the backup trajectories ahead of time, or as the failures happen, with minimal downtime. In case the remaining robots cannot continue to cover all of the tasks, but at least some of the robots can be moved (e.g., along linear rails), the model can also be used, with layout optimization, to quickly determine the optimal new placements for the remaining robots.

In terms of real-world execution of the generated trajectories, although the models already respect joint velocity and acceleration constraints, it would be prudent to perform postprocessing on the generated trajectory for smoothing and for compliance with more complex constraints such as jerk and torque. Movie 2 shows trajectories generated by the model postprocessed by fifth-order spline interpolation to 1 kHz running on Franka Panda robots. In this case, the network has also been trained to respect a 1.0-cm collision margin for additional safety.

High-density robotic task and motion planning for reaching is a problem that expert humans can manually solve at great time investment and manpower costs, using human experience and intuition to navigate the huge combinatorial search space. However, this problem has proven to be very challenging for automated algorithmic solutions. Machine learning provides methods to imitate human intuition that is hard to express as hand-coded heuristics. However, a naive machine learning–based solution would suffer the same fate of combinatorially increasing cost if the model architecture does not take into account the relational inductive biases that can be exploited to reduce the required learning efforts and improve generalization. Our combination of a structured state representation and a model learning approach that exploits the structure mirrors how expert humans solve this problem using their intuitive understanding of the construction of the problem.

The physical world is inherently structured, and one of the most intuitive computational representations of those structures is as graphs. In our case, we chose to represent the scene as a graph made out of robot, task, and obstacle nodes, exploiting the fact that each type of interaction (e.g., driving a robot to a task or avoiding an obstacle) only needs to be learned once, regardless of the complexity of the scene. This gives us better data efficiency and fewer parameters to learn compared with naive network architectures that do not exploit these inductive biases. The fact that we can learn from every interaction of the same type (e.g., robot to task) to train one fixed-size function turns the challenge of combinatorial complexity into an advantage and gives us the ability to efficiently train small and high-quality models that do not overfit to the training data.

Limitations and future work

In practical applications, the key challenge we identified in task and motion planning for dense robotics is to achieve scalability, and it is the main factor preventing existing solutions from being applicable to real-world applications. Our approach is demonstrated in a general reaching setting and a formulation where any robot can complete any task in any order within a shared workspace. To specifically focus on the scalability challenge, the configuration in this paper represents the largest search space, allowing us to tackle the fundamental scalability issues before introducing additional constraints. In practical/industrial applications, there are often constraints, such as intertask dependencies and heterogeneous robot capabilities, that are not considered in this work. Our approach could be extended to handle these constraints, for example, by disabling edges between incompatible robots and tasks or adding edges between tasks to signal dependencies. The resulting problem is theoretically simpler to solve than the fully unconstrained problem, but this is to be demonstrated in future work. Similarly, on the motion

planning level, we chose the abstract setting where the tasks are fully constrained end-effector poses, which makes motion planning the most constrained and difficult. In many settings such as spot welding or painting, the task poses are only semiconstrained, e.g., some rotational constraints of the end effector in spot welding are soft constraints. We believe that this can be accomplished by augmentation of the reward function, and that is also left to be demonstrated in future work.

The favorable scaling properties demonstrated in this paper are sufficient to enable many applications and industrial use cases, but if scalability to even larger settings is required, it would be computationally more efficient to apply some heuristics to prune graph connectivity. For example, a preprocessing step can determine reachability between all robot and task pairs, and edges are only added where a robot can reach a task. This takes $O(N_{robot}N_{task})$ time and would not increase the time complexity of the overall algorithm. Similarly, where a robot's work volume does not overlap with an obstacle, the robot can never collide with the obstacle, and the pair would not need a connecting edge. We did not implement these simple optimizations because we wanted to explore the limit of machine learning and keep the algorithm as simple and generally applicable as possible, but in a real-world application, it would be prudent to include them. A graph attention mechanism (*18*) may also improve training in very large cells to automatically reduce the relevant pairs of entities the network has to consider.

For simplicity, we chose a minimal representation of obstacles—cuboids. Although this representation works well for our evaluation environments, it may not be suitable for environments where most obstacles cannot be decomposed into a reasonable number of cuboid shapes. In that case, and especially if the robots must get very close to the obstacles (e.g., for spot welding), it may be beneficial to add more primitive types and pick the best primitive type for each decomposed obstacle component. This approach would require an adjustment in training to randomly place obstacles of all primitive types. Another possibility is to support arbitrary polyhedrons by representing them as graphs themselves and using another GNN to summarize their properties into fixed-length embeddings to be used in the main neural networks. The embedding and policy/value networks can be trained end-to-end to ensure that the embeddings capture useful information for motion planning.

To illustrate the core capabilities of our approach, we focus on reaching tasks. However, the algorithm can be readily extended to more complex operations, such as pick-and-place tasks. One possible formulation is to factor pick-and-place tasks into separate picking and placing task poses, adding and removing task dependencies and compatibility as required using edge features in the state graph, and have the simulator attach and detach objects from end effectors where necessary. In this formulation, a separate controller or a grasp pose predictor is required to generate feasible pose targets for picking up arbitrary objects, both at training time and at inference time. Extension to manipulation tasks presents further challenges, including object collisions, robot-environment constraints, and occlusion handling in multirobot coordination, requiring future work on both algorithmic development and experimental validation.

Going beyond existing capabilities into potential future applications, although our method is more than fast enough at inference time for online planning, online planning comes with other challenges such as perception and robot trajectory tracking accuracy without postprocessing. Solving these issues, together with our fast planner, would enable industrial capabilities that are not currently available and would be a valuable extension to this work.

As an extension to our workcell layout optimization example, using RoboBallet combined with a black box optimizer as the outer loop, a designer can use it to quickly find the optimal number, types, and positioning of robots in a workcell. For example, they may need to make a decision between using a six-DoF robot mounted on a linear rail versus a seven-DoF robot or maybe a heterogeneous team of robots. Here, we present a simplified version where only the placements are optimized. If the choice of robots is also to be optimized, the model would be trained with robot morphology randomization at training time, in addition to the other axes of randomization, so that the resulting model would be compatible with multiple robot morphologies.

Last, bimanual manipulation is the most natural way for humans to interact with the environment, and it enables capabilities that are out of reach of single manipulators. One of the main challenges in multiarm manipulation is the combinatorial complexity in performing motion planning in such a high-dimensional action space. Our approach shows that this scalability can be achieved using a structured network architecture, which can provide more insights and pave the way for developing future technologies in bimanual and multimanual task settings, extending the nature of multiarm coordination from collision avoidance, which is the focus of our work, to cooperative task completion in many other domains of robot manipulation.

In this work, we present a method for multirobot task and motion planning (demonstrated on but not fundamentally restricted to reaching) that is able to perform automated planning on industrial-scale workcells, a task that is currently performed manually at great cost and great latency. We showed that a fast planner like ours not only allows for acceleration of this process but also makes previously unknown capabilities such as workcell optimization and online planning feasible.

MATERIALS AND METHODS

Our goal is to train a neural network–based controller that, given the states of all robots, tasks, and obstacle primitives, will generate a set of joint velocities for each robot such that, when followed over time, produces the most efficient trajectory to solve all of the tasks. To that end, we designed an RL formulation, where the observations are represented as graphs in which nodes represent robots, tasks, and obstacle primitives and edges represent the relationships between them (e.g., relative pose). The trained policy network is a GNN (*19*) that takes the observation graphs as input and produces commanded joint velocities for all robots at each time step. This in effect allows the agent to reason relationally and combinatorially while only taking in the raw state as inputs. The joint velocities are integrated to produce next robot configurations, except where the commanded velocities would cause a collision, in which case the joint velocities for the robot are zeroed. The agent is rewarded for solving tasks and avoiding collisions. We used a slightly modified asynchronous version of the TD3 (*20*) RL algorithm to train the policy network and two auxiliary discounted state-action return prediction Q networks, also known as critics. The overall training and evaluation setup is shown in Fig. 1.

RL setup

Environment

The environment is implemented using a kinematic simulator. Each action applied advances the simulation by 0.1 s, and each action consists of desired joint velocities for all joints of all robots. We apply acceleration and velocity limiting on the basis of robot kinematic limits. A task is considered solved if the end effector of a robot moves to within a Euclidean distance of 2.5 cm and an angular distance of 15° of the task pose. Although this may seem coarse, there is a trade-off between tolerance and learning speed, and in practice, 2.5 cm is close enough to generate good plans that can be postprocessed to arbitrary precision. When a task is solved, the solving robot is frozen for 0.5 s of dwell time to simulate performing some work at the site and to encourage even task distribution between robot arms. If the joint velocities generated for any of the robots cause it to go into collision with itself, another robot, or the environment, the joint velocities for the robot in question are zeroed, and thus the robot stands still for that time step. Therefore, the environment does not allow for collisions. This is a desirable property for transfer to real-world setups, where collisions must be avoided at all costs. We use a collision penalty to discourage the policy from generating joint velocities that would cause collisions (see the "Reward" section below for more details).

Observation

We represent environment states as graphs, with robots, tasks, and obstacle primitives as nodes. The primary benefit of using this structured representation rather than simply flattening the state information into a vector is that it provides a way to incorporate relational information between entities in a reusable way and allows the network to reason about them relationally. For example, the model only has to learn to reason about relative pose between a robot and a task once, and it automatically transfers via the graph to all combinations of robots and tasks. By contrast, conventional architectures would be forced to discover these relationships for every combination of entities, leading to exponential compute and memory scaling as well as poor data efficiency.

In GNNs, edges can be directional—we used bidirectional edges between robot nodes to allow for collision avoidance and coordination, whereas unidirectional edges from the task nodes and obstacle nodes to robot nodes give the necessary information to robot nodes for motion planning. The directionality of edges determines how information flows during the graph inference step, and our choice was designed to propagate information to the robot nodes to allow reading out of the actions for each robot.

Obstacle representation

Our goal was to train a generalized planner that can plan in environments it has not seen during training and support obstacles with arbitrary geometries. That requires breaking obstacles down to primitives, where each primitive can be added to the graph as a node. During training, we generated cuboid obstacles of random sizes, positions, and orientations to add to the graphs as obstacle nodes, filtered only to exclude collisions between them and the robots in their initial poses.

At evaluation time, with externally defined meshes as obstacles, the obstacle meshes are decomposed into approximate convex hulls, and the oriented bounding boxes of those convex hulls are used as obstacle nodes (see Fig. 6). The approximate convex decomposition is performed using the V-HACD library (*21*), whereas the oriented bounding box search uses our implementation of the algorithm described in (*22*), which consists of a combination of genetic and Nelder-Mead (*23*) algorithms to efficiently find the best rotation for minimum-volume bounding boxes through unconstrained optimization on the 3D rotation group $SO(3)$. Because each cuboid has four equivalent representations that can be created by exchanging axes and making the corresponding change in pose, we randomly selected between the representations at each time step as a simple augmentation.

Task sampling

At training time, the tasks (desired end-effector poses) were sampled uniformly in position from the surfaces of the obstacle cuboids and in orientation randomly such that the desired orientation corresponding to the end-effector direction was maximally 22.5° from the normal direction of the face (that is, to complete the task, the end effector pointed approximately into the obstacle). At evaluation time, the tasks were generated in the same way on the externally defined obstacle geometry. We checked that each task pose had at least one collision-free IK solution from one robot. For efficiency, we were not verifying that there was a collision-free path from the start position to the task pose, only that a collision-free IK solution existed. Therefore, some of the tasks may not have had a valid path solution. The IK solutions were not stored, and no information from this task-filtering process was used in the training process.

Node features

The environment observations are in the form of graphs, where robot and task nodes have different features, whereas obstacle nodes have no node features. The node features for different types of nodes are then concatenated in a block-diagonal fashion to create the combined node features for the scene graph $\mathbf{G}_n$.

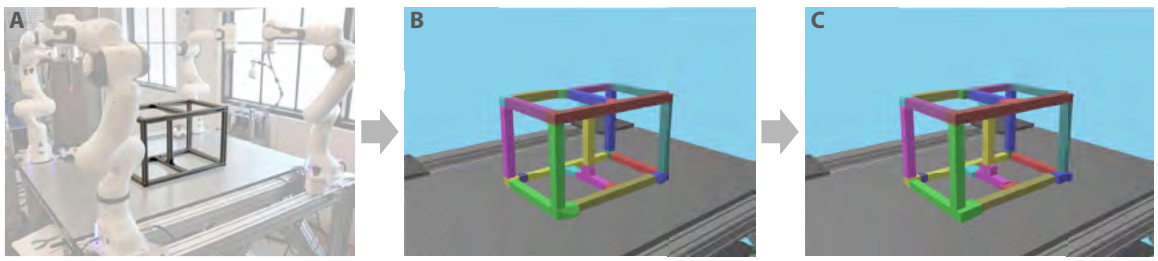


Fig. 6. Obstacle mesh decomposition pipeline. (**A**) Starting from the computer-aided design (CAD) model of the obstacle. (**B**) Result of decomposition into convex polyhedrons using V-HACD. (**C**) Oriented bounding boxes computed from the convex polyhedrons using (*22*).

For each robot, the robot node consists of the joint configuration for each of the seven DoFs of the robot, joint velocity for each DoF, and dwell time remaining, which is nonzero if the robot has reached a task pose and is executing the task (we froze the robot in the task pose for 0.5 s of dwell time to simulate executing some task at the site). Each task node consists of a single feature, task status, which records whether a task has been solved.

Edge features

Edge features encode relative information between each pair of nodes where an edge exists. Similarly to the node features, we have different edge features for each edge type, concatenated in a block-diagonal fashion to create the combined edge features for the scene graph $\mathbf{G}_e$. In a graph net, edges are directional and represent direction of information flow, and we used "sender" and "receiver" to denote the vertices of an edge relative to the direction of information flow. All poses were encoded using a 3D translation and a 6D representation of the rotation (*24*).

The robot → robot edges encode the sending robot base pose relative to the receiving robot tip pose. The task → robot edges similarly encode the task end-effector pose relative to the receiving robot tip pose. Last, the obstacle → robot edges have the following features: centroid translation of the obstacle primitive relative to the receiving robot tip pose and rotation matrix of the obstacle primitive relative to the receiving robot tip pose, with each basis vector scaled by the corresponding span of the primitive in that direction.

Global features

The global features $\mathbf{G}_n$ of the graph consist of the following: episode time normalized to timeout, the episode time at which we terminated the episode if the tasks had not been completed, and the current score as defined in the "Reward" section.

Reward

To calculate the reward for a step, we first calculated a score $S(\mathbf{s})$ for the state $\mathbf{s}$, which is simply the fraction of tasks N_{total} that have already been solved N_{done} (Eq. 1)

$$S(\mathbf{s}) = \frac{N_{\text{done}}(\mathbf{s})}{N_{\text{total}}(\mathbf{s})} \tag{1}$$

The reward R consists of two components. The first component R_{score} is the difference in score between the new state (state $\mathbf{s}$ with action $\mathbf{a}$ applied) and the previous state (Eq. 2)

$$R_{\text{score}}(\mathbf{s}, \mathbf{a}) = S(\mathbf{s}+\mathbf{a}) - S(\mathbf{s}) \tag{2}$$

The second component $R_{\text{collision}}$ is a penalty for actions that would have put the robots in collision (if it was not stopped by the environment). This is the number of robots that would be put in collision $N_{\text{robots_colliding}}$, scaled by a coefficient C_{col} (Eq. 3). That said, in theory, we do not need to add an additional penalty for collisions because the increase in execution time, combined with the discounting, would encourage the agent not to command joint velocities that would cause collisions, which are not optimal. However, in practice, adding an explicit penalty for collisions is effective to both reduce collisions and improve solution execution time

$$R_{\text{collision}}(\mathbf{s}, \mathbf{a}) = C_{\text{col}} N_{\text{robots_colliding}}(\mathbf{s}, \mathbf{a}) \tag{3}$$

The total reward is the sum of the two components (Eq. 4)

$$R(\mathbf{s}, \mathbf{a}) = R_{\text{score}}(\mathbf{s}, \mathbf{a}) + R_{\text{collision}}(\mathbf{s}, \mathbf{a}) \tag{4}$$

Hindsight experience replay

Given that the reward as specified above is sparse, it is unlikely that an RL agent early on during the training process would be able to find good rewards through random exploration. The conventional solution to this problem is to engineer a shaped reward by giving partial credit when the agent moves robots closer to completing tasks. However, designing a good shaped reward that encourages fast and reliable learning is not trivial and is also specific to the type of task being performed. Hindsight experience replay (HER), as proposed by Andrychowicz *et al.* (*25*), provides another solution to the sparse reward problem. With HER, instead of providing partial credits to aid learning when the agent is not yet reaching tasks, with each failed episode, we also generated a corresponding imaginary episode where we moved a randomly selected subset of tasks to randomly selected points of the robot trajectories within the episode, therefore providing positive training episodes that, in essence, show the learner that it would have been a successful episode if the tasks were actually along the robot trajectories the agent generated. This provides rich training signals with a healthy mix of successes and failures for efficient learning. Refer to Andrychowicz *et al.* (*25*) for more details on the general concept of HER. In our experiments, we found that HER performs approximately as well as even a well-tuned shaped reward, so we relied entirely on HER for the experiments above for simplicity.

Modified TD3

Twin-delayed deep deterministic policy gradient (TD3) (*20*) is an RL algorithm based on the deterministic policy gradient algorithm (*26*) in an actor-critic framework, where two networks are trained simultaneously—a policy network $\pi(\mathbf{s})$ that predicts the best action to take from a state and a critic network $Q(\mathbf{s},\mathbf{a})$ that predicts the discounted total future return (Q value) given a state and an action taken in that state. The policy network is trained to produce actions that result in a high Q value

$$\pi(\mathbf{s}) \Rightarrow \text{argmax}_a Q(\mathbf{s}, \mathbf{a}) \tag{5}$$

and the critic network is trained to predict the sum of the immediate reward from applying action $\mathbf{a}$ in state $\mathbf{s}$ and the discounted Q value from the resulting state $\mathbf{s}'$ with the best action in that state. Because we do not know the best action in state $\mathbf{s}'$, we use the output of the policy network $\pi(\mathbf{s})$ as an approximation

$$Q(\mathbf{s}, \mathbf{a}) \Rightarrow r(\mathbf{s}, \mathbf{a}) + \lambda Q\left(\mathbf{s}', \pi\left(\mathbf{s}'\right)\right) \tag{6}$$

where $(\mathbf{s}, \mathbf{a}, r, \mathbf{s}')$ is a transition from applying action $\mathbf{a}$ to state $\mathbf{s}$, arriving at state $\mathbf{s}'$ with reward r, and λ is the discount factor.

Deterministic policy gradient algorithms are a newer class of RL algorithms where the policy network predicts an action directly as opposed to a distribution over actions that would be predicted by policy networks in earlier stochastic policy algorithms. They are more efficient to train but are prone to get stuck in local minima. TD3 is the current state of the art of a lineage of deterministic policy gradient algorithms that each added some tweaks to stabilize training. We refer the reader to Fujimoto *et al.* (*20*) for a thorough description of the algorithm.

In our implementation of TD3, we removed the delayed target network updates for simplicity because, similarly to what the authors reported, we found the stability contribution of the delayed updates to be minimal, and the benefit is mainly a computational one. However, because our network architecture is much more

complex, the target network updates do not constitute a notable part of the compute requirement.

We also decoupled critic and policy Polyak averaging (*27*) constants so they can be tuned independently. On the actor side, we introduced exponential decay in exploration variance to encourage strong exploration in the beginning and more accurate task solutions later on in the training process. Similarly, we introduced two exponential decays—λ_{η_π} and λ_{η_Q} for the policy and Q function learning rates, respectively.

Network architecture

The system consists of three learned functions—two Q value estimators and a policy function. All three functions use a GNN at their core, with the same architecture, but initialized and trained independently without weight sharing.

GNN core

The core GNN takes as input graph G with nodes N with features matrix $\mathbf{G}_n$, edges E with features matrix $\mathbf{G}_e$, and graph-global feature vector $\mathbf{G}_g$ and produces an output graph $\mathbf{G}'$ similarly with node, edge, and global features $\mathbf{G}'_n$, $\mathbf{G}'_e$, and $\mathbf{G}'_g$. The GNN contains three learned update functions—a node update function U_n, an edge update function U_e, and a global update function U_g. In all of the following equations, $\bigoplus$ denotes concatenation.

The GNN first gathers, for each edge $e \in E$, the sending and receiving node features $\mathbf{G}_n[\mathbf{n}_s]$ and $\mathbf{G}_n[\mathbf{n}_r]$ (where $\mathbf{n}_s$ is the sending node and $\mathbf{n}_r$ is the receiving node), edge features $\mathbf{G}_e[\mathbf{n}_s \rightarrow \mathbf{n}_r]$, and global features $\mathbf{G}_g$ and concatenates them for input into the edge update function U_e, which generates new edge features $\mathbf{G}'_e$ for the output graph (Eq. 7)

$$\mathbf{G}'_e = U_e\left(\mathbf{G}_n[\mathbf{n}_s] \oplus \mathbf{G}_n[\mathbf{n}_r] \oplus \mathbf{G}_e[\mathbf{n}_s \rightarrow \mathbf{n}_r] \oplus \mathbf{G}_g\right) \forall e(\mathbf{n}_s \rightarrow \mathbf{n}_r) \in E \tag{7}$$

Then, for each node $n \in N$, all of the edge output features for edges coming into the node are element-wise aggregated by summation to generate an aggregated view of information arriving into the node. This is concatenated with the node input features and the global input features and passed as input into the node update function U_n, which generates node output features $\mathbf{G}'_n$ (Eq. 8)

$$\mathbf{G}'_n = U_n\left(\left(\sum\nolimits_{e \in E(* \rightarrow n)} \mathbf{G}'_e[e]\right) \oplus \mathbf{G}_n[n] \oplus \mathbf{G}_g\right) \forall n \in N \tag{8}$$

Last, the global update function U_g takes as input the aggregated node embeddings and edge embeddings, as well as the input global features $\mathbf{G}_g$, and generates new global embeddings $\mathbf{G}'_g$ (Eq. 9)

$$\mathbf{G}'_g = U_g\left(\sum\nolimits_{n \in N} \mathbf{G}'_n[n] \oplus \sum\nolimits_{e \in E} \mathbf{G}'_e[e] \oplus \mathbf{G}_g\right) \tag{9}$$

In the discussion below, we use Core($\mathbf{G}_n$,$\mathbf{G}_e$,$\mathbf{G}_g$) to denote the operations in Eqs. 7 to 9 above for brevity (Eq. 10)

$$\mathbf{G}'_n, \mathbf{G}'_e, \mathbf{G}'_g = \text{Core}(\mathbf{G}_n, \mathbf{G}_e, \mathbf{G}_g) \tag{10}$$

For a more thorough introduction to GNNs, please refer to Battaglia *et al.* (*19*).

Critic networks (Q value estimators)

The two state-value estimation networks Q have identical architecture but independent weight initialization. Given a state $\mathbf{s}$ with features encoded in graph tuple ($\mathbf{G}_n$,$\mathbf{G}_e$,$\mathbf{G}_g$) and an action $\mathbf{a}$, each Q network first concatenates the action (commanded joint velocities) with the corresponding robot node features, and the new node features are embedded into a higher-dimensional embedding space using the node embedding function E_n to produce the node embedding inputs to the core GNN $\mathbf{L}_n$. The edge features and global features are also embedded into a higher-dimensional embedding space using the learned edge embedding function E_e and global embedding function E_g, into embeddings $\mathbf{L}_e$ and $\mathbf{L}_g$, respectively (Eq. 11)

$$\mathbf{L}_n, \mathbf{L}_e, \mathbf{L}_g = E_n(\mathbf{G}_n \oplus \mathbf{a}), E_e(\mathbf{G}_e), E_g(\mathbf{G}_g) \tag{11}$$

Then, the resulting graph embedding made up of the feature embeddings is passed through the core GNN for processing, and the output Q value is predicted from the global features of the output graph with a scalar prediction function P_Q (Eq. 12)

$$Q(\mathbf{s}, \mathbf{a}) = P_Q\left(\text{Core}(\mathbf{L}_n, \mathbf{L}_e, \mathbf{L}_g)_g\right) \tag{12}$$

Policy network

The policy network uses a similar embedding process as the Q networks but without the action concatenation (Eq. 13)

$$\mathbf{L}_n, \mathbf{L}_e, \mathbf{L}_g = E_n(\mathbf{G}_n), E_e(\mathbf{G}_e), E_g(\mathbf{G}_g) \tag{13}$$

After passing through the core GNN, the output actions are predicted from the node features corresponding to the robot nodes N_{robots}, using another prediction function P_π (Eq. 14)

$$\pi(\mathbf{s}) = P_\pi\left(\text{Core}(\mathbf{L}_n, \mathbf{L}_e, \mathbf{L}_g)_n[\mathbf{n}_r]\right) \forall \mathbf{n}_r \in N_{\text{robots}} \tag{14}$$

All of the learned functions (three embedding functions and three update functions) have the same architecture. Each is a fully connected network where each layer consists of a matrix multiplication followed by a layer normalization operation (*28*) and Gaussian error linear unit (GELU) activation (*29*).

Both the value (P_Q) and policy (P_π) prediction networks are similar in architecture to the main learned update functions but much smaller in size. The experiments were conducted with JAX (*30*), with models built in the Jraph library (*31*) and the Flax library (*32*). Refer to table S1 for the model shapes.

Training

During training, 64 actors were used to continuously and asynchronously generate training episodes for insertion into a first-in–first-out replay buffer implemented using the Reverb library (*33*), updating their local copies of the policy network from the learner after each episode. The learner samples uniformly from the replay buffer. The replay buffer enforces a fixed sample-to-insert ratio by blocking insertion or sampling as necessary. The learner uses a single NVIDIA A100 GPU, and each actor uses an Intel Ice Lake Xeon CPU core. The training process takes 6 to 24 hours depending on problem size to achieve a 95% task solution rate, and the quality of trajectories continues to improve (solving more tasks, reducing execution time, and reducing number of collisions) for an additional 2 to 6 days. Please refer to table S1 for the main hyperparameters used in the experiments. These hyperparameters were tuned with the Google Vizier (*16*) black box optimizer.

Supplementary Materials

The PDF file includes:
Table S1

Other Supplementary Material for this manuscript includes the following:
Movies S1 to S4

REFERENCES AND NOTES

1. S. Pellegrinelli, N. Pedrocchi, L. M. Tosatti, A. Fischer, T. Tolio, Multi-robot spot-welding cells for car-body assembly: Design and motion planning. *Robot. Comput. Integr. Manuf.* **44**, 97–116 (2017).
2. Y. Yamada, S. Nagamatsu, Y. Sato, "Development of multi-arm robots for automobile assembly" in *Proceedings of 1995 IEEE International Conference on Robotics and Automation* (IEEE, 1995), vol. 3, pp. 2224–2229.
3. H. Chen, T. Fuhlbrigge, X. Li, "Automated industrial robot path planning for spray painting process: A review" in *2008 IEEE International Conference on Automation Science and Engineering* (IEEE, 2008), pp. 522–527.
4. V. N. Hartmann, A. Orthey, D. Driess, O. S. Oguz, M. Toussaint, Long-horizon multi-robot rearrangement planning for construction assembly. *IEEE Trans. Robot.* **39**, 239 (2022).
5. Z. Xian, P. Lertkultanon, Q.-C. Pham, Closed-chain manipulation of large objects by multi-arm robotic systems. *IEEE Robot. Autom. Lett.* **2**, 1832–1839 (2017).
6. S. LaValle, Rapidly-exploring random trees: A new tool for path planning (Tech. Rep. 98-11, Iowa State University, 1998).
7. S. M. LaValle, J. J. Kuffner, "Rapidly-exploring random trees: Progress and prospects" in *Algorithmic and Computational Robotics* (Taylor-Francis, 2001), pp. 303–307.
8. S. Karaman, E. Frazzoli, Sampling-based algorithms for optimal motion planning. *Int. J. Robot. Res.* **30**, 846–894 (2011).
9. J. Canny, *The Complexity of Robot Motion Planning* (MIT Press, 1988).
10. W. Vega-Brown, N. Roy, "Task and motion planning is pspace-complete" in *Proceedings of the AAAI Conference on Artificial Intelligence* (AAAI, 2020), vol. 34, pp. 10385–10392.
11. G. B. Mathews, On the partition of numbers. *Proc. Lond. Math. Soc.* **s1-28**, 486–490 (1896).
12. H. Ha, J. Xu, S. Song, Learning a decentralized multi-arm motion planner. arXiv:2011.02608 [cs.RO] (2020).
13. T. Pan, A. M. Wells, R. Shome, L. E. Kavraki, "A general task and motion planning framework for multiple manipulators" *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2021), pp. 3168–3174.
14. Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, L. D. Jackel, "Handwritten digit recognition with a back-propagation network" in *Advances in Neural Information Processing Systems* (MIT Press, 1989), pp. 396–404.
15. J. J. Kuffner, S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning" in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)* (IEEE, 2000), vol. 2, pp. 995–1001.
16. D. Golovin, B. Solnik, S. Moitra, G. Kochanski, J. Karro, D. Sculley, "Google Vizier: A service for black-box optimization" in *KDD '17: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery, 2017), pp. 1487–1495.
17. X. Song, Q. Zhang, C. Lee, E. Fertig, T.-K. Huang, L. Belenki, G. Kochanski, S. Ariafar, S. Vasudevan, S. Perel, D. Golovin, The Vizier gaussian process bandit algorithm. arXiv:2408.11527 [cs.LG] (2024).
18. P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks. *stat* **1050**, 10 (2017).
19. P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, C. Gulcehre, F. Song, A. Ballard, J. Gilmer, G. Dahl, A. Vaswani, K. Allen, C. Nash, V. Langston, C. Dyer, N. Heess, D. Wierstra, P. Kohli, M. Botvinick, O. Vinyals, Y. Li, R. Pascanu, Relational inductive biases, deep learning, and graph networks. arXiv:1806.01261 [cs.LG] (2018).
20. S. Fujimoto, H. Hoof, D. Meger, "Addressing function approximation error in actor-critic methods" in *Proceedings of the 35th International Conference on Machine Learning* (PMLR, 2018), pp. 1587–1596.
21. J. W. Ratcliff, K. Mamou, Voxelized hierarchical convex decomposition - V-HACD, version 4, GitHub (2023); https://github.com/kmammou/v-hacd.
22. C.-T. Chang, B. Gorissen, S. Melchior, Fast oriented bounding box optimization on the rotation group $SO(3, \mathbb{R})$. *ACM Trans. Graph.* **30**, 1–16 (2011).
23. D. M. Olsson, L. S. Nelson, The Nelder-Mead simplex procedure for function minimization. *Technometrics* **17**, 45–51 (1975).
24. Y. Zhou, C. Barnes, J. Lu, J. Yang, H. Li, "On the continuity of rotation representations in neural networks" in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2019), pp. 5745–5753.
25. M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, P. Abbeel, W. Zaremba, "Hindsight experience replay" in *Advances in Neural Information Processing Systems* (Curran Associates, 2017), pp. 5048–5058.
26. D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, M. Riedmiller, "Deterministic policy gradient algorithms" in *Proceedings of the 31st International Conference on Machine Learning* (MLResearchPress, 2014), pp. 387–395.
27. B. T. Polyak, A new method of stochastic approximation type. *Avtom. Telemek.* **7**, 98–107 (1990).
28. J. L. Ba, J. R. Kiros, G. E. Hinton, Layer normalization. arXiv:1607.06450 [stat.ML] (2016).
29. D. Hendrycks, K. Gimpel, Gaussian error linear units (GELUs). arXiv:1606.08415 [cs.LG] (2016).
30. J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, Q. Zhang, JAX: Composable transformations of Python+NumPy programs, version 0.3.13, GitHub (2018); https://github.com/jax-ml/jax.
31. J. Godwin, T. Keck, P. Battaglia, V. Bapst, T. Kipf, Y. Li, K. Stachenfeld, P. Veličković, A. Sanchez-Gonzalez, Jraph: A library for graph neural networks in jax, version 0.0.1.dev, GitHub (2020); http://github.com/google-deepmind/jraph.
32. J. Heek, A. Levskaya, A. Oliver, M. Ritter, B. Rondepierre, A. Steiner, M. van Zee, Flax: A neural network library and ecosystem for JAX, version 0.11.1, GitHub (2023); http://github.com/google/flax.
33. A. Cassirer, G. Barth-Maron, E. Brevdo, S. Ramos, T. Boyd, T. Sottiaux, M. Kroiss, Reverb: A framework for experience replay. arXiv:2102.04736 [cs.LG] (2021).
34. E. Todorov, T. Erez, Y. Tassa, "Mujoco: A physics engine for model-based control" in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2012), pp. 5026–5033.

Acknowledgments
Funding: This work was funded by Google DeepMind and Intrinsic. **Author contributions:** M.L. and K.A. conceptualized, designed, and conducted the experiments. M.L., K.A., and J.S. developed the core algorithmic framework. M.L. and K.G. developed and implemented the experimental infrastructure. M.L., K.A., K.G., J.S., and S.S. proposed improvements to the system. M.L., K.A., S.S., and J.S. prepared and authored the paper. J.S., K.A., and Z.L. provided research guidance and planning. K.A., J.S., S.S., and Z.L. edited and revised the paper. T.K. managed the project and provided organizational support and funding. **Competing interests:** Google DeepMind has submitted US and WO patent applications for the techniques described in this paper. All authors affiliated with Google DeepMind and Intrinsic own Alphabet shares as part of their employment. All authors declare that they have no other competing interests. **Data and materials availability:** All data needed to support the conclusions of this manuscript are included in the main text or Supplementary Materials. The open sourced version of the system described in this paper, which is implemented with the MuJoCo physics simulator (*34*), is available at https://doi.org/10.5281/zenodo.16546403.

Submitted 4 November 2024
Accepted 5 August 2025
Published 3 September 2025
10.1126/scirobotics.ads1204

Share **Your Robotics Research** with the **World**.





Transforming the Future of Robotics in Research !

As a multidisciplinary online-only journal, *Science Robotics* publishes original, peer-reviewed, research articles that advance the field of robotics. The journal provides a central forum for communication of new ideas, general principles, and original developments in research and applications of robotics for all environments.

Submit your research today. Learn more at: **ScienceRobotics.org**

Science Robotics

AAAS