

SPECIAL EDITION

www.ScienceRobotics.org

Science Robotics




SOMETIMES THE GRASS REALLY IS GREENER SOMEPLACE NEW.



Find a new job at [ScienceCareers.org/robotics](https://www.sciencecareers.org/robotics)

There's scientific proof that when you're happy with what you do, you're better at what you do. Access career opportunities, see who's hiring and take advantage of our proprietary career-search tools. Get tailored job alerts, post your resume and manage your applications all in one place. Start here: [sciencecareers.org](https://www.sciencecareers.org)

ScienceCareers

FROM THE JOURNAL SCIENCE  AAAS

SPECIAL EDITION: *Science Robotics*



Credit: adobestocks.com/sdcoret

This booklet features eleven articles published in *Science Robotics* encompassing recent advances in aerial, medical, bioinspired and swarm robotics as well as timely commentary on the use of artificial intelligence in robotics. The journal covers traditional disciplines of robotics such as robotic grasping and legged locomotion. It also covers emerging trends such as electronic skins and soft robotics.

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

Holden Thorp, Ph.D.
Science family of journals
AAAS, Washington, DC

Chief Scientific Advisor and Founding Editor

Guang-Zhong Yang, Ph.D.
Shanghai Jiao Tong University

Editor

Michael M. Lee, DPhil
AAAS, Cambridge, UK

Editorial Team

Trista Wagoner
Brianna Costache

Editorial Board

Howie Choset

Robotics Institute, Carnegie Mellon University

Steven H. Collins

Department of Mechanical Engineering,
Stanford University

Peer Fischer

Max Planck Institute for Intelligent Systems,
and Institute of Physical Chemistry,
University of Stuttgart, Germany

Ken Goldberg

Industrial Engineering & Operations
Research, University of California Berkeley

Neil Jacobstein

Singularity University and MediaX Program,
Stanford University

Danica Kragic

Royal Institute of Technology,
Stockholm, Sweden

Cecilia Laschi

BioRobotics Institute, Scuola Superiore
Sant'Anna, Pisa, Italy

Robin R. Murphy

Department of Computer Science and
Engineering, Texas A&M University

Bradley J. Nelson

Department of Mechanical and
Process Engineering
ETH Zurich, Switzerland

Robert J. Wood

Wyss Institute for Biologically Inspired
Engineering and School of Engineering
and Applied Sciences, Harvard University

Science Robotics Scientific Advisory Board

James G. Bellingham

Center for Marine Robotics
Woods Hole Oceanographic Institution

Henrik I. Christensen

Contextual Robotics Institute
Jacobs School of Engineering,
University of California, San Diego

Paolo Dario

The BioRobotics Institute, Scuola
Superiore Sant'Anna, Pisa, Italy

Pierre E. Dupont

Boston Children's Hospital,
Harvard Medical School

Toshio Fukuda

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

Robert J. Full

Center for Interdisciplinary
Bio-Inspiration in Education and
Research (CIBER), University of
California at Berkeley

Hugh Herr

Harvard-MIT Division of Health
Sciences and Technology and
MIT Media Lab

Robert D. Howe

BioRobotics Laboratory,
Harvard University

Koji Ikuta

University of Tokyo, Japan

Vijay Kumar

University of Pennsylvania

Marcia McNutt

National Academy of Sciences
National Research Council

Yoshihiko Nakamura

University of Tokyo

Stuart Russell

University of California, Berkeley

Russell H. Taylor

Johns Hopkins University

IN THIS BOOKLET:

EDITORIAL

4

Autonomous vehicles: An imperfect path to saving millions of lives

Neil Jacobstein

FOCUS ARTICLES

6

Task-agnostic self-modeling machines

Robert Kwiatkowski and Hod Lipson

8

XAI—Explainable artificial intelligence

David Gunning *et al.* (Guang-Zhong Yang)

10

Forging global cooperation and collaboration

Guang-Zhong Yang *et al.* (Junku Yuh)

RESEARCH ARTICLES

12

Learning ambidextrous robot grasping policies

Jeffrey Mahler *et al.* (Ken Goldberg)

23

Learning agile and dynamic motor skills for legged robots

Jemin Hwangbo *et al.* (Marco Hutter)

36

Perching and resting—A paradigm for UAV maneuvering with modularized landing gears

Kaiyu Hang *et al.* (Fu Zhang)

46

Autonomous robotic intracardiac catheter navigation using haptic vision

G. Fagogenis *et al.* (P. E. Dupont)

58

A neuro-inspired artificial peripheral nervous system for scalable electronic skins

Wang Wei Lee *et al.* (Benjamin C. K. Tee)

73

Biomimetic sensory feedback through peripheral nerve stimulation improves dexterous use of a bionic hand

J. A. George *et al.* (G. A. Clark)

84

Insect-scale fast moving and ultrarobust soft robot

Yichuan Wu *et al.* (Liwei Lin)

93

A robot made of robots: Emergent transport and control of a smarticle ensemble

William Savoie *et al.* (Daniel I. Goldman)

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

Sponsorship Opportunities:

Tracy Holmes, *Science*

tholmes@science-int.co.uk | (44) 1223 326525 ext 5525
1200 New York Ave NW, Washington DC 20005

© 2020 by The American Association for the Advancement of Science. All Rights Reserved.

Learn more and submit your research today: **ScienceRobotics.org**

AUTONOMOUS VEHICLES

Autonomous vehicles: An imperfect path to saving millions of lives

Autonomous vehicles (AVs) are imperfect, but they are likely to eventually become safer drivers than humans. According to the World Health Organization in 2018, 1.35 million humans died in automotive fatalities, with tens of millions more injuries and disabilities (1). Few of those deaths were the result of part failure or bad luck; the majority resulted from intoxication, texting while driving, and other distractions. Although autonomous vehicles still have a long way to develop, they already have a track record of fewer crashes than humans per million miles driven, albeit mostly under good conditions. People may disagree on the precise road conditions and safety differentials between humans and AVs, but it seems likely that eventually AVs will save millions of lives. They do not have to be perfect, in spite of the furor when one is involved in an accident. They just have to be safer, perhaps a lot safer for adoption than the currently available alternatives. AVs will transform the insurance and automotive industries, reshape transportation and delivery alternatives, and alter social behavior and the urban landscape. No amount of bad news from AV accidents over the next few years will change that outcome, independent of the timing of acceptance in different applications and jurisdictions. What remains is a lot of research, development, engineering, and testing work to continuously improve autonomous vehicles with the goal of utilizing them as soon as possible to save lives.

This special section includes a research paper titled “Neural network vehicle models for high-performance automated driving” (2). The Stanford University team trained a neural network structure using a sequence of past states and inputs motivated by a physical model. The neural network achieved better performance than the physical model when implemented in the same feedforward-feedback control architecture on an experimental vehicle. Further, when trained on a combination of data from dry roads and snow, the system was able to make appropriate predictions for the road surface on which the vehicle was traveling without the need for explicit road friction estimation. This work contributes to eventually expanding the model-based control of automated vehicles over their full operating range.

The research paper titled “AADS: Augmented autonomous driving simulation using data-driven algorithms” (3) augmented real-world pictures with a simulated traffic flow to create photorealistic simulation images and

renderings. The paper, based on China/U.S. teamwork, utilized LiDAR and cameras to scan street scenes. The system generated plausible traffic flows for cars and pedestrians and composed them into the background. The composite images could then be resynthesized with different viewpoints and sensor models (camera or LiDAR). The resulting images are photorealistic and annotated. The system provides some scenarios of end-to-end training and testing of autonomous driving systems from perception to planning.

The Focus paper “Self-driving cars: A city perspective” (4) points out that, although AVs navigate cities without modifying the roadbed, they will be integrating signals from a wide variety of urban sources. These sources include traffic flows, aggregated smart phone data, car sharing, and more. In addition, AVs will drive changes in parking spaces and garages, real estate prices, commuting patterns (such as sleeper cars and buses for travel greater than 500 miles), and ride-sharing patterns. Overall, AVs will both drive and coevolve with changes in the urban landscape.

The Focus paper “Parallel testing of vehicle intelligence via virtual-real interaction” (5) reports a closed-loop parallel testing system from the longest-lasting Chinese AV competition: the Intelligent Vehicle Future Challenge of China (IVFC). Developers still lack a systematic and standardized way to test the capabilities of autonomous vehicles. The paper addressed three major programmatic objectives in testing autonomous vehicles. First, they generated and classified a standard set of driving tasks. Second, they designed a testing system that integrates simulation and field testing of real-world traffic tasks in various scenarios. Third, they built an integrated and closed-loop system to evaluate the task-specific performance of autonomous vehicles and help to improve the testing system itself.

It is interesting and useful to compare these methods with others used elsewhere. The key is not to claim superiority, but rather to encourage learning across programs over time. Additional information is available for each program via the authors. The AV industry has just begun to share data widely, with some initial caution due to competitive concerns. However, given the critical life-saving importance of AVs, we think that there is much to be gained by data sharing and continuous improvement across platforms.

—Neil Jacobstein



Neil Jacobstein is the Chair of Artificial Intelligence and Robotics, Singularity University, NASA Research Park, Moffett Field, CA, USA, and a Distinguished Visiting Scholar, MediaX Program, Stanford University, Stanford, CA, USA. neil@singularityu.org

Copyright © 2019
The Authors, some rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

REFERENCES

1. World Health Organization, *Global Status Report on Road Safety 2018*; www.who.int/violence_injury_prevention/road_safety_status/2018/en/.
2. N. A. Spielberg, M. Brown, N. R. Kapania, J. C. Kegelmann, J. C. Gerdes, Neural network vehicle models for high-performance automated driving. *Sci. Robot.* **4**, eaaw1975 (2019).
3. W. Li, C. W. Pan, R. Zhang, J. P. Ren, Y. X. Ma, J. Fang, F. L. Yan, Q. C. Geng, X. Y. Huang, H. J. Gong, W. W. Xu, G. P. Wang, D. Manocha, R. G. Yang, AADS: Augmented autonomous driving simulation using data-driven algorithms. *Sci. Robot.* **4**, eaaw0863 (2019).
4. F. Duarte, Self-driving cars: A city perspective. *Sci. Robot.* **4**, eaav9843 (2019).
5. L. Li, X. Wang, K. Wang, Y. Lin, J. Xin, L. Chen, L. Xu, B. Tian, Y. Ai, J. Wang, D. Cao, Y. Lin, C. Wang, N. Zheng, F.-Y. Wang, Parallel testing of vehicle intelligence via virtual-real interaction. *Sci. Robot.* **4**, eaaw4106 (2019).

10.1126/scirobotics.aaw8703

Citation: N. Jacobstein, Autonomous vehicles: An imperfect path to saving millions of lives. *Sci. Robot.* **4**, eaaw8703 (2019).

ARTIFICIAL INTELLIGENCE

Task-agnostic self-modeling machines

Robert Kwiatkowski^{1*} and Hod Lipson^{2,3}**A robot modeled itself without prior knowledge of physics or its shape and used the self-model to perform tasks and detect self-damage.**

Copyright © 2019
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

We humans are masters of self-reflection. We form a mental picture of ourselves by revisiting past experiences and use that mental image to contemplate future scenarios. Our mental self-image contains information about our body configuration in physical space. Our self-image also gives us the ability to link future actions with likely sensations. Your mental self-image allows you to imagine yourself walking on the beach tomorrow, smelling the ocean and feeling the sand under your feet.

An accurate self-image will be key to allowing robots to learn and plan internally without resorting to costly training in physical reality for each new task. The ability to self-simulate can create an illusion of one-shot learning, whereas in actuality, adaptation involves incremental learning or planning inside an internal self-image. A self-image can also be used to identify and track damage, wear, or growth.

Humans likely acquire their self-image early in life and adapt it continuously (1). However, most robots today cannot generate their own self-image. Although recent advances in machine learning have allowed robots to become increasingly adept at understanding the world around them, when it comes to understanding themselves, most robots today still rely on a hard-coded simulator (2,3). These designer-provided simulators are laborious to construct and invariably become outdated.

As an alternative to self-modeling, many robotic systems do without a model altogether by using end-to-end training for a specific task, applying techniques such as model-free reinforcement learning (4). Such task-specific learning may be good for narrow artificial intelligence (AI) but lacks the generality and transferability required for robots capable of continuously learning new tasks throughout their lives.

Here, we suggest that, because the robot itself is persistent across multiple tasks, there

is strong incentive to extract a self-model and then reuse that self-model repeatedly to learn new tasks. Moreover, by separating the self from the task, every future experience can be used to refine a common self-model, leading to continuous self-monitoring.

Early adaptive control methods also attempted to tune parameters of a fixed analytical self-model (5). We previously used evolutionary algorithms to find the morphology most consistent with the robot's recorded action-sensation pairs (6), but both approaches make many assumptions. A key question remained: Can a robot create a self-model with no prior knowledge?

First, we chose a physical robot with four coupled degrees of freedom. The robot recorded action-sensation pairs by moving through 1000 random trajectories (Fig. 1, step 1). Actions correspond to four motor angle commands and sensations correspond to the absolute coordinate of the end effector. This step is not unlike a babbling baby observing its hands. The entire captured dataset is provided in (7).

Importantly, when a robot's motors are commanded to achieve some target angles, they do not necessarily reach those angles due to hysteresis, self-collision, structural flexibility, and other effects. Therefore, a high-fidelity self-model must capture not just the direct geometric transformations from the robot's base to the end effector, but an implicit relationship between current positions, new motor commands, past positions, and past motor commands.

We used deep learning to train a self-model (Fig. 1, step 2). Using the acquired self-model, the robot could apply a simple planning strategy to accomplish a variety of tasks. We tested the performance of the robot on two separate tasks: a pick-and-place task and a handwriting task (Fig. 1, step 3), both in open and in closed loop.

Closed-loop control allows the robot to recalibrate its actual position between each step along the trajectory by using positional sensor feedback. In contrast, open-loop control involves carrying out a task based entirely on the internal self-model, without any external feedback, like reaching for your nose with eyes closed. This test is also frequently used to test human cerebellar dysfunction such as dysmetria (8).

We ran multiple tests with both explicit and implicit representations. Explicit models capture the relationship between motor commands and end-effector position but cannot handle self-collision. Implicit models capture the sequential relationship between state-action pairs and thus are more general. In open-loop tests where planning was completed successfully, the median distance between the physical effector and the target was 4.3 cm. In closed loop, the self-model achieved a median physical error of 0.6 cm, an error lower than the accuracy between the analytical model and the physical robot. The explicit model achieved a median accuracy of 0.65 cm. These results suggest that acquired self-models would be able to successfully execute internal planning and learning on par with a conventional simulator.

The second test involved a combination of subtasks and gripper actuations. The robot used its self-model to plan how to pick and deposit nine red balls, each 20 mm in diameter. In closed loop, the self-models had sufficient accuracy and consistency to execute a pick-and-place task with precision similar to analytical forward kinematics. The open-loop pick rate was 44%, whereas the place rate was 100% of successful picks, and most failures were a result of the planner, not of the self-model.

To demonstrate that the same self-model could be used for other tasks without additional task-specific retraining, we performed a second task involving simple handwriting with a marker. This task was used for qualitative assessment only.

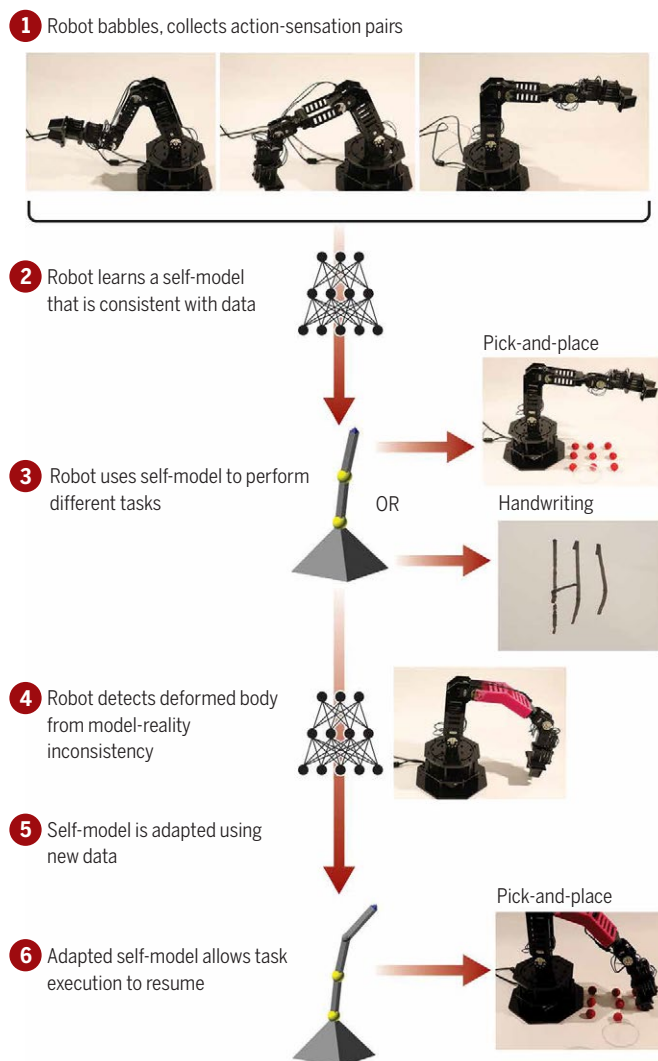
We concluded by replacing one of the robot arms with a longer and slightly deformed

¹Department of Computer Science, Columbia University, New York, NY 10027, USA. ²Department of Mechanical Engineering, Columbia University, New York, NY 10027, USA. ³Data Science Institute, Columbia University, New York, NY 10027, USA.

*Corresponding author. Email: robert.kwiatkowski@columbia.edu

Fig. 1. Self-model generation, usage, and adaptation.

An outline of the self-modeling process from data collection to task planning. (Step 1) The robot recorded action-sensation pairs. (Step 2) The robot used deep learning to create a self-model consistent with the data. (Step 3) The self-model could be used for internal planning of two separate tasks without any further physical experimentation. (Step 4) The robot morphology was abruptly changed to emulate damage. (Step 5) The robot adapted the self-model using new data. (Step 6) Task execution resumed.



part as a proxy for unanticipated morphological damage (Fig. 1, step 4). The robot was able to detect the change and retrain the self-model using 10% additional data (Fig. 1, step 5). The new self-model allowed the robot to resume its original pick-and-place task with little loss of performance (Fig. 1, step 6).

Robotics research has historically split between two camps: model-predictive control and model-free learning. We propose a hybrid where machine learning acquires a self-model that is then reused to perform planning or learning internally. This way, data collected in the course of any task can help refine the self-model and thus transfer to other tasks.

Self-imaging will be key to allowing robots to move away from the confinements of so-called narrow AI toward more general abilities. We conjecture that this separation of self and task may have also been the evolutionary origin of self-awareness in humans.

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/26/eaau9354/DC1

Section S1. Related work

Section S2. Experimental robot platform

Section S3. Training the self-model

Section S4. Results

Section S5. Self-model adaptation to damage

Section S6. Limitations

Fig. S1. The robot and data used for this study.

Fig. S2. Self-modeling process.

Fig. S3. Self-model architecture and training.

Fig. S4. Two tasks.

Fig. S5. Accuracy degradation.

Fig. S6. Diagram of the reach of the WidowX robot arm taken from Trossen Robotics.

Fig. S7. Rendering of the CAD model used to 3D-print the deformed arm length.

Fig. S8. Analytical model position versus physical robot position.

Fig. S9. The distribution of accuracies when using the forward kinematics model.

Fig. S10. The distribution of accuracies visualized onto the reachable space.

Fig. S11. The forward kinematics learner architecture.

Table S1. Summary of results of trajectory planning and pick-and-place tests.

Table S2. Joint position.

Table S3. Table of model parameters in the model used to conduct all self-modeling tests.

Table S4. Table of positions the arm was instructed to go to in the pick-and-place test.

References (9–16).

Movie S1. Overview video.

Data S1. 3D printed part model .stl file.

REFERENCES AND NOTES

- P. Rochat, Five levels of self-awareness as they unfold early in life. *Conscious. Cogn.* **12**, 717–731 (2003).
- N. Koenig, A. Howard, Design and use paradigms for Gazebo, an open-source multi-robot simulator, in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2004), pp. 2149–2154.
- 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.
- J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms. arXiv:1707.06347 [cs.LG] (20 July 2017).
- I. Doré Landau, R. Lozano, M. M'Saad, A. Karimil, *Adaptive Control: Algorithms, Analysis and Applications* (Springer, 2011).
- J. Bongard, V. Zykov, H. Lipson, Resilient machines through continuous self-modeling. *Science* **314**, 1118–1121 (2006).
- R. Kwiatkowski, Source code for Task-Agnostic Self-Modeling Machines, GitHub; https://github.com/rjk2147/Task-Agnostic_Self-Modeling_Machines.
- J. D. Schmahmann, Disorders of the cerebellum: ataxia, dysmetria of thought, and the cerebellar cognitive affective syndrome. *J. Neuropsychiatry Clin. Neurosci.* **16** (3), 367–378 (2004).
- B. Damas, J. Santos-Victor, An online algorithm for simultaneously learning forward and inverse kinematics, in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2012), pp. 1499–1506.
- J. Sturm, C. Stachniss, W. Burgard, A probabilistic framework for learning kinematic models of articulated objects. arXiv:1405.7705 [cs.RO] (16 January 2014).
- N. Haber, D. Mrowca, L. Fei-Fei, D. L. K. Yamins, Emergence of structured behaviors from curiosity-based intrinsic motivation. arXiv:1802.07461 [cs.LG] (21 February 2018).
- B. Amos, L. Dinh, S. Cabi, T. Rothörl, S. Gómez Colmenarejo, A. Muldal, T. Erez, Y. Tassa, N. de Freitas, M. Denil, Learning awareness models. arXiv:1804.06318 [cs.AI] (17 April 2018).
- J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv:1412.3555 [cs.NE] (11 December 2014).
- A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, in *Advances in Neural Information Processing Systems 25* (NIPS 2012), pp. 1097–1105.
- S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv:1502.03167 [cs.LG] (11 February 2015).
- D. P. Kingma, J. Ba, Adam: A method for stochastic optimization. arXiv:1412.6980 [cs.LG] (22 December 2014).

Acknowledgments: Supported by DARPA MTO grant HR0011-18-2-0020.

10.1126/scirobotics.aau9354

Citation: R. Kwiatkowski, H. Lipson, Task-agnostic self-modeling machines. *Sci. Robot.* **4**, eaau9354 (2019).

ARTIFICIAL INTELLIGENCE

XAI—Explainable artificial intelligence

David Gunning^{1*†}, Mark Stefik², Jaesik Choi³, Timothy Miller⁴, Simone Stumpf⁵, Guang-Zhong Yang^{6†}**Explainability is essential for users to effectively understand, trust, and manage powerful artificial intelligence applications.**

Copyright © 2019
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

Recent successes in machine learning (ML) have led to a new wave of artificial intelligence (AI) applications that offer extensive benefits to a diverse range of fields. However, many of these systems are not able to explain their autonomous decisions and actions to human users. Explanations may not be essential for certain AI applications, and some AI researchers argue that the emphasis on explanation is misplaced, too difficult to achieve, and perhaps unnecessary. However, for many critical applications in defense, medicine, finance, and law, explanations are essential for users to understand, trust, and effectively manage these new, artificially intelligent partners [see recent reviews (1–3)].

Recent AI successes are largely attributed to new ML techniques that construct models in their internal representations. These include support vector machines (SVMs), random forests, probabilistic graphical models, reinforcement learning (RL), and deep learning (DL) neural networks. Although these models exhibit high performance, they are opaque in terms of explainability. There may be inherent conflict between ML performance (e.g., predictive accuracy) and explainability. Often, the highest performing methods (e.g., DL) are the least explainable, and the most explainable (e.g., decision trees) are the least accurate. Figure 1 illustrates this with a notional graph of the performance-explainability tradeoff for some of the ML techniques.

WHAT IS XAI?

The purpose of an explainable AI (XAI) system is to make its behavior more intelligible to humans by providing explanations. There are some general principles to help create effective, more human-understandable AI systems: The XAI system should be able to explain its capabilities and understandings; explain what it has done, what it is doing now, and what will happen next; and disclose the salient information that it is acting on (4).

However, every explanation is set within a context that depends on the task, abilities, and expectations of the user of the AI system. The definitions of interpretability and explainability are, thus, domain dependent and may not be defined independently from a domain. Explanations can be full or partial. Models that are fully interpretable give full

and completely transparent explanations. Models that are partially interpretable reveal important pieces of their reasoning process. Interpretable models obey “interpretability constraints” that are defined according to the domain (e.g., monotonicity with respect to certain variables and correlated variables obey particular relationships), whereas black box or unconstrained models do not necessarily obey these constraints. Partial explanations may include variable importance measures, local models that approximate global models at specific points and saliency maps.

EXPECTATION FROM USERS

XAI assumes that an explanation is provided to an end user who depends on the decisions, recommendations, or actions produced by an AI system, yet there could be many different

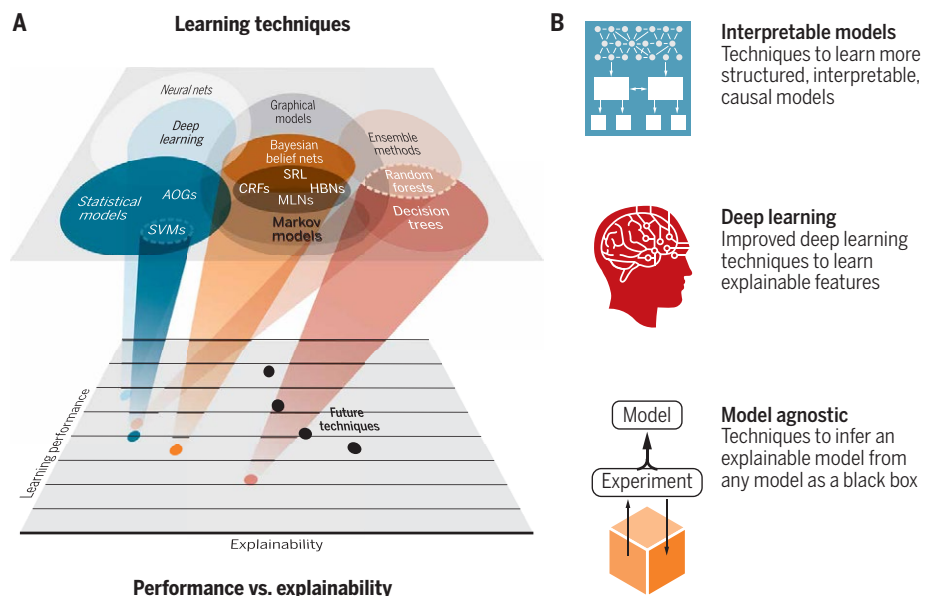


Fig. 1. Performance versus explainability tradeoff for ML techniques. (A) Learning techniques and explainability. Concept adapted from (9). (B) Interpretable models: ML techniques that learn more structured, interpretable, or causal models. Early examples included Bayesian rule lists, Bayesian program learning, learning models of causal relationships, and using stochastic grammars to learn more interpretable structure. Deep learning: Several design choices might produce more explainable representations (e.g., training data selection, architectural layers, loss functions, regularization, optimization techniques, and training sequences). Model agnostic: Techniques that experiment with any given ML model, as a black box, to infer an approximate explainable model.

¹Defense Advanced Research Projects Agency (DARPA), 675 North Randolph Street, Arlington, VA 22201, USA.

²Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA 94304, USA. ³Graduate School of Artificial Intelligence, Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea. ⁴School of Computing and Information Systems, University of Melbourne, Victoria 3010, Australia. ⁵Centre for HCI Design, School of Mathematics, Computer Science and Engineering, City, University of London, London EC1V 0HB, UK. ⁶Institute of Medical Robotics, Shanghai Jiao Tong University, Shanghai, China.

*Present address: Facebook AI Research, 770 Broadway, New York, NY 10003, USA.

†Corresponding author. Email: dgunning@fb.com (D.G.); gzyang@sjtu.edu.cn (G.-Z.Y.)

kinds of users, often at different time points in the development and use of the system (5). For example, a type of user might be an intelligence analyst, a judge, or an operator. However, other users who demand an explanation of the system may be developers or test operators who need to understand where there might be areas of improvement. Yet, another user might be policy makers, who are trying to assess the fairness of the system. Each user group may have a preferred explanation type that is able to communicate information in the most effective way. An effective explanation will take the target user group of the system into account, who might vary in their background knowledge and needs for what should be explained.

EXPLAINABILITY—EVALUATION AND MEASUREMENT

A number of ways of evaluating and measuring the effectiveness of an explanation have been proposed; however, there is currently no common means of measuring whether an XAI system is more intelligible to a user than a non-XAI system. Some of these measures are subjective measures from the user's point of view, such as user satisfaction, which can be measured through a subjective rating of the clarity and utility of an explanation. More objective measures for an explanation's effectiveness might be task performance; i.e., does the explanation improve the user's decision-making? Reliable and consistent measurement of the effects of explanations is still an open research question. Evaluation and measurement for XAI systems include evaluation frameworks, common ground [different thinking and mutual understanding (6)], common sense, and argumentation [why (7)].

XAI—ISSUES AND CHALLENGES

There remain many active issues and challenges at the intersection of ML and explanation.

1) Starting from computers versus starting from people (8). Should XAI systems

tailor explanations to particular users? Should they consider the knowledge that users lack? How can we exploit explanations to aid interactive and human-in-the-loop learning, including enabling users to interact with explanations to provide feedback and steer learning?

2) Accuracy versus interpretability. A major thread of XAI research on explanation explores techniques and limitations of interpretability. Interpretability needs to consider tradeoffs involving accuracy and fidelity and to strike a balance between accuracy, interpretability, and tractability.

3) Using abstractions to simplify explanations. High-level patterns are the basis for describing big plans in big steps. Automating the discovery of abstractions has long been a challenge, and understanding the discovery and sharing of abstractions in learning and explanation are at the frontier of XAI research today.

4) Explaining competencies versus explaining decisions. A sign of mastery by highly qualified experts is that they can reflect on new situations. It is necessary to help end users to understand the competencies of the AI systems in terms of what competencies a particular AI system has, how the competencies should be measured, and whether an AI system has blind spots; that is, are there classes of solutions it can never find?

From a human-centered research perspective, research on competencies and knowledge could take XAI beyond the role of explaining a particular XAI system and helping its users to determine appropriate trust. In the future, XAIs may eventually have substantial social roles. These roles could include not only learning and explaining to individuals but also coordinating with other agents to connect knowledge, developing cross-disciplinary insights and common ground, partnering in teaching people and other agents, and drawing on previously discovered knowledge to accelerate the further discovery and application of

knowledge. From such a social perspective of knowledge understanding and generation, the future of XAI is just beginning.

REFERENCES AND NOTES

1. W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, K. R. Muller, *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning* (Springer Nature, 2019).
2. H. J. Escalante, S. Escalera, I. Guyon, X. Baró, Y. Güçlütürk, U. Güçlü, M. van Gerven, *Explainable and Interpretable Models in Computer Vision and Machine Learning* (Springer, 2018).
3. O. Biran, C. Cotton, Explanation and justification in machine learning: A survey, paper presented at the IJCAI-17 Workshop on Explainable AI (XAI), Melbourne, Australia, 20 August 2017.
4. V. Bellotti, K. Edwards, Intelligibility and accountability: Human considerations in context-aware systems. *Hum. Comput. Interact.* **16**, 193–212 (2009).
5. T. Kulesza, M. Burnett, W. Wong, S. Stumpf, Principles of explanatory debugging to personalize interactive machine learning, in *Proceedings of the 20th International Conference on Intelligent User Interfaces* (ACM, 2015), pp. 126–137.
6. H. H. Clark, S. E. Brennan, Grounding in communication, in *Perspectives on Socially Shared Cognition*, L. B. Resnick, J. M. Levine, S. D. Teasley, Eds. (American Psychological Association, 1991), pp. 127–149.
7. D. Wang, Q. Yang, A. Abdul, B. Y. Lim, Designing theory-driven user-centric explainable AI, in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (ACM, 2019), paper no. 601.
8. T. Miller, Explanation in artificial intelligence: Insights from the social sciences. *Artif. Intell.* **267**, 1–38 (2018).
9. D. Gunning, Explainable artificial intelligence (XAI), DARPA/120; [www.cc.gatech.edu/~alanwags/DLAI2016/\(Gunning\)%20IJCAI-16%20DLAI%20WS.pdf](http://www.cc.gatech.edu/~alanwags/DLAI2016/(Gunning)%20IJCAI-16%20DLAI%20WS.pdf).

Funding: J.C. was supported by an Institute for Information and Communications Technology Planning and Evaluation (IITP) grant (no. 2017-0-01779; A machine learning and statistical inference framework for explainable artificial intelligence). Material within this technical publication is based on the work supported by the Defense Advanced Research Projects Agency (DARPA) under contract FA8650-17-C-7710 (to M.S.). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the official policy or position of the Department of Defense or the U.S. government.

10.1126/scirobotics.aay7120

Citation: D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, G.-Z. Yang, XAI—Explainable artificial intelligence. *Sci. Robot.* **4**, eaay7120 (2019).

ROBOTS AND SOCIETY

Forging global cooperation and collaboration

Guang-Zhong Yang^{1*}, Tong Boon Quek^{2,3}, Stefano Stramigioli^{4,5}, Han Ding⁶, Dong Sun⁷, Junku Yuh⁸**As researchers create better robots, major robotics initiatives and government funding programs need better international cooperation and collaboration.**Copyright © 2020
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

The year 2020 will witness a continuing increase in global robot installations. The five major markets for industrial robots—the United States, China, Japan, South Korea, and Germany—now account for nearly 75% of the total installations (Fig. 1), and the predicted per-year increase in average between 2020 and 2022 is likely to reach +12%. Whereas new manufacturing technologies such as industry 4.0 are driving the growing demand for robots, demographic shift associated with the aging population will see the use of robots for assistive and domestic applications jump significantly. The World Health Organization (WHO) predicts that by 2050, the proportion of the world's population over 60 years will reach 22%, which, in absolute number, will hit 2 billion (1).

In 2011, the U.S. government announced a new initiative called the National Robotics Initiative with investments for major advances in next-generation robotics. A new solicitation for fiscal year 2020, the National Robotics Initiative 2.0: Ubiquitous Collaborative Robots (NRI-2.0), has been announced as a joint effort of the National Science Foundation, the National Aeronautics and Space Administration, the National Institute for Occupational Safety and Health, and the U.S. Department of Agriculture National Institute of Food and Agriculture. The first NRI was established as the result of many years of effort, and the first U.S. National Robotics Roadmap was published in 2009 and updated in 2013 and 2016 (2). A further update is due in August 2020. It identified three major factors that drive the adoption of robots: economic growth, quality of life, and safety of first responders. For the first two, they are clearly interlinked—improved productivity in what is now a

fiercely competitive international market and in the presence of an increasingly aging society. This pressure is clearly felt by many developed and developing countries.

With more than 800 robots per 10,000 employees, Singapore is now among the highest in the world in terms of industrial robot density (3). This high density reflects the city-state's reliance on technology, such as robotics, as a resource multiplier to mitigate the impact of its aging population due to consistently low birth rates. This complements its relatively open immigrant policy. With 45% of residents who are foreign born, Singapore is now one of the most cosmopolitan nations in the world. The National Robotics Programme (NRP) of Singapore has been established to encourage greater adoption of robotics and to ensure a more coordinated, coherent, and concerted effort in building Singapore's robotics capabilities. It is focused on technology and expertise development in robotics enablers, such as those related to end effectors, high-performance tactile sensors, indoor navigation in dense human environments, human-robotics interactions, soft robotics, and configurable robots. NRP is also working with relevant agencies to identify and close the gaps in standards, testing, and certification for robots and their supporting infrastructure especially for service robots. Given its resource constraints, particularly the demographic challenges, it is inevitable that Singapore will be a major robotics user. The NRP aims to make it a robotics innovator as well.

South Korea has also been ranked among the highest in robot density according to the International Federation of Robotics (IFR) (3). These rankings reflect the large investment in robotics by the government and emerging and well-established companies. The South

Korean government has dedicated about \$150 million to \$200 million in its annual budget toward research and development (R&D) in robotics, and plans are in place to expand the robotics market size to \$15 billion in 2023. The Ministry of Trade, Industry and Energy (MOTIE) spearheaded this R&D effort in robotics and established the Intelligent Robots Development and Distribution Promotion Act in 2008, which requires that a nationwide strategy on intelligent robots be presented every 5 years. The South Korean government has sponsored large R&D programs in robotics, such as the Underwater Construction Robotics R&D Center sponsored by the Ministry of Oceans and Fisheries (www.ucrc.re.kr/page_home_eng), Biomimetic Mechatronics (Bionic Arm) sponsored by the Ministry of Science and ICT (www.kist.re.kr/rmi), and Disaster Robotics R&D Center sponsored by MOTIE (<http://drc.re.kr/>). The South Korean government also offers various funding programs for international collaboration.

Europe is, in statistics, often presented as separate countries but it is more meaningful to treat it as a single entity for comparisons with, for example, the United States, considering that the largest research programs are coordinated at the European level. Already in 1998, the various scientific robotic stakeholders coordinated to form the EURON network. In the years after, the industrial network EUROP was formed; during the Seventh Research Framework of the European Commission (FP7), the two initiatives were joined into the euRobotics association (www.eu-robotics.net).

euRobotics has functioned as the private side in the public-private partnerships (PPPs) called SPARC (www.eu-robotics.net/sparc), which runs in the eighth framework called Horizon 2020 that ends in 2020. The European Commission is working to define the new framework program. The new program will have a part called Horizon Europe, which will be targeting research, and a second part called Digital Europe, which will be addressing industry and deployment of new technologies

¹Institute of Medical Robotics, Shanghai Jiao Tong University, Shanghai, China. ²National Robotics Programme, Agency for Science, Technology and Research, Singapore, Singapore. ³Ministry of Trade and Industry, Singapore, Singapore. ⁴euRobotics, Brussels, Belgium. ⁵University of Twente, Twente, Netherlands. ⁶State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, China. ⁷City University of Hong Kong, Hong Kong, China. ⁸Korea Institute of Robotics and Technology Convergence, Pohang, Korea.

*Corresponding author. Email: gzyang@sjtu.edu.cn

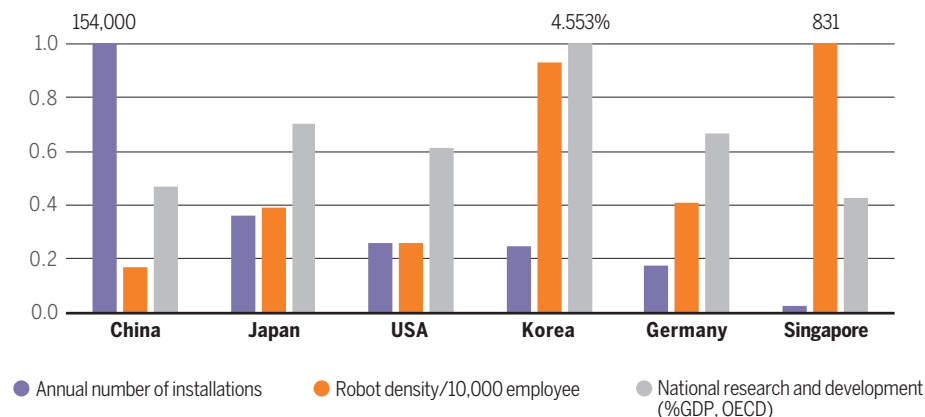


Fig. 1. Robotics and R&D investments by country. Annual installations of industrial robots (China, 154,000), robot density per 10,000 employees (Singapore, 831), and domestic spending on research and development (percentage of GDP; Korea, 4.553%), normalized to the maximum values in each category for the countries shown. Data sources: International Federation of Robotics (IFR) and Organisation for Economic Co-operation and Development (OECD).

for the digital economy and society. Horizon Europe will be divided further in three pillars: (i) Excellent Science, (ii) Global Challenges and European Industrial Competitiveness, and (iii) Innovative Europe. For pillar two, which addresses the cooperation between academia and industry to boost technologies and solutions underpinning EU policies and sustainable development goals, a budget of €52.7 billion has been proposed. New PPPs will be proposed, and euRobotics has taken the lead together with the Big Data Value Association to create a new partnership on artificial intelligence (AI) for which euRobotics will clearly cover the essential and most difficult physical part of AI. This new AI PPP will form the basis for the robotics investment until 2027. Furthermore, not related to PPP, other important initiatives are also forming to support the work of euRobotics as a private association from the content side, which include, for example, CENTRIS (www.centris-ai.eu), which gathers top scientists and researchers in Europe to

create an even stronger knowledge base for European robotics.

We have seen, in recent years, increasing development of cooperative robots—robots that are able to cooperate with human operators, learn from demonstrations, interact intelligently with the environment, and collaborate seamlessly with their robot peers of similar or different architectures. To support this, the National Natural Science Foundation of China launched an 8-year, \$200 million national research initiative called the Tri-Co (Coexisting-Cooperative-Cognitive) Robot in 2016. Coexistence will allow robots to ubiquitously and safely work alongside humans, whereas cooperation will enable robots to coordinate and collaborate effectively with other agents (either people or robots). Last, cognition will provide robots with the means to gather information, perceive, learn, and predict environmental changes and the behaviors of other agents. From deep sea to space exploration, from microrobotics to megascale infrastructure

robots, the pace of development and government support for robotics in China is accelerating. There is also increasingly strong synergy among fundamental science, technology development, and applications of robots.

Research efforts are making robots more intelligent, cooperative, and interconnected with improved learning and adaptation, interoperability, autonomy, and human-robot interaction, and major robotics initiatives and government funding programs need better international cooperation and collaboration. This needs to be at both strategic and tactical levels because we are all faced with similar socioeconomic challenges and the need for developing new talents. From the research and development standpoint, future generations of robotics need to be supported by a wide range of science and engineering disciplines as identified in (4). It is only through such cooperation and collaboration that we can make a true difference in the societal value of robotics.

REFERENCES AND NOTES

1. Ageing and Life Course, WHO; www.who.int/ageing/en/.
2. A Roadmap for UK Robotics (2016); <https://cra.org/ccc/wp-content/uploads/sites/2/2016/11/roadmap3-final-rs-1.pdf>.
3. World Robotics 2019 Edition, IFR; <https://ifr.org>.
4. 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).

Acknowledgments: This article is based on the presentation materials from the Government Forum of IROS 2019.

10.1126/scirobotics.aba5894

Citation: G.-Z. Yang, T. B. Quek, S. Stramigioli, H. Ding, D. Sun, J. Yuh, Forging global cooperation and collaboration. *Sci. Robot.* **5**, eaba5894 (2020).

ARTIFICIAL INTELLIGENCE

Learning ambidextrous robot grasping policies

Jeffrey Mahler^{1,2*}, Matthew Matl¹, Vishal Satish¹, Michael Danielczuk¹, Bill DeRose², Stephen McKinley², Ken Goldberg^{1,2}

Universal picking (UP), or reliable robot grasping of a diverse range of novel objects from heaps, is a grand challenge for e-commerce order fulfillment, manufacturing, inspection, and home service robots. Optimizing the rate, reliability, and range of UP is difficult due to inherent uncertainty in sensing, control, and contact physics. This paper explores “ambidextrous” robot grasping, where two or more heterogeneous grippers are used. We present Dexterity Network (Dex-Net) 4.0, a substantial extension to previous versions of Dex-Net that learns policies for a given set of grippers by training on synthetic datasets using domain randomization with analytic models of physics and geometry. We train policies for a parallel-jaw and a vacuum-based suction cup gripper on 5 million synthetic depth images, grasps, and rewards generated from heaps of three-dimensional objects. On a physical robot with two grippers, the Dex-Net 4.0 policy consistently clears bins of up to 25 novel objects with reliability greater than 95% at a rate of more than 300 mean picks per hour.

INTRODUCTION

Universal picking (UP), or the ability of robots to rapidly and reliably grasp a wide range of novel objects, can benefit applications in warehousing, manufacturing, medicine, retail, and service robots. UP is highly challenging because of inherent limitations in robot perception and control. Sensor noise and occlusions obscure the exact geometry and position of objects in the environment. Parameters governing physics such as center of mass and friction cannot be observed directly. Imprecise actuation and calibration lead to inaccuracies in arm positioning. Thus, a policy for UP cannot assume precise knowledge of the state of the robot or objects in the environment.

One approach to UP is to create a database of grasps on three-dimensional (3D) object models using grasp performance metrics derived from geometry and physics (1, 2) with stochastic sampling to model uncertainty (3, 4). This analytic method requires a perception system to register sensor data to known objects and does not generalize well to a large variety of novel objects in practice (5, 6). A second approach uses machine learning to train function approximators such as deep neural networks to predict the probability of success of candidate grasps from images using large training datasets of empirical successes and failures. Training datasets are collected from humans (7–9) or physical experiments (10–12). Collecting such data may be tedious and prone to inaccuracies due to changes in calibration or hardware (13).

To reduce the cost of data collection, we explored a hybrid approach that uses models from geometry and mechanics to generate synthetic training datasets. However, policies trained on synthetic data may have reduced performance on a physical robot due to inherent differences between models and real-world systems. This simulation-to-reality transfer problem is a long-standing challenge in robot learning (14–17). To bridge the gap, the hybrid method uses domain randomization (17–22) over objects, sensors, and physical parameters. This encourages policies to learn grasps that are robust to imprecision in sensing, control, and physics. Furthermore, the method plans grasps based on depth images, which can be simulated accurately using ray tracing (18, 19, 23) and are invariant to object color (24).

The hybrid approach has been used to learn reliable UP policies on a physical robot with a single gripper (25–28). However, different grasp modalities are needed to reliably handle a wide range of objects in practice. For example, vacuum-based suction-cup grippers can easily grasp objects with nonporous, planar surfaces such as boxes, but they may not be able to grasp small objects, such as paper clips, or porous objects, such as cloth.

In applications such as the Amazon Robotics Challenge, it is common to expand range by equipping robots with more than one end effector (e.g., both a parallel-jaw gripper and a suction cup). Domain experts typically hand-code a policy to decide which gripper to use at runtime (29–32). These hand-coded strategies are difficult to tune and may be difficult to extend to new cameras, grippers, and robots.

Here, we introduce “ambidextrous” robot policy learning using the hybrid approach to UP. We propose the Dexterity Network (Dex-Net) 4.0 dataset generation model, extending the gripper-specific models of Dex-Net 2.0 (19) and Dex-Net 3.0 (19). The framework evaluates all grasps with a common metric: expected wrench resistance, or the ability to resist task-specific forces and torques, such as gravity, under random perturbations.

We implement the model for a parallel-jaw gripper and a vacuum-based suction cup gripper and generate the Dex-Net 4.0 training dataset containing more than 5 million grasps associated with synthetic point clouds and grasp metrics computed from 1664 unique 3D objects in simulated heaps. We train separate Grasp Quality Convolutional Neural Networks (GQ-CNNs) for each gripper and combine them to plan grasps for objects in a given point cloud.

The contributions of this paper are as follows:

- 1) A partially observable Markov decision process (POMDP) framework for ambidextrous robot grasping based on robust wrench resistance as a common reward function.
- 2) An ambidextrous grasping policy trained on the Dex-Net 4.0 dataset that plans a grasp to maximize quality using a separate GQ-CNN for each gripper.
- 3) Experiments evaluating performance on bin picking with heaps of up to 50 diverse, novel objects and an ABB YuMi robot with a parallel-jaw and suction-cup gripper in comparison with hand-coded and learned baselines.

Experiments suggest that the Dex-Net 4.0 policy achieves 95% reliability on a physical robot with 300 mean picks per hour (MPPH) (successful grasps per hour).

¹Department of Electrical Engineering and Computer Sciences, UC Berkeley, Berkeley, CA 94720, USA. ²Department of Industrial Engineering and Operations Research, UC Berkeley, Berkeley, CA 94720, USA.

*Corresponding author. Email: jmahler@berkeley.edu

RESULTS

Ambidextrous robot grasping

We consider the problem of ambidextrous grasping of a wide range of novel objects from cluttered heaps using a robot with a depth camera and two or more available grippers, such as a vacuum-based suction-cup gripper and/or a parallel-jaw gripper. To provide context for the metrics and approaches considered in experiments, we formalize this problem as a POMDP (33) in which a robot plans grasps to maximize expected reward (probability of grasp success) given imperfect observations of the environment.

A robot with an overhead depth camera views a heap of novel objects in a bin. On grasp attempt t , a robot observes a point cloud \mathbf{y}_t from the depth camera. The robot uses a policy $\mathbf{u}_t = \pi(\mathbf{y}_t)$ to plan a grasp action \mathbf{u}_t for a gripper g consisting of a 3D rigid position and orientation of the gripper $\mathbf{T}_g = (\mathbf{R}_g, \mathbf{t}_g) \in \text{SE}(3)$. Upon executing \mathbf{u}_t , the robot receives a reward $R_t = 1$ if it successfully lifts and transports exactly one object from the bin to a receptacle and $R_t = 0$ otherwise. The observations and rewards depend on a latent state \mathbf{x}_t that is unknown to the robot and describes geometry, pose, center of mass, and material properties of each object. After either the bin is empty or T total grasp attempts, the process terminates.

These variables evolve according to an environment distribution that reflects sensor noise, control imprecision, and variation in the initial bin state:

1) Initial state distribution. Let $p(\mathbf{x}_0)$ be a distribution over possible states of the environment that the robot is expected to handle due to variation in objects and tolerances in camera positioning.

2) Observation distribution. Let $p(\mathbf{y}_t|\mathbf{x}_t)$ be a distribution over observations given a state due to sensor noise and tolerances in the camera optical parameters.

3) Transition distribution. Let $p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t)$ be a distribution over next states given the current state and grasp action due to imprecision in control and physics.

The goal is to learn a policy π to maximize the rate of reward, or MPPH ρ , up to a maximum of T grasp attempts:

$$\rho(\pi) = \mathbb{E} \left[\left(\sum_{t=0}^{T-1} R_t \right) / \left(\sum_{t=0}^{T-1} \Delta_t \right) \right]$$

where T is the number of grasp attempts and Δ_t is the duration of executing grasp action \mathbf{u}_t in hours. The expectation is taken with respect to the environment distribution:

$$p(\mathbf{x}_0, \mathbf{y}_0, \dots, \mathbf{x}_T, \mathbf{y}_T|\pi) = p(\mathbf{x}_0) \prod_{t=0}^{T-1} p(\mathbf{y}_t|\mathbf{x}_t) p(\mathbf{x}_{t+1}|\mathbf{x}_t, \pi(\mathbf{y}_t))$$

It is common to measure performance in terms of the mean rate ν and reliability Φ (also known as the success rate) of a grasping policy for a given range of objects:

$$\nu = 1/\mathbb{E}[\Delta_t] \quad \Phi(\pi) = \mathbb{E} \left[\frac{1}{T} \sum_{t=0}^{T-1} R_t \right]$$

If the time per grasp is constant, the MPPH is the product of rate and reliability: $\rho = \nu\Phi$.

Learning from synthetic data

We propose a hybrid approach to ambidextrous grasping that learns policies on synthetic training datasets generated using analytic models

and domain randomization over a diverse range of objects, cameras, and parameters of physics for robust transfer from simulation to reality (17, 20, 22). The method optimizes for a policy to maximize MPPH under the assumption of a constant time per grasp: $\pi^* = \arg\max_{\pi} \Phi(\pi)$.

To learn a policy, the method uses a training dataset generation distribution based on models from physics and geometry, μ , to computationally synthesize a massive training dataset of point clouds, grasps, and reward labels for heterogeneous grippers. The distribution μ consists of two stochastic components: (i) a synthetic training environment $\xi(\mathbf{y}_0, R_0, \dots, \mathbf{y}_T, R_T|\pi)$ that can sample paired observations and rewards given a policy and (ii) a data collection policy $\tau(\mathbf{u}_t|\mathbf{x}_t, \mathbf{y}_t)$ that can sample a diverse set of grasps using full-state knowledge. The synthetic training environment simulates grasp outcomes by evaluating rewards according to the ability of a grasp to resist forces and torques due to gravity and random perturbations. The environment also stochastically samples heaps of 3D objects in a bin and renders depth images of the scene using domain randomization over the camera position, focal length, and optical center pixel. The dataset collection policy evaluates actions in the synthetic training environment using algorithmic supervision to guide learning toward successful grasps.

We explore large-scale supervised learning on samples from μ to train the ambidextrous policy π_{θ} across a set of two or more available grippers \mathcal{G} , as illustrated in Fig. 1. First, we sample a massive training dataset $\mathcal{D} = \{(R_i, \mathbf{y}_i, \mathbf{u}_i)\}_{i=1}^N$ from a software implementation of μ . Then, we learn a GC-CNN $Q_{\theta,g}(\mathbf{y}, \mathbf{u}) \in [0, 1]$ to estimate the probability of success for a grasp with gripper g given a depth image. Specifically, we optimize the weights θ_g to minimize the cross entropy loss \mathcal{L} between the GQ-CNN prediction and the true reward over the dataset \mathcal{D} :

$$\theta_g^* = \arg\min_{\theta_g \in \Theta} \sum_{(R_i, \mathbf{y}_i, \mathbf{u}_i) \in \mathcal{D}_g} \mathcal{L}(R_i, Q_{\theta_g}(\mathbf{y}_i, \mathbf{u}_i))$$

where \mathcal{D}_g denotes the subset of the training dataset \mathcal{D} containing only grasps for gripper g . We construct a robot policy π_{θ} from the GQ-CNNs by planning the grasp that maximizes quality across all available grippers:

$$\pi_{\theta}(\mathbf{y}_t) = \arg\max_g \left\{ \max_{\mathbf{u}_g \in \mathcal{U}_g} Q_{\theta,g}(\mathbf{y}_t, \mathbf{u}_g) \right\}$$

where \mathcal{U}_g is a set of candidate grasps for gripper g sampled from the depth image.

To evaluate the method, we learn the Dex-Net 4.0 ambidextrous policy on the Dex-Net 4.0 training dataset, which contains 5 million synthetic point clouds, grasps, and reward labels. Dex-Net 4.0 is generated from 5000 unique 3D object heaps with about 2.5 million data points each for a vacuum-based suction-cup gripper and a parallel-jaw gripper. Figure S1 analyzes the features learned by the Dex-Net 4.0 GQ-CNNs.

Physical experiments

We executed more than 2500 grasp attempts on a physical robot system with a parallel-jaw and suction-cup gripper to characterize the reliability of the Dex-Net 4.0 policy on a bin-picking task with 50 novel test objects. The experiments aimed to evaluate (i) the reliability and range of the Dex-Net 4.0 policy compared with a set of baselines; (ii) the effect of training dataset diversity, neural network architecture, and learning from real data; and (iii) the failure modes of the Dex-Net 4.0 policy.

To analyze performance, we selected a dataset of 50 objects with diverse shapes, sizes, and material properties. The dataset, described in the

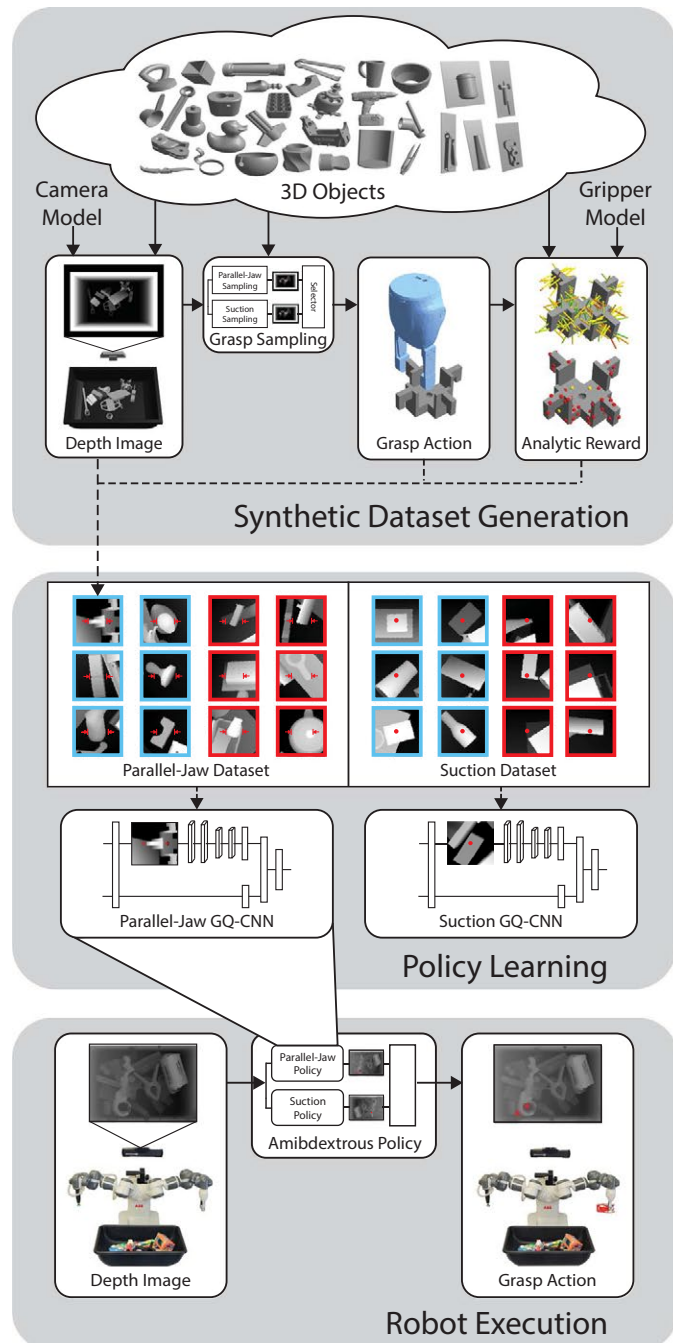


Fig. 1. Learning ambidextrous grasping policies for UP. (Top) Synthetic training datasets of depth images, grasps, and rewards are generated from a set of 3D computer-aided design (CAD) models using analytic models based on physics and domain randomization. Specifically, a data collection policy proposes actions given a simulated heap of objects, and the synthetic training environment evaluates rewards. Reward is computed consistently across grippers by considering the ability of a grasp to resist a given wrench (force and torque) based on the grasp wrench space, or the set of wrenches that the grasp can resist through contact. (Middle) For each gripper, a policy is trained by optimizing a deep GQ-CNN to predict the probability of grasp success given a point cloud over a large training dataset containing millions of synthetic examples from the training environment. Data points are labeled as successes (blue) or failures (red) according to the analytic reward metric. (Bottom) The ambidextrous policy is deployed on the real robot to select a gripper by maximizing grasp quality using a separate GQ-CNN for each gripper.

Supplementary Materials, includes retail products, groceries, tools, office supplies, toys, and 3D-printed industrial parts. We separated objects into two difficulty levels with 25 objects each, illustrated in Fig. 2:

1) Level 1. Prismatic and circular solids (e.g., rectangular prisms, spheres, and cylinders).

2) Level 2. Common household objects including examples with flat cardboard backing and clear plastic covers (“blisterpack”), varied geometry, and masses up to 500 g (the payload of the ABB YuMi).

For each trial, we placed all objects in the bin and allowed the robot to iteratively attempt grasps until either no objects remained or a maximum number of attempts were reached. Each grasp was planned on the basis of a depth image from an overhead 3D camera. For details on the experimental setup and procedure, see Materials and Methods. A video of each grasp attempt is available in the Supplementary Materials.

Comparison with baseline policies

We evaluated the Dex-Net 4.0 ambidextrous policy against three baselines in five independent trials. To compare with hand-coded methods used in practice, we implemented a best-effort suction-only policy and an ambidextrous policy based on geometric heuristics similar to those used in the Amazon Robotics Challenge (29, 30, 32). To study the importance of the consistent reward model, we also evaluated a policy that

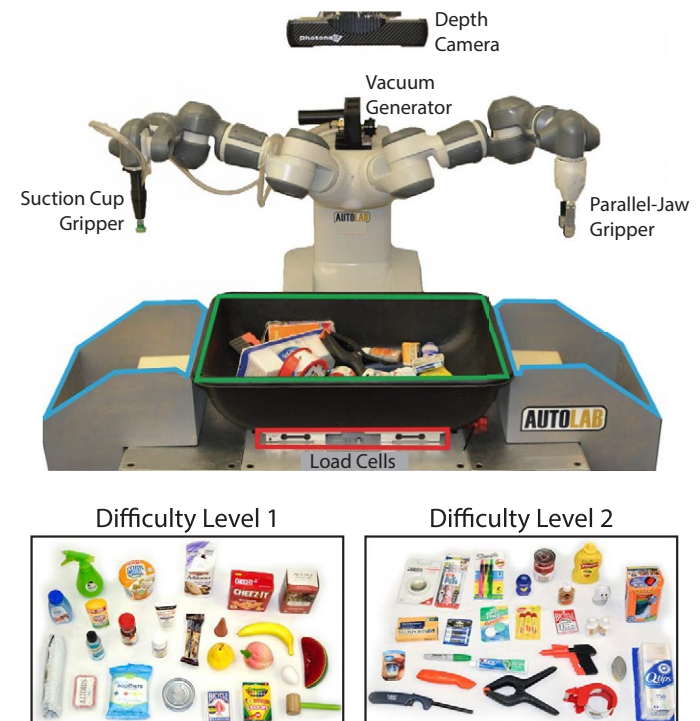


Fig. 2. Physical benchmark for evaluating UP policies. (Top) The robot plans a grasp to iteratively transport each object from the picking bin (green) to a receptacle (blue) using either a suction-cup or a parallel-jaw gripper. Grasp planning is based on 3D point clouds from an overhead Photoneo PhoXi S industrial depth camera. (Bottom) Performance is evaluated on two datasets of novel test objects not used in training. (Left-Bottom) Level 1 objects consist of prismatic and circular solids (e.g., boxes and cylinders) spanning groceries, toys, and medicine. (Right-Bottom) Level 2 objects are more challenging, including common objects with clear plastic and varied geometry, such as products with cardboard blisterpack packaging.

ranks grasps using Dex-Net 2.0 and 3.0 fine-tuned on simulated heaps with separate reward metrics for each gripper (see Materials and Methods for details).

Figure 3 shows the performance on the two object datasets. Dex-Net 4.0 achieves the highest success rate across all object datasets with a reliability of 97 and 95% on the level 1 and level 2 objects, respectively. The policy uses the suction cup gripper on 82% of grasps. The best baseline method has a reliability of 93 and 80%, respectively. Analysis of the number of objects picked versus the number of attempts suggests that the baseline methods take longer to clear the last few objects from the bin, sometimes failing to clear several of the objects.

We detail additional metrics for each policy in Table 1, including the reliability and MPPH of the learned quality functions. The Dex-Net 4.0 policy has the highest reliability on both level 1 and level 2 objects. The policy has a slightly lower MPPH than the suction heuristic on the level 1 objects because the heuristic can be evaluated more rapidly than the GQ-CNN.

We analyze the per-object reliability of each policy in fig. S2. The results suggest that differences in reliability between the policies on the level 1 objects may be due to specific configurations of objects (e.g., a thin object leaning against a wall of the bin) rather than the objects themselves. Figure S3 details the difficulty of each object according to the reliability across all policies. The most difficult objects were a box of Q-tips, a bottle of mustard, and the “bialetti,” an espresso filter in a thin blisterpack package.

To further quantify the range of the Dex-Net 4.0 ambidextrous policy, we measured the performance of grasping each of the 50 objects from level 1 and level 2 in isolation in the bin for five attempts each. Dex-Net 4.0 achieved 98% reliability versus 52 and 94% reliability for the Dex-Net 2.0 and 3.0 policies, respectively.

Performance with large heaps

To investigate whether heap size affects performance, we benchmarked the policy on a dataset of 50 test objects combining all objects from the level 1 and level 2 datasets. Figure 3 displays the results for five independent trials with each policy. Dex-Net 4.0 has the highest reliability at 90%. In comparison, the performance of the heuristics is relatively unchanged, with success rates near 80%. Some failures of Dex-Net 4.0 are due to attempts to lift objects from underneath others.

Effects of training dataset diversity

We quantified the importance of dataset diversity by training the GQ-CNNs on three alternative synthetic training datasets:

- 1) Fewer unique objects: 100 unique 3D objects in 2500 unique heaps.
- 2) Very few unique heaps: 1664 unique 3D objects in 100 unique heaps.
- 3) Fewer unique heaps: 1664 unique 3D objects in 500 unique heaps.

Figure 3 displays the performance on level 1 and level 2 objects. The policies have reduced reliability and appeared to be particularly sensitive to the number of unique heaps used in training.

Varying the neural network architecture

We studied whether changes to the neural network architecture affect the performance of the resulting policy by training on the Dex-Net 4.0 dataset using the “Improved GQ-CNN” architecture (34). As seen in Fig. 3, the architecture has comparable performance with the standard GQ-CNN architecture.

Training on physical grasp outcomes

We also explored whether performance can be improved by training on labeled grasp attempts on a physical system. We used a dataset of more than 13,000 labeled grasp attempts collected over 6 months of experiments and demonstrations of the system. About 5000 data points were labeled by human operators, and the remaining 8000 were labeled automatically using weight differences measured by load cells.

We trained 10 variants of the Dex-Net 4.0 policy on varying ratios of synthetic and real data using fine-tuning on the fully connected (FC) layers, including a model trained on only empirical data (see Materials

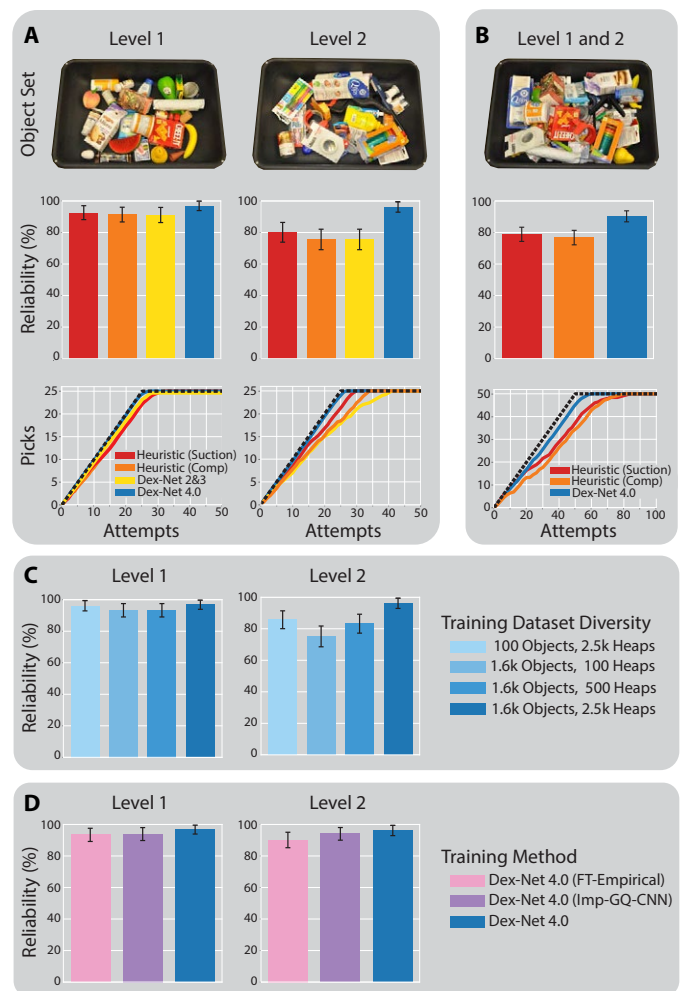


Fig. 3. Performance of the Dex-Net 4.0 ambidextrous policy on the bin picking benchmark. Error bars show the 95% confidence interval on reliability using the standard error of the mean (SEM). (A) Comparison with three baseline methods on level 1 and level 2 objects on heaps of 25 novel objects: (i) a hand-coded heuristic for the suction cup [Heuristic (suction)], (ii) a hand-coded heuristic for selecting between suction-cup and parallel-jaw grippers [Heuristic (comp)], and (iii) an ambidextrous policy fine-tuned on simulated heaps from the Dex-Net 2.0 and 3.0 base GQ-CNNs and reward metrics. For reference, the best possible performance (succeeding on every pick until the bin is cleared) is illustrated with a dashed-dotted black line. (B) Performance with large heaps of 50 novel objects. (C) Ablation study measuring the effect of training on less diverse datasets with either fewer unique heaps or fewer unique 3D object models. (D) Performance of two training alternatives: the improved GQ-CNN (Imp-GQ-CNN) architecture (34) and fine-tuning (FT) on 13,000 real data points.

Table 1. Detailed performance analysis of the Dex-Net 4.0 and baseline policies on the bin-picking benchmark for five trials on level 1 and level 2 datasets of 25 novel objects each. We report the reliability, MPPH, average precision (AP), total number of grasps attempts (minimum of 125), and total number of failures.

Policy	Level 1					Level 2				
	Reliability (%)	MPPH	AP (%)	No. of attempts	No. of failures	Reliability (%)	MPPH	AP (%)	No. of attempts	No. of failures
Heuristic (suction)	93	331	95	135	10	80	304	87	156	31
Heuristic (composite)	91	281	93	139	14	76	238	83	168	43
Dex-Net 2 and 3 composite	91	306	93	135	10	76	255	64	168	43
Dex-Net 4.0	97	309	100	129	4	95	312	99	131	6

and Methods for details). The best-performing empirically trained policy had comparable reliability with the original Dex-Net 4.0 policy on the physical benchmark, as shown in Fig. 3, and did not lead to substantial performance increases.

Adversarial objects

To probe the boundaries of the range of the Dex-Net 4.0 policy, we evaluated its performance on a third object dataset that contained 25 novel objects with few accessible and perceptible grasps due to adversarial geometry, transparency, specularly, and deformability. The results are illustrated in Fig. 4. Dex-Net 4.0 was still the highest performing policy, but the reliability was reduced to 63%.

Failures of the Dex-Net 4.0 policy often occur several times in sequence. To characterize these sequential failures, we explored a first-order memory-based policy to encourage exploration upon repeated failures, a technique that has been used to improve performance in the Amazon Robotics Challenge (32). The policy uses an instance recognition system to match object segments to previous failures in a database (12) and pushes objects to create accessible grasps when none are available (see Materials and Methods for details on the memory-based policy). The addition of memory increased the reliability to 80% at 242 MPPH.

DISCUSSION

Experiments suggest that ambidextrous policies trained on Dex-Net 4.0 achieve high reliability on novel objects on a physical robot, with more than 95% reliability on heaps of 25 novel objects at more than 300 MPPH. Dex-Net 4.0 outperforms hand-coded baselines similar to those used in applications such as the Amazon Robotics Challenge and also outperforms an ambidextrous policy based on previous versions of Dex-Net that use separate reward functions for each gripper. This suggests that learning with consistent reward functions across grippers can lead to increased reliability on a physical robot.

Experiments also suggest that performance is sensitive to several factors. Heaps containing more objects lead to decreased reliability because the policy attempts to lift objects that are occluded by others in the heap. Performance also depends on the diversity of the training dataset, with more diverse datasets leading to higher performance on a physical robot. Last, performance varies based on the test objects, with more complex geometries and material properties leading to reduced reliability. Use of a memory system can help compensate for repeated failures, increasing reliability on adversarial objects from 63 to 80%.

Benefits of ambidextrous grasping

The experimental results highlight the advantage of using a set of two or more heterogeneous grippers. Although a policy with only a single suction cup can achieve high reliability on the level 1 prismatic and circular objects, performance drops to 80% on the level 2 objects with more complex geometry. In comparison, the ambidextrous grasping policy uses the parallel jaws on 20% of grasp attempts to achieve 95% reliability on the level 2 objects. Furthermore, a consistent reward appears to be important for learning an ambidextrous policy to reliably decide between grippers. However, this study only considers a single combination of grippers. Future research could study applications to new grippers, such as two-finger underactuated hands or multi-suction-cup arrays. Future work could also consider extensions of ambidextrous grasping, such as simultaneous grasping with multiple arms or planning grasps for three or more grippers.

Physics-based reward design

The results of this paper also suggest that analytic quasi-static grasp quality metrics (35, 36) with domain randomization can be used as a computationally efficient reward function for learning ambidextrous grasping policies that are robust to sensor noise and imprecision. This stands in contrast to past research (5, 6) that has criticized quasi-static metrics for making strong assumptions and considering only a necessary, not sufficient, condition for dynamic grasp stability. Experiments suggest that the Dex-Net 4.0 policy generalizes to objects with deformable surfaces, moving parts, and transparencies that do not satisfy the assumptions of the analytic metrics. This may be because grasps with high analytic quality over a diverse range of 3D objects tend to correlate with grasp affordances: geometric features of objects that facilitate grasping, such as handles or flat suctionable surfaces. Further studies may be necessary to understand why grasps are often dynamically stable in practice. One hypothesis is that material compliance in the fingertips acts as a passive stability controller. Future research could investigate whether this result generalizes to additional grippers such as multifingered (20) or soft hands (37).

Bias-variance tradeoff in dataset collection

Experiments suggest that a policy fine-tuned on 13,000 examples collected from physical experiments does not substantially improve the Dex-Net 4.0 ambidextrous grasping policy trained on only synthetic data. This may appear counterintuitive, because the model used to generate synthetic training data cannot possibly model the exact behavior of the real-world system and therefore may induce bias (38). This may

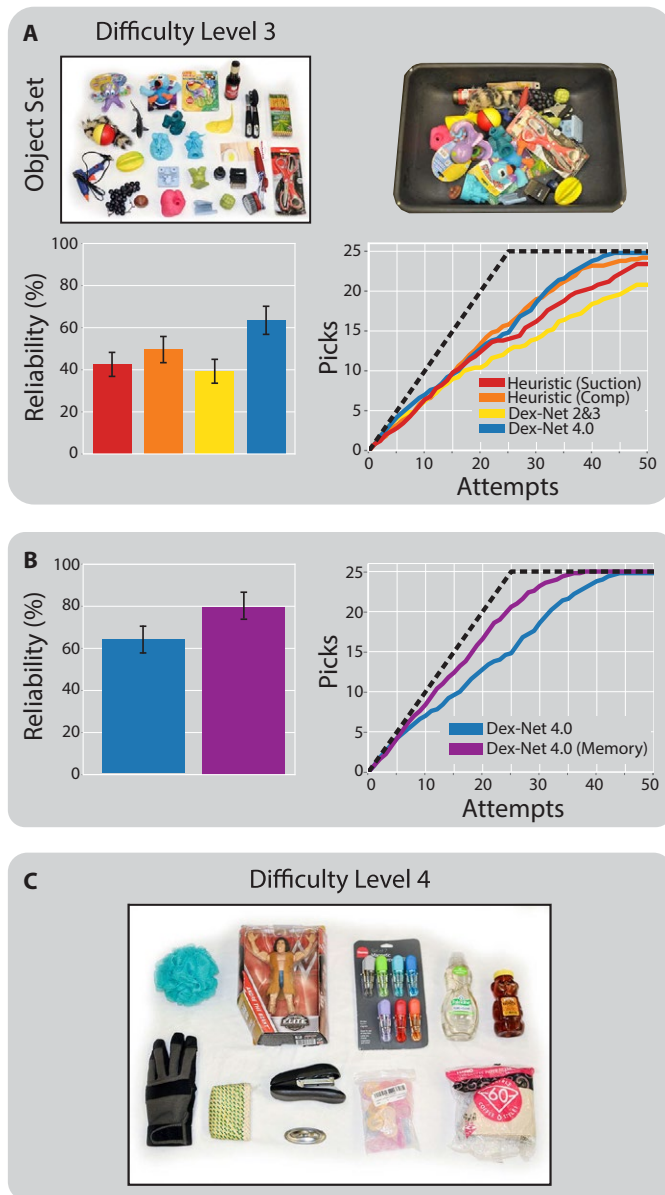


Fig. 4. Failure modes of the Dex-Net 4.0 policy. Error bars show the 95% confidence interval on reliability using the SEM. **(A)** Performance on level 3: a dataset of 25 novel objects with adversarial geometry and material properties. **(B)** Evaluation of a first-order memory-based policy for UP that masks regions of the point cloud with an instance recognition system to avoid repeated failures. **(C)** Pathological objects that cannot be grasped with Dex-Net 4.0 due to reflectance properties such as transparency, which affect depth sensing, and material properties such as porosity and deformability (e.g., loose packaging), which affect the ability to form a vacuum seal on the object surface.

relate to the bias-variance tradeoff in machine learning (39). Although the tradeoff is typically analyzed in terms of the function class, the results of this paper suggest that the training data distribution is also relevant. Using a biased model for rapid data collection may improve the scale and consistency of training datasets, leading to better performance on a physical system in comparison with methods based on smaller training datasets with high rates of mislabeled examples. Fu-

ture research could consider novel methods for learning with a combination of synthetic and real data, such as using analytic models to guide empirical data collection.

Sequential learning for UP

Finding a policy to maximize MPPH is inherently a sequential learning problem, in which grasp actions affect future states of the heap. Theory on imitation learning (40) and reinforcement learning (41) suggests that policies should take actions that lead to states with high expected future reward to guarantee high reliability. However, experiments in this paper suggest that the Dex-Net 4.0 policy performs well on the sequential task of bin picking, although it was trained with supervised learning to greedily maximize the probability of success for a single timestep. This suggests that performance is not particularly sensitive to the sequence of states of the object heap. This may be due to the random configuration of objects, which often have one or more exposed graspable surfaces in every state of the heap. Furthermore, performance may be increased on difficult objects by augmenting the policy with a memory system that avoids repeated mistakes.

Application to different sensors and grippers

The Dex-Net 4.0 method for training UP policies could be applied to other objects, cameras, and grippers by implementing a new dataset generation distribution and training a new GQ-CNN on samples from this distribution. For example, objects could be placed in structured configurations, such as packed in boxes or placed on shelves, and camera intrinsic parameters could be set to model a different sensor. However, the experiments in this paper are limited in scope. This study only evaluates performance on heaps of 50 unique, randomly arranged objects, which do not represent all possible object geometries. Furthermore, the hardware benchmark uses only one industrial high-resolution depth camera positioned directly overhead. The experiments only test a single parallel-jaw and vacuum-based suction cup gripper. Last, the constant time assumption that relates MPPH maximization to supervised learning may not be applicable to all robot picking systems. For example, there may be a time cost for switching grippers due to time spent physically mounting each tool. Future studies could evaluate performance in new applications with variations in objects, cameras, grippers, and robots.

Opportunities for future research

The most common failure modes of the policy are (i) attempting to grasp objects that are occluded due to overlap in the heap and (ii) repeated failures on objects with adversarial geometry and material properties. A subset of objects that cannot yet be reliably grasped with Dex-Net 4.0 is pictured in Fig. 4. One category includes objects imperceptible to a depth camera, such as those with transparent or specular surfaces. Another category is characterized by structured surface variations, such as parallel lines or buttons on a remote, which can trigger false positives in the suction network. Other classes include porous objects and objects with loose packaging.

Some failure modes could be addressed by increasing the diversity of objects in the training dataset or improving the dataset generation model. Rendering synthetic color images using domain randomization (17) could enable the system to grasp transparent, specular, or highly textured objects. Models of deformation and porosity could be used to reduce suction failures due to incorrect assumptions of the Dex-Net 4.0 model. The reward model could also be extended to compute the wrench set from all contacts between objects instead of only

considering the grippers and gravity, which could reduce failures due to object overlap.

Other extensions could substantially increase the reliability and range. The observed performance increase on the level 3 objects using a first-order memory system suggests that reinforcement learning could be used to reduce repeated failures. Furthermore, training on larger datasets of empirically collected data could reduce the simulation-to-reality gap. Another way to increase rate is to use feedback policies that actively regrasp dropped objects based on visual servoing (10, 28, 42), force sensing (43–45), or tactile sensing (46–49).

MATERIALS AND METHODS

Synthetic training environment

The Dex-Net 4.0 synthetic training environment is based on the following assumptions: (i) quasi-static physics (e.g., inertial terms are negligible) with Coulomb friction, (ii) objects are rigid and made of nonporous material, (iii) the robot has a single overhead depth sensor with known intrinsics, and (iv) the robot has two end effectors with known geometry—a vacuum-based gripper consisting of a single disc-shaped linear-elastic suction-cup and a parallel-jaw gripper (see the Supplementary Materials for detailed values of parameters). Dex-Net 4.0 uses the POMDP definition described in the following sections.

States

Let $\mathbf{x} = (\mathcal{O}_1, \dots, \mathcal{O}_m, \mathcal{C}, \mathbf{w}_1, \dots, \mathbf{w}_m)$ denote the state of the environment at time t , consisting of a single overhead depth camera, a set of objects, and a perturbation wrench on each object (e.g., gravity and disturbances). Each object state \mathcal{O}_i specifies the geometry \mathcal{M}_i , pose $\mathbf{T}_{o,i}$, friction coefficient γ_i , and center of mass \mathbf{z}_i . The camera state \mathcal{C} specifies the intrinsic parameters \mathcal{I} and pose \mathbf{T}_c . Each wrench \mathbf{w}_i is specified as a vector $\mathbf{w}_i \in \mathbb{R}^6$.

Grasp actions

Let $\mathbf{u}_s \in \mathcal{U}_s$ denote a suction grasp in 3D space defined by a suction gripper \mathcal{G}_s and a rigid pose of the gripper $\mathbf{T}_s = (\mathbf{R}_s, \mathbf{t}_s) \in \text{SE}(3)$, where the rotation $\mathbf{R}_s \in \text{SO}(3)$ defines the orientation of the suction tip and the translation $\mathbf{t}_s \in \mathbb{R}^3$ specifies the target location for the center of the suction disc. Let $\mathbf{u}_p \in \mathcal{U}_p$ denote a parallel-jaw grasp in 3D space defined by a parallel-jaw gripper \mathcal{G}_p and a rigid pose of the gripper $\mathbf{T}_p = (\mathbf{R}_p, \mathbf{t}_p) \in \text{SE}(3)$, where the rotation $\mathbf{R}_p \in \text{SO}(3)$ defines the grasp axis and approach direction and the translation $\mathbf{t}_p \in \mathbb{R}^3$ specifies the target center point of the jaws. The set of all possible grasps is $\mathcal{U} = \mathcal{U}_s \cup \mathcal{U}_p$.

Point clouds

Let $\mathbf{y} = \mathbb{R}_+^{H \times W}$ be a 2.5D point cloud represented as a depth image with height H and width W taken by a camera with known intrinsics (50).

State distribution

The initial state distribution $\xi(\mathbf{x}_0)$ is the product of distributions on (26):

- 1) Object count (m): Poisson distribution with mean λ truncated to $[1, 10]$.
- 2) Object heap (\mathcal{O}): Uniform distribution over 3D object models and the pose from which each model is dropped into the heap. Objects are sampled without replacement.
- 3) Depth camera (\mathcal{C}): Uniform distribution over the camera pose and intrinsic parameters.
- 4) Coulomb friction (γ): Truncated Gaussian constrained to $[0, 1]$.

The initial state is sampled by drawing an object count m , drawing a subset of m objects, dropping the objects one by one from a fixed

height above the bin, and running dynamic simulation with pybullet (51) until all objects have about zero velocity. The 3D object models are sampled from a dataset of 1664 3D objects models selected to reflect a broad range of products that are commonly encountered in applications such as warehousing, manufacturing, or home decluttering. The dataset was augmented with synthetic blisterpack meshes to reflect cardboard-backed products encountered in retail applications. Augmentation was performed by placing each source mesh in a quasi-static stable resting pose (52) on an infinite planar work surface and attaching a thin, flat segment to the mesh at the triangle(s) touching the work surface.

Observation distribution

Depth image observations are rendered using the open source Python library meshrender using randomization in the camera focal length and optical center pixel. No noise was added to the rendered images, because experiments used a high-resolution Photoneo PhoXi S industrial depth camera.

Reward distribution

Binary rewards occur when a quasi-static equilibrium is feasible between the grasp and an external wrench perturbation (e.g., due to gravity or inertia). Let $\mathcal{O}_i \in \mathbf{x}_t$ be an object contacted by the gripper when executing action \mathbf{u}_t . Then, we measure grasp success with a binary-valued metric $S(\mathbf{x}_t, \mathbf{u}_t) \in \{0, 1\}$ that evaluates the following conditions.

- 1) The gripper geometry in the pose specified by \mathbf{u}_t is collision free.
- 2) The gripper contacts exactly one object \mathcal{O}_i when executing the grasp parameterized by \mathbf{u}_t .
- 3) The grasp can resist a random disturbing force and torque (wrench) $\mathbf{w}_t = \mathbf{w}_g + \epsilon_w$ on the grasped object with more than 50% probability, where \mathbf{w}_g is the fixed wrench due to gravity and ϵ_w is a random wrench sampled from a zero-mean Gaussian $\mathcal{N}(0, \sigma_w^2 \mathbf{I})$.

Given an object consisting of a geometry \mathcal{M} in pose \mathbf{T}_o , the gripper g (geometry and physical parameters such as friction) and grasp pose \mathbf{T}_g are used to determine the contacts \mathbf{c} , or set of points and normals between the fingers and object. This set of contacts is used to compute the set of wrenches Λ that the grasp can apply to the object under quasi-static physics and a point contact model. Specifically, the wrench space for grasp \mathbf{u} using a contact model with m basis wrenches is $\Lambda(\mathbf{u}) = \{\mathbf{w} \in \mathbb{R}^6 | \mathbf{w} = G(\mathbf{u})\alpha \text{ for some } \alpha \in \mathcal{F}(\mathbf{u})\}$, as defined in (27). The grasp matrix $G(\mathbf{u}) \in \mathbb{R}^{6 \times m}$ is a set of basis wrenches in the object coordinate frame specifying the set of wrenches that the grasp can apply through contact via active (e.g., joint torques) and passive (e.g., inertia) means. The wrench constraint set $\mathcal{F}(\mathbf{u}) \subseteq \mathbb{R}^m$ limits contact wrench magnitudes based on the capabilities of the gripper (1). Last, the grasp wrench space is used to measure grasp reward based on wrench resistance, or the ability of the grasp to resist a perturbation wrench \mathbf{w} (e.g., due to gravity) as defined in (27). The grasp reward R is 1 if the probability of wrench resistance is greater than a threshold over M samples from the stochastic model.

Data collection policy

The dataset collection policy $\tau(\mathbf{u}_t | \mathbf{x}_t, \mathbf{y}_t)$ samples a mixture of actions from the point cloud and from an algorithmic supervisor $\Omega(\mathbf{x})$ that guides data toward successful grasps. Grasp actions are sampled from the point cloud using the sampling techniques of (19) and (27) to model the set of actions that the policy will evaluate when tested on real point clouds. Because this distribution may contain a very small percentage of successful actions, we sample actions with high expected reward from an algorithmic supervisor that evaluates grasps using full-state

information (26). The supervisor precomputes grasps on a set of known 3D objects in a database [such as in Dex-Net 1.0 (53)] that are robust to different possible orientations of each object. Because the state of each object in the heap is not known ahead of time, the supervisor estimates the probability of success, or quality, for each grasp over a large range of possible object orientations using the Monte Carlo grasp computation methods of Dex-Net 2.0 (19) and Dex-Net 3.0 (27). Given a full state of the heap, the supervisor computes the set of collision-free grasps with quality above a threshold for each object and then samples a grasp uniformly at random from the candidate set. Formally, the supervisor-guided data collection policy is

$$\tau(\mathbf{u}_t | \mathbf{x}_t, \mathbf{y}_t) = \begin{cases} \Omega(\mathbf{x}_t) & \text{with prob. } \varepsilon \\ \text{Unif}(\mathcal{U}_g(\mathbf{y}_t)) & \text{otherwise} \end{cases}$$

where $\mathcal{U}_g(\mathbf{y})$ is the set of candidate actions sampled from the point cloud with equal numbers of suction and parallel-jaw grasps. We use $\varepsilon = 1\%$ to favor actions sampled from the policy's own action space.

Training details

The Dex-Net 4.0 training dataset contains a large set of labeled actions for each point cloud to improve the computational efficiency of generating a single data point. Specifically, data points were generated using a one-time step Monte Carlo evaluation of reward for a large set of grasp actions on each unique object state. This leads to faster dataset collection and can eliminate the need for fine-tuning, which is prone to a phenomenon known as “catastrophic forgetting” that can lead to unpredictable failures of the grasping policy (54). Every sampled state from $\xi(\mathbf{x})$ has five associated depth images in Dex-Net 4.0, representing 3D point clouds captured from randomized camera poses and intrinsic optical parameters. Each image sampled from $\xi(\mathbf{y}|\mathbf{x})$ has a set of labeled actions for each gripper with associated quality metrics. The intrinsic parameters for the virtual cameras were sampled around the nominal values of a Photoneo PhoXi S industrial depth sensor. Images were converted to 96 pixel-by-96 pixel training thumbnails translated to move the grasp center to the thumbnail center pixel and rotated to align the grasp approach direction or axis with the middle row of pixels for the suction and parallel-jaw grippers, respectively.

The GQ-CNN architectures are similar to those used in Dex-Net 2.0 (19) and Dex-Net 3.0 (27) with two primary changes. First, we removed local response normalization because experiments suggest that it does not affect training performance. Second, we modified the sizes and pooling of the following layers: conv1_1 (16 9 by 9 filters, 1× pooling), conv1_2 (16 5 by 5 filters, 2× pooling), conv2_1 (16 5 by 5 filters, 1× pooling), conv2_2 (16 5 by 5 filters, 2× pooling), fc3 (128 output neurons), pc1 (16 output neurons), and fc4 (128 output neurons).

We trained each GQ-CNN using stochastic gradient descent with momentum for 50 epochs using an 80-20 training-to-validation image-wise split of the Dex-Net 4.0 dataset. We used a learning rate of 0.01 with an exponential decay of 0.95 every 0.5 epochs, a momentum term of 0.9, and an ℓ_2 weight regularization of 0.0005. We initialized the weights of the model by sampling from a zero-mean Gaussian with variance $\frac{2}{n_i}$, where n_i is the number of inputs to the i th network layer (55). To augment the dataset during training, we reflected each image about its vertical and horizontal axes and rotated each image by 180° because these lead to equivalent grasps. Training took about 24 hours on a single NVIDIA TITAN Xp graphics processing unit (GPU). The learned GQ-CNNs achieved 96 and 98% classification accuracy for the suction cup and parallel-jaw grippers, respectively, on the held-out validation set.

Implementation of policies

We used the trained GQ-CNNs to plan grasps from point clouds on a physical robot with derivative-free optimization to search for the highest-quality grasp across both grippers. The policy optimizes for the highest-quality grasp for each gripper separately, using the cross-entropy method (CEM) (10, 19, 27, 56), and then selects the grasps with the highest estimated quality across the grippers. To avoid grasping the bin, we constrained grasps to the foreground by subtracting out the background pixels of the bin using a reference depth image of an empty bin. Grasps were also constrained to be collision free with the bin to avoid damage to the robot. Given the constraints, CEM sampled a set of initial candidate grasps uniformly at random from a point cloud and then iteratively resampled grasps from a Gaussian mixture model fit to the grasps with the highest estimated quality. For the suction-cup gripper, initial candidate grasps were sampled by selecting a 3D point and choosing an approach direction along the inward-facing surface normal. For the parallel-jaw gripper, initial candidate grasps were sampled by finding antipodal point pairs using friction cone analysis.

Study design

The objective of the UP benchmark is to measure the rate, reliability, and range of the Dex-Net 4.0 policy in reference to baseline methods. The number of trials and objects used in the benchmark was chosen to maximize the number of unique grasp attempts and baseline policies that could be evaluated in a fixed time budget. The objects were divided into categories of 25 objects each based on difficulty to quantify the range of each policy. Rather than experiment on hundreds of unique objects, we used a reduced subset of 75 to evaluate a larger number of grasps for each object given a 3-week time budget for experiments. A grasp was considered successful if it lifted and transported exactly one object from the bin to the receptacle. Successes and failures were labeled by a human operator to avoid labeling errors due to hardware failures or sensor calibration.

Experimental hardware setup

The experimental workspace is illustrated in Fig. 2. The benchmark hardware system consists of an ABB YuMi bimanual industrial collaborative robot with an overhead Photoneo PhoXi S industrial 3D scanner, a custom suction gripper, and custom 3D-printed parallel-jaw fingers with silicone fingertips (57). The suction gripper consists of a 20-mm-diameter silicone single-bellow suction cup seated in a 3D-printed housing mounted to the end of the right arm. The vacuum was created by supplying compressed air from a JUN-AIR 18-40 quiet air compressor to a VacMotion MSV-35 vacuum generator. The payload of the suction system was about 0.9 kg with a vacuum flow of about 8 standard cubic feet/min. Objects were grasped from a bin mounted on top of a set of Loadstar load cells that measured the weight with a resolution of about 5 g. Each gripper has a separate receptacle to drop the objects into on the side of the bin.

Experimental procedure

Each experiment consisted of five independent trials in which the bin was filled with a random configuration of one or more objects and the robot attempted to pick each object from the bin and transport it to a receptacle. Before each experiment, the camera position and orientation relative to the robot were measured using a chessboard. In each trial, the operator set a full dataset of objects in the bin by shaking the objects in a box, placing the box upside down in the bin, and mixing the bin manually to ensure that objects rested below the rim of the bin. Then, the

robot iteratively attempted to pick objects from the bin. On each attempt, the grasping policy received as input a point cloud of the objects in the bin and returned a grasp action for exactly one of the grippers, consisting of a pose for the gripper relative to the base of the robot. Then, the ABB RAPID linear motion planner and controller were used to move to the target pose, establish contact with the object, and drop the object in the receptacle. The operator labeled the grasp as successful if the robot lifted and transported the object to the receptacle on the side of the workspace. The operator also labeled the identity of each grasped object. A trial was considered complete after all objects were removed, 75 total attempts, or 10 consecutive failures. All experiments ran on a desktop running Ubuntu 16.04 with a 3.4-GHz Intel Core i7-6700 quad-core central processing unit and an NVIDIA TITAN Xp GPU [see the Supplementary Materials for a characterization of variables in the benchmark (58)].

Description of baselines

We compared performance with three baselines:

1) Heuristic (suction). Ranked planar grasps based on the inverse distance to the centroid of an object (30), where the object centroid was estimated as the mean pixel of an object instance segmask from a Euclidean clustering segmentation algorithm from the Point Cloud Library (PCL) (59). Planarity was determined by evaluating the mean squared error (MSE) of all 3D points within a sphere with a radius of 10 mm (based on the suction cup size) to the best-fit plane for the points. Grasps were considered planar if either (i) the MSE was less than an absolute threshold or (ii) the MSE was within the top 5% of all candidate grasps. The hyperparameters were hand-coded to optimize performance on the physical robot.

2) Heuristic (composite). Ranked grasps planned with the suction heuristic above and a parallel-jaw heuristic based on antipodality. The parallel-jaw heuristic ranked antipodal grasps based on the inverse distance to the estimated centroid of an object, determining antipodality based on estimated point cloud surface normals. The height of the gripper above the bin surface was a constant offset from the highest point within the region of the grasp. The grasp closest to the estimated object centroid across both grippers was selected for execution.

3) Dex-Net 2.0 and 3.0 composite. Ranked grasps based on the estimated quality from separate GQ-CNNs trained to estimate the quality of parallel-jaw and suction-cup grasps in clutter. The GQ-CNNs were trained by fine-tuning the Dex-Net 2.0 and 3.0 base networks on simulated heaps with imitation learning from an algorithmic supervisor (26).

Details of empirical training

We train 10 variants of the Dex-Net 4.0 ambidextrous grasping policy on a dataset of 13,000 real grasp attempts: training from scratch and fine-tuning either all FC layers or only the last FC layer (fc5) on varying ratios of real to simulated data: 1:0, 1:1, 1:10, and 1:100. Each variant was evaluated on the adversarial level 3 objects on the physical robot, and the highest performing policy was the Dex-Net 4.0 policy with the last FC layer fine-tuned on the 1:10 combined real and synthetic dataset.

Details of memory system

To avoid repeated grasp failures, we implemented a first-order memory system that associates regions of the point cloud with past failures. A grasp was considered a failure if the weight reading from the load cells changed less than 5 g after a grasp attempt. When a failure occurred,

the point cloud was segmented using PCL (59), and the segment corresponding to the grasped object was associated with a region in a grayscale image. The segmented image patch was featurized using the VGG-16 network and stored in a failure database corresponding to the gripper. On the next grasp attempt, the point cloud segments were matched to the failure database using the VGG-16 featurization. If a match was found, the region in the current image was marked as forbidden to the grasp sampler for the Dex-Net 4.0 policy. Furthermore, if more than three consecutive failures occurred, then the memory system rejected the planned grasp and used a pushing policy (60) to perturb the objects in the bin.

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/26/eaau4984/DC1

Text

Fig. S1. Analysis of features learned by the GQ-CNNs from the ambidextrous grasping policy.

Fig. S2. Per-object reliability of each policy on each test object.

Fig. S3. Difficulty of each object from the test object datasets characterized by the overall reliability averaged across methods.

Movie S1. Summary.

Raw data, code for data analysis, videos, and listing of objects used in experiments (.zip file)

REFERENCES AND NOTES

1. R. M. Murray, Z. Li, S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation* (CRC Press, 1994).
2. D. Prattichizzo, J. C. Trinkle, *Springer Handbook of Robotics* (Springer, 2008), pp. 671–700.
3. B. Kehoe, A. Matsukawa, S. Candido, J. Kuffner, K. Goldberg, Cloud-based robot grasping with the google object recognition engine, in *2013 IEEE International Conference on Robotics and Automation* (IEEE, 2013), pp. 4263–4270.
4. J. Weisz, P. K. Allen, Pose error robust grasping from contact wrench space metrics, in *2012 IEEE International Conference on Robotics and Automation* (IEEE, 2012), pp. 557–562.
5. R. Balasubramanian, L. Xu, P. D. Brook, J. R. Smith, Y. Matsuoka, Physical human interactive guidance: Identifying grasping principles from human-planned grasps. *IEEE Trans. Robot.* **28**, 899–910 (2012).
6. J. Bohg, A. Morales, T. Asfour, D. Kragic, Data-driven grasp synthesis—A survey. *IEEE Trans. Robot.* **30**, 289–309 (2014).
7. D. Kappler, J. Bohg, S. Schaal, Leveraging big data for grasp planning, in *2015 IEEE International Conference on Robotics and Automation* (ICRA, 2015), pp. 4304–4311.
8. I. Lenz, H. Lee, A. Saxena, Deep learning for detecting robotic grasps. *Int. J. Robot. Res.* **34**, 705–724 (2015).
9. A. Saxena, J. Driemeyer, A. Y. Ng, Robotic grasping of novel objects using vision. *Int. J. Rob. Res.* **27**, 157–173 (2008).
10. S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, D. Quillen, Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. J. Rob. Res.* **37**, 421–436 (2018).
11. L. Pinto, A. Gupta, Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours, in *IEEE International Conference on Robotics and Automation* (ICRA, 2016), pp. 3406–3413.
12. A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, T. Funkhouser, Learning synergies between pushing and grasping with self-supervised deep reinforcement learning, in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (IROS, 2018).
13. D. Ma, A. Rodriguez, Friction variability in planar pushing data: Anisotropic friction and data-collection bias. *IEEE Robot. Autom. Lett.* **3**, 3232–3239 (2018).
14. S. James, A. J. Davison, E. Johns, Transferring end-to-end visuomotor control from simulation to real world for a multi-stage task, in *1st Conference on Robot Learning* (CoRL, 2017), pp. 334–343.
15. X. B. Peng, M. Andrychowicz, W. Zaremba, P. Abbeel, Sim-to-real transfer of robotic control with dynamics randomization, in *IEEE International Conference on Robotics and Automation* (ICRA) (IEEE, 2018), pp. 3803–3810.
16. A. A. Rusu, M. Večerík, T. Rothörl, N. Heess, R. Pascanu, R. Hadsell, Sim-to-real robot learning from pixels with progressive nets, in *1st Conference on Robot Learning* (CoRL, 2017), pp. 262–270.
17. 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 *IEEE/RSJ International Conference on Intelligent Robots and Systems* (IROS) (IEEE, 2017), pp. 23–30.

18. E. Johns, S. Leutenegger, A. J. Davison, Deep learning a grasp function for grasping under gripper pose uncertainty, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2016), pp. 4461–4468.
19. J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, K. Goldberg, Dex-Net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics, in *Proceedings of Robotics: Science and Systems* 10.15607/RSS.2017.XIII.058 (2017).
20. OpenAI, Learning dextrous in-hand manipulation. arXiv:1808.00177 [cs.LG] (1 August 2018).
21. A. ten Pas, M. Gualtieri, K. Saenko, R. Platt, Grasp pose detection in point clouds. *Int. J. Robot. Res.* **36**, 1455–1473 (2017).
22. F. Sadeghi, S. Levine, CAD2RL: Real single-image flight without a single real image, in *Proceedings of Robotics: Science and Systems* 10.15607/RSS.2017.XIII.034 (2017).
23. M. Danielczuk, M. Matl, S. Gupta, A. Li, A. Lee, J. Mahler, K. Goldberg, Segmenting unknown 3D objects from real depth images using mask R-CNN trained on synthetic point clouds. arXiv:1809.05825 [cs.CV] (16 September 2018).
24. D. Seita, N. Jamali, M. Laskey, R. Berenstein, A. K. Tanwani, P. Baskaran, S. Iba, J. Canny, K. Goldberg, Robot bed-making: Deep transfer learning using depth sensing of deformable fabric. arXiv:1809.09810 [cs.RO] (26 September 2018).
25. K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige, S. Levine, V. Vanhoucke, Using simulation and domain adaptation to improve efficiency of deep robotic grasping, in *IEEE International Conference on Robotics and Automation (ICRA)* (ICRA, 2018), pp. 4243–4250.
26. J. Mahler, K. Goldberg, Learning deep policies for robot bin picking by simulating robust grasping sequences, in *1st Conference on Robot Learning (CoRL)* (2017), pp. 515–524.
27. J. Mahler, M. Matl, X. Liu, A. Li, D. Gealy, K. Goldberg, Dex-Net 3.0: Computing robust vacuum suction grasp targets in point clouds using a new analytic model and deep learning, in *IEEE International Conference on Robotics and Automation (ICRA)* (ICRA, 2018), pp. 5620–5627.
28. U. Vierendeck, A. ten Pas, K. Saenko, R. Platt, Learning a visuomotor controller for real world robotic grasping using easily simulated depth images, in *1st Conference on Robot Learning (CoRL)* (2017), pp. 291–300.
29. C. Hernandez, M. Bharathesha, W. Ko, H. Gaiser, J. Tan, K. van Deurzen, M. de Vries, B. Van Mil, J. van Egmond, R. Burger, M. Morariu, J. Ju, X. Germann, R. Ensing, J. Van Frankenhuyzen, M. Wisse, Team Delft's robot winner of the amazon picking challenge 2016. arXiv:1610.05514 [cs.RO] (18 October 2016).
30. D. Morrison, A. W. Tow, M. McTaggart, R. Smith, N. Kelly-Boxall, S. Wade-McCue, J. Erskine, R. Grinover, A. Gurman, T. Hunn, D. Lee, A. Milan, T. Pham, G. Rallos, A. Razjigaev, T. Rowntree, K. Vijay, Z. Zhuang, C. Lehnert, I. Reid, P. Corke, J. Leitner, Cartman: The low-cost cartesian manipulator that won the amazon robotics challenge, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2018), pp. 7757–7764.
31. K.-T. Yu, N. Fazeli, N. Chavan-Dafle, O. Taylor, E. Donlon, G. D. Lankenau, A. Rodriguez, A summary of team MIT's approach to the amazon picking challenge 2015. arXiv:1604.03639 [cs.RO] (13 April 2016).
32. A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauzá, D. Ma, O. Taylor, M. Liu, E. Romo, N. Fazeli, F. Alet, N. Chavan-Dafle, R. Holladay, I. Morona, P. Q. Nair, D. Green, I. Taylor, W. Liu, T. A. Funkhouser, A. Rodriguez, Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2018), pp. 3750–3757.
33. L. P. Kaelbling, M. L. Littman, A. R. Cassandra, Planning and acting in partially observable stochastic domains. *Artif. Intell.* **101**, 99–134 (1998).
34. M. Jaśkowski, J. Świątkowski, M. Zajac, M. Klimek, J. Potiuk, P. Rybicki, P. Polatowski, P. Walczyk, K. Nowicki, M. Cygan, Improved GQ-CNN: Deep learning model for planning robust grasps. arXiv:1802.05992 [cs.LG] (16 February 2018).
35. R. Krug, Y. Bekiroglu, M. A. Roa, Grasp quality evaluation done right: How assumed contact force bounds affect wrench-based quality metrics, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017), pp. 1595–1600.
36. F. T. Pokorny, D. Kragic, Classical grasp quality evaluation: New algorithms and theory, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IIRS)* (IEEE, 2013), pp. 3493–3500.
37. R. Deimel, O. Brock, A novel type of compliant and underactuated robotic hand for dexterous grasping. *Int. J. Robot. Res.* **35**, 161–185 (2016).
38. A. Gupta, A. Murali, D. Gandhi, L. Pinto, Robot learning in homes: Improving generalization and reducing dataset bias. arXiv:1807.07049 [cs.RO] (18 July 2018).
39. J. Friedman, T. Hastie, R. Tibshirani, *The Elements of Statistical Learning*, vol. 1 (Springer Series in Statistics, Springer, 2001).
40. S. Ross, G. Gordon, D. Bagnell, A reduction of imitation learning and structured prediction to no-regret online learning, in *Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS)* (2011), pp. 627–635.
41. A. G. Barto, *Reinforcement Learning: An Introduction* (MIT Press, 1998).
42. D. Morrison, J. Leitner, P. Corke, Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach, in *Proceedings of Robotics: Science and Systems* 10.15607/RSS.2018.XIV.021 (2018).
43. F. R. Hogan, E. R. Grau, A. Rodriguez, Reactive planar manipulation with convex hybrid MPC, in *IEEE International Conference on Robotics and Automation (ICRA)* (ICRA, 2018), pp. 247–253.
44. L. Righetti, M. Kalakrishnan, P. Pastor, J. Binney, J. Kelly, R. C. Voorhies, G. S. Sukhatme, S. Schaal, An autonomous manipulation system based on force control and optimization. *Auton. Robots* **36**, 11–30 (2014).
45. M. R. Tremblay, M. R. Cutkosky, Estimating friction using incipient slip sensing during a manipulation task, in *Proceedings IEEE International Conference on Robotics and Automation (IEEE, 1993)*, pp. 429–434.
46. R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, S. Levine, More than a feeling: Learning to grasp and regrasp using vision and touch, in *IEEE Robotics and Automation Letters (IEEE, 2018)*, pp. 3300–3307.
47. F. R. Hogan, M. Bauza, O. Canal, E. Donlon, A. Rodriguez, Tactile regrasp: Grasp adjustments via simulated tactile transformations. arXiv:1803.01940 [cs.RO] (5 March 2018).
48. R. D. Howe, Tactile sensing and control of robotic manipulation. *Adv. Robot.* **8**, 245–261 (1993).
49. A. Molchanov, O. Kroemer, Z. Su, G. S. Sukhatme, Contact localization on grasped objects using tactile sensing, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2016), pp. 216–222.
50. R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision* (Cambridge Univ. Press, 2003).
51. E. Coumans, Bullet physics library (2013); bulletphysics.org.
52. K. Goldberg, B. V. Mirtich, Y. Zhuang, J. Craig, B. R. Carlisle, J. Canny, Part pose statistics: Estimators and experiments. *IEEE Trans. Rob. Autom.* **15**, 849–857 (1999).
53. J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, K. Goldberg, Dex-Net 1.0: A cloud-based network of 3D objects for robust grasp planning using a multi-armed bandit model with correlated rewards, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2016), pp. 1957–1964.
54. J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, R. Hadsell, Overcoming catastrophic forgetting in neural networks. *Proc. Natl. Acad. Sci. U.S.A.* **13**, 3521–3526 (2017).
55. K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in *IEEE International Conference on Computer Vision (ICCV)* (IEEE, 2015), pp. 1026–1034.
56. R. Y. Rubinstein, A. Ridder, R. Vaisman, *Fast Sequential Monte Carlo Methods for Counting and Optimization* (John Wiley & Sons, 2013).
57. M. Guo, D. V. Gealy, J. Liang, J. Mahler, A. Goncalves, S. McKinley, J. A. Ojea, K. Goldberg, Design of parallel-jaw gripper tip surfaces for robust grasping, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017), pp. 2831–2838.
58. J. Mahler, R. Platt, A. Rodriguez, M. Ciocarlie, A. Dollar, R. Detry, M. A. Roa, H. Yanco, A. Norton, J. Falco, K. van Wyk, E. Messina, J. J. Leitner, D. Morrison, M. Mason, O. Brock, L. Odhner, A. Kurenkov, M. Matl, K. Goldberg, Guest editorial open discussion of robot grasping benchmarks, protocols, and metrics. *IEEE Trans. Autom. Sci. Eng.* **15**, 1440–1442 (2018).
59. R. B. Rusu, S. Cousins, 3D is here: Point Cloud Library (PCL), in *IEEE International Conference on Robotics and Automation (IEEE, 2011)*, pp. 1–4.
60. M. Danielczuk, J. Mahler, C. Correa, K. Goldberg, Linear push policies to increase grasp access for robot bin picking, in *IEEE 14th International Conference on Automation Science and Engineering (CASE)* (IEEE, 2018), pp. 1249–1256.

Acknowledgments: This research was performed at the AUTOLAB at UC Berkeley in affiliation with the Berkeley AI Research (BAIR) Lab, the Real-Time Intelligent Secure Execution (RISE) Lab, and the CITRIS “People and Robots” (CPAR) Initiative. We thank those who aided in experiments: L. Amladi, C. Correa, S. Dolasia, D. Gealy, and M. Guo. We also thank our colleagues who provided helpful feedback, code, and suggestions, particularly R. Bajcsy, O. Brock, M. Laskey, S. Krishnan, L. Manuelli, J. A. Ojea, P. Puchwein, A. Rodriguez, and D. Seita.

Funding: We were supported, in part, by donations from Siemens, Google, Amazon Robotics, Toyota Research Institute, Autodesk, ABB, Samsung, Knapp, Loccioni, Honda, Intel, Comcast, Cisco, and Hewlett-Packard; by equipment grants from Photoneo and NVIDIA, by the U.S. Department of Defense (DOD) through the National Defense Science and Engineering Graduate Fellowship (NDSEG) Program; and by the Scalable Collaborative Human-Robot Learning (SCHoRL) Project, NSF National Robotics Initiative Award 1734633. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsors. **Author contributions:** J.M. devised the training method, designed the experiments, and wrote the manuscript. M.M. helped implement physics simulations, implemented the memory system, conducted

experiments, and edited the manuscript. V.S. implemented the neural network training, conducted experiments, and edited the manuscript. M.D. helped implement physics simulations, implemented the pushing policy used in the memory system, conducted experiments, and edited the manuscript. B.D. designed the experimental object datasets, conducted experiments, and edited the manuscript. S.M. designed and constructed the robotic picking cell used in experiments, conducted experiments, and edited the manuscript. K.G. supervised the project, advised the design and experiments, and edited the manuscript. **Competing interests:** J.M., M.M., S.M., and K.G. have equity in Ambidextrous Robotics Inc. J.M., M.M., and K.G. are co-inventors on a patent application (no. PCT/US18/26122) related to this work. **Data and materials availability:**

All data needed to evaluate the conclusions in the paper are present in the paper or the Supplementary Materials.

Submitted 4 September 2018

Accepted 12 December 2018

Published 16 January 2019

10.1126/scirobotics.aau4984

Citation: J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, K. Goldberg, Learning ambidextrous robot grasping policies. *Sci. Robot.* **4**, eaau4984 (2019).

ARTIFICIAL INTELLIGENCE

Learning agile and dynamic motor skills for legged robots

Jemin Hwangbo^{1*}, Joonho Lee¹, Alexey Dosovitskiy², Dario Bellicoso¹, Vassilios Tsounis¹, Vladlen Koltun³, Marco Hutter¹

Copyright © 2019
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

Legged robots pose one of the greatest challenges in robotics. Dynamic and agile maneuvers of animals cannot be imitated by existing methods that are crafted by humans. A compelling alternative is reinforcement learning, which requires minimal craftsmanship and promotes the natural evolution of a control policy. However, so far, reinforcement learning research for legged robots is mainly limited to simulation, and only few and comparably simple examples have been deployed on real systems. The primary reason is that training with real robots, particularly with dynamically balancing systems, is complicated and expensive. In the present work, we introduce a method for training a neural network policy in simulation and transferring it to a state-of-the-art legged system, thereby leveraging fast, automated, and cost-effective data generation schemes. The approach is applied to the ANYmal robot, a sophisticated medium-dog-sized quadrupedal system. Using policies trained in simulation, the quadrupedal machine achieves locomotion skills that go beyond what had been achieved with prior methods: ANYmal is capable of precisely and energy-efficiently following high-level body velocity commands, running faster than before, and recovering from falling even in complex configurations.

INTRODUCTION

Legged robotic systems are attractive alternatives to tracked/wheeled robots for applications with rough terrain and complex cluttered environments. The freedom to choose contact points with the environment enables them to overcome obstacles comparable to their leg length. With such capabilities, legged robots may one day rescue people in forests and mountains, climb stairs to carry payloads in construction sites, inspect unstructured underground tunnels, and explore other planets. Legged systems have the potential to perform any physical activity humans and animals are capable of.

A variety of legged systems are being developed in the effort to take us closer to this vision of the future. Boston Dynamics introduced a series of robots equipped with hydraulic actuators (1, 2). These have advantages in operation because they are powered by conventional fuel with high energy density. However, systems of this type cannot be scaled down (usually >40 kg) and generate smoke and noise, limiting them to outdoor environments. Another family of legged systems is equipped with electric actuators, which are better suited to indoor environments. Massachusetts Institute of Technology's (MIT) Cheetah (3) is one of the most promising legged systems of this kind. It is a fast, efficient, and powerful quadrupedal robot designed with advanced actuation technology. However, it is a research platform optimized mainly for speed and has not been thoroughly evaluated with respect to battery life, turning capability, mechanical robustness, and outdoor applicability. Boston Dynamics' newly introduced robot, SpotMini, is also driven by electric actuators and is designed for both indoor and outdoor applications. Although details have not been disclosed, public demonstrations and media releases (4) are convincing evidence of its applicability to real-world operation. The platform used in this work, ANYmal (5), is another promising quadrupedal robot powered by electric actuators. Its bioinspired actuator design makes it robust against impact while allowing accurate torque measurement at the joints. However, the complicated actuator design increases cost and compromises the power output of the robot.

Designing control algorithms for these hardware platforms remains exceptionally challenging. From the control perspective, these robots are high-dimensional and nonsmooth systems with many physical constraints. The contact points change over the course of time and depending on the maneuver being executed and, therefore, cannot be prespecified. Analytical models of the robots are often inaccurate and cause uncertainties in the dynamics. A complex sensor suite and multiple layers of software bring noise and delays to information transfer. Conventional control theories are often insufficient to deal with these problems effectively. Specialized control methods developed to tackle this complex problem typically require a lengthy design process and arduous parameter tuning.

The most popular approach to controlling physical legged systems is modular controller design. This method breaks the control problem down into smaller submodules that are largely decoupled and therefore easier to manage. Each module is based on template dynamics (6) or heuristics and generates reference values for the next module. For example, some popular approaches (7–10) use a template-dynamics-based control module that approximates the robot as a point mass with a massless limb to compute the next foothold position. Given the foothold positions, the next module computes a parameterized trajectory for the foot to follow. The last module tracks the trajectory with a simple proportional-integral-derivative (PID) controller. Because the outputs of these modules are physical quantities, such as body height or foot trajectory, each module can be individually hand-tuned. Approaches of this type have achieved impressive results. Kalakrishnan *et al.* (11) demonstrated robust locomotion over challenging terrain with a quadrupedal robot: To date, this remains the state of the art for rough-terrain locomotion. Recently, Bellicoso *et al.* (12) demonstrated dynamic gaits, smooth transitions between them, and agile outdoor locomotion with a similar controller design. Yet, despite their attractive properties, modular designs have limitations. First, limited detail in the modeling constrains the model's accuracy. This inherent drawback is typically mitigated by limiting the operational state domain of each module to a small region where the approximations are valid. In practice, such constraints lead to substantial compromises in performance, such as slow acceleration, fixed upright pose of

¹Robotic Systems Lab, ETH Zurich, Zurich, Switzerland. ²Intelligent Systems Lab, Intel, Munich, Germany. ³Intelligent Systems Lab, Intel, Santa Clara, CA, USA.
*Corresponding author. Email: jhwangbo@ethz.ch

the body, and limited velocity of the limbs. Second, the design of modular controllers is extremely laborious. Highly trained engineers spend months to develop a controller and to arduously hand-tune the control parameters per module for every new robot or even for every new maneuver. For example, running and climbing controllers of this kind can have markedly different architectures and are designed and tuned separately.

More recently, trajectory optimization approaches have been proposed to mitigate the aforementioned problems. In these methods, the controller is separated into two modules: planning and tracking. The planning module uses rigid-body dynamics and numerical optimization to compute an optimal path that the robot should follow to reach the desired goal. The tracking module is then used to follow the path. In general, trajectory optimization for a complex rigid-body model with many unspecified contact points is beyond the capabilities of current optimization techniques. Therefore, in practice, a series of approximations are used to reduce complexity. Some methods approximate the contact dynamics to be smooth (13, 14), making the dynamics differentiable. Notably, Neunert *et al.* (13) demonstrated that such methods can be used to control a physical quadrupedal robot. Other methods (15) prespecify the contact timings and solve for sections of trajectories where the dynamics remain smooth. A few methods aim to solve this problem with little to no approximation (16, 17). These methods can discover a gait pattern (i.e., contact sequence) with hard contact models and have demonstrated automatic motion generation for two-dimensional (2D) robotic systems, but, like any other trajectory optimization approach, the possible contact points are specified a priori. Although more automated than modular designs, the existing optimization methods perform worse than state-of-the-art modular controllers. The primary issue is that numerical trajectory optimization remains challenging, requires tuning, and in many cases, can produce suboptimal solutions. Besides, optimization has to be performed at execution time on the robot, making these methods computationally expensive. This problem is often solved by reducing precision or running the optimization on a powerful external machine, but both solutions introduce their own limitations. Furthermore, the system still consists of two independent modules that do not adapt to each other's performance characteristics. This necessitates hand-tuning of the tracker; yet, accurately tracking fast motion by an underactuated system with many unexpected contacts is nearly impossible.

Data-driven methods, such as reinforcement learning (RL), promise to overcome the limitations of prior model-based approaches by learning effective controllers directly from experience. The idea of RL is to collect data by trial and error and automatically tune the controller to optimize the given cost (or reward) function representing the task. This process is fully automated and can optimize the controller end to end, from sensor readings to low-level control signals, thereby allowing for highly agile and efficient controllers. On the down side, RL typically requires prohibitively long interaction with the system to learn complex skills—typically weeks or months of real-time execution (18). Moreover, over the course of training, the controller may exhibit sudden and chaotic behavior, leading to logistical complications and safety concerns. Direct application of learning methods to physical legged systems is therefore complicated and has only been demonstrated on relatively simple and stable platforms (19) or in a limited context (20).

Because of the difficulties of training on physical systems, most advanced applications of RL to legged locomotion are restricted to

simulation. Recent innovations in RL make it possible to train locomotion policies for complex legged models. Levine and Koltun (21) combined learning and trajectory optimization to train a locomotion controller for a simulated 2D walker. Schulman *et al.* (22) trained a locomotion policy for a similar 2D walker with an actor-critic method. More recent work obtained full 3D locomotion policies (23–26). In these papers, animated characters achieve remarkable motor skills in simulation.

Given the achievements of RL in simulated environments, a natural question is whether these learned policies can be deployed on physical systems. Unfortunately, such simulation-to-reality transfer is hindered by the reality gap—the discrepancy between simulation and the real system in terms of dynamics and perception. There are two general approaches to bridging the reality gap. The first is to improve simulation fidelity either analytically or in a data-driven way; the latter is also known as system identification (27–32). The second approach is to accept the imperfections of simulation and aim to make the controller robust to variations in system properties, thereby allowing for better transfer. This robustness can be achieved by randomizing various aspects of the simulation: using a stochastic policy (33), randomizing the dynamics (34–37), adding noise to the observations (38), and perturbing the system with random disturbances. Both approaches lead to improved transfer; however, the former is cumbersome and often impossible, and the latter can compromise the performance of the policy. Therefore, in practice, both are typically used in conjunction. For instance, the recent work of Tan *et al.* (35) demonstrated successful sim-to-real transfer of locomotion policies on a quadrupedal system called Minitaur via the use of an accurate analytical actuator model and dynamic randomization. Although it achieved impressive results, the method of Tan *et al.* (35) crucially depended on accurate analytical modeling of the actuators, which is possible for direct-drive actuators (as used in Minitaur) but not for more complex actuators, such as servomotors, series elastic actuators (SEAs), and hydraulic cylinders, which are commonly used in larger legged systems.

In this work, we developed a practical methodology for autonomously learning and transferring agile and dynamic motor skills for complex and large legged systems, such as the ANYmal robot (5). Compared with the robots used in (35), ANYmal has a much larger leg length relative to footprint, making it more dynamic, less statically stable, and therefore more difficult to control. In addition, it features 12 SEAs, which are difficult to control and for which sufficiently accurate analytical models do not exist. Gehring *et al.* (39) have attempted analytical modeling of an SEA, but, as we will show, their model is insufficiently accurate for training a high-performance locomotion controller.

Our approach is summarized in Fig. 1. Our key insight on the simulation side is that efficiency and realism can be achieved by combining classical models representing well-known articulated system and contact dynamics with learning methods that can handle complex actuation (Fig. 1, steps 1 and 2). The rigid links of ANYmal, connected through high-quality ball bearings, closely resemble an idealized multibody system that can be modeled with well-known physical principles (40). However, this analytical model does not include the set of mechanisms that map the actuator commands to the generalized forces acting on the rigid-body system: the actuator dynamics, the delays in control signals introduced by multiple hardware and software layers, the low-level controller dynamics, and compliance/damping at the joints. Because these mechanisms are nearly impossible

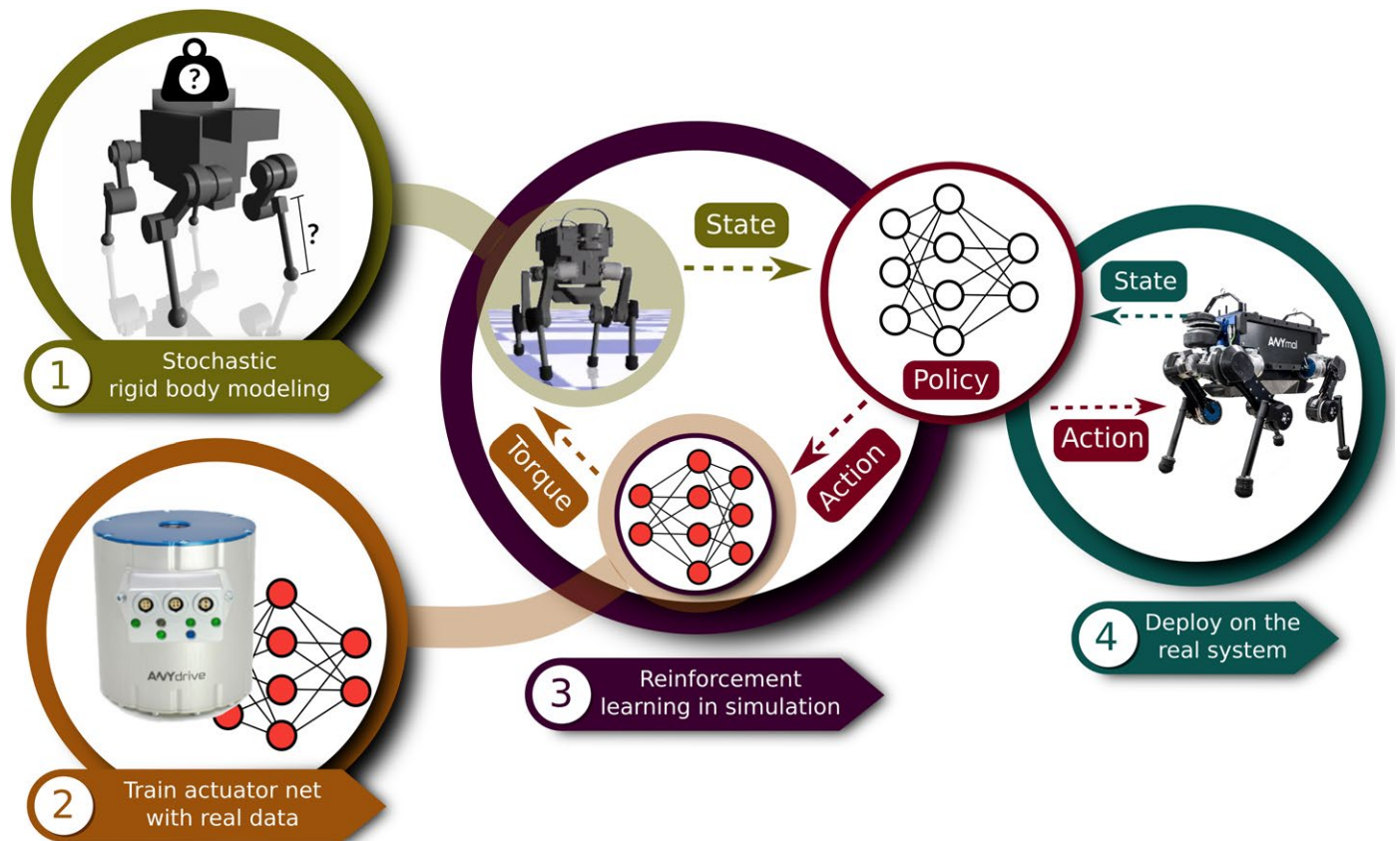


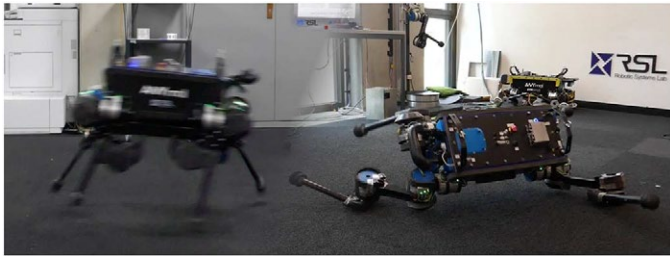
Fig. 1. Creating a control policy. In the first step, we identify the physical parameters of the robot and estimate uncertainties in the identification. In the second step, we train an actuator net that models complex actuator/software dynamics. In the third step, we train a control policy using the models produced in the first two steps. In the fourth step, we deploy the trained policy directly on the physical system.

to model accurately, we learned the corresponding mapping in an end-to-end manner—from commanded actions to the resulting torques—with a deep network. We learned this “actuator net” on the physical system via self-supervised learning and used it in the simulation loop to model each of the 12 joints of ANYmal. Crucially, the full hybrid simulator, including a rigid-body simulation and the actuator nets, runs at nearly 500,000 time steps per second, which allows the simulation to run roughly a thousand times faster than real time. About half of the run time was used to evaluate the actuator nets, and the remaining computations were efficiently performed via our in-house simulator, which exploits the fast contact solver of Hwangbo *et al.* (41), efficient recursive algorithms for computing dynamic properties of articulated systems (composite rigid-body algorithm and recursive Newton-Euler algorithm) (40), and a fast collision-detection library (42). Thanks to efficient software implementations, we did not need any special computing hardware, such as powerful servers with multiple central processing units (CPUs) and graphics processing units (GPUs), for training. All training sessions presented in this paper were done on a personal computer with one CPU and one GPU, and none lasted more than 11 hours.

We used the hybrid simulator for training controllers via RL (Fig. 1, step 3). The controller is represented by a multilayer perceptron (MLP) that took as input the history of the robot’s states and produced as output the joint position target. Specifying different reward functions for RL yielded controllers for different tasks of interest.

The trained controller was then directly deployed on the physical system (Fig. 1, step 4). Unlike the existing model-based control approaches, our proposed method is computationally efficient at run time. Inference of the simple network used in this work took 25 μ s on a single CPU thread, which corresponds to about 0.1% of the available onboard computational resources on the robot used in the experiments. This is in contrast to model-based control approaches that often require an external computer to operate at sufficient frequency (13, 15). Also, by simply swapping the network parameter set, the learned controller manifested vastly different behaviors. Although these behaviors were trained separately, they share the same code base: Only the high-level task description changed depending on the behavior. In contrast, most of the existing controllers are task-specific and have to be developed nearly from scratch for every new maneuver.

We applied the presented methodology to learning several complex motor skills that were deployed on the physical quadruped. First, the controller enabled the ANYmal robot to follow base velocity commands more accurately and energy-efficiently than the best previously existing controller running on the same hardware. Second, the controller made the robot run faster, breaking the previous speed record of ANYmal by 25%. The controller could operate at the limits of the hardware and push performance to the maximum. Third, we learned a controller for dynamic recovery from a fall. This maneuver is exceptionally challenging for existing methods because it involves multiple



Movie 1. Summary of the results and the method.

unspecified internal and external contacts. It requires fine coordination of actions across all limbs and must use momentum to dynamically flip the robot. To the best of our knowledge, such recovery skill has not been achieved on a quadruped of comparable complexity.

RESULTS

Movie 1 summarizes the results and the method of this work. In the following subsections, we describe the results in detail.

Command-conditioned locomotion

In most practical scenarios, the motion of a robot is guided by high-level navigation commands, such as the desired direction and the speed of motion. These commands can be provided, for instance, by an upper-level planning algorithm or by a user via teleoperation. Using our method, we trained a locomotion policy that could follow such commands at runtime, adapting the gait as needed, with no prior knowledge of command sequence and timing. A command consists of three components: forward velocity, lateral velocity, and yaw rate.

We first qualitatively evaluated this learned locomotion policy by giving random commands using a joystick. In addition, the robot was disturbed during the experiment by multiple external pushes to the main body. The resulting behavior is shown in movie S1. The video shows about 40 s of robust command following. We also tested the policy for 5 min without a single failure, which manifests the robustness of the learned policy.

The trained policy performed stably within the command distribution that it was trained on, with any random combination of the command velocities. Although the forward command velocity was sampled from $U(-1, 1)$ m/s during training, the trained policy reached 1.2 m/s of measured forward velocity reliably when the forward command velocity was set slightly higher (1.23 m/s), and the other command velocities were set to zero.

Next, we quantitatively evaluated this learned locomotion policy by driving the robot with randomly sampled commands. The commands were sampled as described in section S2. The robot received a new command every 2 s, and the command was held constant in between. The test was performed for 30 s, and a total of 15 random transitions were performed, including the initial transition from zero velocity. The base velocity plot is shown in fig. S1. The average linear velocity error was 0.143 m/s, and the average yaw rate error was 0.174 rad/s.

We next compared the learned controller with the best-performing existing locomotion controller available for ANYmal (12). For this experiment, we used a flying trot gait pattern (trot with full flight phase) because this is the only gait that stably reached 1.0 m/s forward velocity. We used the same velocity command profile, which resulted in the base velocity shown in fig. S2. The average linear velocity error

was 0.231 m/s, and the average yaw rate error was 0.278 rad/s. Given the same command profile, the tracking error of the model-based controller is about 95% higher than our learned controller with respect to linear velocity and about 60% higher with respect to yaw rate. In addition, our learned controller used less torque (8.23 N·m versus 11.7 N·m) and less mechanical power (78.1 W versus 97.3 W) on average. Movie S2 illustrates the experiments for both the learned policy and the model-based policy.

The control performance was also evaluated and compared in forward running. To this end, we sent a step input of four different speed commands (0.25, 0.5, 0.75, and 1.0 m/s) for 4.5 s each. The results, including a comparison to the prior method (12), are shown in Fig. 2. Figure 2A shows the flying trot pattern discovered by the learned controller. Note that this flight phase disappeared for low-velocity commands, and ANYmal displayed walking trot as shown in movie S1. Even without specifying the gait pattern, the learned policy manifested trot, a gait pattern that is commonly observed in quadrupedal animals. Figure 2B shows the velocity tracking accuracy of the policy both in simulation and on the real robot. Note that the oscillation of the observed velocity around the commanded one is a well-known phenomenon in legged systems, including humans (43). In terms of average velocity, the learned policy has an error of 2.2% on the real robot, 1.1% higher than in a simulation.

Figure 2 (C to E) compares the performance of the learned controller with the approach of Bellicoso *et al.* (12) in terms of accuracy and efficiency. We used two gaits from (12) for the comparison: flying trot, the only gait that can achieve 1 m/s, and dynamic lateral walk, the most efficient gait. First, we compared the velocity error at various commanded velocities in Fig. 2C. The learned controller is more accurate than the prior controller for all commanded velocities: by a factor of 1.5 to 2.5 compared with the dynamic lateral walk and by a factor of 5 to 7 compared with the flying trot, depending on the speed. Figure 2D shows the mechanical power output as a function of the measured velocity. The learned controller performed similarly to the dynamic lateral walk and more efficiently than the flying trot by a factor of 1.2 to 2.5, depending on the speed. Last, Fig. 2E plots the average measured torque magnitude against the measured velocity. The learned controller is more efficient in this respect than both prior gaits, using 23 to 36% less torque depending on the velocity. This large improvement in efficiency is possible because the learned controller walks with a nominal knee posture that is 10° to 15° straighter than prior gaits. The nominal posture cannot be adjusted to this level in the approach of Bellicoso *et al.* because this would markedly increase the rate of failure (falling).

Next, we compared our method with ablated alternatives: training with an ideal actuator model and training with an analytical actuator model. The ideal actuator model assumes that all controllers and hardware inside the actuator have infinite bandwidth and zero latency. The analytical model uses the actual controller code running on the actuator in conjunction with identified dynamic parameters from experiments and computer-aided design (CAD) tools. Some parameters, such as latency, damping, and friction, were hand-tuned to increase the accuracy of predicted torque in relation to data obtained from experiments. The policy training procedure for each method was identical to ours.

Neither alternative method could make a single step without falling. The resulting motions are shown in movies S3 and S4. We observed violent shaking of the limbs, probably due to not accounting for various delays properly. Although the analytical model contained multiple delay

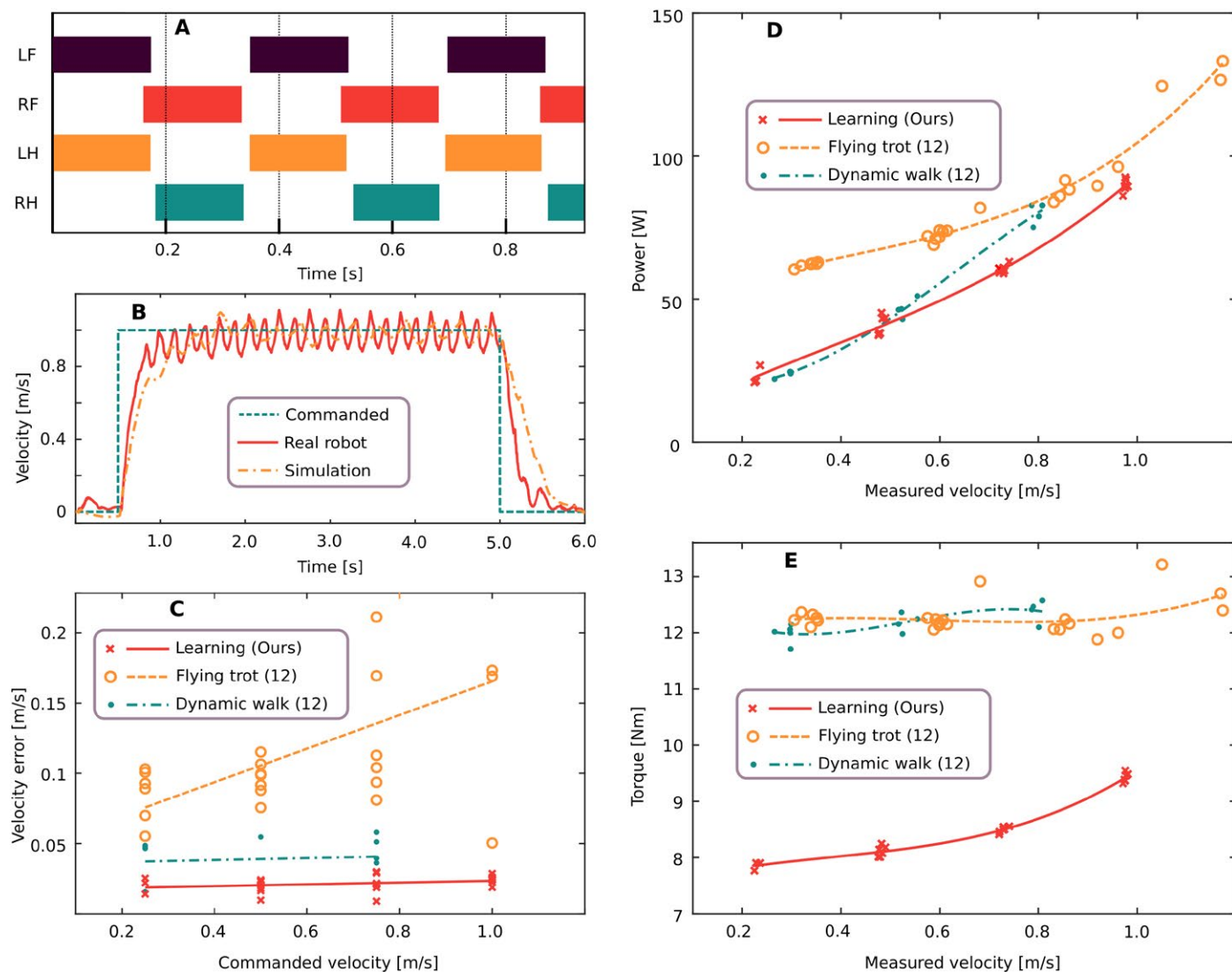


Fig. 2. Quantitative evaluation of the learned locomotion controller. (A) The discovered gait pattern for 1.0 m/s forward velocity command. LF, left front leg; RF, right front leg; LH, left hind leg; RH, right hind leg. (B) The accuracy of the base velocity tracking with our approach. (C to E) Comparison of the learned controller against the best existing controller, in terms of power efficiency, velocity error, and torque magnitude, given forward velocity commands of 0.25, 0.5, 0.75, and 1.0 m/s.

sources that were tuned using real data, accurately modeling all delay sources is complicated when the actuator has limited bandwidth. SEA mechanisms generate amplitude-dependent mechanical response time, and manual tuning of latency parameters becomes challenging. We tuned the analytical model for more than a week without much success.

High-speed locomotion

In the previous section, we evaluated the generality and robustness of the learned controller. Here, we focus on operating close to the limits of the hardware to reach the highest possible speed. The notion of high speed is, in general, hardware dependent. There are some legged robots that are exceptional in this regard. Park *et al.* (44) demonstrated full 3D legged locomotion at over 5.0 m/s with the MIT Cheetah. The Boston Dynamics WildCat has been reported to reach 8.5 m/s (45). These robots are designed to run as fast as possible, whereas ANYmal is designed to be robust, reliable, and versatile. The current speed record on ANYmal is 1.2 m/s and was set using the flying trot gait

(12). Although this may not seem high, it is 50% faster than the previous speed record on the platform (39). Such velocities are challenging to reach via conventional controller design while respecting all limits of the hardware.

We used the presented methodology to train a high-speed locomotion controller. This controller was tested on the physical system by slowly increasing the commanded velocity to 1.6 m/s and lowering it to zero after 10 m. The forward speed and joint velocities/torques are shown in Fig. 3. ANYmal reached 1.58 m/s in simulation and 1.5 m/s on the physical system when the command was set to 1.6 m/s. All speed values were computed by averaging over at least three gait cycles. The controller used both the maximum torque (40 N·m) and the maximum joint velocities (12 rad/s) on the physical system (Fig. 3, B and C). This shows that the learned policy can exploit the full capacity of the hardware to achieve the goal. For most existing methods, planning while accounting for the limitations of the hardware is very challenging, and executing the plan on the real

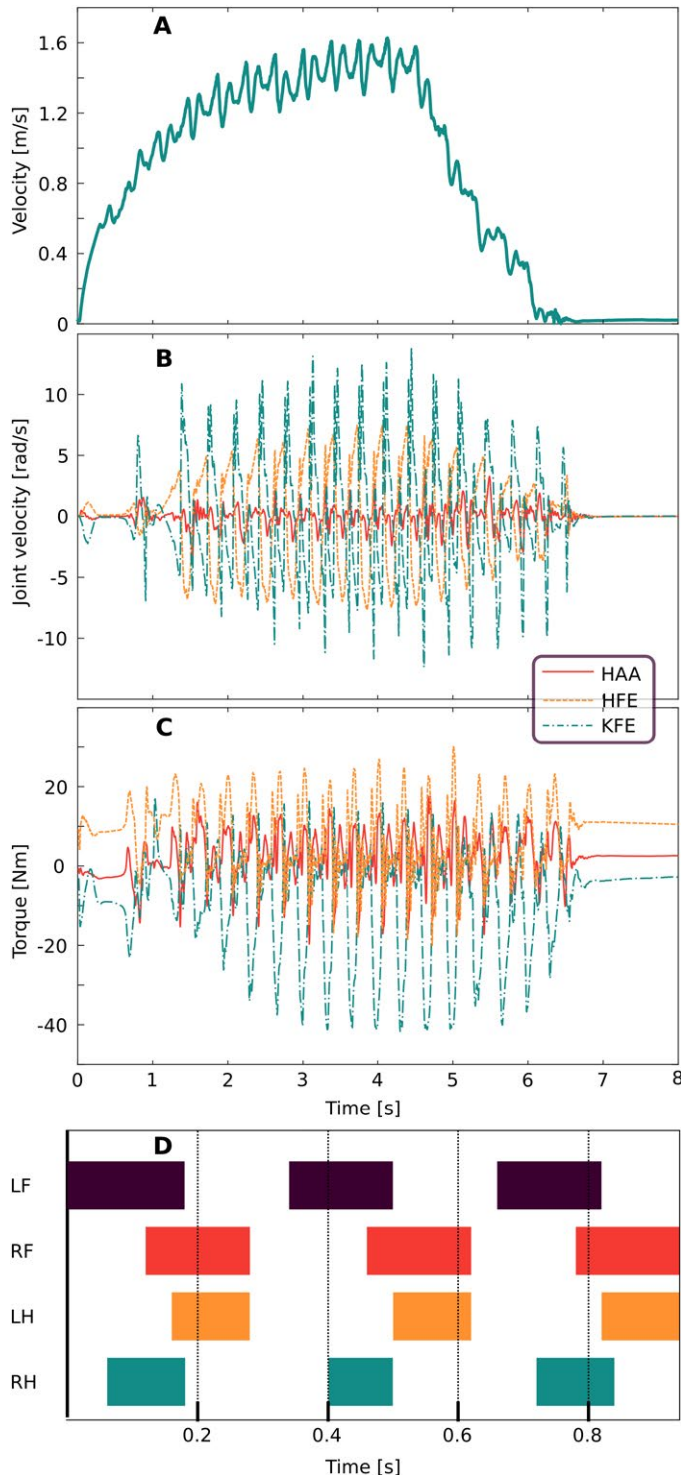


Fig. 3. Evaluation of the trained policy for high-speed locomotion. (A) Forward velocity of ANYmal. (B) Joint velocities. (C) Joint torques. (D) Gait pattern.

system reliably is harder still. Even state-of-the-art methods (12, 46) cannot limit the actuation during planning owing to limitations of their planning module. Modules in their controllers are not aware of the constraints in the later stages; consequently, their outputs may not be realizable on the physical system.

The gait pattern produced by our learned high-speed controller (Fig. 3D) is distinct from the one exhibited by the command-conditioned locomotion controller. It is close to a flying trot but with notably longer flight phase and asymmetric flight phase duration. This is not a commonly observed gait pattern in nature, and we suspect that it is among multiple near-optimal solution modes for this task. The behavior of the policy is illustrated in movie S5.

Recovery from a fall

Legged systems change contact points as they move and are thus prone to falling. If a legged robot falls and cannot autonomously restore itself to an upright configuration, then a human operator must intervene. Autonomous recovery after a fall is thus highly desirable. One possibility is to represent recovery behaviors as well-tuned joint trajectories that can simply be replayed—an approach that has been taken in some commercial systems (47). Such trajectories have required laborious manual tuning. They also take a very long time to execute because they do not take dynamics into account in the motion plan or the control. Some robots are designed such that recovery is either unnecessary or trivial (48, 49). However, such a design may not be possible for bigger and more complex machines. Morimoto *et al.* (50) demonstrated that a standing-up motion can be learned on a real robot. However, a simple three-link chain was used for demonstration, and the method has not been scaled to realistic systems.

Fast and flexible recovery after a fall, as seen in animals, requires dynamic motion with multiple unspecified contact points. The collision model for our quadruped is highly complicated: It consists of 41 collision bodies, such as boxes, cylinders, and spheres (Fig. 1, step 1). Planning a feasible trajectory for such a model is extremely complicated. Even simulating such a system is challenging because there are many internal contacts. We used the approach of Hwangbo *et al.* (41) owing to its ability to handle such simulation in numerically stable fashion.

Using the presented methodology, we trained a recovery policy and tested it on the real robot. We placed ANYmal in nine random configurations and activated the controller as shown in movie S6. Many challenging configurations were tested, including a nearly entirely upside-down configuration (pose 8) and more complex contact scenarios where ANYmal was resting on its own legs (poses 2 and 4). In all tests, ANYmal successfully flipped itself upright. An example motion is shown in Fig. 4. These agile and dynamic behaviors demonstrate that our approach is able to learn performant controllers for tasks that are difficult or impossible to address with prior methods.

DISCUSSION

The learning-based control approach presented in this paper achieved a high level of locomotion skill based purely on training in simulation and without tedious tuning on the physical robot. The system achieved more precise and energy-efficient motions than the prior state of the art. It outperformed the previous speed record by 25% and learned to consistently restore the robot to an operational configuration by dynamically rolling over its body.

Existing controllers are created by engineers. A model with adequate complexity has to be designed, and a control strategy has to be developed, tested, and tuned. This process typically takes months and has to be repeated for every distinct maneuver. In contrast, the simulation and learning framework used in this work are applicable to any rigid-body system. For applications to new tasks, our method

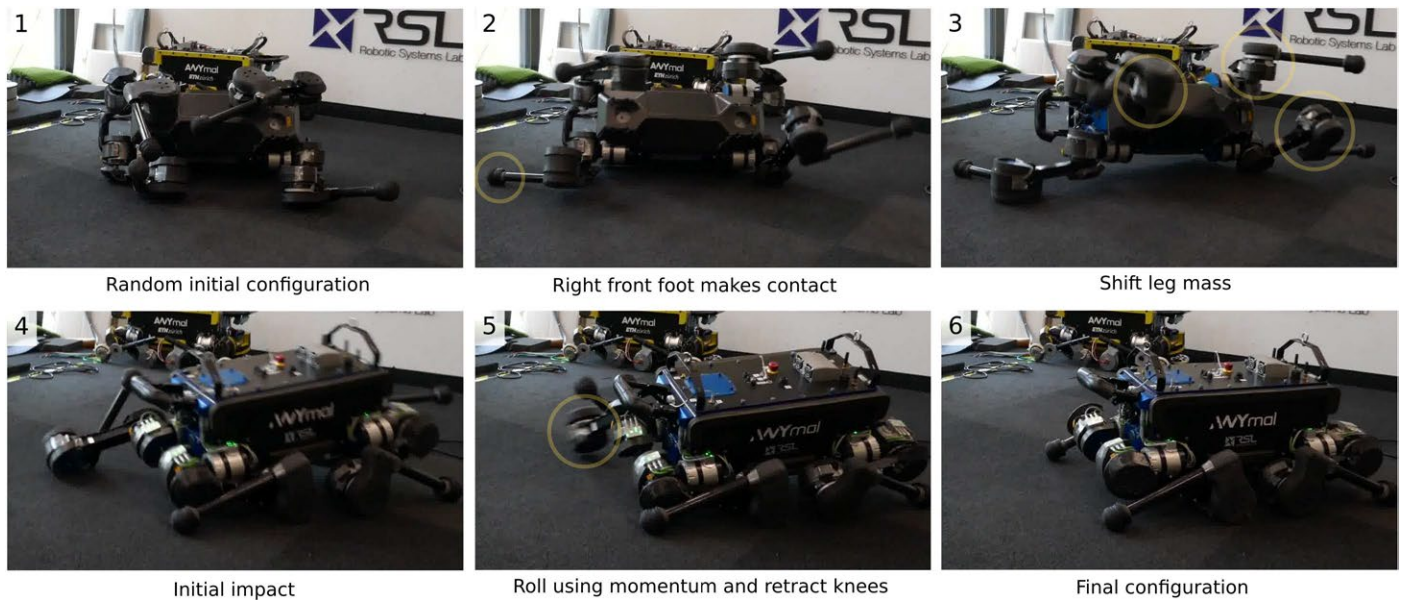


Fig. 4. A learned recovery controller deployed on the real robot. The learned policy successfully recovers from a random initial configuration in less than 3 s.

only requires a task description, which consists of the cost function, the initial state distribution, and randomization.

In our method, learned actuator dynamics effectively reduce the reality gap, whereas stochastic modeling guides the policy to be sufficiently conservative. The recovery task was successful on the very first attempt on the hardware. We then further improved the success rate to 100% by relaxing the joint velocity constraints. The results presented here were obtained on the second day of experiments on the physical system. In contrast, because of many model abstraction layers, which are necessary to make the computation tractable, prior methods often cannot exploit a sophisticated actuator model in controlling a complex legged system. Consequently, they often compromise performance or rely on well-tuned low-level controllers. For example, low-level controllers (e.g., the tracking controllers and the whole-body controller) have to be extensively tuned in the tested model-based controller (12) to mitigate imperfections of the actuators.

The learned policies are also robust to changes in hardware, such as those caused by wear and tear. All control policies have been tested for more than 3 months on the real robot without any modification. Within this period, the robot was heavily used with many controllers, including the ones presented here. Many hardware changes were introduced as well: different robot configurations, which roughly contribute 2.0 kg to the total weight, and a new drive which has a spring three times stiffer than the original one. All of the policies presented in this paper have performed robustly even under such conditions.

In terms of computational cost, our approach has an advantage over prior methods. Although it requires several hours of training with an ordinary desktop PC, the inference on the robot requires less than 25 μ s using a single CPU thread. Our method shifts nearly all computational costs to the training phase, where we can use external computational resources. Prior controllers often require two orders of magnitude more onboard computation. These demanding requirements limit the level of sophistication and thus the overall performance of the controller.

Using a policy network that directly outputs a joint-level command brings another advantage to our method. In contrast to many prior

methods that have numerical issues at singular configurations of the robot, our policies can be evaluated at any configuration. Consequently, our method is free from using ad hoc methods (e.g., branching conditions) in resolving such issues.

Although our approach allows for largely automated discovery of performant policies, it still requires some human expertise. A cost function and an initial state distribution have to be designed and tuned for each task. For a person with good understanding of both the task and RL, this process takes about 2 days for the locomotion policies presented in this work. Although this is still substantial amount of time, all the necessary tuning happens in simulation. Therefore, the development time will keep decreasing as computational technology evolves. In contrast, the prior controllers that use model abstractions inevitably require more development time and often extensive tuning on the real systems. Developing the recovery policy took about a week largely owing to the fact that some safety concerns (i.e., high impacts, fast swing legs, collisions with fragile components, etc.) are not very intuitive to embed in a cost function. Achieving a stand-up behavior was as simple as other tasks. However, for achieving the safe and robust behaviors that are demonstrated in this work, the cost function had to be tweaked several times. Longer development time was also attributed to the fact that it was trained by a person who had no previous experience with any real robot.

To train policies for a new robot, necessary modeling effort has to be made. This includes rigid-body modeling using the CAD model and actuator modeling using an actuator network. The former is often automated by modern CAD software, and the latter is easy if all necessary software/hardware infrastructures (e.g., logging, regression, and torque measurements) are in place. If not, it will also take a substantial portion of the development time. In addition, there are a few actuation types that manifest coupled dynamics (e.g., hydraulic actuators sharing a single accumulator). Learning actuators independently might not result in a sufficient accuracy for these systems. With a good understanding on the actuator dynamics, an appropriate history configuration can be estimated a priori and tuned further with respect to the validation error. In contrast, constructing an analytical actuator model

for ANYmal takes at least 3 weeks even if there is a very similar model studied in literature (39). The model also has many more parameters, many of which cannot be accurately obtained from measurements or the data sheet. Consequently, it requires more tuning than constructing an actuator network.

Another limitation of our approach was observed over the course of this study. A single neural network trained in one session manifests single-faceted behaviors that do not generalize across multiple tasks. Introducing hierarchical structure in the policy network can remedy this and is a promising avenue for future work (25).

The presented approach is not fundamentally limited to known and simple environments. We see the results presented in this paper

as a step toward comprehensive locomotion controllers for resilient and versatile legged robots.

MATERIALS AND METHODS

This section describes in detail the simulation environment, the training process, and the deployment on the physical system. An overview of our training method is shown in Fig. 5. The training loop proceeds as follows: The rigid-body simulator outputs the next state of the robot given the joint torques and the current state. The joint velocity and the position error are buffered in a joint state history within a finite time window. The control policy, implemented by an MLP with two hidden

layers, maps the observation of the current state and the joint state history to the joint position targets. Last, the actuator network maps the joint state history and the joint position targets to 12 joint torque values, and the loop continues. In what follows, we describe each component in detail.

Modeling rigid-body dynamics

To efficiently train a complex policy within a reasonable time and transfer it to the real world, we needed a simulation platform that is both fast and accurate. One of the biggest challenges with walking robots is the dynamics at intermittent contacts. To this end, we used the rigid-body contact solver presented in our previous work (41). This contact solver uses a hard contact model that fully respects the Coulomb friction cone constraint. This modeling technique can accurately capture the true dynamics of a set of rigid bodies making hard contacts with their environment. The solver is not only accurate but also fast, generating about 900,000 time steps per second for the simulated quadruped on an ordinary desktop machine.

The inertial properties of the links were estimated from the CAD model. We expected up to about 20% error in the estimation due to unmodeled cabling and electronics. To account for such modeling inaccuracies, we robustified the policy by training with 30 different ANYmal models with stochastically sampled inertial properties. The center of mass positions, the masses of links, and joint positions were randomized by adding a noise sampled from $U(-2, 2)$ cm, $U(-15, 15)\%$, and $U(-2, 2)$ cm, respectively.

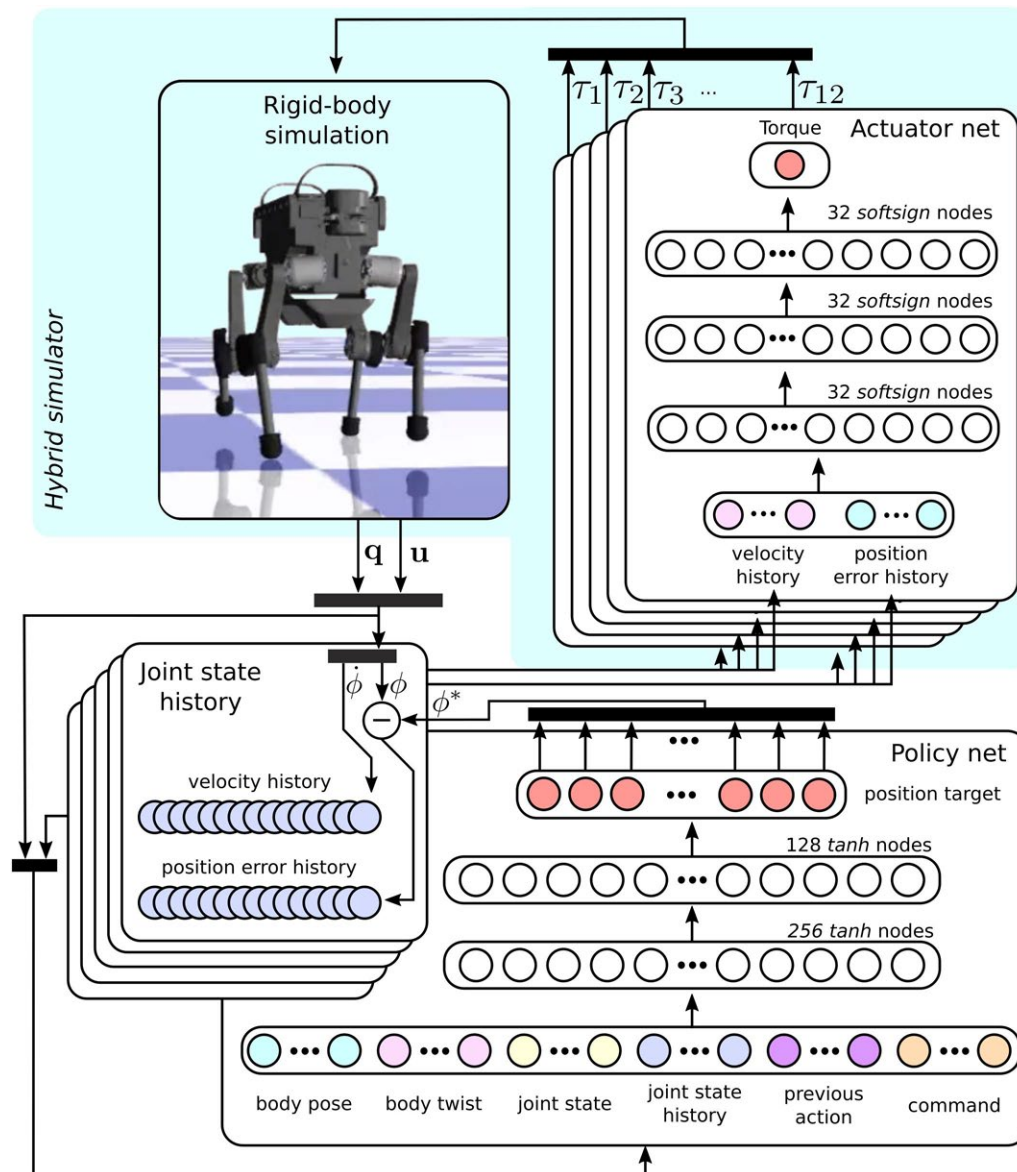


Fig. 5. Training control policies in simulation. The policy network maps the current observation and the joint state history to the joint position targets. The actuator network maps the joint state history to the joint torque, which is used in rigid-body simulation. The state of the robot consists of the generalized coordinate q and the generalized velocity u . The state of a joint consists of the joint velocity $\dot{\phi}$ and the joint position error, which is the current position ϕ subtracted from the joint position target ϕ^* .

Modeling the actuation

Actuators are an essential part of legged systems. Fast, powerful, lightweight, and high-accuracy actuators typically translate to dynamic, versatile, and agile robots. Most legged systems are driven by hydraulic actuators (51) or electric motors with gears (3), and some even include dedicated mechanical compliance (5, 52). These actuators have one thing in common: They are extremely difficult to model accurately. Their dynamics involve nonlinear and nonsmooth dissipation, and they contain cascaded feedback loops and a number of internal states that are not even directly observable. Gehring *et al.* (39) extensively studied SEA actuator modeling. The model of Gehring *et al.* includes nearly 100 parameters that have to be estimated from experiments or assumed to be correct from data sheets. This process is error prone and time consuming. In addition, many manufacturers do not provide sufficiently detailed descriptions of their products; consequently, an analytical model may not be feasible.

To this end, we used supervised learning to obtain an action-to-torque relationship that included all software and hardware dynamics within one control loop. More precisely, we trained an actuator network that output an estimated torque at the joints given a history of position errors (the actual position subtracted from the commanded position) and velocities. In this work, we assumed that the dynamics of the actuators are independent to each other such that we could learn a model for each actuator separately. This assumption might not be valid for other types of actuation. For example, hydraulic actuators with a single common accumulator might manifest coupled dynamics, and a single large network, representing multiple actuators together, might be more desirable.

The states of the actuators are only partially observable because the internal states of the actuators (e.g., states of the internal controllers and motor velocity) cannot be measured directly. We assumed that the network could be trained to estimate the internal states given a history of position errors and velocities, because otherwise the given information is simply insufficient to control the robot adequately. The actuator used in this work is revolute and radially symmetric, and the absolute angular position is irrelevant given the position error. We use a history consisting of the current state and two past states that correspond to $t - 0.01$ and $t - 0.02$ s. Note that too-sparse input configuration might not effectively capture the dynamics at high frequency (>100 Hz). This issue was partially mitigated by introducing a smoothness cost term, which penalizes abrupt changes in the output of the policy. Too-dense history can also have adverse effects: It is more prone to overfitting and computationally more expensive. The length of the history should be chosen such that it is sufficiently longer than the sum of all communication delays and the mechanical response time. In practice, the exact input configuration is tuned with respect to the validation error. This tuning process often takes less than a day because the network is very small.

To train the network, we collected a dataset consisting of joint position errors, joint velocities, and the torque. We used a simple parameterized controller that generates foot trajectories in the form of a sine wave; the corresponding joint positions were computed using inverse kinematics. The feet constantly made or broke a contact with the ground during data collection so that the resulting trajectories roughly mimicked the trajectories followed by a locomotion controller. To obtain a rich set of data, we varied the amplitude (5 to 10 cm) and the frequency (1 to 25 Hz) of the foot trajectories and disturbed the robot manually during data collection. We found that the excitation must cover a wide range of frequency spectra; otherwise, the trained model

generated unnatural oscillation even during the training phase. Data collection took less than 4 min because the data could be collected, in parallel, from the 12 identical actuators on ANYmal. Data were collected at 400 Hz; therefore, the resulting dataset contains more than a million samples. About 90% of the generated data were used for training, and the rest were used for validation.

The actuator network is an MLP with three hidden layers of 32 units each (Fig. 5, actuator net box). After testing with two common smooth and bounded activation functions—tanh and softsign (53)—we chose the softsign activation function because it is computationally efficient and provides a smooth mapping. Evaluating the actuator network for all 12 joints took 12.2 μ s with softsign and 31.6 μ s with tanh. As shown here, the tanh activation function resulted in a higher computational cost and is therefore less preferred. The two activation functions resulted in about the same validation error [0.7 to 0.8 N·m in root mean square (RMS)]. The validation result with the softsign function is shown in Fig. 6. The trained network nearly perfectly predicted the torque from the validation data, whereas the ideal actuator model failed to produce a reasonable prediction. Here, the ideal actuator model assumes that there is no communication delay and that the actuator can generate any commanded torque instantly (i.e., infinite actuator bandwidth). The trained model has an average error of 0.740 N·m on the validation set, which is not far from the resolution of the torque measurement (0.2 N·m) and much smaller than the error of the ideal actuator model (3.55 N·m). Its prediction error on test data (i.e., collected using the trained locomotion policies) is notably higher (0.966 N·m) but still far less than that of the ideal model (5.74 N·m).

Reinforcement learning

We represent the control problem in discretized time. At every time step t , the agent obtains an observation $o_t \in \mathcal{O}$, performs an action $a_t \in \mathcal{A}$, and achieves a scalar reward $r_t \in \mathcal{R}$. We refer to reward and cost interchangeably, with cost being the negative of the reward. We denote by $\mathbf{O}_t = \langle o_t, o_{t-1}, \dots, o_{t-h} \rangle$ the tuple of recent observations. The agent selects actions according to a stochastic policy $\pi(a_t | \mathbf{O}_t)$, which is a distribution over actions conditioned on the recent observations. The aim is to find a policy that maximizes the discounted sum of rewards over an infinite horizon:

$$\pi^* = \arg \max_{\pi} \mathbb{E}_{\tau(\pi)} \left[\sum_{t=0}^{\infty} \gamma^t r_t \right] \quad (1)$$

where $\gamma \in (0, 1)$ is the discount factor, and $\tau(\pi)$ is the trajectory distribution under policy π (the distribution depends on both the policy and the environment dynamics). In our setting, the observations are the measurements of robot states provided to the controller, the actions are the position commands to the actuators, and the rewards are specified so as to induce the behavior of interest.

A variety of RL algorithms can be applied to the specified policy optimization problem. We chose Trust Region Policy Optimization (TRPO) (22), a policy gradient algorithm that has been demonstrated to learn locomotion policies in simulation (54). It requires almost no parameter tuning; we used only the default parameters [as provided in (22, 54)] for all learning sessions presented in this paper. We used a fast custom implementation of the algorithm (55). This efficient implementation and fast rigid-body simulation (41) allowed us to generate and process about a quarter of a billion state transitions in roughly 4 hours. A learning session terminates if the average performance of

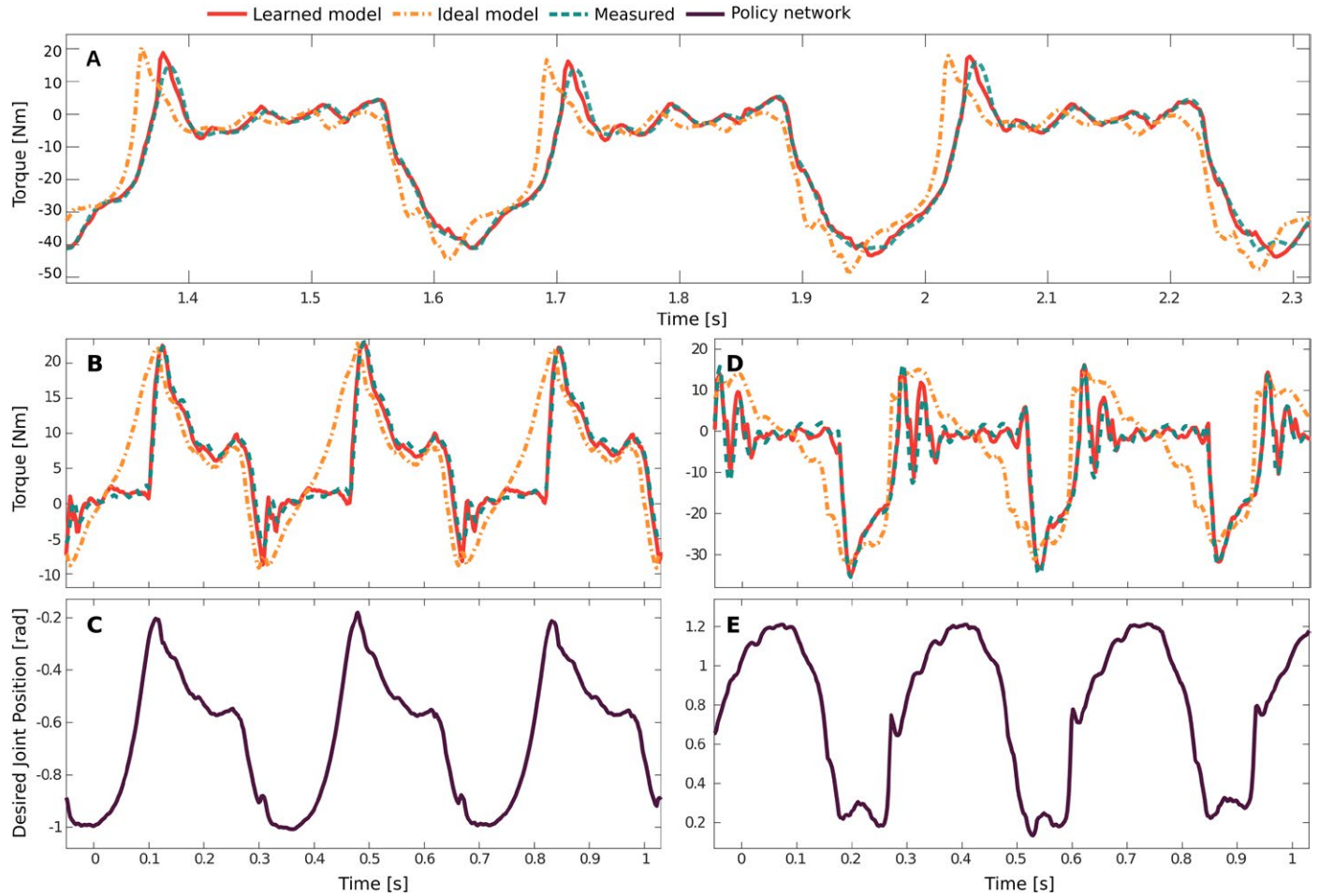


Fig. 6. Validation of the learned actuator model. The measured torque and the predicted torque from the trained actuator model are shown. The “ideal model” curve is computed assuming an ideal actuator (i.e., zero communication delay and zero mechanical response time) and is shown for comparison. (A) Validation set. Data from (B) a command-conditioned policy experiment with 0.75 m/s forward command velocity and (C) its corresponding policy network output. Data from (D) a high-speed locomotion policy experiment with 1.6 m/s forward command velocity and (E) its corresponding policy network output. Note that the measured ground truth in (A) is nearly hidden because the predicted torque from the trained actuator network accurately matches the ground-truth measurements. Test data were collected at one of the knee joints.

the policy does not improve by more than a task-specific threshold within 300 TRPO iterations.

Observation and action

The observations in our method should be observable (i.e., can be inferred from measurements) on the real robot and relevant for the task. The joint angles, velocities, and body twists are all observable and highly relevant. Measuring the body orientation is not straightforward because only two degrees of freedom in the orientation are observable with an inertial measurement unit (IMU). The set of observable degrees in the orientation is in bijection with S^2 , or with a unit vector, which can be interpreted as the direction of the gravity vector expressed in the IMU frame. We denote this unit vector as ϕ^g . The height of the base is not observable, but we can estimate it from the leg kinematics, assuming the terrain is flat. A simple height estimator based on a 1D Kalman filter was implemented along with the existing state estimation (56). However, this height estimator cannot be used when the robot is not on its feet, so we removed the height observation when training for recovery from a fall. The whole observation at $t = t_k$ is

defined as $o_k = \langle \phi^g, r_z, v, \omega, \phi, \dot{\phi}, \Theta, a_{k-1}, C \rangle$, where r_z , v , and ω are height, linear, and angular velocities of the base, ϕ and $\dot{\phi}$ are positions and velocities of the joints, Θ is a sparsely sampled joint state history, a_{k-1} is the previous action, and C is the command. The joint state history is sampled at $t = t_k - 0.01$ s and $t = t_k - 0.002$ s.

The joint state history was essential in training a locomotion policy. We hypothesize that this is due to the fact that it enables contact detection. An alternative way to detect contacts is to use force sensors, which give a reliable contact state estimate. However, such sensors increase the weight of the end effectors and consequently lower the energy efficiency of the robot. The exact history configuration was found empirically by analyzing the final performance of the policy.

Our policy outputs low-impedance joint position commands, which we find to be very effective in many tasks. Peng and van de Panne (57) found that such a controller can outperform a torque controller in both training speed and final control performance. Although there is always a bijective map between them, the two action parameterizations have different smoothness and thus different training difficulty. In addition, a position policy has an advantage in training

because it starts as a standing controller, whereas a torque controller initially creates many trajectories that result in falling. Thus, we use the policy network as an impedance controller. Our network outputs a single position reference, which is converted to torque using fixed gains ($k_p = 50 \text{ N}\cdot\text{m rad}^{-1}$ and $k_d = 0.1 \text{ N}\cdot\text{m rad}^{-1} \text{ s}^{-1}$) and zero target velocity. The position gain is chosen to be roughly the nominal range of torque ($\pm 30 \text{ N}\cdot\text{m}$) divided by the nominal range of motion ($\pm 0.6 \text{ rad}$). This ensures that the policy network has similar output range for torque and position. The velocity gain is chosen to be sufficiently high to prevent unwanted oscillation on the real robot. From our experience, the final locomotion performance is robust against a small change in gains. For instance, increasing the position gain to $80 \text{ N}\cdot\text{m rad}^{-1}$ does not noticeably change the performance.

Note that the position policy we use here is different from position controllers commonly used in robotics. Position controllers are sometimes limited in performance when the position reference is time indexed, which means that there is a higher-level controller that assumes that the position plan will be followed at high accuracy. This is the main reason that torque controllers have become popular in legged robotics. However, as in many other RL literature, our control policy is state indexed and does not suffer from the limitations of common PD controllers. The policy is trained to foresee that position errors will occur and even uses them to generate acceleration and interaction forces. In addition, thanks to kinematic randomization, a trained policy does not solely rely on kinematics: The policy inevitably has to learn to exert appropriate impulse on the environment for locomotion. This makes our policy more robust because impulse-based control approaches are known to be more robust against system changes and model inaccuracies (44).

Policy training details

The control policies presented in this work were trained only in simulation. To train performant policies using only simulated data, we followed both standard and problem-specific training procedures. Here, we describe them in detail and explain the rationale behind them.

Training control policies for locomotion have been demonstrated multiple times in literature. (22, 24, 25). However, many of the trained policies do not manifest natural motions, and it is highly questionable whether they will work on physical systems. Some researchers have noticed that naive methods cannot generate natural-looking and energy-efficient locomotion behaviors (58). Low penalty on joint torque and velocity results in unnatural motions, whereas high penalty on them results in a standing behavior. The main reason for the standing behavior is that such a behavior is already a good local minimum when there is high penalty associated with motion.

We solved this problem by introducing a curriculum: Using a curriculum, we shape the initial cost landscape such that the policy is strongly attracted to a locomotion policy and then later polish the motion to satisfy the other criteria. A simple curriculum was generated by modulating the coefficients of the cost terms and the disturbance via a multiplicative curriculum factor. We define a curriculum factor that describes the progression of the curriculum: $k_c = k_0 \in (0, 1)$ corresponds to the start of the curriculum and $k_c = 1$ corresponds to the final difficulty level. The intermediate values are computed as $k_{c,j+1} \leftarrow (k_{c,j})^{k_d}$, where $k_d \in (0, 1)$ is the advance rate, which describes how quickly the final difficulty level is reached, and j is the iteration index of RL training. The sequence of curriculum factors is monotonically increasing and asymptotically converging to 1 within the given

parameter intervals. We suspect that many other update rules adhering to these criteria will result in similar learning performance. All of cost terms are multiplied by this curriculum factor, except the cost terms related to the objective (i.e., base velocity error cost in the command-conditioned and high-speed locomotion task and base orientation cost in recovery task). This way, the robot first learns how to achieve the objective and then how to respect various constraints. This technique is related to curriculum learning introduced by Bengio *et al.* (59), which incrementally introduces samples of more difficulties. Instead of altering the samples, we alter the objective to control the training difficulty. For all training sessions, we use $k_0 = 0.3$ and $k_d = 0.997$. The parameter k_0 should be chosen to prevent the initial tendency to stand still. It can be easily tuned by observing the first 100 iterations of the RL algorithm. The parameter k_d is chosen such that the curriculum factor almost reaches 1 (or ~ 0.9) at the end of training. Although the required number iterations are not known a priori, there are sufficient publications on RL applications (including this one) to provide necessary insights to the users.

We tuned the discount factor γ (Eq. 1) separately for each task based on the qualitative performance of the trained controllers in simulation. For training the command-conditioned controller and the high-speed controller, we used $\gamma = 0.9988$, which corresponds to a half-life of 5.77 s. We also successfully trained almost equally performant policies with a lower half-life (2 s), but they manifest a less natural standing posture. For training the recovery controller, we used $\gamma = 0.993$, which corresponds to a half-life of 4.93 s. A sufficiently high discount factor shows more natural standing posture owing to the fact that it penalizes standing torque more than motion (torque, joint velocities, and other quantities incurring due to motion). However, a too-high discount factor might result in a slow convergence, so it should be tuned appropriately depending on the task. For training command-conditioned and high-speed locomotion, TRPO finished training in 9 days of simulated time, which corresponds to 4 hours of computation in real time. For training for recovery from a fall, TRPO took 79 days of simulated time, which corresponds to 11 hours of computation in real time.

For command-conditioned and high-speed locomotion, we represent a command by three desired body velocity values: forward velocity, lateral velocity, and the turning rate. During training, the commands are sampled randomly from predefined intervals (see tables S1 and S2 for details), and the cost defined in section S3 is used. The initial state of the robot is sampled either from a previous trajectory or a random distribution, shown in table S3, with equal probability. This initialization procedure generates data containing complicated state transitions and robustifies the trained controller. Each trajectory lasts 6 s unless the robot reaches a terminal state earlier. There are two possibilities for termination: violating joint limits and hitting the ground with the base. Upon termination, agent receives a cost of 1 and is reinitialized. The value of the termination cost is not tuned: Because only the ratio between the cost coefficients is important for the final performance, we tune other cost terms to work with this terminal value.

For training recovery from a fall, the collision bodies of the ANYmal model are randomized in size and position. Samples that result in unrealistic internal collisions are removed. The cost function and the initial state distribution are described in section S4 and fig. S3, respectively. The special initialization method in section S4 is needed to train for this task, because naive sampling often results in interpenetration and the dynamics become unrealistic. To this end, we

dropped ANYmal from a height of 1.0 m with randomized orientations and joint positions, ran the simulation for 1.2 s, and used the resulting state as initialization.

Another crucial detail is that joint velocities cannot be directly measured on the real robot. Rather, they are computed by numerically differentiating the position signal, which results in noisy estimates. We modeled this imperfection by injecting a strong additive noise [$U(-0.5, 0.5)$ rad/s] to the joint velocity measurements during training. This way, we ensured that the learned policy is robust to inaccurate velocity measurements. We also added noise during training to the observed linear velocity [$U(-0.08, 0.08)$ m/s] and angular velocity [$U(-0.16, 0.16)$ m/s] of the base. The rest of the observations were noise free. Removing velocities from the observation altogether led to a complete failure to train, although in theory, the policy network could infer velocities as finite differences of observed positions. We explain this by the fact that nonconvexity of network training makes appropriate input preprocessing important. For similar reasons, input normalization is necessary in most learning procedures.

We implemented the policy with an MLP with two hidden layers, with 256 and 128 units each and tanh nonlinearity (Fig. 5). We found that the nonlinearity has a strong effect on performance on the physical system. Performance of two trained policies with different activation functions can be very different in the real world even when they perform similarly in simulation. Our explanation is that unbounded activation functions, such as rectified linear unit, can degrade performance on the real robot, because actions can have very high magnitude when the robot reaches states that were not visited during training. Bounded activation functions, such as tanh, yield less aggressive trajectories when subjected to disturbances. We believe that this is true for softsign as well, but it was not tested in policy networks owing to an implementation issue in our RL framework (55).

Deployment on the physical system

We used the ANYmal robot (5), shown in step 4 of Fig. 1, to demonstrate the real-world applicability of our method. ANYmal is a dog-sized quadrupedal robot weighing about 32 kg. Each leg is about 55 cm long and has three actuated degrees of freedom, namely, hip abduction/adduction, hip flexion/extension, and knee flexion/extension.

ANYmal is equipped with 12 SEAs (60, 61). An SEA is composed of an electric motor, a high gear ratio transmission, an elastic element, and two rotary encoders to measure spring deflection and output position. In this work, we used a joint-level PD controller with low feedback gains on the joint-level actuator module of the ANYmal robot. The dynamics of the actuators contain multiple components in succession, as follows. First, the position command is converted to the desired torque using a PD controller. Subsequently, the desired current is computed using a PID controller from the desired torque. The desired current is then converted to phase voltage using a field-oriented controller, which produces the torque at the input of the transmission. The output of the transmission is connected to an elastic element whose deflection finally generates torque at the joint (39). These highly complex dynamics introduce many hidden internal states that we do not have direct access to and complicate our control problem.

After acquiring a parameter set for a trained policy from our hybrid simulation, the deployment on the real system was straightforward. A custom MLP implementation and the trained parameter set were ported to the robot's onboard PC. This network was evaluated at 200 Hz for command-conditioned/high-speed locomotion and at 100 Hz for recovery from a fall. We found that performance was un-

expectedly insensitive to the control rate. For example, the recovery motion was trained at 20 Hz but performance was identical when we increased the control rate up to 100 Hz. This was possible because the flip-up behaviors involve low joint velocities (mostly below 6 rad/s). More dynamic behaviors (e.g., locomotion) often require a much higher control rate to have an adequate performance. A higher frequency (100 Hz) was used for experiments because it made less audible noise. Even at 100 Hz, evaluation of the network uses only 0.25% of the computation available on a single CPU core.

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/26/eaau5872/DC1

Section S1. Nomenclature

Section S2. Random command sampling method used for evaluating the learned command-conditioned controller

Section S3. Cost terms for training command-conditioned locomotion and high-speed locomotion tasks

Section S4. Cost terms for training recovery from a fall

Fig. S1. Base velocity tracking performance of the learned controller while following random commands.

Fig. S2. Base velocity tracking performance of the best existing method while following random commands.

Fig. S3. Sampled initial states for training a recovery controller.

Table S1. Command distribution for training command-conditioned locomotion.

Table S2. Command distribution for training high-speed locomotion.

Table S3. Initial state distribution for training both the command-conditioned and high-speed locomotion.

Movie S1. Locomotion policy trained with a learned actuator model.

Movie S2. Random command experiment.

Movie S3. Locomotion policy trained with an analytical actuator model.

Movie S4. Locomotion policy trained with an ideal actuator model.

Movie S5. Performance of a learned high-speed policy.

Movie S6. Performance of a learned recovery policy.

REFERENCES AND NOTES

1. M. Raibert, K. Blankespoor, G. Nelson, R. Playter, BigDog, the rough-terrain quadruped robot. *IFAC Proc. Vol.* **41**, 10822–10825 (2008).
2. G. Nelson, A. Saunders, N. Neville, B. Swilling, J. Bondaryk, D. Billings, C. Lee, R. Playter, M. Raibert, PETMAN: A humanoid robot for testing chemical protective clothing. *J. Robot. Soc. Japan* **30**, 372–377 (2012).
3. S. Seok, A. Wang, M. Y. Chuah, D. Otten, C. J. Lang, S. Kim, Design principles for highly efficient quadrupeds and implementation on the MIT Cheetah robot, in *2013 IEEE International Conference on Robotics and Automation (IEEE, 2013)*, pp. 3307–3312.
4. Spotmini autonomous navigation, https://youtu.be/Ve9kWX_KXus. [accessed 11 August 2018].
5. 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. Hoepfner, ANYmal – a highly mobile and dynamic quadrupedal robot, in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE, 2016)*, pp. 38–44.
6. R. J. Full, D. E. Koditschek, Templates and anchors: Neuromechanical hypotheses of legged locomotion on land. *J. Exp. Biol.* **202**, 3325–3332 (1999).
7. M. H. Raibert, J. J. Craig, Hybrid position/force control of manipulators. *J.Dyn. Sys. Meas. Control* **103**, 126–133 (1981).
8. J. Pratt, J. Carff, S. Drakunov, A. Goswami, Capture point: A step toward humanoid push recovery, in *2006 6th IEEE-RAS International Conference on Humanoid Robots (IEEE, 2006)*, pp. 200–207.
9. A. Goswami, B. Espiau, A. Keramane, Limit cycles in a passive compass gait biped and passivity-mimicking control laws. *Auton. Robots* **4**, 273–286 (1997).
10. W. J. Schwind, Spring loaded inverted pendulum running: A plant model, thesis, University of Michigan (1998).
11. M. Kalakrishnan, J. Buchli, P. Pastor, M. Mistry, S. Schaal, Fast, robust quadruped locomotion over challenging terrain, in *2010 IEEE International Conference on Robotics and Automation (IEEE, 2010)*, pp. 2665–2670.
12. C. D. Bellicoso, F. Jenelten, C. Gehring, M. Hutter, Dynamic locomotion through online nonlinear motion optimization for quadrupedal robots. *IEEE Robot. Autom. Lett.* **3**, 2261–2268 (2018).
13. M. Neunert, F. Farshidian, A. W. Winkler, J. Buchli, Trajectory optimization through contacts and automatic gait discovery for quadrupeds. *IEEE Robot. Autom. Lett.* **2**, 1502–1509 (2017).

14. I. Mordatch, E. Todorov, Z. Popović, Discovery of complex behaviors through contact-invariant optimization. *ACM Trans. Graph.* **31**, 43 (2012).
15. F. Farshidian, M. Neunert, A. W. Winkler, G. Rey, J. Buchli, An efficient optimal planning and control framework for quadrupedal locomotion, in *2017 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017), pp. 93–100.
16. M. Posa, C. Cantu, R. Tedrake, A direct method for trajectory optimization of rigid bodies through contact. *Int. J. Robot. Res.* **33**, 69–81 (2014).
17. J. Carius, R. Ranftl, V. Koltun, M. Hutter, Trajectory optimization with implicit hard contacts. *IEEE Robot. Autom. Lett.* **3**, 3316–3323 (2018).
18. S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, D. Quillen, Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. J. Robot. Res.* **37**, 421–436 (2018).
19. R. Tedrake, T. W. Zhang, H. S. Seung, Stochastic policy gradient reinforcement learning on a simple 3d biped, in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2004), pp. 2849–2854.
20. J. Yosinski, J. Clune, D. Hidalgo, S. Nguyen, J. C. Zagal, H. Lipson, Evolving robot gaits in hardware: The hyperNEAT generative encoding vs. parameter optimization, in *Proceedings of the 20th European Conference on Artificial Life* (Springer MIT Press, 2011), pp. 890–897.
21. S. Levine, V. Koltun, Learning complex neural network policies with trajectory optimization, in *Proceedings of the 31st International Conference on Machine Learning, PMLR* (PMLR, 2014), pp. 829–837.
22. J. Schulman, S. Levine, P. Abbeel, M. Jordan, P. Moritz, Trust region policy optimization, in *International Conference on Machine Learning (ICML)* (Omnipress, 2015), pp. 1889–1897.
23. J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms, arXiv:1707.06347 (2017).
24. N. Heess, S. Sriram, J. Lemmon, J. Merel, G. Wayne, Y. Tassa, T. Erez, Z. Wang, A. Eslami, M. Riedmiller, D. Silver, Emergence of locomotion behaviours in rich environments, arXiv:1707.02286 (2017).
25. X. B. Peng, G. Berseeth, K. Yin, M. Van De Panne, Deeploco: Dynamic locomotion skills using hierarchical deep reinforcement learning. *ACM Trans. Graph.* **36**, 41 (2017).
26. Z. Xie, G. Berseeth, P. Clary, J. Hurst, M. van de Panne, Feedback control for Cassie with deep reinforcement learning, arXiv:1803.05580 (2018).
27. T. Lee, F. C. Park, A geometric algorithm for robust multibody inertial parameter identification. *IEEE Robot. Autom. Lett.* **3**, 2455–2462 (2018).
28. M. Neunert, T. Boaventura, J. Buchli, Why Off-The-Shelf Physics Simulators Fail in Evaluating Feedback Controller Performance: A Case Study for Quadrupedal Robots. *Advances in Cooperative Robotics* (World Scientific, 2017), pp. 464–472.
29. J. Bongard, V. Zykov, H. Lipson, Resilient machines through continuous self-modeling. *Science* **314**, 1118–1121 (2006).
30. D. Nguyen-Tuong, M. Seeger, J. Peters, Model learning with local gaussian process regression. *Adv. Robot.* **23**, 2015–2034 (2009).
31. D. Nguyen-Tuong, J. Peters, Learning robot dynamics for computed torque control using local gaussian processes regression, in *2008 ECIS Symposium on Learning and Adaptive Behaviors for Robotic Systems (LAB-RS)* (IEEE, 2008), pp. 59–64.
32. K. Nikzad, J. Ghaboussi, S. L. Paul, Actuator dynamics and delay compensation using neurocontrollers. *J. Eng. Mech.* **122**, 966–975 (1996).
33. R. S. Sutton, A. G. Barto, *Reinforcement Learning: An Introduction* (MIT Press, 1998), vol. 1.
34. I. Mordatch, K. Lowrey, E. Todorov, Ensemble-cio: Full-body dynamic motion planning that transfers to physical humanoids, in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2015), pp. 5307–5314.
35. J. Tan, T. Zhang, E. Coumans, A. Isen, Y. Bai, D. Hafner, S. Bohez, V. Vanhoucke, Sim-to-real: Learning agile locomotion for quadrupedal robots. *Proc. Robot.* 10.15607/RSS.2018.XIV.010 (2018).
36. X. B. Peng, M. Andrychowicz, W. Zaremba, P. Abbeel, Sim-to-real transfer of robotic control with dynamics randomization. arXiv:1710.06537 (2017).
37. N. Jakobi, P. Husbands, I. Harvey, *Noise and The Reality Gap: The Use of Simulation in Evolutionary Robotics*, *European Conference on Artificial Life* (Springer, 1995), pp. 704–720.
38. A. Dosovitskiy, V. Koltun, Learning to act by predicting the future, in *International Conference on Learning Representations (ICLR)* (PMLR, 2017).
39. C. Gehring, S. Coros, M. Hutter, C. D. Bellicoso, H. Heijnen, R. Diethelm, M. Bloesch, P. Fankhauser, J. Hwangbo, M. Hoepfner, R. Siegwart, Practice makes perfect: An optimization-based approach to controlling agile motions for a quadrupedal robot. *IEEE Robot. Autom. Mag.* **23**, 34–43 (2016).
40. R. Featherstone, *Rigid Body Dynamics Algorithms* (Springer, 2014).
41. J. Hwangbo, J. Lee, M. Hutter, Per-contact iteration method for solving contact dynamics. *IEEE Robot. Autom. Lett.* **3**, 895–902 (2018).
42. R. Smith, Open dynamics engine (2005).
43. D. A. Winter, *The Biomechanics and Motor Control of Human Gait: Normal, Elderly and Pathological* (University of Waterloo Press, ed. 2, 1991).
44. H.-W. Park, S. Park, S. Kim, Variable-speed quadrupedal bounding using impulse planning: Untethered high-speed 3d running of MIT cheetah 2, in *2015 IEEE International Conference on Robotics and Automation* (IEEE, 2015), pp. 5163–5170.
45. Introducing wildcat, <https://youtu.be/wE3fmFTp9g> [accessed 6 August 2018].
46. A. W. Winkler, F. Farshidian, D. Pardo, M. Neunert, J. Buchli, Fast trajectory optimization for legged robots using vertex-based zmp constraints. *IEEE Robot. Autom. Lett.* **2**, 2201–2208 (2017).
47. S. Shamsuddin, L. I. Ismail, H. Yusoff, N. I. Zahari, S. Bahari, H. Hashim, A. Jaffar, Humanoid robot Nao: Review of control and motion exploration, in *2011 IEEE International Conference on Control System, Computing and Engineering* (IEEE, 2011), pp. 511–516.
48. U. Saranlı, M. Buehler, D. E. Koditschek, Rhex: A simple and highly mobile hexapod robot. *Int. J. Robot. Res.* **20**, 616–631 (2001).
49. E. Ackerman, Boston dynamics sand flea robot demonstrates astonishing jumping skills. *IEEE Spectrum Robotics Blog* **2**, (2012).
50. J. Morimoto, K. Doya, Acquisition of stand-up behavior by a real robot using hierarchical reinforcement learning. *Robot. Auton. Syst.* **36**, 37–51 (2001).
51. C. Semini, N. G. Tsagarakis, E. Guglielmino, M. Focchi, F. Cannella, D. G. Caldwell, Design of HyQ – a hydraulically and electrically actuated quadruped robot. *Proc. Inst. Mec. Eng. Part I J. Syst. Control Eng.* **225**, 831–849 (2011).
52. N. G. Tsagarakis, S. Morfe, G. M. Cerda, L. Zhibin, D. G. Caldwell, COMpliant huMANoid COMAN: Optimal joint stiffness tuning for modal frequency control, in *2013 IEEE International Conference on Robotics and Automation* (IEEE, 2013), pp. 673–678.
53. J. Bergstra, G. Desjardins, P. Lamblin, Y. Bengio, Quadratic polynomials learn better image features. Technical report, 1337 (2009).
54. J. Schulman, P. Moritz, S. Levine, M. Jordan, P. Abbeel, High-dimensional continuous control using generalized advantage estimation, in *Proceedings of the International Conference on Learning Representations (ICLR)* (2016).
55. J. Hwangbo, I. Sa, R. Siegwart, M. Hutter, Control of a quadrotor with reinforcement learning. *IEEE Robot. Autom. Lett.* **2**, 2096–2103 (2017).
56. M. Bloesch, M. Hutter, M. A. Hoepfner, S. Leutenegger, C. Gehring, C. D. Remy, R. Siegwart, State estimation for legged robots-consistent fusion of leg kinematics and imu. *Robotics* **17**, 17–24 (2013).
57. X. B. Peng, M. van de Panne, Learning locomotion skills using deeprl: Does the choice of action space matter? in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (ACM, 2017), p. 12.
58. W. Yu, G. Turk, C. K. Liu, Learning symmetric and low-energy locomotion. *ACM Trans. Graph.* **37**, 144 (2018).
59. Y. Bengio, J. Louradour, R. Collobert, J. Weston, Curriculum learning, in *Proceedings of the 26th annual International Conference on Machine Learning* (ACM, 2009), pp. 41–48.
60. G. A. Pratt, M. M. Williamson, Series elastic actuators, in *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots* (IEEE, 1995), pp. 399–406.
61. M. Hutter, K. Bodie, A. Lauber, J. Hwangbo, Joint unit, joint system, robot for manipulation and/or transportation, robotic exoskeleton system and method for manipulation and/or transportation EP16181251 (2016).

Acknowledgments: We thank ANYbotics for responsive support on ANYmal. **Funding:** The project was funded, in part, by the Intel Network on Intelligent Systems and the Swiss National Science Foundation (SNF) through the National Centre of Competence in Research Robotics and Project 200021-166232. The work has been conducted as part of ANYmal Research, a community to advance legged robotics. **Author contributions:** J.H. conceived the main idea of the train and control methods, set up the simulation, and trained networks for the command-conditioned locomotion and the high-speed locomotion. J.L. trained a network for recovery from a fall. J.H., A.D., M.H., and V.K. refined ideas and contributed in the experiment design. J.H. and J.L. performed experiments together. D.B. and V.T. helped setting up the hardware for the experiments. J.H., A.D., M.H., and V.K. analyzed the data and prepared the manuscript. **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 and/or the Supplementary Materials. Other materials can be found at https://github.com/junja94/anyml_science_robotics_supplementary.

Submitted 3 September 2018

Accepted 12 December 2018

Published 16 January 2019

10.1126/scirobotics.aau5872

Citation: J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, M. Hutter, Learning agile and dynamic motor skills for legged robots. *Sci. Robot.* **4**, eaau5872 (2019).

ANIMAL ROBOTS

Perching and resting—A paradigm for UAV maneuvering with modularized landing gears

Kaiyu Hang^{1*†}, Ximin Lyu^{2*}, Haoran Song^{2*}, Johannes A. Stork^{3,4*}, Aaron M. Dollar¹, Danica Kragic³, Fu Zhang⁵

Copyright © 2019
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

Perching helps small unmanned aerial vehicles (UAVs) extend their time of operation by saving battery power. However, most strategies for UAV perching require complex maneuvering and rely on specific structures, such as rough walls for attaching or tree branches for grasping. Many strategies to perching neglect the UAV's mission such that saving battery power interrupts the mission. We suggest enabling UAVs with the capability of making and stabilizing contacts with the environment, which will allow the UAV to consume less energy while retaining its altitude, in addition to the perching capability that has been proposed before. This new capability is termed "resting." For this, we propose a modularized and actuated landing gear framework that allows stabilizing the UAV on a wide range of different structures by perching and resting. Modularization allows our framework to adapt to specific structures for resting through rapid prototyping with additive manufacturing. Actuation allows switching between different modes of perching and resting during flight and additionally enables perching by grasping. Our results show that this framework can be used to perform UAV perching and resting on a set of common structures, such as street lights and edges or corners of buildings. We show that the design is effective in reducing power consumption, promotes increased pose stability, and preserves large vision ranges while perching or resting at heights. In addition, we discuss the potential applications facilitated by our design, as well as the potential issues to be addressed for deployment in practice.

INTRODUCTION

With recent advances in lightweight, low-power sensor technology and onboard computation, unmanned aerial vehicles (UAVs) are now engaging in missions with an unprecedented degree of autonomy (1–3). Onboard sensors such as cameras, ultrasonic sensors, and accelerometers not only provide advanced perception capabilities that allow increasingly complex missions but also enable more powerful control methods (4–8). Even commercial off-the-shelf (COTS) UAVs can reliably fulfill missions such as aerial videography, autonomous surveillance, object delivery, and construction site inspection (9–13) and are deployed in crisis response to provide on-site measurements (2, 14–16) or set up ad hoc data networks (17).

Autonomous UAVs are often deployed to conduct long-duration missions that require watching an area on the ground from heights for an extended period of time, such as in an autonomous surveillance task (12, 18). For this reason, energy consumption is one of the primary concerns in the operation of lightweight UAVs because mission duration is limited by battery power. Because UAVs require constant motor action to create lift to stay in the air, more energy-efficient control and aircraft design are therefore of high interest to reduce the energy consumption during flight (19–24). However, the most effective way of saving energy is to directly reduce the required lift during execution of the mission.

Exploiting contacts to save energy

In this work, we try to learn from nature and take inspiration from the behavior and anatomy of birds and bats. However, we propose a de-

sign that is simpler and more optimized for the specific task of saving energy than what we observe in nature. Figure 1 displays several ways in which animals with powered flight have adapted to temporarily exploit contacts with structures in their habitat for saving energy. For example, birds can be observed placing their feet on supports while still flapping their wings, and bats are known to hang upside down while grasping suitable surfaces. In all of these cases, some suitably shaped part of the animal's foot interacts with a structure in the environment and facilitates that less lift needs to be generated or that power flight can be completely suspended.

Our goal is to use the same concept, which is commonly referred to as "perching," for UAVs. Perching requires attaching and detaching from a structure in the surroundings on command and relies on the availability of certain structures in the surroundings, such as tree branches. It is therefore limited to a small set of mission environments; when the perching location does not provide a good view range, it will result in mission interruptions. For addressing the problem of allowing UAVs to reduce their power consumption in a mission, we propose to enable UAVs with the capability of making and stabilizing contacts with the environment to obtain force support. With this capability, UAVs require less lift generated by the motors and can save energy. Moreover, it enables UAVs to be able to exploit a much larger range of structures in the environment to conduct missions without interruptions. We term this kind of action "resting" (Fig. 1, left and right). Perching or resting on elevated locations allows continuation of a large range of UAV missions with reduced, or even suspended, motor action and therefore extends the UAV's operation time and allows long-duration missions, such as in the most common perch-and-stare missions (25). Additionally, perching and resting remove degrees of freedom from the UAV's motion and can therefore reduce the required attention from operators and can improve safety.

The need for perching capabilities in UAVs has led to research in a wide range of different forms of landing gears (26–44), control for the required flight regimes, and the generation and optimization of

¹Department of Mechanical Engineering and Material Science, Yale University, New Haven, CT, USA. ²Hong Kong University of Science and Technology, Hong Kong, China. ³RPL, KTH Royal Institute of Technology, Stockholm, Sweden. ⁴Centre for Applied Autonomous Sensor Systems (AASS), Örebro University, Örebro, Sweden. ⁵The University of Hong Kong, Hong Kong, China.

*These authors contributed equally to this work.

†Corresponding author. Email: kaiyu.hang@yale.edu

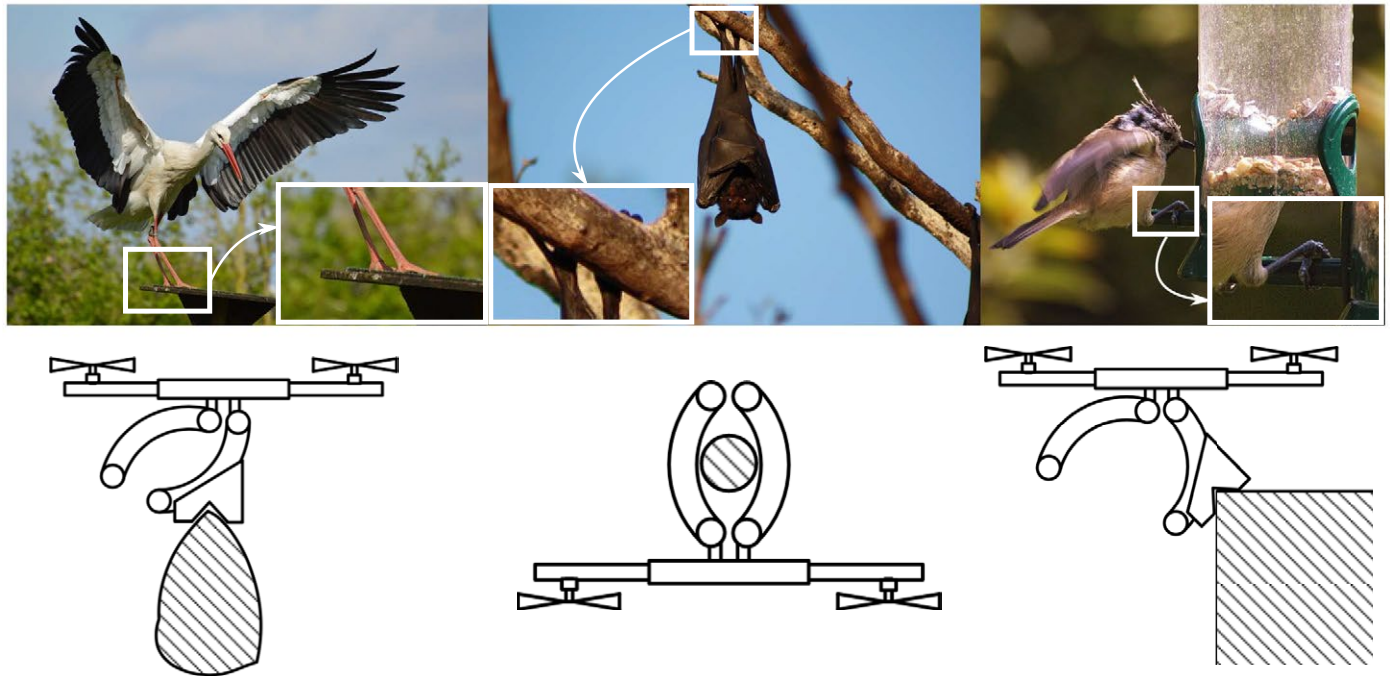


Fig. 1. Example perching and resting actions in nature. Flying animals such as birds or bats often make use of structures in the environment to save energy. In choosing, they select locations that can be approached and evacuated by simply maneuvering in the air while still allowing them to execute a mission such as observing the environment or looking for prey.

approach trajectories (30–32, 37, 38, 45–47). Surface contacts were established and maintained with dry adhesive technology, such as electrostatic surfaces (41–43) or fibers (44). A collection of small needles were used for perching on rough surfaces (26–31) or combined to form bioinspired claw-like grippers (33). Also, multiple tensile anchors were launched to fixed structures (48) to mechanically stabilize the UAV for high-accuracy operation in a three-dimensional (3D) workspace. Other UAV-mounted grippers took design inspiration from the feet of songbirds for perching on branch-shaped structures (37–40). Furthermore, grippers were used to attach to flat surfaces (32, 46) and, in some cases, also served as landing skids when opened (36). In general, passive and compliant grippers can wrap around structures (34, 39–41), whereas actuated grippers can actively grasp a structure to attach the UAV (33, 36).

Challenges

Approaches based on dry adhesive (41–44) or small needles (26–31) have only been demonstrated for extremely lightweight UAVs and require specific UAV design to allow proper positioning of the landing gear for perching. Therefore, these approaches are difficult to adapt to COTS UAVs or UAVs that carry a heavy sensor payload, such as a high-resolution camera. Also, although avian-inspired grippers can be mounted on COTS UAVs, most gripper-based approaches are limited to perching on cylindrical structures of a certain diameter (36).

As another very important component for perching, control has to address a challenging problem because the UAV needs to be positioned close to a structure. Different from flight in open space, this is often done with flight regimes involving high angle of attack (47), post stall (45), or aggressive (32, 46) maneuvering to bring the landing gear to the required attitude and location while the UAV reaches a flight condition that allows safe contacting on the structure.

In bioinspired approaches, this can be done directly from feedback without optimizing the flight trajectory explicitly before the flight (37, 38). For maneuvering while in contact with a pivot point on a structure, dynamic modeling of the different flight phases is necessary (49).

However, flight regimes for attaching and detaching are, in many cases, complex and are not covered by control for COTS UAVs. For instance, approaches that perch on walls and have the landing gear mounted below the UAV have to fly toward the wall and turn the bottom side forward for attachment (26–32, 46). Failure to attach will result in a critical flight condition close to an obstacle. These risks are shared with approaches that use a high angle of attack (47) and post stall (45) maneuvers for perching. Perching on walls can also require a mechanism-supported takeoff strategy that puts the UAV in a critical flight condition after detachment (26).

Many approaches are not focused on continuing the UAV's mission and can therefore lead to mission interruption when perching. For instance, in approaches that rely on surfaces for perching, the UAV has to comply to the surface's orientation (26–32, 46), which might obstruct sensors or communication devices. As a result, it is still challenging to enable perching capabilities in COTS UAVs under a wide range of circumstances without disrupting the mission or requiring risky and complex maneuvering that involves critical flight conditions.

A new paradigm for perching and resting

As mentioned above, we observed in nature that (perching) birds and bats have adapted to their habitats by developing prehensility and claws in their feet, which allows them to use a large variety of structures for support when perching (see Fig. 1). Instead of directly imitating, for instance, the feet of perching birds (passerine birds),

we propose a simplified and specialized solution for COTS UAVs. On the basis of four design principles, we designed a modularized and actuated landing gear framework for rotary-wing UAVs consisting of an actuated gripper module and a set of contact modules that are mounted on the gripper's fingers. The gripper module was mounted on the bottom side of the UAV and, for its weight and size, was compliant with a large range of COTS UAVs. Unlike previous approaches with grippers (37–40), our approach was not limited to cylindrical structures and did not require complex attachment maneuvers, such as a sideways approach (32, 46).

If a horizontal surface was available, the gripper module was opened and the stiff fingers were used as landing skids, similar to a bird landing on a flat rooftop. If a cylindrical structure was available, the UAV approached it from above such that the gripper module could grasp the structure, after which all motors could be suspended. This was directly inspired by how birds land on branches of trees onto which they then hold. For other types of structures, such as edges or corners of a building, strut, bar, or street sign, we relied on modularization, allowing us to flexibly design and fabricate contact modules that matched the specific structure. Through gripper actuation and position control, we then brought a suitable contact module to rest on the structure, and all or a part of the UAV's weight was supported by the structure, reducing the required lift. This modularization substantially increased the range of possible structures that can be exploited for perching and resting as compared with avian-inspired grippers. Although not inspired by nature and much more simple than the foot of a bird, the stiff fingers and contact modules were easier to manufacture and more robust and durable than avian-inspired grippers with several joints per finger.

Takeoff and landing are critical phases in a flight; for example, pigeons show complex patterns of wing strokes for acceleration and deceleration during maneuvers (50). Although we took inspiration from how birds and bats rest, we did not imitate their maneuvering for landing or taking off because the UAV as a rotary wing aircraft has substantial different flight characteristics from birds and bats with flapping wings. In contrast to previous approaches (32, 37, 38, 45–47), we developed an approach that relied on position control and reference poses only, without requiring complex control strategies. For perception, we present a proof-of-concept method that identified suitable structures for perching and resting from point cloud data of the environment.

Overall, we investigated four fundamental questions of UAV maneuvering in terms of the exploitation of external contacts: (i) how to design landing gears to facilitate UAVs to exploit contacts for perching and resting, (ii) how energy consumption and pose stability are affected by perching and resting, (iii) how the mission-relevant view ranges of UAVs are affected by different perching and resting actions, and (iv) the use cases and limitations of the proposed paradigm.

In experiments, we mounted our landing gear framework on a COTS UAV and demonstrated the efficacy of our design in enabling the desired perching and resting capabilities in a controlled laboratory environment. The experiments included perception of and perching and resting on different structures. During the experiments, the UAV was globally localized with an external measurement system. In this setting, we evaluated power consumption and pose stability during perching and resting for empirical comparison with hovering. Furthermore, we qualitatively studied the view ranges of different perching and resting actions on locations at heights and discuss other

potential usage in terms of the features enabled by perching and resting. Our experiment results show that the proposed paradigm not only reduces energy consumption but also enables UAVs to exploit external contacts with a variety of structures to facilitate mission execution, which, to the best of our knowledge, has not been extensively studied.

RESULTS

Our modularized and actuated landing gear framework is designed to be flexible and accommodate a wide range of applications. To demonstrate and evaluate the principles and efficacy of our design, we present a proof-of-concept study in which we designed and fabricated a landing gear for a DJI F450 quadrotor platform and tested the resulting perching and resting capabilities in a number of scenarios with different structures. Because most recent UAV applications involve load-carrying for videography or surveillance, we evaluated the perching and resting states in terms of (i) power consumption, (ii) pose stability, and (iii) view ranges.

We fabricated the gripper module's base and fingers using carbon fiber to keep the landing gear rigid and lightweight. The contact modules were 3D printed using the soft TangoBlackPlus material to facilitate contact compliance and stability for a wide range of environments. The weight of each part of our landing gear framework is listed in table S1. In the experiments, the environment was perceived using an externally placed Kinect One sensor, which provided point clouds in which we detected structures that allowed perching and resting. Once contact locations in the environment were identified, as shown by colored points in Figs. 2 and 4, the UAV was autonomously navigated on the basis of the localization provided by a VICON system. An example laboratory setup for our experiments is shown in Fig. 2.

Landing gear design

In this section, we first describe the design principles of the proposed modularized landing gear framework. On the basis of the principles, we demonstrate our example design and evaluate its performance.

Principles of landing gear design for perching and resting with COTS UAVs

To enable perching and resting under various circumstances while keeping the design versatile, we proposed to design landing gears obeying four principles:

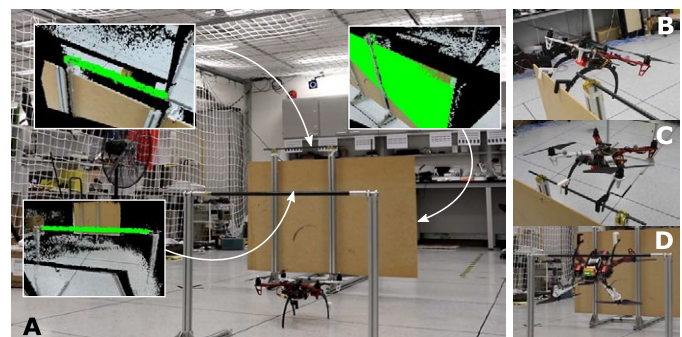


Fig. 2. Example actions with vision-based perching and resting location detection. (A) Laboratory environment and detected perching and resting locations. (B) Perching by hooking on a thin board (P_H). (C) Resting by hooking on a stick (R_H). (D) Perching by grasping around a stick (P_G).

1) The landing gear should be usable for landing on flat surfaces, mirroring the capabilities of most standard landing gears for COTS UAVs. This allows the UAVs to land and take off as usual COTS UAVs.

2) The landing gear should allow the UAVs to grasp or hook around structures of different scales. This allows the UAVs to turn all rotors off when perched

3) The landing gear should allow the UAVs to rest on different structures to provide lift support in the vertical direction. This allows the UAV to slow down or completely stop some of the rotors when resting by establishing stable contacts with the environment.

4) The landing gear should be mountable on a COTS UAV and be minimalistic in hardware, actuators, and control. This allows the user to design and replace parts of the landing gear without the need of reprogramming when working in different scenarios.

Following the principles, we demonstrate an example design consisting of an actuated gripper module that features principles 1, 2, and 4 and a set of contact modules that features principles 3 and 4.

Actuated gripper module

The actuated gripper module consists of servomotors, a set of fingers, and a base platform that attached to the bottom of the UAV. Our landing gear design for the DJI F450 UAV with three fingers is shown in Fig. 3. On the base platform, the three servomotors were installed to actuate open and close motions of the fingers. To ensure sufficient grasping forces, the three servomotors were adopted to actuate the fingers separately. However, the motors were controlled jointly for open and close actions with only one degree of freedom. In practice, all fingers can be actuated by a single motor as long as the provided torque is sufficient for the grasping actions. When the gripper was opened, the fingers enabled normal landing and takeoff from the ground because the fingertips were in level position under the UAV.

As seen in Fig. 3, the size of the landing gear is approximately identical to the UAV's dimensions. This enables the UAV to grasp structures of up to 0.2 m in radius. However, deciding on the dimensions of the gripper module involves trade-offs in the size of potential perching

structures, the gripper weight, undesired aerodynamical side effects, and collision-free maneuvering. A larger gripper can accommodate larger structures but can lead to loose contact for small structures. Our design made finger replacement easy, and it is recommended to design the fingers in appropriate sizes to achieve the tasks while avoiding undesired side effects. Additionally, the design of the gripper fingers should guarantee that it makes a closed loop when in close position, which ensures perching ability on all structures within the scale of the landing gear.

Contact modules

According to the design principles, we equipped the UAV with different contact modules that were easy to use, design, and fabricate. Inspired by the claws of birds, we designed the contact modules such that they were able to stabilize the UAV with different structures in the environment by contacting their modeled side, which acts similar to claws to hold onto small or thin structures. As shown in Fig. 3, contact modules were installed at the distal ends of the fingers, making them accessible to structures below the UAV. For resting, the gripper module was actuated to bring the contact module to the desired pose. This could be an open pose for contacts on one side of the UAV (Fig. 4A) or a closed pose for contacts below the center of the UAV (Fig. 4B). The contact modules themselves were not actuated for actively stabilizing the contacts. Instead, their shapes were adapted to achieve stable contacts against certain structures. On the basis of the minimalistic and modular design principles, the contact modules were exchangeable to provide more contact possibilities with a large variety of geometries.

In this work, we exemplify a few contact module designs that were based on the concept of contact primitives and fingertip surface optimization (51). The algorithm synthesized contact modules based on a set of example structures. As long as the provided examples sufficiently represented potential contact structures, the synthesized contact modules were able to stabilize the contacts. Figure 3 shows two contact modules (II and III) that were synthesized by the algorithm. Additionally, similar to claws that birds use to grasp and perch, we designed another L-shaped contact module, which, together with the finger, created a U-shaped claw. As shown in Fig. 3, this design

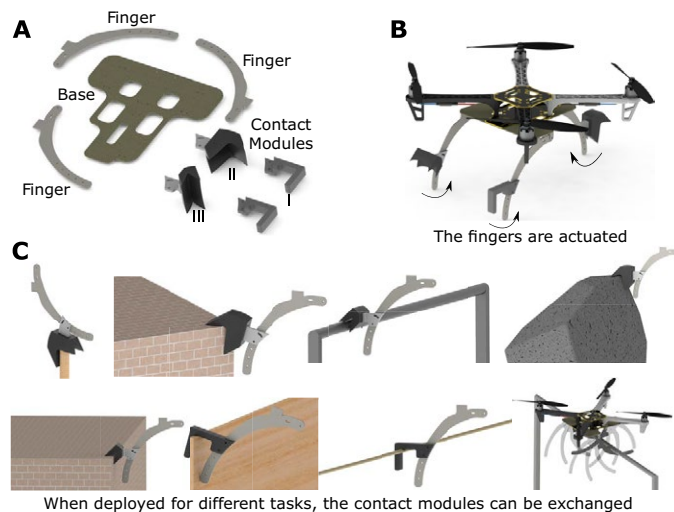


Fig. 3. An example landing gear design for DJI F450. (A) Example of the modularized landing gear design consisting of a base, three fingers, and three different contact modules. (B) Example of the installation of the designed modules on a DJI F450 platform. (C) Example perching and resting actions using different contact modules or the actuated gripper module.

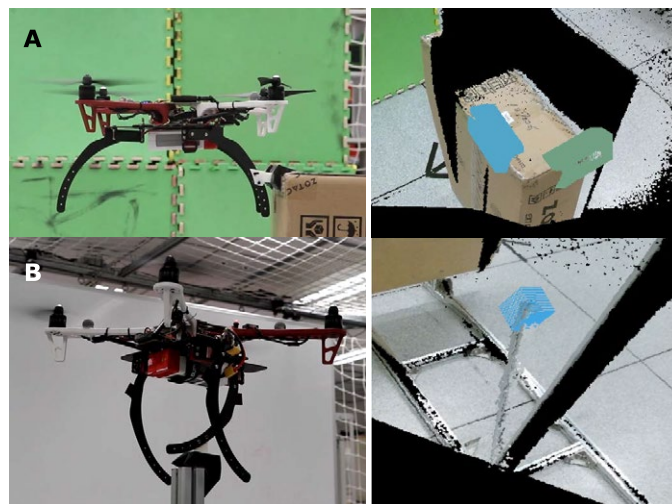


Fig. 4. Example resting actions with vision-based perching and resting location detection. (A) Resting on a box's edge (R_E). (B) Stand-resting on a stick (R_S).

allowed perching on thin structures onto which a UAV can hook itself using gravity.

Saving power by reducing motor action

In this work, we exemplify five perching or resting actions using the experimental UAV for demonstration and evaluation. As seen in Figs. 2 and 4, the actions were perching by hooking (P_H), perching by grasping (P_G), resting by hooking (R_H), resting on an edge (R_E), and stand-resting on a stick (R_S).

Power consumption is one of the major concerns for many UAV applications, and the main goal of our design was to save battery power by reducing motor action for generating lift. For this reason, we analyzed energy consumption in examples of perching and resting and compared them with the energy consumption while hovering in the air or above the floor.

If the UAV was perching by grasping around a structure (P_G) or hooking on a thin structure (P_H), as seen in Fig. 2, its weight was fully supported by the structure, and all the rotors could be turned off. Therefore, the energy consumption was 0.

When using a contact module below the center of the UAV for resting, as seen by the action R_S in Fig. 4, all the rotors still needed to be used for maintaining the balance. However, the rotors could be markedly slowed down because the load was mainly supported by the structure. When using contact modules for resting on a structure below the side of the UAV, as shown by the actions R_E in Fig. 4 and R_H in Fig. 2, the UAV had to only maintain two degrees of freedom, which were the rotation about the contact line and sliding along the contact line. In those cases, two rotors could be completely turned off.

Empirical results are reported in Fig. 5. The power consumption data were recorded from the point when the UAV had stabilized itself and ended when the UAV took off again. As can be seen from the figure, the stand-resting action R_S consumed the least energy because almost all the load was supported by the contact. When resting on the box edge R_E or on the stick R_H , power consumption was higher, which was up to about half of the energy consumption of the hovering action. It is worthwhile noting that these two resting actions consumed a bit more power than half of the hovering. This is because the UAV needed to counteract the ground effect when it was very close to other objects. Last, we could see that, when hovering near ground, due to the ground effect, the UAV consumed a little less energy than hovering in the air. In comparison with hovering in the air, R_S , R_E , and R_H saved 69, 46, and 41% power consumption, respectively.

Evaluating stability and view range

For many applications such as videography, surveillance, or object delivery at heights, stable positioning of the UAV over a period of

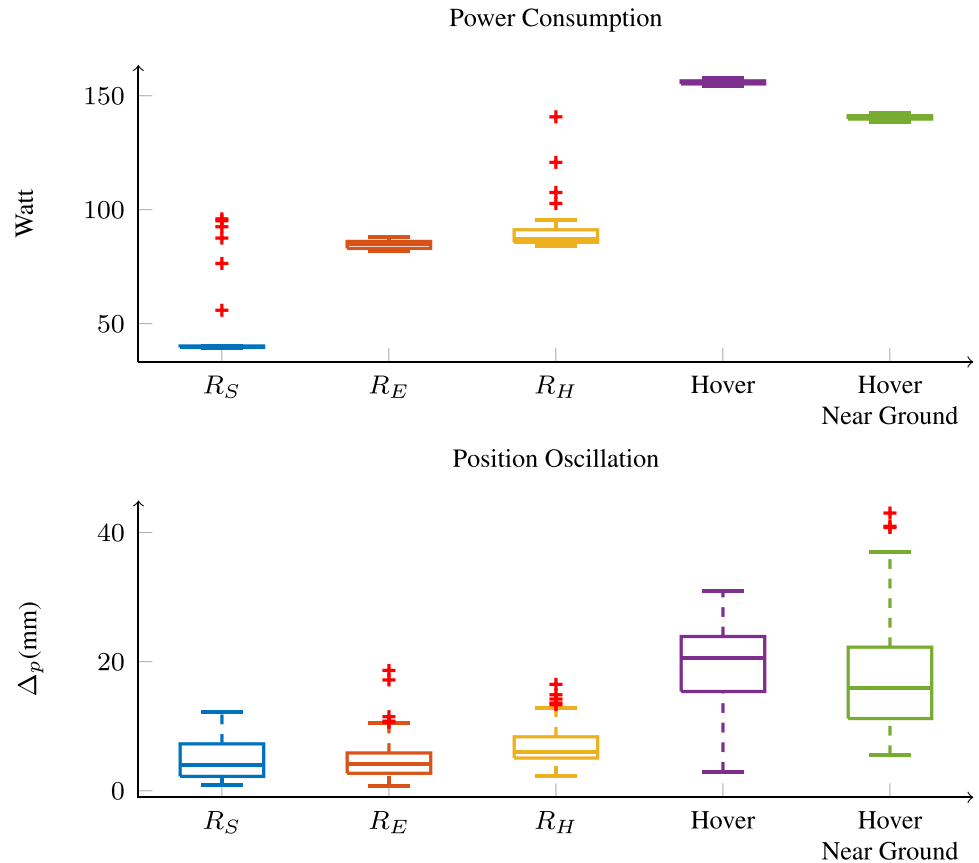


Fig. 5. Power consumption and stability evaluation results. For measuring the power consumption, we took the measurement directly from the motors without considering the power consumed by other electronics. On each box plot, the central mark indicates the median, and the bottom and top edges of the box indicate the 25th and 75th percentiles, respectively. The whiskers extend to the most extreme data points not considered outliers, and the outliers are plotted individually using the “+” symbol.

time is necessary. For this reason, we evaluate position oscillation Δ_p with respect to a reference location $\bar{p} = (\bar{x}, \bar{y}, \bar{z})$ for different perching and resting scenarios. For this, we define $\Delta_p = \frac{1}{T} \sum_{i=1}^T \sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2 + (z_i - \bar{z})^2}$, where (x_i, y_i, z_i) is the location sampled at time i , $1 \leq i \leq T$. Because the UAV's position was passively determined when all rotors were turned off, we only evaluated the position oscillation for resting actions when the stability was actively determined by the control of rotors.

As reported in Fig. 5, hovering results in oscillation were within a small range of about 2 cm. However, resting was even more stable and maintained the desired pose within about 5 mm. By checking the standard deviations, we could see that the standard deviations of resting were less than half of those of the hovering actions. These results show that resting can provide more stability while at the same time reducing power consumption.

Especially in perch-and-stare missions, the UAV's view range is a crucial concern when it is tasked to stare or watch over a certain area. However, landing on a flat elevated position such as a rooftop can significantly reduce the UAV's view range. Figure 6 shows how the rooftop occludes most of the view ranges below when a UAV lands on it. Compared with that, perching or resting as offered by the modularized landing gear framework can improve the view range. In

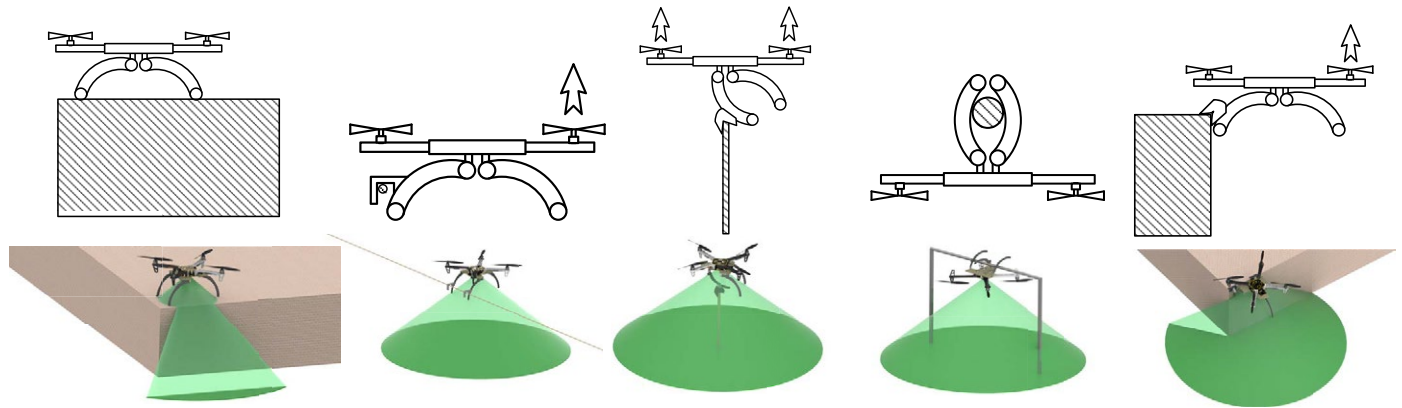


Fig. 6. Example view ranges of different perching and resting actions. The top row shows various perching and resting actions, with arrows indicating which rotors are still working for generating lift. The bottom row shows the corresponding view ranges rendered by green cones.

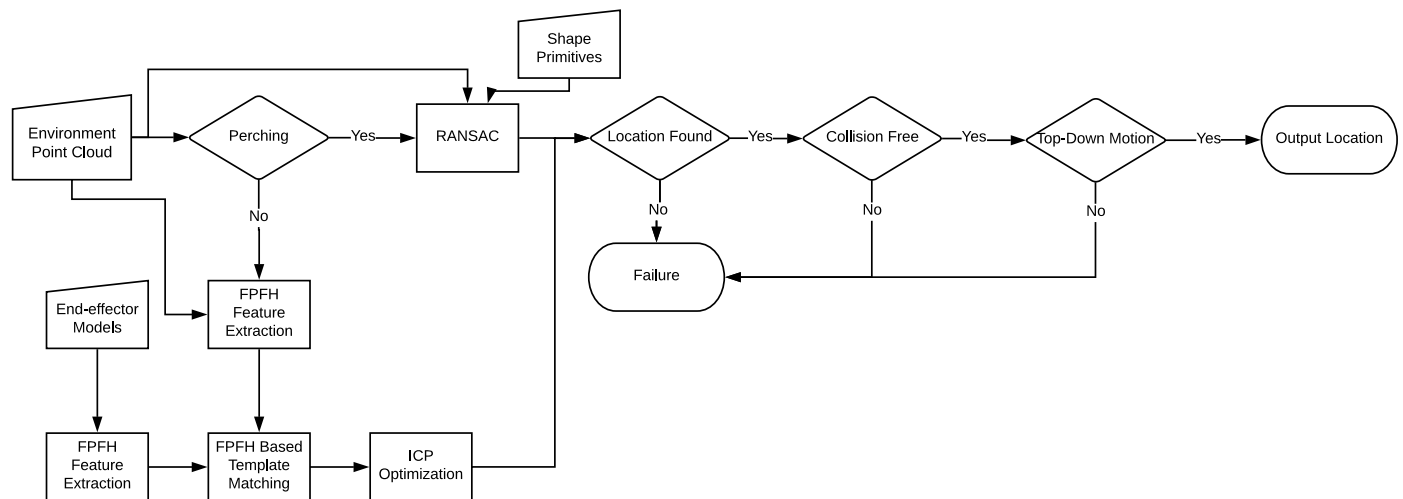


Fig. 7. Flowchart of the hybrid system for perching and resting location detection.

most cases, the UAV could fully observe the area below it without any occlusions. An exception was seen when a UAV rested on the edge of a building, which occluded about half of the view below. Nonetheless, it was still much better than a normal roof landing.

Upon using different perching or resting actions, the onboard camera could be configured accordingly to optimize the view. For example, when perching on a stick by using the actuated fingers to grasp, the UAV will finally be stabilized after it turns over around the stick and stops all its rotors. Hence, unlike most UAVs, which have the camera installed below the main frame, the camera, or an extra camera, should be installed on top of the UAV to achieve the view range when the UAV turns over and faces down.

DISCUSSION

In this section, we first give a brief summary of what has been proposed and evaluated. Thereafter, we discuss the limitations and implementation concerns of our design, the concerns in pose stability and energy consumption in relation to our paradigm, and the use cases of the proposed design framework.

In this work, we focused on the problem of enabling perching and resting for rotary-wing UAVs. First, we proposed to enable UAVs with

the capability of making and stabilizing contacts with the environment so as to obtain force supports from the contacts to be able to consume less battery energy while retaining the heights. For this, we developed a design framework of modularized and actuated landing gears consisting of an actuated gripper module and customized contact modules. The goal was to permit lower power consumption, better stability, and larger view ranges when the task was to be executed at fixed locations at heights. Following the four design principles, we designed an example landing gear for a DJI F450 quadrotor. The example design is composed of a base platform, three actuated fingers that were fabricated using carbon fibers, and three customized contact modules that were 3D printed using soft materials. The design resembles the basic functionalities of normal landing gears allowing landing and takeoff actions and is lightweight for the UAV to carry on board while not introducing much more extra power consumption.

We validated the example design by demonstrating perching and resting under laboratory conditions, such as perching by grasping and hooking, resting on an edge or stick, and stand-resting on a stick. The stability and power consumption of demonstrated actions were evaluated, and the results indicate that the featured actions can significantly reduce the power consumption while providing better stability comparing with normal actions. Additionally, we have

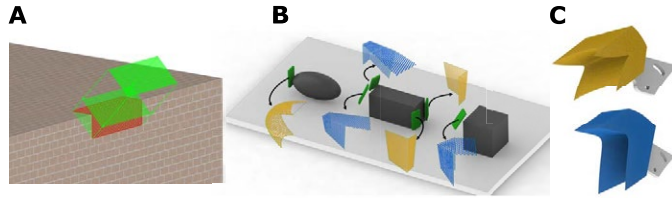


Fig. 8. Example contact area extraction for automatic contact module design. (A) Extraction of a contact area (red) based on the specified contact pose and size. (B) Shape primitives (black) and extracted contact areas (clustered in blue and yellow) used in the contact module design in this work. (C) Contact modules designed in terms of the clustered contact areas.

qualitatively shown that the featured actions provide much larger view ranges when working at heights, which can hardly be achieved by normal landing actions.

Limitations and implementation in practice

In this work, our experimental quadrotor was not equipped with onboard vision capability. The perching and resting locations were detected on the basis of the point cloud obtained by an external Kinect One sensor beforehand, and the UAV was navigated by a VICON system in the laboratory environment. In practice, when maneuvering a UAV in outdoor environments, the onboard visual perception is important to help the human operator to navigate or to enable the UAV to be more autonomous. When a UAV is tasked to autonomously execute perching or resting actions, an onboard visual sensor is required to enable the UAV to understand the environment, as well as to detect the locations where desired actions can be applied. As will be described shortly, given that the modularized landing gears were flexibly customized for accommodating a certain range of task requirements, the vision algorithms can be designed using template-based approaches that match geometrical features between the environment and the designs. Nevertheless, the vision-based detection approach is intrinsically limited in that it is not easy to acquire physical properties of the environment, such as the rigidity of the detected locations, which can affect the action stability. For addressing this problem, learning-based algorithms can be adopted to predict the physical properties. More reliably, active perception algorithms can be developed to conduct physical estimation by enabling the UAV to actively interact with the environment; for example, a UAV can use its contact modules to touch and press certain locations to acquire knowledge, which can potentially be obtained by additional sensors installed on the contact modules.

Additionally, in our experiments, our control strategy is to always navigate the UAV to a point above the perching or resting locations and then execute the action from top-down. However, in many tasks in reality, one can imagine a UAV working in confined environments, in which the perching or resting actions cannot be executed without a trajectory planning algorithm. As discussed in (38, 52), we could enable the UAV with trajectory planning to perch or rest in more difficult scenarios by bringing the UAV to the desired location without the top-down motion constraint. For instance, a UAV could perch on a tree branch by approaching it from the side and grasping it with the fingers when the region above that tree branch is occluded.

Pose stability and energy consumption

As the main goals of the proposed paradigm, pose stability and energy consumption have been evaluated using an example design in a labo-

ratory environment. The experimental results have shown that both perching and resting actions can significantly reduce the power consumption by exploiting force support from external contacts. In addition, using the same flight controller, we have seen that pose stability has been improved when external contacts were made. This can be explained by the fact that, when contacts were made between the contact modules and external structures, the degrees of freedom of the UAV's movement were reduced. Hence, the potential external disturbance was reduced, and more importantly, the flight controller could focus on balancing only the remaining degrees of freedom, mitigating the trade-off of keeping pose stabilities between different moving dimensions.

Nonetheless, we can foresee a variety of factors that can affect these two performance concerns. When a UAV is tasked to work in outdoor environments, wind disturbance and other aerodynamic uncertainties can be a major factor that affects pose stability. In this case, the flight controller will have to regulate the actuation inputs more intensively to keep the stability at a similar level, resulting in increased energy consumption. Moreover, the rigidity or mobility of contact locations can be another concern that affects the pose stability. For example, when resting by making contacts at a thin tree branch, although the UAV can gain force support to reduce energy consumption, stability is more difficult to keep because of the passive movement of the contacts, and the UAV will consume more motor energy in comparison with making contacts at rigid locations. To reduce the effect of physical uncertainties and to improve the energy performance, although not included in this work, we plan to design a tilt-pan connector between the main body of the UAV and the modular landing gear. By mechanically decoupling the movement of the UAV's main body from the landing gear or by actively compensating the disturbances at the connector, the pose stability can be further improved. Limited by the scope of this study, we leave this development to our future work.

Use cases

A UAV with perching and resting capabilities may enable many applications that are not possible otherwise. Besides that perching and resting can provide lower power consumption, better stability, and larger view ranges in many cases, which are very useful for perch-and-stare applications, the physical interaction with the environment enabled by such actions may additionally empower many more applications. For example, in aerial grasping (53–55), the maximum load is limited by the power provided by the rotors. However, once a UAV is perched, it will be able to lift markedly larger loads without requiring any power from the rotors.

When delivering objects to workers at heights, a UAV can perch or rest at some location near the worker for object pickup, or it can carry a pair of pulley and rope to perch at a certain location, such that object delivery can be achieved from both ends of the rope. While resting at the edge of a windowsill, a UAV will be able to deliver objects to someone inside, without the need to keep the rotors at the window side still working, so as to reduce the risk for humans to interact with a UAV. Overall, the ability of making contacts by resting or fixing itself by perching at heights may empower many applications that are related to load-lifting and that demand interaction.

MATERIALS AND METHODS

In addition to the design of the proposed modularized landing gears, this section briefly describes how to enable a UAV with such landing

gears to execute the perching and resting actions in reality. Concretely, we introduce how we implemented the vision algorithm to detect perching and resting locations, how the UAV was controlled, and how to automatically design contact modules based on example contacts.

Perching location detection and navigation

As the main focus, we concentrated on the design of modularized landing gears and evaluated our example design installed on a DJI F450 platform. For the experiments, we did not install an onboard camera for the UAV to detect perching locations or other sensors to navigate it in the environment. Instead, we 3D-scanned the laboratory environment beforehand and saved a point cloud of the environment. For perching location detection, we implemented a hybrid system based on the PCL (Point Cloud Library) (56) to detect feasible perching and resting locations. Concretely, as shown in Fig. 7, the system takes the environment point cloud as the input and first needs to decide whether a perching location or a resting location is desired. In practice, we always tried to find perching locations first and then looked for resting locations if the former was not available.

If some perching structures are desired, in addition to the environment point cloud, the system was provided with a set of shape primitives that were preferable for perching actions. In our examples, we showed perching by grasping on a stick and perching by hooking on a thin board. To detect such locations in a point cloud, we used the Random Sample Consensus (RANSAC) algorithm based on parameterized shape templates, and the results are shown in Fig. 2. For detecting resting locations, given that those actions rely on the customized contact modules, the detection is also based on the shapes of contact modules. As depicted in Fig. 7, the fast point feature histograms (FPFHs) were extracted from both the environment point cloud and the contact modules. Thereafter, we tried to register the contact modules to feasible locations in the environment and optimized the results using the iterative closest point (ICP) algorithm.

Once a perching or resting location was found, it needed to be verified by two additional steps. Because the UAV always approached those locations from top-down in an upright pose, we checked whether the surrounding area was collision free and whether the area allowed top-down approaching motions. As a negative example, for resting on an edge, the UAV could not stabilize itself by making contacts on the side edges of a box or a building; the edge had to be on top and approximately horizontal. Once a perching or resting location was confirmed, the coordinates of it were transformed to the VICON system, and the UAV navigated to apply the action.

Perching and resting control

To execute the actions for perching and resting, we applied a flight controller that mixed the position control and the attitude control of the UAV in a cascaded manner. The details of the controller design are provided in appendix S1. For perching actions, the UAV first navigated to the desired location; once the grasping or hooking actions were applied, the UAV turned off all rotors and stayed in the perching mode. If the UAV needed to turn over, we applied a proportional angular velocity controller to realize a smooth motion.

For resting actions, the UAV also needed to first navigate to the desired location. However, differently from the perching actions, the UAV only turned off or slowed down some of the rotors. In cases when one side of the UAV could totally rest on some structures, such as edge resting, the rotors at the corresponding side were turned off, and the rest of the rotors could still work to support the weight. In

another case when the UAV could not totally rest on any side, such as the stand-resting, the UAV could slow down the rotors but still needed some lift to keep the balance.

In both of the above cases, we aimed at minimizing the power consumption to stabilize the UAV at the desired pose. This was achieved through the cascaded controller using a shifted reference point. Concretely, denoted by $\mathbf{p} \in \mathbb{R}^3$, the location of the UAV at the resting location, if we command the UAV to stay at \mathbf{p} , the rotors will still work at full speed to realize the precise pose control. To automatically slow down the rotors while keeping the UAV at the desired resting pose to stabilize contacts, we introduced a shifting factor $\Delta_r \in \mathbb{R}^3$ to shift the reference point toward the direction from which the UAV will obtain the resting support. Once the UAV has reached the resting location \mathbf{p} , the reference point for the controller will be shifted to $\mathbf{p} - \Delta_r$, and the rotors at the supported side will be stopped. Because of the physical contacts, the UAV in practice was not able to achieve the shifted reference. However, it slowed down the rotors to try to approach $\mathbf{p} - \Delta_r$ while keeping the pose upright. Additionally, as the UAV tries to approach the shifted reference point, it will actively exert force at the contacts; this effect can further improve the stability of resting actions.

Contact module design

The contact modules were used to passively stabilize the contacts between the landing gear and the resting locations. Therefore, we aimed at generating contact modules with shapes that could maximally resemble the typical contact geometries available in the environment. To keep the design general enough to accommodate as many scenarios as possible, we adopted the fingertip design algorithm from (51).

Concretely, the contact module design was formulated as an optimization problem addressed in three steps. First, given a working environment of the UAV, we provided the algorithm with a set of example shape primitives, which were representatives for describing typical shape geometries in the working environment. Thereafter, as shown in Fig. 8, by specifying a set of example contact poses, the algorithm extracted a set of contact areas that could be potentially used for resting contacts in the environment and represented them as point clouds. Second, the algorithm automatically determined the number of clusters and then clustered the extracted contact areas into different groups in terms of the geometric similarities between them. Last, modeled by a parameterized 3D surface for each contact module, the algorithm optimized the module's surface shape by minimizing the differences between the surface and all the contact areas in the corresponding cluster. Hence, the optimized contact module's surface maximally resembled the geometric features of the potential contacts and improved the stability of contacts for the UAV to rest at the corresponding locations. For a more detailed explanation of this algorithm, we refer the readers to (51).

In this procedure, the more example contact areas that were provided to the algorithm, the more potential clusters of contact areas were produced, and so the number of designed contact modules. This enabled the UAVs to rest at a variety of different locations, because the contact modules could be exchanged when working in different environments. Additionally, although the designs are maximally resembling the geometric features of contact areas, there were always differences between the designed contact module and the real contact locations in the environment. To minimize the effects given by this difference, we suggest fabricating the contact modules using soft materials so that some small differences at contacts can be compensated to improve the stability of contacts.

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/28/eaau6637/DC1

Appendix S1. Flight controller design.

Table S1. Weights of parts.

Movie S1. Perching and resting actions test.

References (57, 58)

REFERENCES AND NOTES

1. C. Goerzen, Z. Kong, B. Mettler, A survey of motion planning algorithms from the perspective of autonomous UAV guidance. *J. Intell. Robot. Appl.* **57**, 65–100 (2010).
2. T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. L. Grix, F. Ruess, M. Suppa, D. Burschka, Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue. *IEEE Robot. Autom. Mag.* **19**, 46–56 (2012).
3. S. Rathinam, P. Almeida, Z. Kim, S. Jackson, A. Tinka, W. Grossman, R. Sengupta, Autonomous searching and tracking of a river using an uav, in *American Control Conference* (IEEE, 2007), pp. 359–364.
4. A. Ryan, M. Zennaro, A. Howell, R. Sengupta, J. K. Hedrick, An overview of emerging results in cooperative uav control, in *IEEE Conference on Decision and Control* (IEEE, 2004), vol. 1, pp. 602–607.
5. J.-H. Kim, S. Sukkarieh, S. Wishart, Real-time navigation, guidance, and control of a uav using low-cost sensors, in *Field and Service Robotics* (Springer, 2003), pp. 299–309.
6. S. Lange, N. Sunderhauf, P. Protzel, A vision based onboard approach for landing and position control of an autonomous multirotor uav in gps-denied environments, in *IEEE International Conference on Advanced Robotics* (IEEE, 2009), pp. 1–6.
7. W. Pisano, D. Lawrence, P. Gray, Autonomous uav control using a 3-sensor autopilot, in *AIAA Infotech@Aerospace Conference and Exhibit* (AIAA, 2007), p. 2756.
8. S. Ross, N. Melik-Barkhudarov, K. S. Shankar, A. Wendel, D. Dey, J. A. Bagnell, M. Hebert, Learning monocular reactive uav control in cluttered natural environments, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2013), pp. 1765–1772.
9. G. Zhou, J. Yuan, I.-L. Yen, F. Bastani, Robust real-time uav based power line detection and tracking, in *IEEE International Conference on Image Processing* (IEEE, 2016), pp. 744–748.
10. I. Maza, K. Kondak, M. Bernard, A. Ollero, Multi-uav cooperation and control for load transportation and deployment, in *International Symposium on UAVs* (Springer, 2009), pp. 417–449.
11. P. E. Pounds, D. R. Bersak, A. M. Dollar, Stability of small-scale UAV helicopters and quadrotors with added payload mass under PID control. *Autonom. Robot.* **33**, 129–142 (2012).
12. E. Semsch, M. Jakob, D. Pavlicek, M. Pechoucek, Autonomous uav surveillance in complex urban environments, in *IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology* (IEEE Computer Society, 2009), pp. 82–85.
13. D. Santano, H. Esmaili, Aerial videography in built heritage documentation: The case of post-independence architecture of Malaysia, in *International Conference on Virtual Systems & Multimedia* (IEEE, 2014), pp. 323–328.
14. P. Doherty, P. Rudol, A uav search and rescue scenario with human body detection and geolocalization, in *Australasian Joint Conference on Artificial Intelligence* (Springer, 2007), pp. 1–13.
15. G. Li, X. Zhou, J. Yin, Q. Xiao, An uav scheduling and planning method for postdisaster survey, in *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* (2014), vol. 40, p. 169.
16. Q. Wen, H. He, X. Wang, W. Wu, L. Wang, F. Xu, P. Wang, T. Tang, Y. Lei, Uav remote sensing hazard assessment in Zhouqu debris flow disaster, in *Remote Sensing of the Ocean, Sea Ice, Coastal Waters, and Large Water Regions* (International Society for Optics and Photonics, 2011), vol. 8175, p. 817510.
17. I. Bekmezci, O. K. Sahingoz, Ş. Temel, Flying ad-hoc networks (FANETs): A survey. *Ad Hoc Networks* **11**, 1254–1270 (2013).
18. A. Puri, “A survey of unmanned aerial vehicles (UAV) for traffic surveillance” (Technical Report, Department of Computer Science and Engineering, University of South Florida, 2005), pp. 1–29.
19. A. Chakrabarty, J. Langelaan, Flight path planning for uav atmospheric energy harvesting using heuristic search, in *AIAA Guidance, Navigation, and Control Conference* (AIAA, 2010), p. 8033.
20. B. Sumantri, N. Uchiyama, S. Sano, Least square based sliding mode control for a quad-rotor helicopter and energy saving by chattering reduction. *Mech. Syst. Sig. Process.* **66**, 769–784 (2016).
21. F. Morbidi, R. Cano, D. Lara, Minimum-energy path generation for a quadrotor uav, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2016), pp. 1492–1498.
22. X. Lyu, H. Gu, J. Zhou, Z. Li, S. Shen, F. Zhang, A hierarchical control approach for a quadrotor tail-sitter vtol uav and experimental verification, in *IEEE International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2017).
23. X. Lyu, H. Gu, J. Zhou, Z. Li, S. Shen, F. Zhang, Design and implementation of a quadrotor tail-sitter vtol uav, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017), pp. 3924–3930.
24. H. Gu, X. Cai, J. Zhou, Z. Li, S. Shen, F. Zhang, A coordinate descent method for multidisciplinary design optimization of electric-powered winged uavs, in *International Conference on Unmanned Aircraft Systems* (IEEE, 2018), pp. 1189–1198.
25. T. W. Danko, A. Kellas, P. Y. Oh, Robotic rotorcraft and perch-and-stare: Sensing landing zones and handling obscurants, in *IEEE International Conference on Advanced Robotics* (IEEE, 2005), pp. 296–302.
26. M. T. Pope, C. W. Kimes, H. Jiang, E. W. Hawkes, M. A. Estrada, C. F. Kerst, M. R. Cutkosky, A multimodal robot for perching and climbing on vertical outdoor surfaces. *IEEE Trans. Robot.* **33**, 38–48 (2017).
27. M. Kovač, J. Germann, C. Hürzeler, R. Y. Siegwart, D. Floreano, A perching mechanism for micro aerial vehicles. *J. Micro-Nano Mechatron.* **5**, 77–91 (2009).
28. A. L. Desbiens, M. R. Cutkosky, Landing and perching on vertical surfaces with microspines for small unmanned air vehicles. *J. Intell. Robot. Syst.* **57**, 313 (2010).
29. A. Lussier Desbiens, A. T. Asbeck, M. R. Cutkosky, Landing, perching and taking off from vertical surfaces. *Int. J. Robot. Res.* **30**, 355–370 (2011).
30. A. L. Desbiens, A. T. Asbeck, M. R. Cutkosky, Scansorial landing and perching, in *Robotics Research* (Springer, 2011), pp. 169–184.
31. D. Mehanovic, J. Bass, T. Courteau, D. Rancourt, A. L. Desbiens, Autonomous thrust-assisted perching of a fixed-wing uav on vertical surfaces, in *Conference on Biomimetic and Biohybrid Systems* (Springer, 2017), pp. 302–314.
32. J. Thomas, M. Pope, G. Loianno, E. W. Hawkes, M. A. Estrada, H. Jiang, M. R. Cutkosky, V. Kumar, Aggressive flight with quadrotors for perching on inclines surfaces. *J. Mech. Robot.* **8**, 051007 (2016).
33. W. R. Roderick, H. Jiang, S. Wang, D. Lentink, M. R. Cutkosky, Bioinspired grippers for natural curved surface perching, in *Conference on Biomimetic and Biohybrid Systems* (Springer, 2017), pp. 604–610.
34. I.-W. Park, T. Smith, H. Sanchez, S. W. Wong, P. Piacenza, M. Ciocarlie, Developing a 3-dof compliant perching arm for a free-flying robot on the international space station, in *IEEE International Conference on Advanced Intelligent Mechatronics* (IEEE, 2017), pp. 1135–1141.
35. C. Luo, L. Yu, P. Ren, A vision-aided approach to perching a bioinspired unmanned aerial vehicle. *IEEE Trans. Ind. Electron.* **65**, 3976–3984 (2018).
36. M. A. Erbil, S. D. Prior, A. J. Keane, Design optimisation of a reconfigurable perching element for vertical take-off and landing unmanned aerial vehicles. *Int. J. Micro Air Veh.* **5**, 207–228 (2013).
37. Z. Zhang, P. Xie, O. Ma, Bio-inspired trajectory generation for uav perching, in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (IEEE, 2013), pp. 997–1002.
38. Z. Zhang, P. Xie, O. Ma, Bio-inspired trajectory generation for UAV perching movement based on tau theory. *Int. J. Adv. Robot. Syst.* **11**, 141 (2014).
39. C. E. Doyle, J. J. Bird, T. A. Isom, C. J. Johnson, J. C. Kallman, J. A. Simpson, R. J. King, J. J. Abbott, Avian-inspired passive perching mechanism for robotic rotorcraft, in *IEEE International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2011), pp. 4975–4980.
40. C. E. Doyle, J. J. Bird, T. A. Isom, An avian-inspired passive mechanism for quadrotor perching. *IEEE/ASME Trans. Mechatron.* **18**, 506–517 (2013).
41. H. Prahlad, R. Pelrine, S. Stanford, J. Marlow, R. Kornbluh, Electrodeposited robots—Wall climbing robots enabled by a novel, robust, and electrically controllable adhesion technology, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2008), pp. 3028–3033.
42. M. A. Graule, P. Chirattananon, S. B. Fuller, N. T. Jafferis, K. Y. Ma, M. Spenko, R. Kornbluh, R. J. Wood, Perching and takeoff of a robotic insect on overhangs using switchable electrostatic adhesion. *Science* **352**, 978–982 (2016).
43. H. Jiang, M. T. Pope, E. W. Hawkes, D. L. Christensen, M. A. Estrada, A. Parlier, R. Tran, M. R. Cutkosky, Modeling the dynamics of perching with opposed-grip mechanisms, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2014), pp. 3102–3108.
44. L. Daler, A. Klaptocz, A. Briod, M. Sitti, D. Floreano, A perching mechanism for flying robots using a fibre-based adhesive, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2013), pp. 4433–4438.
45. J. Moore, R. Tedrake, Control synthesis and verification for a perching uav using lqr-trees, in *IEEE Annual Conference on Decision and Control* (IEEE, 2012), pp. 3707–3714.
46. J. Thomas, G. Loianno, M. Pope, E. W. Hawkes, M. A. Estrada, H. Jiang, M. R. Cutkosky, V. Kumar, Planning and control of aggressive maneuvers for perching on inclined and vertical surfaces, in *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (American Society of Mechanical Engineers, 2015).
47. R. Cory, R. Tedrake, Experiments in fixed-wing uav perching, in *AIAA Guidance, Navigation and Control Conference and Exhibit* (AIAA, 2008), p. 7256.
48. K. Zhang, P. Chermprayong, T. Alhina, R. Siddall, M. Kovac, Spidermav: Perching and stabilizing micro aerial vehicles with bio-inspired tensile anchoring systems, in *IEEE International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2017), pp. 6849–6854.

49. Q. Delamare, P. R. Giordano, A. Franchi, Toward aerial physical locomotion: The contact-fly-contact problem. *IEEE Robot. Autom. Lett.* **3**, 1514 (2018).
50. A. M. Berg, A. A. Biewener, Wing and body kinematics of takeoff and landing flight in the pigeon (*Columba livia*). *J. Exp. Biol.* **213**, 1651–1658 (2010).
51. H. Song, M. Y. Wang, K. Hang, Fingertip surface optimization for robust grasping on contact primitives. *IEEE Robot. Autom. Lett.* **3**, 742–749 (2018).
52. D. Mellinger, N. Michael, V. Kumar, Trajectory generation and control for precise aggressive maneuvers with quadrotors. *Int. J. Robot. Res.* **31**, 664 (2012).
53. D. Mellinger, M. Shomin, N. Michael, V. Kumar, Cooperative grasping and transport using multiple quadrotors, in *Distributed Autonomous Robotic Systems* (Springer, 2013), pp. 545–558.
54. S.-J. Kim, D.-Y. Lee, G.-P. Jung, K.-J. Cho, An origami-inspired, self-locking robotic arm that can be folded flat. *Sci. Robot.* **3**, eaar2915 (2018).
55. P. E. I. Pounds, D. R. Bersak, A. M. Dollar, Grasping from the air: Hovering capture and load stability, in *IEEE International Conference on Robotics and Automation (ICRA)* (2011), pp. 2491–2498.
56. R. B. Rusu, S. Cousins, 3D is here: Point Cloud Library (PCL), in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2011), pp. 1–4.
57. D. Mellinger, V. Kumar, Minimum snap trajectory generation and control for quadrotors, in *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2011), pp. 2520–2525.
58. J. Solà, Quaternion kinematics for the error-state Kalman filter. arXiv:1711.02508 [cs.RO] (3 November 2017).
- Funding:** This work was supported by the Knut and Alice Wallenberg Foundation, the Swedish Research Council, HKUST Initiation grant 16EG09, and the Hong Kong Innovation Technology Fund (ITS/334/15FP). **Author contributions:** K.H. proposed the design principles, designed the modular landing gear, implemented the vision detector, analyzed data, and wrote part of the manuscript. X.L. designed part of the landing gear and performed all the experiments. H.S. designed contact modules and fabricated the landing gear. J.A.S. advised for the design principles and the experiment design, formulated the scientific questions, analyzed data, and wrote part of the paper. A.M.D. advised and supervised the project. D.K. provided funding and supervised the project. F.Z. provided funding and supervised the project. **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 or the Supplementary Materials. Additional requests for information should be addressed to K.H.
- Submitted 3 July 2018
Accepted 30 January 2019
Published 13 March 2019
10.1126/scirobotics.aau6637
- Citation:** K. Hang, X. Lyu, H. Song, J. A. Stork, A. M. Dollar, D. Kragic, F. Zhang, Perching and resting—A paradigm for UAV maneuvering with modularized landing gears. *Sci. Robot.* **4**, eaau6637 (2019).

MEDICAL ROBOTS

Autonomous robotic intracardiac catheter navigation using haptic vision

G. Fagogenis¹, M. Mencattelli¹, Z. Machaidze¹, B. Rosa², K. Price¹, F. Wu³, V. Weixler¹, M. Saeed¹, J. E. Mayer¹, P. E. Dupont^{1*}

Copyright © 2019
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

Although all minimally invasive procedures involve navigating from a small incision in the skin to the site of the intervention, it has not been previously demonstrated how this can be performed autonomously. To show that autonomous navigation is possible, we investigated it in the hardest place to do it—inside the beating heart. We created a robotic catheter that can navigate through the blood-filled heart using wall-following algorithms inspired by positively thigmotactic animals. The catheter uses haptic vision, a hybrid sense using imaging for both touch-based surface identification and force sensing, to accomplish wall following inside the blood-filled heart. Through in vivo animal experiments, we demonstrate that the performance of an autonomously controlled robotic catheter rivaled that of an experienced clinician. Autonomous navigation is a fundamental capability on which more sophisticated levels of autonomy can be built, e.g., to perform a procedure. Similar to the role of automation in a fighter aircraft, such capabilities can free the clinician to focus on the most critical aspects of the procedure while providing precise and repeatable tool motions independent of operator experience and fatigue.

INTRODUCTION

Minimally invasive surgery reduces the trauma associated with traditional open surgery, resulting in faster recovery time, fewer wound infections, reduced postoperative pain, and improved cosmesis (1). The trauma of open-heart surgery is particularly acute because it involves cutting and spreading the sternum to expose the heart. Nonetheless, an important additional step to reducing procedural trauma and risk in cardiac procedures is to develop ways to perform repairs without stopping the heart and placing the patient on cardiopulmonary bypass.

To this end, many specialized devices have been designed that replicate the effects of open surgical procedures, but which can be delivered by catheter. These include transcatheter valves (2), mitral valve neochords (3), occlusion devices (4), stents (5), and stent grafts (6). To deploy these devices, catheters are inserted either into the vasculature (e.g., femoral vein or artery) or, via a small incision between the ribs, directly into the heart through its apex.

From the point of insertion, the catheter must be navigated to the site of the intervention inside the heart or its vessels. Beating-heart navigation is particularly challenging because blood is opaque and cardiac tissue is moving. Despite the difficulties of navigation, however, the most critical part of the procedure is device deployment. This is the component when the judgment and expertise of the clinician are most crucial. Much like the autopilot of a fighter jet, autonomous navigation can relieve the clinician from performing challenging, but routine, tasks so that they can focus on the mission-critical components of planning and performing device deployment.

To safely navigate a catheter, it is necessary to be able to determine its location inside the heart and to control the forces it applies to the tissue. In current clinical practice, forces are largely controlled by touch, whereas catheter localization is performed using fluoroscopy. Fluoroscopy provides a projective view of the catheter, but it does

not show soft tissue and exposes the patient and clinician to radiation. Ultrasound, which enables visualization of soft tissue and catheters, is often used during device deployment, but the images are noisy and of limited resolution. In conjunction with heart motion, this makes it difficult to precisely position the catheter tip with respect to the tissue.

The limitations of existing cardiac imaging prompted us to seek an alternate approach. In nature, wall following—tracing object boundaries in one's environment—is used by certain insects and vertebrates as an exploratory mechanism in low-visibility conditions to ameliorate their localization and navigational capabilities in the absence of visual stimuli. Positively thigmotactic animals, which attempt to preserve contact with their surroundings, use wall following in unknown environments as an incremental map-building function to construct a spatial representation of the environment. Animals initially localize new objects found by touch in an egocentric manner, i.e., the object's relative position to the animal is estimated; however, later, more complex spatial relations can be learned, functionally resembling a map representation (7, 8). These animals often sample their environment by generating contact such as through rhythmically controlled whisker motion, as reported in rodents (9), or antennae manipulations in cockroaches (10) and blind crayfish (11).

RESULTS

Inspired by this approach, we designed positively thigmotactic algorithms that achieve autonomous navigation inside the heart by creating low-force contact with the heart tissue and then following tissue walls to reach a goal location. To enable wall following while also locally recapturing the detailed visual features of open surgery, we introduced a sensing modality at the catheter tip that we call “haptic vision.” Haptic vision combines intracardiac endoscopy, machine learning, and image processing algorithms to form a hybrid imaging and touch sensor—providing clear images of whatever the catheter tip is touching while also identifying what it is touching (e.g., blood, tissue, and valve) and how hard it is pressing (Fig. 1A).

¹Boston Children's Hospital, Harvard Medical School, Boston, MA 02115, USA.

²Cube, Université de Strasbourg, CNRS, Strasbourg, France. ³Taipei Veterans General Hospital, Taipei, Taiwan.

*Corresponding author. Email: pierre.dupont@childrens.harvard.edu

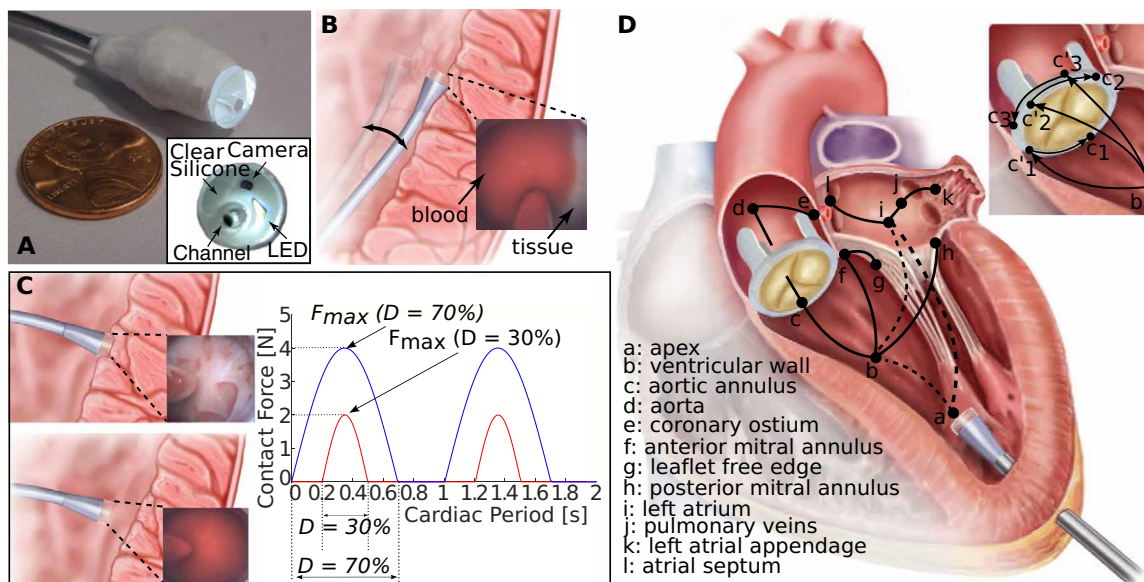


Fig. 1. Autonomous intracardiac navigation using haptic vision. (A) Haptic vision sensor composed of millimeter-scale camera and LED encased in silicone optical window with working channel for device delivery. Endoscope acts as a combined contact and imaging sensor with optical window displacing blood between camera and tissue during contact. (B) Continuous contact mode in which catheter tip is pressed laterally against heart wall over entire cardiac motion cycle. Contact force is controlled on the basis of the amount of tissue visible at the edge of optical window as shown in inset. (C) Intermittent contact mode in which catheter is in contact with heart wall for a specified fraction, D , of the cardiac period (contact duty cycle). Insets show corresponding haptic vision images in and out of contact. Maximum contact force relates to contact duty cycle, D , as shown on plot and is controlled by small catheter displacements orthogonal to the heart wall. (D) Wall-following connectivity graph. Vertices, defined in legend, denote walls and cardiac features. Solid arcs indicate connectivity of walls and features, and dashed arcs indicate that noncontact transition is required. Wall-following paths through heart can be constructed as sequence of connected vertices. Paravalvular leak closure experiments described in the paper consist of paths $a \rightarrow b \rightarrow c_i \rightarrow c_i, i = \{1, 2, 3\}$.

We used haptic vision as the sole sensory input to our navigation algorithms to achieve wall following while also controlling the forces applied by the catheter tip to the tissue. We evaluated autonomous navigation through in vivo experiments and compared it with operator-controlled robot motion and with manual navigation.

For wall following, we exploited the inherent compliance of the catheter to implement two control modes based on continuous and intermittent contact. Continuous contact can often be safely maintained over the cardiac cycle when the catheter tip is pressed laterally against the tissue because catheters are highly compliant in this direction (Fig. 1B). Intermittent contact can be necessary when there is substantial tissue motion and the catheter is pressed against the tissue along its stiffer longitudinal axis (Fig. 1C).

In both the continuous and the intermittent contact modes, the robot acted to limit the maximum force applied to the tissue using a haptic vision-based proxy for force. In the continuous contact mode, catheter position with respect to the tissue surface was adjusted to maintain a specified contact area on the catheter tip (Fig. 1B) corresponding to a desired force. In the intermittent contact mode, catheter position with respect to the tissue surface was adjusted to maintain a desired contact duty cycle—the fraction of the cardiac cycle during which the catheter was in tissue contact (Fig. 1C). The relationship between contact duty cycle and maximum tissue force was investigated experimentally as described in the section below titled “In-vivo calibration of contact duty cycle versus maximum tissue force.” Complex navigation tasks can be achieved by following a path through a connectivity graph (Fig. 1D) and selecting between continuous and intermittent contact modes along that path based on contact compliance and the amplitude of tissue motion.

We have implemented autonomous navigation based solely on haptic vision sensing and demonstrated the potential of the approach in the context of a challenging beating-heart procedure, aortic paravalvular leak closure. Paravalvular leaks occur when a gap opens between the native valve annulus and the prosthetic valve (12, 13). Transcatheter leak closure involves sequentially navigating a catheter to the leak, passing a wire from the catheter through the gap, and then deploying an expanding occluder device inside the gap (Fig. 2A). This procedure is currently manually performed using multimodal imaging (electrocardiogram-gated computed tomographic angiography, transthoracic and transesophageal echo preoperatively, and echocardiography and fluoroscopy intraoperatively) and requires 29.9 ± 24.5 min of fluoroscopic x-ray exposure (14).

To perform paravalvular leak closure, we designed a robotic catheter (Fig. 2B) for entering through the apex of the heart into the left ventricle, navigating to the aortic valve and deploying an occluder into the site of a leak (Fig. 1D). We created a porcine paravalvular leak model by replacing the native aortic valve with a bioprosthesis incorporating three leaks (Fig. 2C). Using leak locations determined from preoperative imaging, the catheter could either navigate autonomously to that location, or the clinician could guide it there (Fig. 2B). Occluder deployment was performed under operator control.

During autonomous catheter navigation to the leak location, both continuous and intermittent contact modes were used (Fig. 1D). For navigation from the heart’s apex to the aortic valve, the robot first located the ventricular wall ($a \rightarrow b$) and then followed it to the aortic valve using the continuous contact mode ($b \rightarrow c$). Because the valve annulus displaces by several centimeters over the cardiac cycle

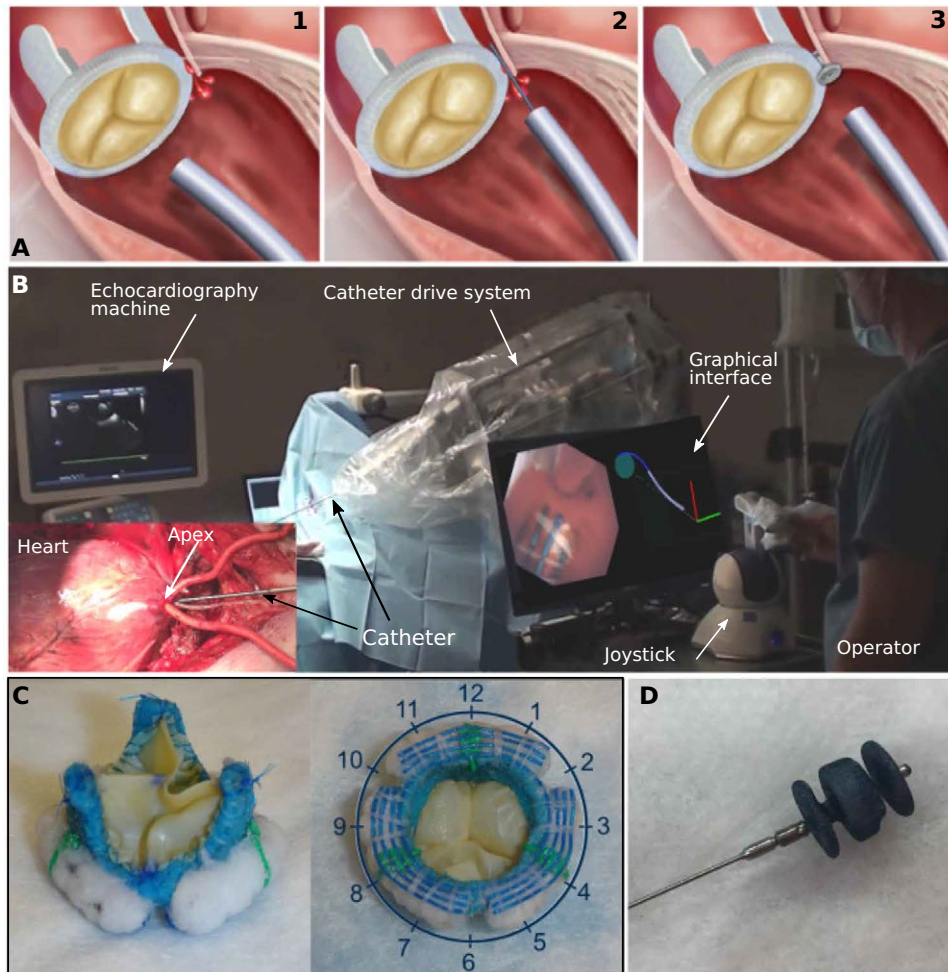


Fig. 2. Paravalvular leak closure experiments. (A) Current clinical approach to paravalvular leak closure: image 1, catheter approaches valve; image 2, wire is extended from catheter to locate leak; image 3, vascular occluder is deployed inside leak channel. Although transapical access is illustrated, approaching the valve from the aorta via transfemoral access is common. (B) Robotic catheter in operating room. Graphical interface displays catheter tip view and geometric model of robot and valve annulus. (C) Two views of bioprosthetic aortic valve designed to produce three paravalvular leaks at 2, 6, and 10 o'clock positions. Blue sutures are used to detect tangent to annulus. Green sutures are used to estimate valve rotation with respect to robot. (D) Vascular occluder (AMPLATZER Vascular Plug II, St. Jude Medical, Saint Paul, MN) used to plug leaks.

along the axis of the catheter, the robot switched to the intermittent contact mode once it detected that it had reached the aortic valve. It then navigated its tip around the perimeter of the valve annulus to the leak location specified from preoperative imaging ($c_i' \rightarrow c_i$; Fig. 1D, inset).

Switching between continuous and intermittent contact modes depends on the robot recognizing the tissue type it is touching. We implemented the capability for the catheter to distinguish the prosthetic aortic valve from blood and tissue using a machine learning classification algorithm. The classification algorithm first identified a collection of “visual words,” which consisted of visual features shared between multiple images in a set of prelabeled training images, and learned the relationship between how often these visual features occurred and what the image depicted—in this case, the prosthetic valve or blood and tissue. Implementation details and performance evaluation of the classification algorithm can be found in the Materials and Methods.

Navigation on the annulus of the aortic valve to the location of a leak requires two capabilities. The first is to maintain the appropriate radial distance from the center of the valve. The second is to be able to move to a specified angular location on the annulus. For robust control of radial distance, we integrated colored sutures into the bioprosthetic valve annulus that enable the navigation algorithm to compute the tangent direction of the annulus (Fig. 2C). Moving to a specific angular location requires the robot to estimate its current location on the annulus, to determine the shortest path around the valve to its target location, and to detect when the target has been reached. We programmed the robot to build a geometric model of the valve as it navigates. On the basis of the estimated tangent direction of the valve annulus, as well as basic knowledge of the patient and robot position on the operating table, the robot could estimate its clock face position on the valve (Fig. 2C). To account for valve rotation relative to the robot due to variability in patient anatomy and positioning, we incorporated radially oriented colored registration sutures spaced 120° apart. As the catheter navigated along the annulus and detected the registration sutures, it updated its valve model to refine the estimate of its location on the valve.

To evaluate the autonomous navigation algorithms, we performed *in vivo* experiments comparing autonomous navigation with teleoperated (i.e., joystick-controlled) robotic navigation. We also compared these two forms of robotic navigation with our results in (15), describing manual navigation of a handheld

catheter. In all cases, the only sensing used consisted of the video stream from the tip-mounted endoscope, kinesthetic sensing of the robot/human, and force sensing of the human (handheld). At the end of each experiment, we opened the heart, examined the ventricular walls for bruising or other tissue damage, and found none.

We first compared success rate and navigation time for autonomous navigation (Fig. 1D, $a \rightarrow b \rightarrow c_i'$) from the apex of the left ventricle to the aortic annulus (five animals, 90 trials), with teleoperated control (three animals, 9 trials) and with manual control (three animals, 13 trials; Fig. 3A). Autonomous navigation consisted of first moving to a wall of the ventricle specified by clock position (Fig. 2C) and then following that wall to the valve using the continuous contact control mode. Autonomous navigation was successful 99% of the time (89 of 90 trials). Autonomous control was faster than teleoperated control and with a smaller variance but slower than manual control.

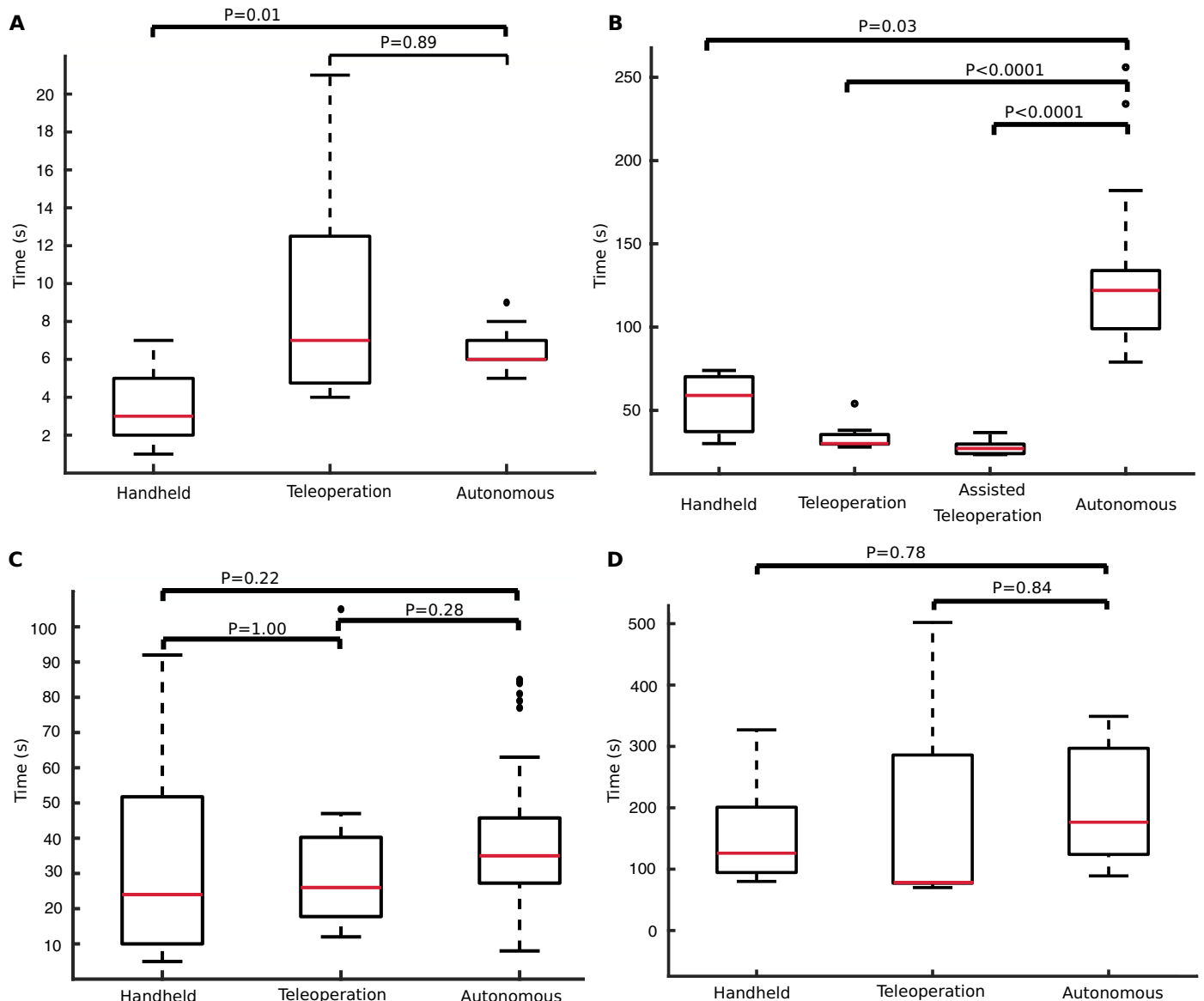


Fig. 3. In vivo navigation completion times. (A) Navigation from apex of the left ventricle to the aortic annulus (Fig. 1D, $a \rightarrow b \rightarrow c$). (B) Circumnavigation of the entire aortic valve annulus (e.g., $c_1' \rightarrow c_1 \rightarrow c_2 \rightarrow c_3' \rightarrow c_2' \rightarrow c_3 \rightarrow c_1'$; Fig. 1D, inset). (C) Navigation from apex to paravalvular leak (Fig. 1D, $a \rightarrow b \rightarrow c_i' \rightarrow c_i$). (D) Deployment of vascular occluder. Red bars indicate median, box edges are 25th and 75th percentiles, whiskers indicate range, and dots denote outliers. *P* values computed as described in the “Statistical analysis” section.

Next, we investigated the ability of the controller to navigate completely around the valve annulus using the intermittent contact mode (e.g., $c_1' \rightarrow c_1 \rightarrow c_2 \rightarrow c_3' \rightarrow c_2' \rightarrow c_3 \rightarrow c_1'$; Fig. 1D, inset). This is substantially more challenging than what is required for paravalvular leak closure because, for leak closure, our algorithms enable the catheter to follow the ventricular wall in a direction that positions the catheter at an angular position on the valve that is close to the leak. For example, to reach c_1 in Fig. 1D, the catheter could follow the path $a \rightarrow b \rightarrow c_1' \rightarrow c_1$. For circumnavigation of the annulus, we compared autonomous control (three animals, 65 trials) with a handheld catheter (three animals, 3 trials) and with two forms of teleoperation (Fig. 3B). The first consisted of standard teleoperated control (one animal, 9 trials). The second corresponds to autono-

mous operator assistance (one animal, 10 trials). In the latter, the robot automatically controlled motion perpendicular to the plane of the valve to achieve a desired contact duty cycle, whereas the human operator manually controlled motion in the valve plane. Autonomous valve circumnavigation was successful 66% of the time (43 of 65 trials). Manual and teleoperated control had 100% success rates because the human operator, a clinician, could interpret and respond to unexpected situations. For this task, teleoperation was faster than autonomous and manual navigation, with assisted teleoperation being the fastest (Fig. 3B). Autonomous control was the slowest, taking over twice as long as manual control.

We then compared controller performance for the complete paravalvular leak navigation task (Fig. 1D; $a \rightarrow b \rightarrow c_i' \rightarrow c_i$), in

which the catheter started at the heart's apex, approached the ventricular wall in a user-provided direction, moved to the aortic annulus along the ventricular wall, and then followed the annulus to the prespecified leak position (five animals, 83 trials). We chose the direction along the ventricular wall so that the catheter would arrive on the valve at a point c_i' , close to the leak c_i , but such that it would still have to pass over at least one registration marker to reach the leak. Autonomous navigation was successful in 95% of the trials (79 of 83) with a total time of 39 ± 17 s compared with times of 34 ± 29 s for teleoperation (three animals, 9 trials) and 31 ± 27 s for manual navigation (three animals, 13 trials) (see Fig. 3C). Note that for teleoperated and manual navigation, the operator was not required to follow a particular path to a leak.

For autonomous navigation, we also evaluated how accurately the catheter was able to position its tip over a leak. In the first three experiments, valve rotation with respect to the robot was estimated by an operator before autonomous operation. In the last two experiments, valve rotation was estimated by the robot based on its detection of the registration sutures. The distance between the center of the catheter tip and the center of each leak was 3.0 ± 2.0 mm for operator-based registration (three animals, 45 trials) and 2.9 ± 1.5 mm for autonomous estimation (two animals, 38 trials) with no statistical difference between methods ($P = 0.8262$, Wilcoxon rank sum). This error is comparable with the accuracy to which a leak can be localized on the basis of preoperative imaging.

To ensure that autonomous navigation did not affect occluder delivery, we performed leak closure after autonomous, teleoperated, and manual navigation. The time to close a leak was measured from the moment either the robot or the human operator signaled that the working channel of the catheter was positioned over the leak. Any time required by the operator to subsequently adjust the location of the working channel was included in closure time (Fig. 3D). Leak closure was successful in 8 of 11 trials (autonomous navigation), 7 of 9 trials (teleoperation), and 11 of 13 trials (manual navigation). The choice of navigation method produced no statistical difference in closure success or closure time.

DISCUSSION

Our primary result is that autonomous navigation in minimally invasive procedures is possible and can be successfully implemented using enhanced sensing and control techniques to provide results comparable with expert manual navigation in terms of procedure time and efficacy. Furthermore, our experiments comparing clinician-controlled robotic navigation with manual navigation echo the results obtained for many medical procedures—robots operated by humans often provide no better performance than manual procedures except for the most difficult cases and demanding procedures (16, 17). Medical robot autonomy provides an alternative approach and represents the way forward for the field (18–21).

Benefits of autonomous navigation

Automating such tasks as navigation can provide important benefits to clinicians. For example, when a clinician is first learning a procedure, a significant fraction of their attention is allocated to controlling instruments (e.g., catheters and tools) based on multimodal imaging. Once a clinician has performed a large number of similar procedures with the same instruments, the amount of attention devoted to instrument control is reduced. By using autonomy to relieve the

clinician of instrument control and navigation, the learning curve involved in mastering a new procedure could be substantially reduced. This would be of significant benefit during initial clinical training, and it may also enable midcareer clinicians to adopt new minimally invasive techniques that would otherwise require too much retraining. In addition, even after a procedure is mastered, there are many situations where an individual clinician may not perform a sufficient number of procedures to maintain mastery of it. In all of these cases, autonomy could enable clinicians to operate as experts with reduced experience- and fatigue-based variability.

There are also many places in the world where clinical specialties are not represented. Although medical robots can provide the capability for a specialist to perform surgery remotely (22), this approach requires dedicated high-bandwidth two-way data transfer. Transmission delays or interruptions compromise safety owing to loss of robot control. In these situations, autonomy may enable stable and safe robot operation even under conditions of low-bandwidth or intermittent communication. Autonomy may also enable the robot to detect and correct for changing patient conditions when communication delays preclude sufficiently fast reaction by the clinician.

Autonomy also enables, to an unprecedented degree, the capability to share, pool, and learn from clinical data (23–25). With teleoperated robots, robot motion data can be easily collected, but motions are being performed by clinicians using different strategies, and the information they are using to guide these strategies may not all be known, let alone recorded. In contrast, the sensor data streaming to an autonomous controller are well defined, as is its control strategy. This combination of well-defined input and output data, together with known control strategies, will make it possible to standardize and improve autonomous technique based on large numbers of procedural outcomes. In this way, robot autonomy can evolve by applying the cumulative experiential knowledge of its robotic peers to each procedure.

Limitations

We demonstrated that haptic vision combined with biologically inspired wall following serve as an enabling method for autonomous navigation inside the blood-filled heart. The haptic vision sensor provides a high-resolution view of the catheter tip—which is exactly the view needed for both wall following and device deployment. The sensor, combined with its machine learning and image processing algorithms, enables the robot to distinguish what it is touching and to control its contact force.

After initial wall-following exploration, blinded animals, as well as people, have been observed to use their environmental map to create shortcuts through free space (7, 8, 26, 27). Although not formally studied, we observed similar behavior in our previously published experiments with a handheld instrument (15). During initial use of the instrument, the clinician navigated from the apex to the aortic annulus using wall following. Once familiar with the procedure, however, the clinician would usually attempt to move directly through the center of the ventricle to the annulus. If the annulus was not found where they expected it to be, then they would retract the instrument, search for the ventricular wall, and follow it to the annulus. Although we did not attempt this more advanced form of navigation, it is worthy of future study because it is likely to lead to faster performance and facilitate safe navigation through valves.

The most challenging part of our in vivo navigation plan was moving along the aortic valve annulus. Whereas the success rate of

the overall navigation task was 95%, complete circumnavigation of the annulus was successful only 66% of the time. This lower success rate is not a limitation of wall following but rather reflects our decision to make annular navigation only as robust as necessary to achieve high overall task success. In particular, a primary failure mode of circumnavigation corresponded to the catheter tip experiencing simultaneous lateral contact with the ventricular wall and tip contact with the valve. This uncontrolled lateral contact led to the catheter becoming stuck against the ventricular wall. By adding a new control mode considering simultaneous tip and lateral contacts, circumnavigation could be made much more robust. Furthermore, although we limited navigational sensing to haptic vision to evaluate its capabilities, the use of additional sensing modalities and more sophisticated modeling and control techniques is warranted for clinical use.

We used transapical access in our experiments because it allowed us to focus on the most challenging navigation problem: performing precise motions inside a pulsating three-dimensional (3D) volume containing complex moving features. For clinical use, the proposed approach should be extended to enable autonomous navigation starting from the femoral artery. Such vascular navigation is a straightforward extension of our proposed approach and can be performed with our continuous contact control mode. It would involve following 1D curves using a wall-following connectivity graph (Fig. 1D) mapping the branching of the vascular network.

Future directions

Wall-following autonomous navigation is extensible to many minimally invasive procedures, including those in the vasculature, airways, gastrointestinal tract, and ventricular system of the brain. Even in the absence of blood or bleeding, haptic vision, potentially augmented with other sensing modalities, can be used to mediate tissue contact. The sequence of wall contacts defining a navigational plan is based on anatomical topology but not on anatomical dimensions. Consequently, wall-following plans are largely patient independent but can be adapted as needed for anatomical variants based on pre-operative imaging. As demonstrated with our bioprosthetic aortic valve, previously deployed devices can serve as navigational way points and can incorporate visual features as navigational aids.

Clinical translation

Clinical translation of autonomy does not require that the robot be capable of completing its task in every possible circumstance. Instead, it needs to satisfy the lesser requirement of knowing when it cannot complete a task and should ask for help. Initially, this framework would enable the robot to perform the routine parts of a procedure, as demonstrated here for navigation, and so enable clinicians to focus on planning and performing the critical procedural components. Ultimately, as autonomous technology matures, the robot can expand its repertoire into more difficult tasks.

MATERIALS AND METHODS

The goal of the study was to investigate the feasibility of performing autonomous catheter navigation for a challenging intracardiac procedure in a preclinical porcine in vivo model. To perform this study, we designed and built a robotic catheter and haptic vision sensor. We also designed and wrote control algorithms, enabling the catheter to navigate either autonomously or under operator control. For our

in vivo experiments, we chose transapical paravalvular leak closure as a demonstration procedure and compared autonomous and operator-controlled navigation times with each other and with previous results using a handheld catheter. For autonomous navigation, we also measured the distance between the final position of the catheter tip and the actual location of the leak.

Robotic catheter

Catheter design

We designed the catheter using concentric tube robot technology in which robots are composed of multiple needle-sized concentrically combined precurved superelastic tubes. A motorized drive system located at the base of the tubes rotated and telescopically extended the tubes with respect to each other to control the shape of the catheter and its tip position (movie S1) (28, 29). The drive system was mounted on the operating room table using a passively adjustable frame that allowed the catheter tip to be positioned and oriented for entry into the apex (Fig. 2B).

Tools and devices were delivered through the lumen of the innermost robot tube, which incorporated a valve and flush system at its proximal end. This system enabled the catheter lumen to be flushed with saline to prevent air entry into the heart and to prevent pressurized blood from the heart from entering the lumen of the catheter. We used a design optimization algorithm to solve for the tube parameters based on the anatomical constraints and clinical task (aortic paravalvular leak closure) (30). The anatomical and task constraints were defined using a 3D model of an adult human left ventricle (Fig. 4). Because the relative dimensions of the human and porcine hearts are similar, the resulting design was appropriate for our in vivo experiments. The design algorithm solved for the tube parameters enabling the catheter tip to reach from the apex of the heart to a set of 25 uniformly sampled points around the aortic valve annulus without the catheter contacting the ventricular wall along its length. The orientation of the catheter tip was further constrained at the 25 points to be within 10° of orthogonal to the valve plane. The resulting design was composed of three tubes forming two telescoping sections with motion as shown in movie S1. The tube parameters of the robotic catheter are given in Table 1.

Haptic vision sensor design

We fabricated the sensor (Fig. 1A) using a 1-mm³ complementary metal-oxide semiconductor camera (NanEye Camera System, AWAIBA) and a 1.6 mm by 1.6 mm light-emitting diode (LED) (XQ-B LED, Cree Inc.) encased in an 8-mm-diameter silicone optical window (QSil 218, Quantum Silicones LLC.) molded onto a stainless-steel body (15). The optical window diameter was selected to provide a field of view facilitating both autonomous and operator-guided navigation. The sensor also incorporated a 2.5-mm-diameter working channel for device delivery. The system clamped to the tip of the catheter such that the lumen of the innermost catheter tube aligned with the endoscope working channel. Although we have also designed catheters in which the sensor wiring was run through the catheter lumen, in these experiments, the wiring for the camera and LED were run outside the catheter so that the sensor could be replaced without disassembling the robotic catheter.

Computer software design

The software was executed on two PCs. One was used for catheter motion control [Intel Core Quad CPU Q9450@2.66 GHz with 4-GB random-access memory (RAM)], whereas the second was used to acquire and process images from the haptic vision sensor (Intel

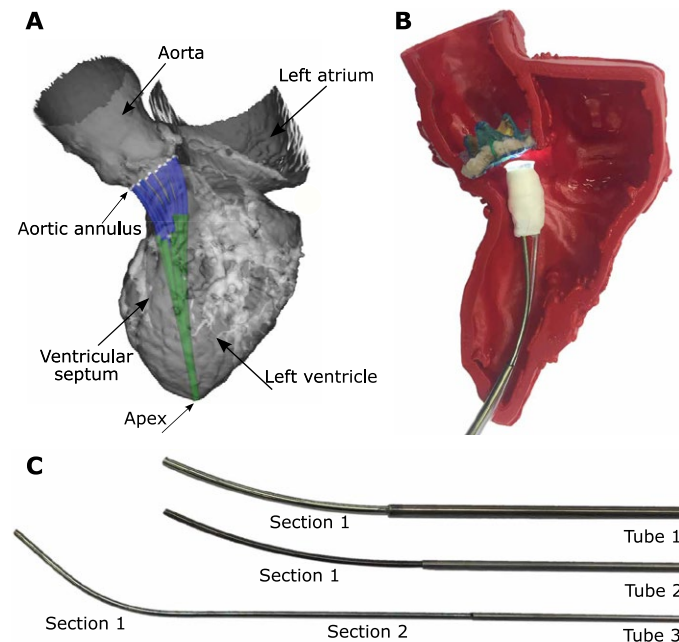


Fig. 4. Algorithmic robotic catheter design. (A) Computer model of optimized design composed of three tubes shown in adult left ventricle. Robot enters heart through apex and is depicted at 12 locations on the aortic annulus. (B) Fabricated catheter with haptic vision sensor shown inside 3D printed model of heart shown in (A). (C) Disassembled catheter showing its three precurved superelastic tubes. Tube parameters are given in Table 1.

Core i7-6700HQ CPU@2.6 GHz with 16-GB RAM). The two computers exchanged information at runtime via transmission control protocol/internet protocol. The motion control computer received real-time heart rate data by serial port (Advisor, SurgiVet) and was also connected through universal serial bus to a six-DOF joystick (Touch, 3D Systems) that was used during teleoperated control of catheter motion. The motion control computer could execute either the autonomous navigation algorithms or the joystick motion commands. In either case, catheter tip motion commands were converted to signals sent to the motor amplifiers of the catheter drive system.

Robotic catheter control

The catheter control code converting desired catheter tip displacements to the equivalent rotations and translations of the individual tubes was written in C++. The code was based on modeling the kinematics using a functional approximation (truncated Fourier series) that was calibrated offline using tip location data collected over the workspace (28). The calibrated functional approximation model had been previously demonstrated to predict catheter tip position more accurately (i.e., smaller average and maximum prediction error) over the workspace compared with the calibrated mechanics-based model (31). Catheter contact with tissue along its length produced unmodeled and unmeasured deformations that must be compensated for via tip imaging. A hierarchical control approach was used (32) to ensure that the desired tip position was given a higher priority than the desired orientation if both criteria could not be satisfied simultaneously.

Haptic vision–based contact classification

To perform wall following, we designed a machine learning–based image classifier that can distinguish between blood (no contact) or

Table 1. Optimized parameter values for three tubes comprising the robotic catheter. Tube sections are labeled in Fig. 4B.

	Tube 1		Tube 2		Tube 3	
	Section 1	Section 1	Section 1	Section 1	Section 2	Section 2
Outer diameter (mm)	2.77	2.40	1.875	1.875		
Inner diameter (mm)	2.54	2.00	1.60	1.60		
Section length (mm)	72.0	72.0	55.0	72.0		
Radius of curvature (mm)	150	150	40.0	∞ (straight)		
Relative bending stiffness	0.995	0.995	0.338	0.338		

ventricular wall tissue and the bioprosthetic aortic valve. The algorithm used the bag-of-words approach (33) to separate images into groups (classes) based on the number of occurrences of specific features of interest. During training, the algorithm determined which features were of interest and the relationship between their number and the image class. For training, we used OpenCV to detect features in a set of manually labeled training images. Next, the detected features were encoded mathematically using LUCID descriptors for efficient online computation (34). To reduce the number of features, we identified the optimal feature representatives using clustering (k -means). The resulting cluster centers were the representative features used for the rest of the training, as well as for runtime image classification. Having identified the set of representative features, we made a second pass through the training data to build a feature histogram for each image by counting how many times each representative feature appeared in the image. The final step was to train a support vector machine (SVM) (35) classifier that learned the relationship between the feature histogram and the corresponding class.

Using the trained algorithm, image classification proceeded by first detecting features and computing the corresponding LUCID descriptors. The features were then matched to the closest representative features, and the resulting feature histogram was constructed. On the basis of the histogram, the SVM classifier predicted the tissue-based contact state. We achieved good results using a small set of training images (~2000 images) with training taking ~4 min. Because image classification took 1 ms, our haptic vision system estimated contact state at the frame rate of the camera (45 frames/s). The contact classification algorithm was accurate 97% of the time (tested on 7000 images not used for training) with type I error (false positive) of 3.7% and type II (false negative) of 2.3%.

Continuous contact navigation algorithm

When the catheter was positioned laterally against cardiac tissue, its flexibility could enable continuous contact to be maintained without applying excessive force to the tissue. We used haptic vision to control the amount of tissue contact by controlling catheter motion in the direction orthogonal to the tissue surface. Catheter motion in the plane of the tissue surface was independently controlled so as to produce wall following at the desired velocity and in the desired direction. The controller was initialized with an estimate of wall location

so that if it was not initially in tissue contact, it moved toward the wall to generate contact. This occurred in our in vivo experiments during navigation from the apex to the aortic valve. The catheter started in the center of the apex with the haptic vision sensor detecting only blood. It would then move in the direction of the desired wall (Fig. 1D, a \rightarrow b), specified using valve clock coordinates (Fig. 2C), to establish contact and then follow that wall until it reached the valve.

When the haptic vision sensor was pressed laterally against the tissue, the tissue deformed around the sensor tip so that it covered a portion of the field of view (Fig. 1B). The navigation algorithm adjusted the catheter position orthogonal to the tissue surface to maintain the centroid of the tissue contact area within a desired range on the periphery of the image, typically 30 to 40%. We implemented tissue segmentation by first applying a Gaussian filter to reduce the level of noise in the image. Next, we segmented the tissue using color thresholding on the hue saturation value (HSV) and the CIE $L^*a^*b^*$ representations of color images. The result was a binary image, where white pixels indicated tissue. Last, we performed a morphological opening operation to remove noise from the binary image. After segmentation, the tissue centroid was computed and sent to the motion control computer. Image processing speed was sufficient to provide updates at the camera frame rate (45 frames/s).

Intermittent contact navigation algorithm

When a catheter is stiff along its longitudinal axis and positioned orthogonal to a tissue surface that moved significantly in the direction of this axis over the cardiac cycle, the contact forces can become sufficiently high so as to result in tissue damage or puncture. To maintain contact forces at safe levels, one approach is to design the catheter so that it can perform high-velocity trajectories that move the robotic catheter tip in synchrony with the tissue (36). We used an alternative technique requiring only slow catheter motion so as to position the tip such that it was in contact with the tissue for a specified fraction of the cardiac cycle, the contact duty cycle, D (Fig. 1C). As described in the section below titled “In-vivo calibration of contact duty cycle versus maximum tissue force,” the contact duty cycle was linearly related to the maximum contact force. The intermittent contact mode was used during navigation around the aortic valve annulus.

We implemented intermittent contact navigation using haptic vision to detect tissue contact and, combined with heart rate data, to compute the contact duty cycle at the frame rate of the camera (45 frames/s). We implemented a controller that adjusted catheter position along its longitudinal axis to drive the contact duty cycle to the desired value. Catheter motion in the plane of the tissue surface was performed either autonomously or by the operator (shared control mode). In the autonomous mode, catheter motion in the tissue plane was performed only during the fraction of the cardiac cycle when the haptic vision sensor indicated that the catheter was not touching tissue. This reduced the occurrence of the catheter tip sticking to the tissue surface during wall following.

Autonomous navigation on the valve annulus

Intermittent contact control was used to control catheter motion orthogonal to the plane of the annulus. The desired value of contact duty cycle was typically set to be $\sim 40\%$. Thus, 40% of the cardiac cycle was available for image processing (during contact), whereas the motion in the plane of the annulus was performed during the 60% noncontact portion of the cardiac cycle. During contact, the robot detected the blue tangent sutures on the valve (Fig. 2C) using

color thresholding in the HSV color space and computed the centroid of the detected sutures. Next, a Hough transform on the thresholded image was used to estimate the tangent of the aortic annulus. During the noncontact portion of the cardiac cycle, the algorithm generated independent motion commands in the radial and tangential directions. In the radial direction, the catheter adjusted its position such that the centroid of the detected sutures was centered in the imaging frame. Motion in the tangential direction was performed at a specified velocity. While navigating around the valve, the robot incrementally built a map of the location of the annulus in 3D space based on the centroids of the detected sutures and the catheter tip coordinates as computed using the robot kinematic model. The model was initialized with the known valve diameter and the specified direction of approach. By comparing the current tangent with the model, the robot estimated its clock position on the annulus. Although not implemented, this model may also be used to estimate the valve tangent and radial position in situations where the sutures are not well detected.

Registration of valve rotation with respect to the robot

We assumed that paravalvular leaks have been identified in pre-operative imaging, which also indicated their location relative to the features of the bioprosthetic valve, e.g., leaflet commissures. In the ventricular view of the valve annulus provided by haptic vision, such features are hidden. Although the model built during annular navigation defines the coordinates of the annulus circle in 3D space, there was no means to refine the initial estimate of where 12 o'clock fell on the circle, i.e., to establish the orientation of the valve about its axis. To enable the robot to refine its orientation estimate, we introduced registration features into the annulus composed of green sutures located at 4, 8, and 12 o'clock. During annular navigation, whenever the robot detected one of these features, it compared its actual location with the current prediction of the model and updated its estimate of valve rotation accordingly.

Endothelialization of blue and green sutures on the valve annulus

In clinical use, the sutures would remain visible for several months before endothelialization. Thus, they could be used for autonomous repair of paravalvular leaks that occur at the time of valve implantation or soon after, as is the case for transcatheter valves (12).

Autonomous navigation for paravalvular leak closure

The algorithm inputs are consisted of the clock-face leak location and the desired ventricular approach direction, also specified as a clock-face position. Starting from just inside the apex of the left ventricle, the catheter moved in the desired approach direction until it detected tissue contact. It then switched to continuous contact mode and performed wall following in the direction of the valve. When the classifier detected the bioprosthetic valve in the haptic vision image, the controller switched to intermittent contact mode and computed the minimum distance direction around the annulus to the leak location based on its initial map of the annulus. As the catheter moved around the annulus in this direction, its map was refined on the basis of the detection of tangent and registration sutures. Once the leak location was reached, the robot controller acted to maintain its position at this location and sent an alert to the operator. Using joystick control, the operator could then reposition the working channel over the leak as needed, and then, the occluder could be deployed.

Software development cycle

To develop and test our autonomous navigation algorithms, we implemented a development cycle composed of three steps: in silico simulation, ex vivo experiments, and in vivo experiments (Fig. 5).

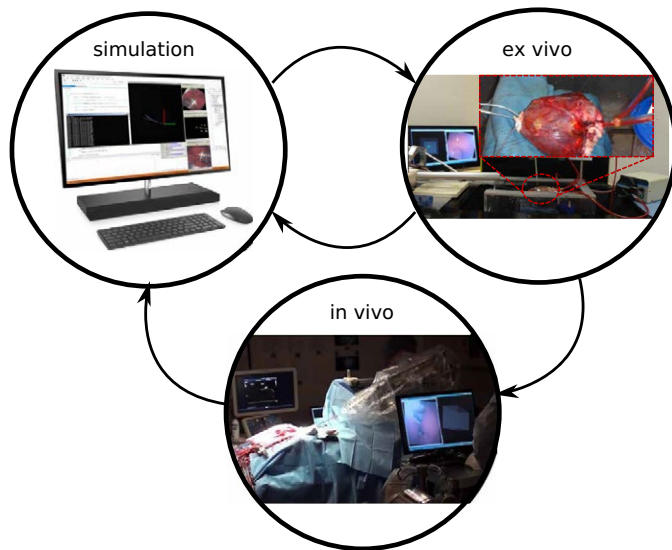


Fig. 5. Software development cycle. In simulation, we replayed data from previous in vivo experiments to evaluate and debug software. New features were first implemented in the simulator either to address previously identified in vivo challenges or to extend robot capabilities. New software was then tested in the ex vivo model to check the desired functionality and to ensure code stability. Identified problems were addressed by iterating between in silico and ex vivo testing. New software features were then assessed with in vivo testing. The design cycle was then completed by importing the in vivo data into the simulator and evaluating algorithm performance.

We created a simulation engine that can replay time-stamped data, comprising haptic vision images and robot trajectories, recorded during in vivo cases. We used the simulation engine to implement new software functionality and to troubleshoot unexpected results from in vivo experiments. After simulation, we tested new functionality on an ex vivo model comprising an explanted porcine heart, pressurized using a peristaltic pump (Masterflex Pump, 115 VAC). We immobilized the pressurized heart using sutures to attach it to a fixture. On the basis of the outcome of the ex vivo tests, we either performed additional simulations to refine the software implementation or proceeded to in vivo testing. This process was repeated iteratively for each algorithm as it was developed.

In vivo experiments

Surgical procedure

We created a porcine paravalvular leak model by implanting a custom bioprosthetic device (Fig. 2C) into the aortic valve position in 84.3 ± 4.7 kg of Yorkshire swine. The device was designed with three sewing ring gaps evenly distributed around its circumference (120° apart) to produce the areas of paravalvular leakage. The bioprosthetic valve consists of a titanium frame covered by a nonwoven polyester fabric. A polypropylene felt sewing ring is sutured to the frame around the annulus. Suture is passed through this ring when the valve is sewn in place inside the heart. Last, glutaraldehyde-fixed porcine pericardium leaflets are sutured to the frame (37).

Animal care followed procedures prescribed by the Institutional Animal Care and Use Committee. To implant the bioprosthetic valve, we premedicated the swine with atropine (0.04 mg/kg intramuscularly), followed by Telazol (4.4 mg/kg) and xylazine (2.2 mg/kg intravenously), and we accessed the thoracic cavity through a median sternotomy

incision. We acquired epicardial echocardiographic images to determine the size of the valve to be implanted. Next, we initiated cardiopulmonary bypass by placing purse-string sutures for cannulation, cross-clamping the aorta, and infusing cardioplegia solution to induce asystole. We incised the aorta to expose the valve leaflets, which were then removed, and the artificial valve was implanted using nine 2-0 ETHIBOND valve sutures supra-annularly. At this point, we closed by suture the aortomy incision, started rewarming, and released the aortic cross-clamp. We maintained cardiopulmonary bypass to provide 35 to 50% of normal cardiac output to ensure hemodynamic and cardiac rhythm stability. The function of the implanted valve, as well as the leak locations and sizes, were determined by transepicaardial short- and long-axis 2D and color Doppler echocardiography. Apical ventriculotomy was then performed, with previous placement of purse-string sutures to stabilize the cardiac apex for the introduction of the robotic catheter. The catheter was introduced through the apex and positioned such that its tip was not in contact with the ventricular walls. All experiments in a group were performed using the same apical catheter position. Throughout the procedure, we continuously monitored arterial blood pressure, central venous pressure, heart rate, blood oxygenation, temperature, and urine output. At the end of the experiment, a euthanasia solution was injected, and we harvested the heart for postmortem evaluation.

Autonomous navigation from apex to valve

We performed experiments on five animals. For each animal, navigation was performed using three valve approach directions corresponding to 6 o'clock (posterior ventricular wall), 9 o'clock (ventricular septal wall), and 12 o'clock (anterior ventricular wall) (Fig. 2C). Of the 90 total trials, the number performed in the 6, 9, and 12 o'clock directions were 31, 32, and 27, respectively.

Autonomous circumnavigation of aortic valve annulus

Experiments were performed on three animals. In the first experiment, a range of contact duty cycles was tested, whereas in the latter two experiments, the contact duty cycle was maintained between 0.3 and 0.4. In all experiments, the tangential velocity was specified as 2 mm/s during those periods when the tip was not in contact with the valve and 0 mm/s when in contact.

Autonomous navigation from apex to paravalvular leaks

We performed experiments on five animals. As an initial step for all experiments, we built a 3D spatial model of the valve by exploring the valve with the catheter under operator control. We used this model, which is separate from the model built by the autonomous controller, to monitor autonomous navigation. For three animals, we also used this model to estimate valve rotation with respect to the robot and provided this estimate as an input to the autonomous navigation algorithm. In two animals, valve rotation was estimated autonomously on the basis of the valve model built by the navigation algorithm and its detection of registration sutures.

In each experiment, navigation trials were individually performed for each of the three leaks located at 2 o'clock ($n = 28$), 6 o'clock ($n = 27$), and 10 o'clock ($n = 28$) (Fig. 2C). For each leak location, we selected a clock direction to follow on the ventricular wall such that the catheter would arrive at the valve annulus close to the leak but far enough away that it would have to pass over registration sutures to reach the leak. In general, this corresponded to approaching the valve at 11, 9, and 1 o'clock to reach the leaks at 2 o'clock (clockwise), 6 o'clock (counter clockwise), and 10 o'clock (counter clockwise), respectively. If we observed that along these paths the

annulus was covered by valve tissue or a suturing pledget, then we instructed the navigation algorithm to approach the leak from the opposite direction. Note that the mitral valve is located from 2 to 5 o'clock; the ventricular wall cannot be followed in these directions to reach the aortic valve. Thus, in one experiment involving operator-specified valve registration, the clockwise-approach path was covered by tissue, and we chose to approach the leak directly from the 2 o'clock direction rather than start farther away at 6 o'clock.

We designed the registration sutures to be 120° apart under the assumption that valve rotation with respect to the robot would be less than $\pm 60^\circ$ from the nominal orientation. In one animal in which valve rotation was estimated autonomously, however, the rotation angle was equal to 60°. In this situation, it is impossible for either man or machine to determine whether the error is +60° or -60°. For these experiments, we shifted the approach direction for the leak at 6 o'clock from 9 to 8 o'clock so that the catheter would only see one set of registration sutures along the path to the leak. This ensured that it would navigate to the correct leak.

Occluder deployment

After navigation to the desired leak location, the operator took control of the catheter and, as needed, centered the working channel over the leak. A three-lobed vascular occluder (AMPLATZER Vascular Plug II, AGA Medical Corporation), attached to a wire and preloaded inside a delivery cannula, was advanced ~3 mm into the leak channel (Fig. 6). The cannula was then withdrawn, allowing the occluder to expand inside the leak channel. We then retracted the wire and robotic catheter until the proximal lobe of the occluder was positioned flush with the valve annulus and surrounding tissue. If positioning was satisfactory, then the device was released by unscrewing it from the wire. If not, then the device was retracted back into the delivery cannula and the procedure was repeated as necessary.

In vivo calibration of contact duty cycle versus maximum tissue force

To investigate the relationship between maximum contact force and contact duty cycle, we designed a handheld instrument integrating haptic vision and force sensing (Fig. 7). Force sensor integration was inspired by (38). The haptic vision sensor is mounted on a stiff tube that is supported by two polymer sliding bearings mounted inside the proximal handle. The proximal end of the shaft is connected to a force sensor. An outer cannula encloses the sensing shaft and extends from the handle to about 6 cm from the sensing tip. When the instrument is inserted into the apex of the heart, the outer cannula is in contact with the apical tissue, but the sensing tube is not. The gap between the outer cannula and sensing tube is filled with silicone grease to prevent blood flow from the heart into the instrument while generating minimal friction on the sensing tube. Calibration experiments indicated that the friction due to the bearings and grease was less than ± 0.2 N.

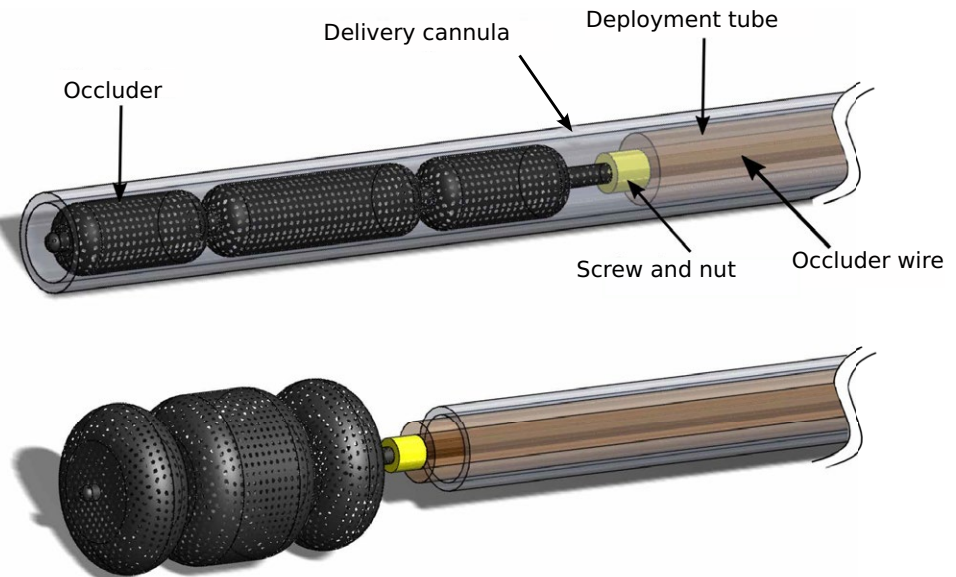


Fig. 6. Occluder deployment system. The occluder, attached to a wire via a screw connection, is preloaded inside a flexible polymer delivery cannula. The delivery cannula is inserted through the lumen of the catheter into the paravalvular leak. A polymer deployment tube is used to push the occluder out of the delivery cannula. Once positioned, the occluder is released by unscrewing the wire.

We performed in vivo experiments in which we positioned the haptic vision sensor on the bioprosthetic valve annulus in locations where we could be sure that the sensor was experiencing contact only on its tip. At these locations, we collected force, heart rate, and haptic vision data (Fig. 7B). By manually adjusting instrument position along its axis, we were able to obtain data for a range of duty cycle values. The image and force data were collected at 46 Hz, and contact duty cycle based on valve contact was computed at each sampling time using a data window of width equal to the current measured cardiac period (~36 images). To remove high-frequency components not present in the force data, we then filtered the computed duty cycle using a 121-sample moving average filter corresponding to ~3.4 heartbeats. The filtered input data and the output data (e.g., Fig. 7B) were then binned using duty cycle intervals of 0.05. Last, we computed the relationship between filtered contact duty cycle and maximum applied force by averaging the maximum forces for each binned duty cycle value (Fig. 7C). We computed the Pearson's coefficient as a measure of linear relationship between the contact duty cycle and the maximum annular force. The Pearson's coefficient was equal to 0.97, which indicates a strong linear relationship. The plot of Fig. 7C indicates that the contact duty cycle range of 0.35 to 0.45 that we used in most of our experiments corresponded to a maximum force of 1.25 to 2.3 N.

Statistical analysis

MATLAB (version R2017b) statistical subroutines were used to analyze the data and perform all statistical tests. We compared time duration for each navigation mode (i.e., handheld, teleoperated, and autonomous) for the tasks of navigating from the apex to the aortic annulus, navigating around the valve annulus, and from the apex to the leak. We also compared occluder deployment times for each navigation mode. Groups, corresponding to different navigation modes, have unequal sample sizes and sample variances. We used Levene's test to evaluate equality of variances. With no evidence of

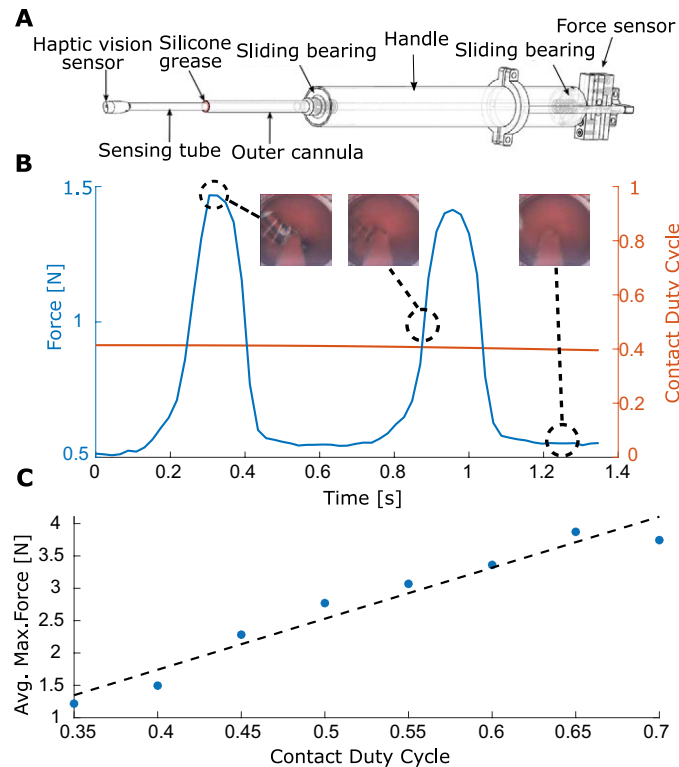


Fig. 7. Maximum tissue force as a function of contact duty cycle. (A) We constructed a handheld instrument for the simultaneous measurement of tip contact force and contact duty cycle that combines a haptic vision sensor with a force sensor. (B) We made in vivo measurements of the temporal variations in contact force as a function of contact duty cycle on the aortic valve annulus. The insets show images from the haptic vision sensor at three points in the cardiac cycle. Note that the minimum force value is not necessarily zero because a small amount of tip contact with ventricular tissue can occur during systole when the valve moves away from the tip, but the ventricle contracts around it. This white (septal) ventricular tissue can be seen on the left side of the rightmost inset. (C) Maximum contact force as a function of duty cycle. Average values of maximum force are linearly related to contact duty cycle for duty cycles in the range of 0.35 to 0.7.

normally distributed time duration and more than two groups, we used the Kruskal-Wallis nonparametric test to check whether there are statistically significant time differences among groups. In experiments with statistical significance, we compared pairs of groups using the Mann-Whitney *U* test with Bonferroni correction. Data less than $Q1 - 1.5 \times IQR$ or greater than $Q3 + 1.5 \times IQR$, where the interquartile range ($IQR = Q3 - Q1$), were considered outliers. Fisher's exact test was used to compare success rates between different groups in the case of paravalvular leak closure. Statistical significance was tested at the 5% confidence level ($P < 0.05$).

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/29/eaaw1977/DC1

Movie S1. Operator-controlled catheter navigation inside a 3D printed heart model.

Movie S2. In vivo autonomous catheter navigation.

REFERENCES AND NOTES

1. M. A. Cooper, S. Hutfless, D. L. Segev, A. Ibrahim, H. Lyu, M. A. Makary, Hospital level under-utilization of minimally invasive surgery in the United States: Retrospective review. *Br. Med. J.* **349**, g4198 (2014).
2. M. J. Reardon, N. M. Van Mieghem, J. J. Popma, N. S. Kleiman, L. Søndergaard, M. Mumtaz, D. H. Adams, G. M. Deeb, B. Maini, H. Gada, S. Chetcuti, T. Gleason, J. Heiser, R. Lange,

- W. Merhi, J. K. Oh, P. S. Olsen, N. Piazza, M. Williams, S. Windecker, S. J. Yakubov, E. Grube, R. Makkari, J. S. Lee, J. Conte, E. Vang, H. Nguyen, Y. Chang, A. S. Mugglin, P. W. Serruys, A. P. Kappetein; SURTAVI Investigators, Surgical or transcatheter aortic-valve replacement in intermediate-risk patients. *N. Engl. J. Med.* **376**, 1321–1331 (2017).
3. P. Kiefer, S. Meier, T. Noack, M. A. Borge, J. Ender, A. Hoyer, F. W. Mohr, J. Seeburger, Good 5-year durability of transapical beating heart off-pump mitral valve repair with neochordae. *Ann. Thorac. Surg.* **106**, 440–445 (2018).
4. J. Tobis, M. Shenoda, Percutaneous treatment of patent foramen ovale and atrial septal defects. *J. Am. Coll. Cardiol.* **60**, 1722–1732 (2012).
5. G. G. Stefanini, D. R. Holmes Jr., Drug-eluting coronary-artery stents. *N. Engl. J. Med.* **368**, 254–265 (2013).
6. F. A. Lederle, J. A. Freischlag, T. C. Kyriakides, J. S. Matsumura, F. T. Padberg Jr., T. R. Kohler, P. Kougiass, J. M. Jean-Claude, D. F. Cikrit, K. M. Swanson; OVER Veterans Affairs Cooperative Study Group, Long-term comparison of endovascular and open repair of abdominal aortic aneurysm. *N. Engl. J. Med.* **367**, 1988–1997 (2012).
7. R. Avni, Y. Tzvaigrach, D. Eilam, Exploration and navigation in the blind mole rat (*Spalax ehrenbergi*): Global calibration as a primer of spatial representation. *J. Exp. Biol.* **211**, 2817–2826 (2008).
8. S. Sharma, S. Coombs, P. Patton, T. Burt de Perera, The function of wall-following behaviors in the Mexican blind cavefish and a sighted relative, the Mexican tetra (*Astyanax*). *J. Comp. Physiol. A* **195**, 225–240 (2009).
9. B. Mitchinson, C. J. Martin, R. A. Grant, T. J. Prescott, Feedback control in active sensing: Rat exploratory whisking is modulated by environmental contact. *Proc. R. Soc. B* **274**, 1035–1041 (2007).
10. R. Creed, J. Miller, Interpreting animal wall-following behavior. *Experientia* **46**, 758–761 (1990).
11. B. W. Patullo, D. L. Macmillan, Corners and bubble wrap: The structure and texture of surfaces influence crayfish exploratory behaviour. *J. Exp. Biol.* **209**, 567–575 (2006).
12. M. F. Eleid, A. K. Cabalka, J. F. Malouf, S. Sanon, D. J. Hagler, C. S. Rihal, Techniques and outcomes for the treatment of paravalvular leak. *Circ. Cardiovasc. Interv.* **8**, e001945 (2015).
13. F. Nietlispach, F. Maisano, P. Sorajja, M. B. Leon, C. Rihal, T. Feldman, Percutaneous paravalvular leak closure: Chasing the chameleon. *Eur. Heart J.* **37**, 3495–3502 (2016).
14. C. Ruiz, V. Jelnin, I. Kronzon, Y. Dudi, R. Del Valle-Fernandez, B. N. Einhorn, P. T. Chiam, C. Martinez, R. Eiros, G. Roubin, H. A. Cohen, Clinical outcomes in patients undergoing percutaneous closure of periprosthetic paravalvular leaks. *J. Am. Coll. Cardiol.* **58**, 2210–2217 (2011).
15. B. Rosa, Z. Machaidze, M. Mencattelli, S. Manjila, B. Shin, K. Price, M. A. Borger, V. Thourani, P. del Nido, D. W. Brown, C. W. Baird, J. E. Mayer Jr., P. E. Dupont, Cardioscopically guided beating heart surgery: Paravalvular leak repair. *Ann. Thorac. Surg.* **104**, 1074–1079 (2017).
16. G. I. Barbash, S. A. Glied, New technology and health care costs—The case of robot-assisted surgery. *N. Engl. J. Med.* **363**, 701–704 (2010).
17. J. C. Hu, X. Gu, S. R. Lipsitz, M. J. Barry, A. V. D'Amico, A. C. Weinberg, N. L. Keating, Comparative effectiveness of minimally invasive vs open radical prostatectomy. *JAMA* **302**, 1557–1564 (2009).
18. 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**, 337ra364 (2016).
19. G.-Z. Yang, J. Cambias, K. Cleary, E. Daimler, J. Drake, P. E. Dupont, N. Hata, P. Kazanides, S. Martel, R. V. Patel, V. J. Santos, R. H. Taylor, Medical robotics—Regulatory, ethical, and legal considerations for increasing levels of autonomy. *Sci. Robot.* **2**, eaam8638 (2017).
20. 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).
21. N. A. Patronik, T. Ota, M. A. Zenati, C. N. Riviere, A miniature mobile robot for navigation and positioning on the beating heart. *IEEE Trans. Robot.* **25**, 1109–1124 (2009).
22. J. Marescaux, J. Leroy, M. Gagner, F. Rubino, D. Mutter, M. Vix, S. E. Butner, M. K. Smith, Transatlantic robot-assisted telesurgery. *Nature* **413**, 379–380 (2001).
23. Z. Obermeyer, E. J. Emanuel, Predicting the future—Big data, machine learning, and clinical medicine. *N. Engl. J. Med.* **375**, 1216–1219 (2016).
24. M. E. Porter, S. Larsson, T. H. Lee, Standardizing patient outcomes measurement. *N. Engl. J. Med.* **374**, 504–506 (2016).
25. L. Maier-Hein, S. S. Vedula, S. Speidel, N. Navab, R. Kikinis, A. Park, M. Eisenmann, H. Feussner, G. Forestier, S. Giannarou, M. Hashizume, D. Katic, H. Kennigott, M. Kranzfelder, A. Malpani, K. März, T. Neumuth, N. Padoy, C. Pugh, N. Schoch,

- D. Stoyanov, R. Taylor, M. Wagner, G. D. Hager, P. Jannin, Surgical data science for next-generation interventions. *Nat. Biomed. Eng.* **1**, 691–696 (2017).
 26. J. Kallai, T. Makary, A. Csatho, K. Karadi, D. Horvath, L. Nadel, B. Kovacs-Labadi, R. Jarai, L. Nadel, J. W. Jacobs, Cognitive and affective aspects of thigmotaxis strategy in humans. *Behav. Neurosci.* **121**, 21–30 (2007).
 27. O. Yaski, J. Portugali, D. Eilam, The dynamic process of cognitive mapping in the absence of visual cues: Human data compared with animal studies. *J. Exp. Biol.* **212**, 2619–2626 (2009).
 28. P. E. Dupont, J. Lock, B. Itkowitz, E. Butler, Design and control of concentric tube robots. *IEEE Trans. Robot.* **26**, 209–225 (2010).
 29. R. Webster III, J. Romano, N. Cowan, Mechanics of precurved tube continuum robots. *IEEE Trans. Robot.* **25**, 67–78 (2009).
 30. C. Bergeles, A. H. Gosline, N. V. Vasilyev, P. J. Codd, P. J. del Nido, P. E. Dupont, Concentric tube robot design and optimization based on task and anatomical constraints. *IEEE Trans. Robot.* **31**, 67–84 (2015).
 31. J. Ha, G. Fagogenis, P. E. Dupont, Effect of path history on concentric tube robot model calibration, in *10th Hamlyn Symposium on Medical Robotics* (The Hamlyn Centre, Faculty of Engineering, Imperial College London, 2017), pp. 77–78.
 32. L. Sentis, O. Khatib, Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. *Int. J. Hum. Robot.* **2**, 505–518 (2005).
 33. F.-F. Li, P. Perona, A Bayesian hierarchical model for learning natural scene categories, in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (IEEE, 2005), vol. 2, pp. 524–531.
 34. A. Ziegler, E. Christiansen, D. Kriegman, S. J. Belongie, Locally uniform comparison image descriptor, in *Advances in Neural Information Processing Systems 25* (Neural Information Processing Systems Foundation, 2012), pp. 1–9.
 35. C. Cortes, V. Vapnik, Support-vector networks. *Mach. Learn.* **20**, 273–297 (1995).
 36. S. B. Kesner, R. D. Howe, Robotic catheter cardiac ablation combining ultrasound guidance and force control. *Int. J. Rob. Res.* **33**, 631–644 (2014).
 37. B. Rosa, Z. Machaidze, B. Shin, S. Manjila, D. W. Brown, C. W. Baird, J. E. Mayer Jr., P. E. Dupont, A low cost bioprosthetic semilunar valve for research, disease modeling and surgical training applications. *Interact. Cardiovasc. Thorac. Surg.* **25**, 785–792 (2017).
 38. S. B. Kesner, “Robotic catheters for beating heart surgery,” thesis, Harvard University, Cambridge, MA (2011).
- Acknowledgments:** We thank R. Howe for suggestions on how to improve the manuscript. We thank A. Nedder, C. Pimental, M. Woomer, K. Connor, M. Bryant, W. Regan, N. Toutenel, and S. Hofferberth for assistance with the in vivo experiments. **Funding:** This work was supported by the National Institutes of Health under grant R01HL124020. B.R. was partially supported by ANR/Investissement d’avenir program (ANR-11-LABX-0004, Labex CAMI).
- Author contributions:** M.M., G.F., B.R., and P.E.D. designed and fabricated the robotic catheter and haptic vision sensor. G.F., B.R., and P.E.D. designed and implemented algorithms and user interface. Z.M., M.M., G.F., and P.E.D. designed experiments. Z.M., G.F., M.M., K.P., F.W., V.W., M.S., J.E.M., and P.E.D. performed experiments. G.F., M.M., Z.M., and P.E.D. evaluated experimental results. G.F., M.M., and P.E.D. prepared the manuscript and figures. All authors edited the manuscript. **Competing interests:** P.E.D., M.M., B.R., and Z.M. are inventors on a U.S. patent application 15/158,475 held by Boston Children’s Hospital that covers the optical imaging technique. All other authors declare that they have no competing interests. **Data and materials availability:** All the data needed to evaluate the study are in the main text or in the Supplementary Materials. Contact P.E.D. for additional data or materials.
- Submitted 27 November 2018
Accepted 22 March 2019
Published 24 April 2019
10.1126/scirobotics.aaw1977
- Citation:** 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).

SENSORS

A neuro-inspired artificial peripheral nervous system for scalable electronic skins

Wang Wei Lee^{1,2}, Yu Jun Tan^{1,2}, Haicheng Yao¹, Si Li^{1,2}, Hian Hian See¹, Matthew Hon³, Kian Ann Ng⁴, Betty Xiong¹, John S. Ho^{2,4,5}, Benjamin C. K. Tee^{1,2,3,4,5*}

Copyright © 2019
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

The human sense of touch is essential for dexterous tool usage, spatial awareness, and social communication. Equipping intelligent human-like androids and prosthetics with electronic skins—a large array of sensors spatially distributed and capable of rapid somatosensory perception—will enable them to work collaboratively and naturally with humans to manipulate objects in unstructured living environments. Previously reported tactile-sensitive electronic skins largely transmit the tactile information from sensors serially, resulting in readout latency bottlenecks and complex wiring as the number of sensors increases. Here, we introduce the Asynchronously Coded Electronic Skin (ACES)—a neuromimetic architecture that enables simultaneous transmission of thermotactile information while maintaining exceptionally low readout latencies, even with array sizes beyond 10,000 sensors. We demonstrate prototype arrays of up to 240 artificial mechanoreceptors that transmitted events asynchronously at a constant latency of 1 ms while maintaining an ultra-high temporal precision of <60 ns, thus resolving fine spatiotemporal features necessary for rapid tactile perception. Our platform requires only a single electrical conductor for signal propagation, realizing sensor arrays that are dynamically reconfigurable and robust to damage. We anticipate that the ACES platform can be integrated with a wide range of skin-like sensors for artificial intelligence (AI)-enhanced autonomous robots, neuroprosthetics, and neuromorphic computing hardware for dexterous object manipulation and somatosensory perception.

INTRODUCTION

Electronic skins (e-skins) are essential for sensing human-machine-environment interactions, with applications in advanced collaborative anthropomorphic robots (1) and neuroprosthetics (2–4). To perform the sensing, e-skins typically consist of numerous tactile sensor elements distributed over a large area substrate that is preferably soft (5), conformable (6), stretchable (7–10), and lightweight (11). Similar to biological skins, the ideal e-skin should be highly responsive and capable of resolving millisecond-precise tactile stimuli to facilitate rapid discrimination (12, 13). This enables emerging dexterous robots the ability to react quickly to highly localized and transient contact events, such as a prick from a needle or slippage of an object. The sensors of the ideal e-skin should also readily scale to thousands in number and be distributed with variable spatial densities depending on the sensing needs while having minimal wiring requirements (14). Furthermore, an e-skin that remains functional while being subject to physical harm enables continuous sensor feedback for safe robotic motor controls and decision making. Although several pioneering efforts have been made to achieve some of these desirable traits, there is currently no technology that encompasses all of the stated attributes.

One major reason is because a vast majority of tactile sensor arrays are currently interfaced via time-divisional multiple access (TDMA), where individual sensors are sampled sequentially and periodically to reconstruct a two-dimensional (2D) map of pressure distribution. Although the serial readout nature of TDMA allows conductor traces to be shared across multiple sensors, which simplifies the wiring of large

arrays, it consequently leads to a decline in readout rates as the number of sensors in the array increases. Existing solutions to boost the sampling rate of TDMA-based e-skins include use of high-speed electronic components (15, 16), intelligent subsampling (17), and data compression techniques (18). Unfortunately, such approaches present challenges to scaling to the thousands of sensors needed to sensorize large exterior areas of a robot and may necessitate impractically large amounts of computation and power (16). Similarly, intelligent sampling and data compression techniques often depend on having a priori knowledge of the tactile stimuli, which limits broad applicability.

A promising alternative to TDMA is event-based signaling. Unlike TDMA, event-based sensors are not periodically polled by a central electronic controller. Instead, data are transmitted by individual sensors only when necessary (19, 20), similar to biological mechanoreceptors. By capitalizing on the temporal sparsity of tactile signals, event-based representations of touch have been shown to use the available communication bandwidth more effectively in comparison with TDMA-based solutions, thus yielding shorter readout latencies (16, 20, 21). Some vision and auditory sensors have also applied such event-driven approaches (22) using asynchronous protocols such as Address Event Representation (AER) (23). However, these systems require monolithic integrations of numerous electrical traces on silicon substrates, which are challenging to do for large-area, distributed skin-like sensing using flexible and stretchable substrates. Hybrid event-based packet forwarding methods are an interesting approach (16, 20), but such data packets require time stamping and rearrangements at the receiver because they are transmitted using communication protocols that do not guarantee fixed latencies.

Moreover, e-skins are expected to make frequent physical contact with the environment. Thus, it is imperative for the e-skin system to withstand mechanical damages, such as abrasion and cuts, with minimal loss of functionality or repair downtime. The typical $n \times m$ sensor matrix (n , rows; m , columns) has limited mechanical robustness because mechanical damage to the electrical trace would disable all the transducers on the entire row/column. Recognizing this limitation,

¹Department of Materials Science and Engineering, National University of Singapore, Singapore 117575, Singapore. ²Institute for Health Innovation and Technology, National University of Singapore, Singapore 117599, Singapore. ³Graduate School of Integrative Sciences and Engineering, National University of Singapore, Singapore 117456, Singapore. ⁴N.I. Institute for Health, National University of Singapore, Singapore 117456, Singapore. ⁵Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117583, Singapore.

*Corresponding author. Email: benjamin.tee@nus.edu.sg

researchers have introduced alternate architectures with redundant components and connections to improve damage resilience of e-skins (24, 25). For instance, sensor nodes configured in a mesh network can reconfigure their routing tables to avoid damaged connections (24). However, individual nodes in such a network require sufficient computational power to handle the overhead of data packet routing and damage repair protocols, thus constraining the level of miniaturization possible. Reconfiguration of larger networks will also incur substantial downtime, potentially impairing sensing when it is needed the most (e.g., during damage events).

In contrast, the human somatosensory system overcomes many of these limitations by coupling action potential signal representations with extensive arrays of nerve fibers in the peripheral nervous system. The ultra-high density of nerve bundles allows mechanoreceptors to use dedicated bioelectronic pathways to the somatosensory cortex (26). In this manner, the latency of human somatosensory system is largely invariant with the number of receptors in the body and limited only by the propagation speed of action potentials or “spikes” (12). When responding to tactile contact events, mechanoreceptors asynchronously transduce ensembles of spikes that represent information through spatiotemporal patterns (12, 13). Unlike TDMA, these spikes are propagated in parallel to the somatosensory cortex (Fig. 1, right) with submillisecond resolution (12). Tactile information is conveyed in a myriad of ways, including spike frequency, spike latency, and phase (27). Damage resilience is excellent because the connections are physically discrete and unaffected by the loss of a particular mechanoreceptor or its axon, enabling continuous functionality of the rest of the system.

Inspired by the human nervous system and motivated by the limitations of existing e-skin technology, we developed a new communication architecture for e-skins that can support thousands of spatially distributed sensors, each capable of asynchronous transmission that requires only a single common conductor for signaling. We term this platform Asynchronously Coded Electronic Skin (ACES). By using a spread spectrum technique, ACES enables an artificial electronic version of the peripheral nervous system for e-skins by multiplexing signals from many sensors to a single receiver (Fig. 1, left).

In our ACES platform, each sensor, referred to as an ACES “receptor,” captures and transmits stimuli information asynchronously as “events” using electrical pulses spaced in time (Fig. 1A and fig. S1A). The temporal arrangement of the pulses, which we refer to as an ACES pulse signature, is unique to each receptor. The spread spectrum (28) nature of the pulse signatures permits multiple sensors to transmit without specific time synchronization, propagating the combined pulse signatures to the decoders via a single electrical conductor (Fig. 1B). The ACES platform is inherently asynchronous due to its robustness to overlapping signatures and does not require intermediate hubs used in existing approaches to serialize or arbitrate the tactile events (19, 20). The ACES signature was designed to be transmitted in 1 ms, similar in duration to a biological action potential (12). At the receiving end, the decoder identifies the transmitting receptor by correlating the received pulses against the known temporal arrangement of pulses for each receptor’s signature. An event from a particular receptor was deemed to be detected if the number of matched pulses exceeded predefined thresholds (Fig. 1C).

RESULTS

Performance of ACES signaling scheme

To replicate the functional role of the biological nerve, the ACES signaling scheme is capable of propagating pulse signatures (events) from

thousands of ACES receptors while preserving the relative time differences between the events. To demonstrate the concept, we developed a prototype system using off-the-shelf components to characterize the performance of our ACES communication architecture (Fig. 2A). Using a physical network of 240 ACES receptors, we established that the temporal resolution, defined as the minimum resolvable time difference between two pulse signatures, is <60 ns (Fig. 2B). The transmission latency is also constant, dependent only on the duration of the pulse signature and, importantly, not on the number of receptors. An exceptionally high temporal precision is potentially achievable even for arrays with $>10,000$ sensors (fig. S1B), despite an increase in temporal uncertainty with an increasing number of overlapping pulse signatures (Fig. 2B), determined using Simulation Program with Integrated Circuit Emphasis (SPICE) simulations (see Materials and Methods).

To establish the reliability of ACES signaling, we observed that by using a decoder threshold of 6 for signatures with 10 pulses ($W = 10$), a false positive probability of 2.3×10^{-5} and missed detection probability of 3.1×10^{-4} could be achieved even with 240 pulse signatures overlapping temporally (Fig. 2C). Using Monte Carlo simulation, which realistically predicts the performance of our experimental prototype (fig. S1C), we further show that false positive and missed detection probabilities increased to 2.4×10^{-2} and 2.1×10^{-3} , respectively, for 1000 overlapping pulse signatures (Fig. 2C). However, the likelihood of signatures overlapping in the thousands was estimated to be rare because of the known sparse nature of tactile stimuli (13, 20). Moreover, we could further improve decoding accuracy by an order of magnitude when the duration of the voltage pulses was halved (Fig. 2D), characterized using the decoded signal-to-interference ratio (DSIR) metric (see Materials and Methods). In addition, we observed that, although larger values of pulse signature weight (W) improved DSIR for <100 overlapping signatures, the opposite was true if a high amount of signature overlap was expected (Fig. 2E). Hence, a value of $W = 10$ appears to be optimal from our experiments.

A major concern of electronic communications in real-world scenarios is the reliability of the system when exposed to electromagnetic interference (EMI). Hence, we evaluated the effects of two common sources of EMI on ACES: (i) the 13.56-MHz band from radio frequency identification (RFID) devices and (ii) the 2.4-GHz band from Wi-Fi, Bluetooth, and microwave ovens. Using only basic shielding, the effects of 13.56-MHz waves did not significantly impair the signal quality of ACES (fig. S1D). Similarly, the EMI of 2.45-GHz waves had little effect on signal quality (fig. S1E). This is also expected given that 2.45-GHz frequency is beyond the bandwidth of the operational amplifier used. Thus, ACES can function reliably in real-world scenarios without the need for extensive shielding.

Neuromimetic tactile representations using ACES

The ACES signaling scheme is uniquely suitable for encoding biomimetic somatosensory representations (13) because it can propagate the sparse stimuli events from numerous receptors with high temporal precision. We designed a set of biomimetic models using our ACES platform to mimic the fast-adapting (FA), slow-adapting (SA), and temperature afferents by integrating flexible tactile and temperature sensors that communicate through ACES pulse signatures (Fig. 3A). Similar to their biological counterparts (13), ACES-FA receptors respond only to dynamic skin deformations (fig. S2A, see Materials and Methods for details) but are insensitive to static forces, whereas ACES-SA receptors respond to static pressure by producing events at frequencies that increases with greater force amplitudes (Fig. 3B and fig. S2B).

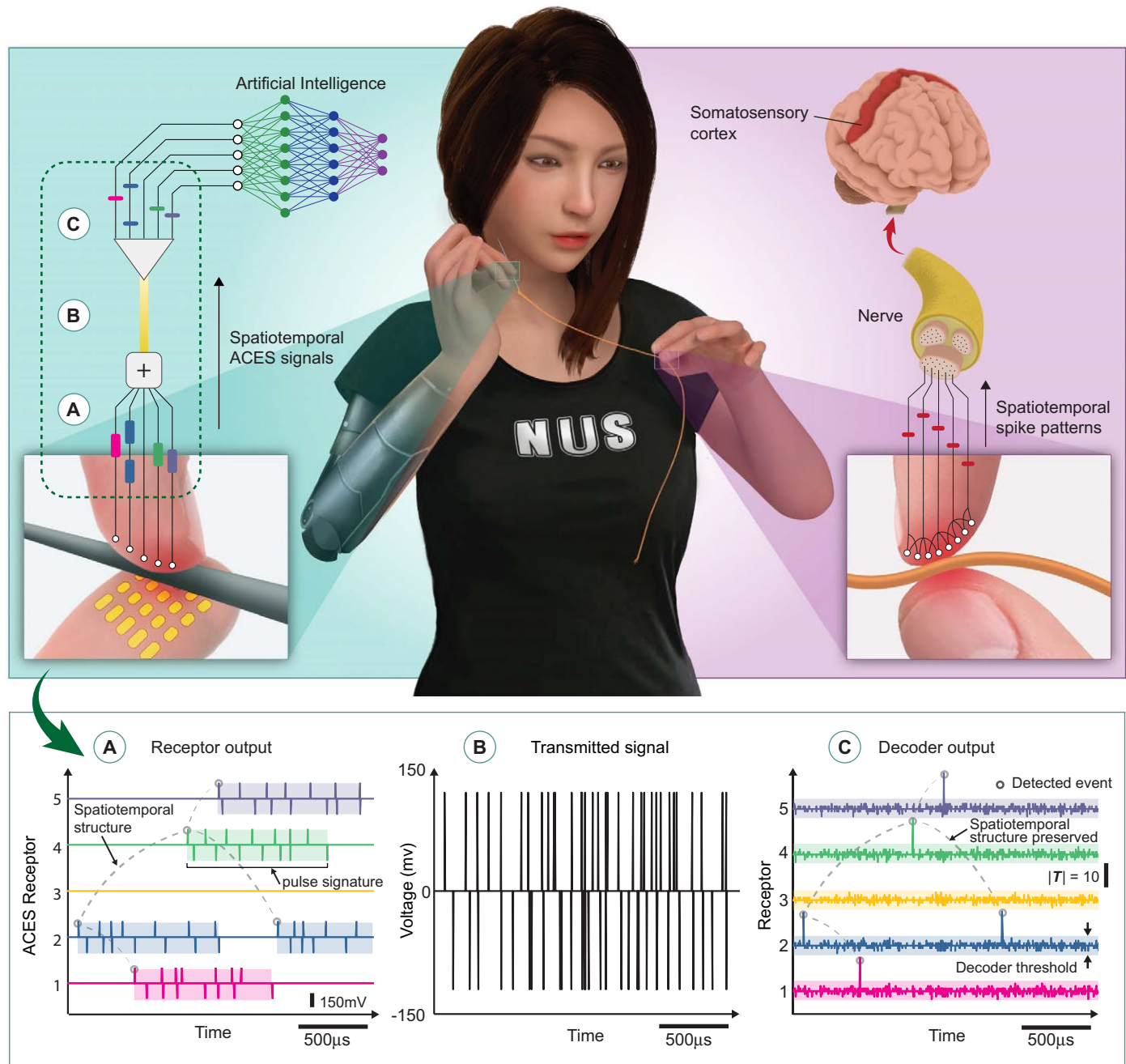


Fig. 1. The ACES architecture for large arrays with single-wire transmission capability. Illustration of ACES artificial receptors on e-skin (left) that independently and asynchronously transduce tactile events into pulse signatures, analogous to biological action potentials, or spikes (right). (A) ACES receptors generate tactile events with spatiotemporal structures (dashed lines) that encode the stimulation sequence. (B) Pulse signatures are combined and propagated via a single conductor. (C) Decoders match pulses in time. A strong match with correlation $|T|$ exceeding a predefined decoder threshold (shaded box) indicates the presence of an event by the particular receptor. The decoded events preserve the spatiotemporal structure (dashed lines) of receptor activation with ultra-high temporal precision, resembling the spike patterns that enable rapid discrimination in the somatosensory system.

The ACES-SA receptors enable accurate reconstruction of applied forces by extracting the inter-event intervals (Fig. 3B). However, ACES-SA receptors alone could not accurately reproduce transient stimuli, such as a prick from a lancet lasting 1 ms, because the stimulus duration was below the interval needed to encode the force intensity. Instead, we reliably detected the applied impulse using the

ACES-FA receptors because they transmit events immediately due to an increase (FA+) or decrease (FA-) in pressure. Our system allows for ACES-FA receptors to also encode for a decrease in stimuli magnitude, as opposed to biological FA receptors that do not discriminate between increase or decrease in force (12). However, ACES-FA output may not accurately track pressure intensities due to the dynamic nature

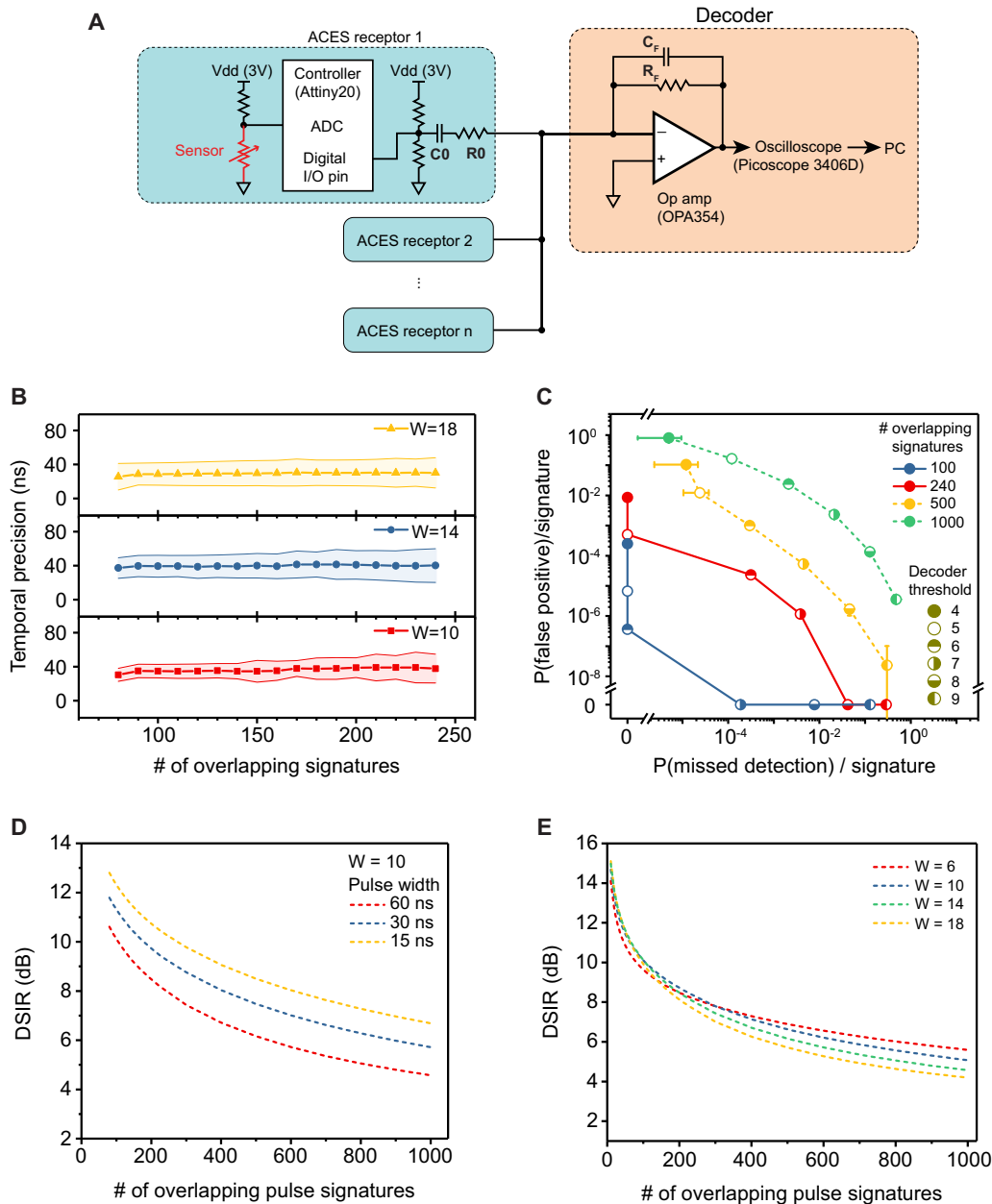


Fig. 2. Architecture and performance characterization of ACES. (A) Multiple ACES receptors (blue blocks) are connected to a decoder (orange block) via a single electrical conductor. The red variable resistor represents the resistive sensors used in the current prototype. (B) Temporal precision of decoded events versus number of overlapping pulse signatures for different numbers of pulses per signature (W). Shaded regions indicate SD. (C) The influence of decoder threshold on the detection error tradeoff graph ($W = 10$). Dashed lines represent simulation results. Error bars indicate 95% confidence bounds for simulated results. (D) Influence of pulse width on DSIR. (E) Effect of W on DSIR (pulse width = 60 ns). Simulations were used to derive (D) and (E).

of the threshold used (Fig. 3B and fig. S2A). Combining both FA and SA behaviors, e-skins based on ACES have the versatility to react to a wide temporal range of tactile stimuli.

To demonstrate a multimodal sensing e-skin using ACES, we developed an e-skin integrated with flexible pressure- and temperature-sensitive transducers (Fig. 3C and fig. S2C). We used pressure transducers that have heterogeneous transduction profiles by altering the Young's modulus of microstructured elastomers (29). This increased

the dynamic range of pressure transduction (fig. S2, D to F). We also designed resistive temperature sensors that are most sensitive between 20° and 50°C (fig. S2, G and H), which is similar to the cold receptor afferents in human skin (30). Akin to biological cold receptors, our ACES thermoreceptor transmitted events at a reduced frequency as the surrounding temperature rose above 25°C (fig. S2I).

Combining both transducers on a single sheet of flexible substrate, we show that our ACES platform could perform simultaneous multimodal sensing. We demonstrate that, when an ACES-equipped prosthetic hand grasped a hot cup of coffee, simultaneous detection of thermotactile sensations could be achieved (Fig. 3, C to F).

Slip detection via spatiotemporal ACES events

The ability to preserve the spatiotemporal profile of tactile stimuli allows rapid detection of object slippage, which is essential for grasp stability during in-hand object manipulation (31). Inspired by the simplicity and computational efficiency of optical flow algorithms in event-based vision sensors (32), we implemented a spiking convolutional network that computes the magnitude and direction of slippage in an event-driven manner (Fig. 4A). As an object is pulled out of grasp, ACES-FA receptors are triggered in particular spatiotemporal sequences that could indicate the onset of slippage. By comparing each received event with other events correlated in space and time (see Materials and Methods), our network obtained movement estimates immediately upon slip onset and accurately identified the downward movement of the object (Fig. 4B).

The system was also capable of detecting the slippage of a needle with minimal latency (Fig. 4C). Despite the fair accuracy of the speeds derived compared with a high-speed camera, the spike timing patterns can indicate and enable control systems to react with low latencies (<10 ms).

Speed and force invariant classification of texture

Humans typically recognize surface textures by sliding their fingers over them laterally. Studies suggest that the humans can discriminate

textures independent of sliding speed (33)—a feat that is difficult to replicate in robotic tactile sensing because controlled tangential speeds are needed to derive stable frequency-based descriptors (21, 34, 35).

We propose that our motion estimation technique (Fig. 4A) can be applied to improve texture recognition when sliding speeds are inconsistent.

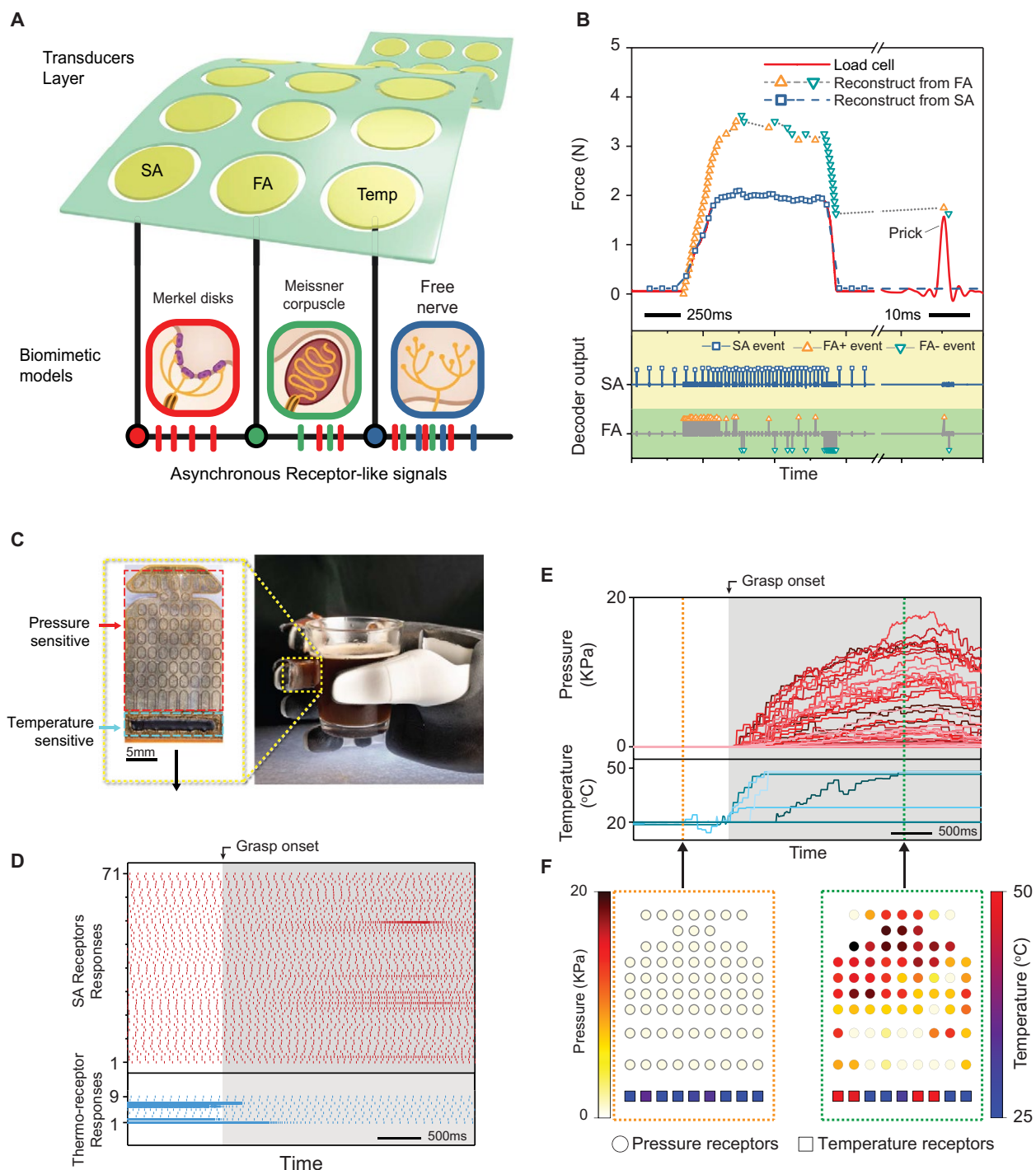


Fig. 3. ACES receptor responses using integrated multimodal sensors. (A) Schematic of a multimodal ACES sensor array, where biomimetic models convert tactile stimuli to events that are concurrently propagated. (B) Top: Estimation of time-varying force intensities based on ACES-FA and ACES-SA receptor output. Center and bottom: Decoded response of ACES-FA and ACES-SA receptors to a force profile. The ACES-SA receptor modulates pressure intensity by event frequency, whereas the ACES-FA receptor transmits FA+ and FA− events whenever it increases and decreases in pressure are experienced, respectively. (C) Photograph of the flexible multimodal sensor, wrapped onto a finger of a robotic hand grasping a cup of hot coffee. Inset: Photograph of the sensor. (D) Raw events generated from pressure-sensitive (top) and temperature-sensitive (bottom) ACES receptors during the grasp. (E) Derived pressure and temperature profiles from (D). (F) Snapshots of pressure and temperature distribution before and after grasp onset.

We constructed a dataset by sliding an array of ACES-FA receptors over gratings of various spatial density by hand, thus ensuring that the sliding speed and normal forces varied within and between trials (Fig. 5, A and B). This substantially increased the problem complexity, because the variability in tangential speed meant that a larger grating pitch did not always trigger receptors less frequently (Fig. 5B). Next, we extracted frequency component information by compiling the distribution of ACES-FA event time intervals into a feature vector for classification by a neural network (see Materials and Methods). Our event-based motion estimation technique was also used to approximate the tangential speed of the recording. We observed that discriminating between the grating sizes was about 14% more accurate when the estimated tangential speed was included as a feature (Fig. 5C). The improvement agrees with our expectation, because the approximated sliding speed tracked the actual speed fairly well. In addition, we observed a decrease in recognition performance from 88 to 50% when temporal resolution of the events decreased from 1 to 40 ms, indicating the importance of having tactile sensors with high temporal fidelity. The sparsity of the ACES-FA representation allowed for exceptional timing precision while consuming only a fraction of the data rates produced by a comparable frame-based im-

plementation (Fig. 5D). Moreover, frequency domain information was derived directly from the inter-event time intervals, unlike frame-based systems where complex operations such as Fourier transformations are typically required (36).

Rapid perception of local curvature and object hardness

Dexterous manipulation tasks often necessitate the ability to swiftly perceive object local curvature and hardness from mechanosensory stimuli. In humans, manipulation tasks are typically executed as action phases delimited by mechanical contact events (13). Meissner corpuscles, with their FA responses of submillisecond precision, play a crucial role in rapid and reliable contact classification to ensure seamless transitions between action phases. Using an array of 69 ACES-FA receptors (Fig. 6A) coupled with a spike-based classification technique (see Materials and Methods), we demonstrated the ability to classify various local curvatures up to 10 times faster (<7 ms for 97% accuracy) than a 100-frame per second (fps) conventional sensor array (Fig. 6, B and C). A marked improvement in classification speed was also observed when the ACES-FA receptor output was used to distinguish geometrically identical shapes of different hardness (Fig. 6, D to F). Our results highlight the importance and effectiveness

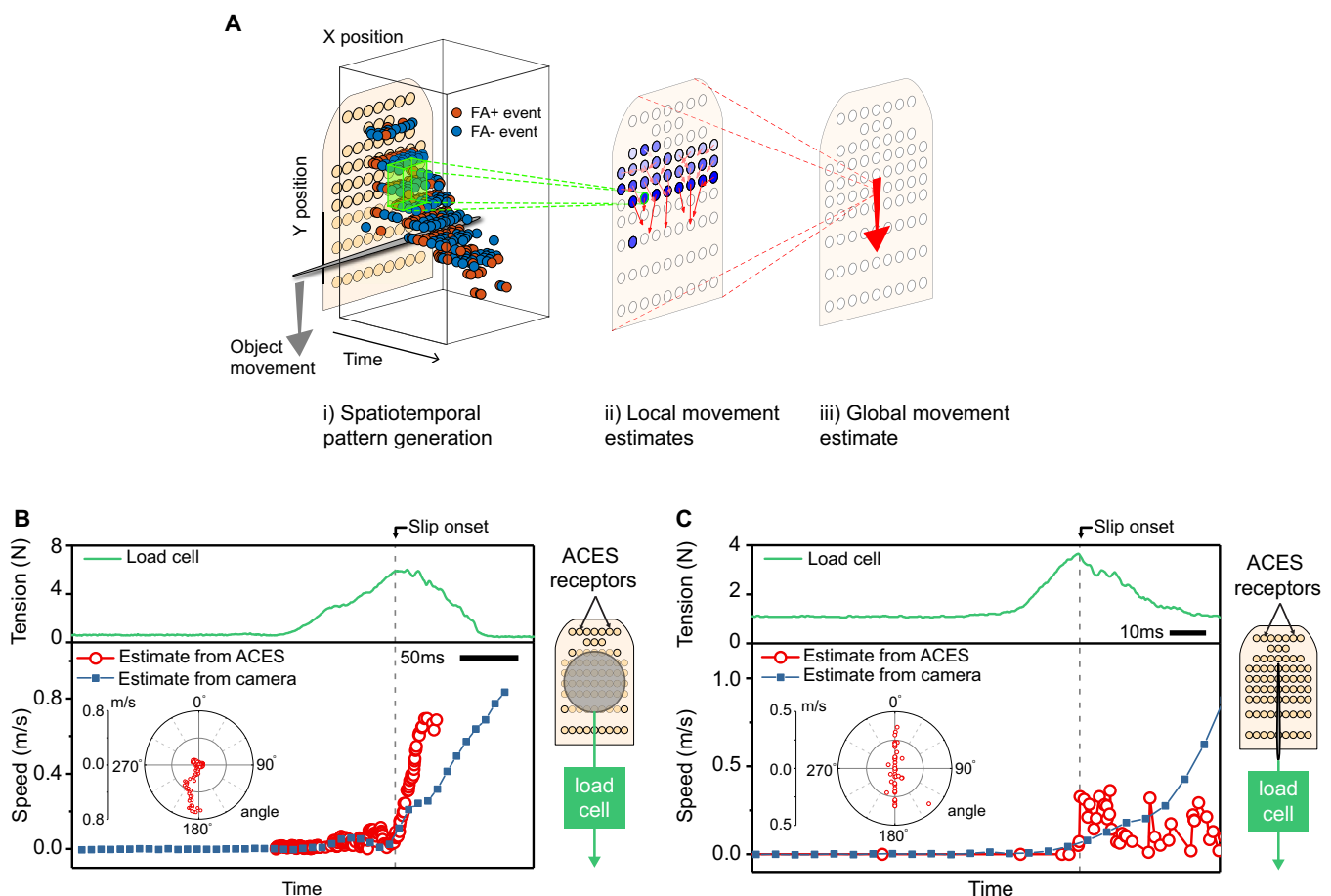


Fig. 4. Spatiotemporal patterns in the ACES-FA receptor response enable rapid slip detection. (A) Schematic of the event-based algorithm for movement estimation. The input consists of spatiotemporal events from ACES-FA receptors. Each intermediate node computes local movement estimates for every event occurring within its sensitive region. The global movement estimate is computed by pooling responses from intermediate nodes. (B) Detection of slip as a disc is pulled out of grasp. A sudden loss of string tension indicates slip onset. Estimates of movement and direction are shown in the bottom and inset panels, respectively. Blue trace indicates speed derived from optical tracking (see Materials and Methods). (C) Detecting slippage of a needle.

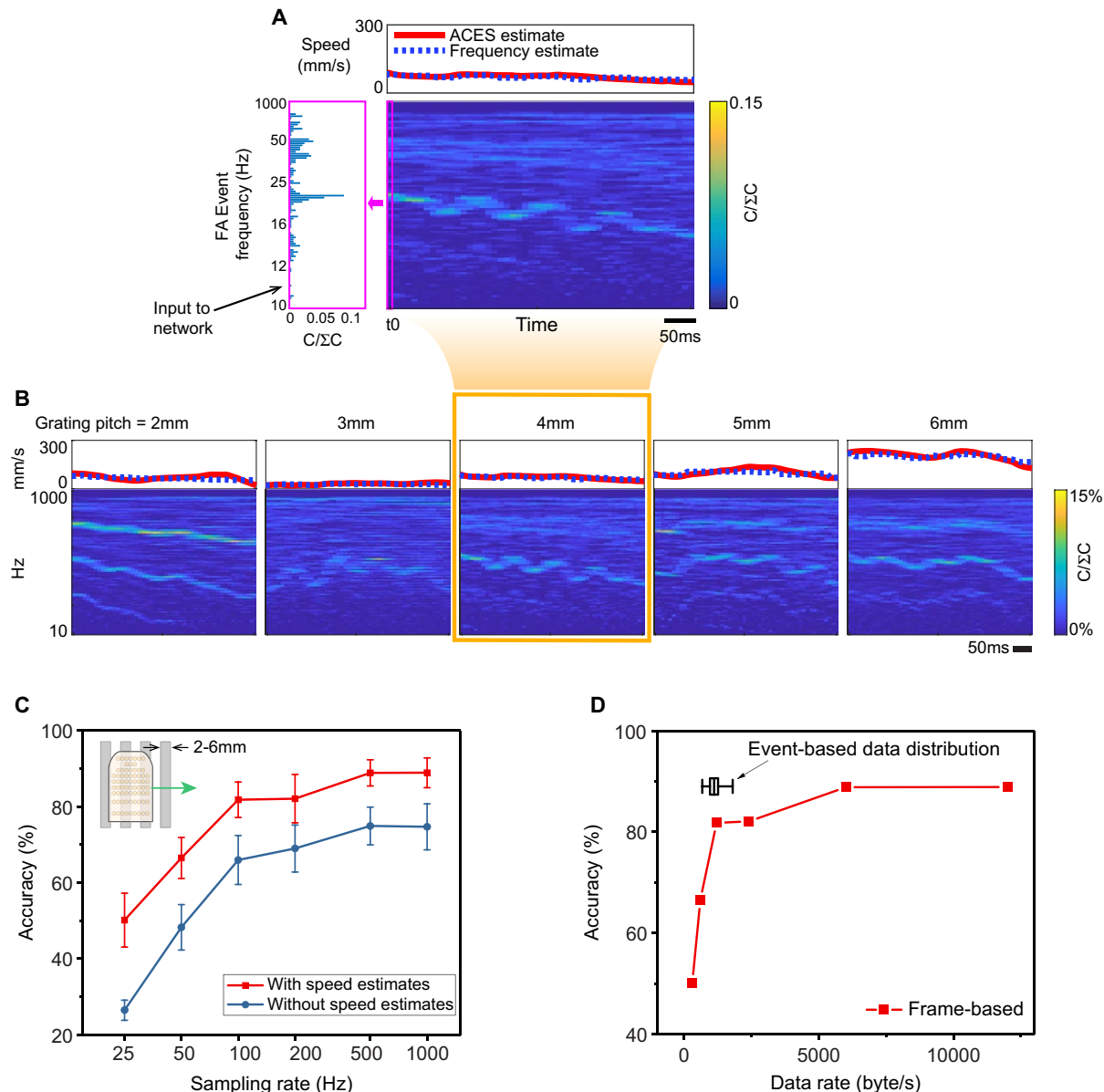


Fig. 5. Feature computation and classification of grating pitch. (A) Features extracted from an example trial where an ACES-FA receptor array is moved over a grating pattern with 4-mm pitch. Red trace in top panel indicates estimated tangential speed using ACES, whereas the dotted blue trace is the ground truth speed derived from frequency components (see Materials and Methods). Bottom panel is a 2D histogram of event frequencies over time. Color codes correspond to bin counts C normalized by total counts ΣC . Magenta box highlights the 100×1 vector of bin counts for classification. (B) Selected trials illustrating outputs from five different grating sizes. (C) Classification accuracy of grating sizes obtained when sampled at different rates. Error bars denote SD. (D) Comparison of classification accuracy achievable (with speed estimates) based on data rate consumed. For event-based output, the vertical line in the box plot is the median data rate, the caps indicate 1st to 99th percentiles, and the sides of the box indicate 25th to 75th percentiles of distribution.

of temporal features in rapid tactile discrimination, which concur with earlier studies in both biological and artificial somatosensory systems (12, 37).

ACES enables flexible arrangements of sensor array

Wiring simplicity in e-skins is critical, especially when routing wires along nonuniform surfaces or curvatures. Our ACES pulse-based communication approach markedly simplifies the problem because it allows signals from all receptors to be transmitted through a single, common electrical conductor. Here, we demonstrate nine ACES-FA

receptors that could be dynamically rearranged to form multiple spatial formations as long as contact with a sheet of conductive fabric was maintained (Fig. 7). Thus, our ACES platform enables highly irregular sensor arrangements that accommodate nonuniform geometries.

Robustness of ACES networks to physical damage

E-skins are vulnerable to wear and tear because they constantly rely on physical contact to sense the environment. Our ACES platform enables sensor arrays to have high signaling redundancy when

interfaced to a common electrical conductor, which translates to exceptional robustness against physical damage. We exemplify this level of resilience by showing that an array of 16 ACES-SA receptors retained full functionality even when the substrate was cut at multiple sites (Fig. 8, A and B, and movie S1). In comparison, a conventional row-column multiplexed tactile sensor array experiencing similarly large damage could not sense at most locations (Fig. 8, C and D, and movie S2). Unlike network-based approaches (24), ACES networks do not require reconfiguration when damaged and hence experience zero down time (movie S1).

DISCUSSION

ACES is designed to be a communication technique analogous to nerve bundles in biological skin. Toward this goal, our ACES electronic peripheral nervous system architecture presents several advances in e-skin technology. First, ACES uses a novel spread-spectrum technique to asynchronously multiplex signals from a large number of transducers while achieving near-constant latency despite increasing the number of transducers. Therefore, larger, full-body e-skins with thousands of spatially distributed sensors can be realized without compromising system responsiveness to stimuli.

Second, being event based, our ACES platform enables tactile signals to be captured with ultra-high temporal fidelity, because the sampling time is not constrained to a central “clock.” This enables precise time recording of tactile events that enable rapid slip detection and shape and material hardness classifications and also provides movement estimates that enhance speed-invariant texture discrimination. Third, ACES allows signals from all receptors to propagate via a single shared conductor, empowering roboticists and sensor system designers with outstanding levels of flexibility when spatially arranging sensors on the e-skin. Furthermore, the high redundancy of physical

connections when a planar conductor is used ensures damage robustness of ACES sensor arrays—the system is shown to operate uninterrupted even when multiple mechanical cuts were being made to the substrate.

Having tactile signals that are millisecond precise may appear excessive and unnecessary, especially if the time frames for physical motion of the intended robot are not particularly fast. However, recent works in both biological and artificial tactile sensing have shown that significantly more information is present in the temporal domain of tactile signals than the rate of movement suggests (12, 13). For instance, vibratory signal above 100 Hz are often used to characterize slip events (15), whereas the high-frequency signal components generated when sliding tactile sensors over surfaces typically describe texture (21, 38). In the human skin, there are specialized mechanoreceptors that respond to transient stimuli from 50 to 500 Hz (13). Therefore, harnessing tactile signals with millisecond precision will enhance texture recognition and dexterous manipulation capabilities while ensuring safety in human-robot interactions. Moreover, the event-based nature of ACES allows e-skins to achieve impressive temporal resolutions at a fraction of the data rates needed by existing systems.

ACES appears to have similarities with AER, which is an asynchronous protocol used in neuromorphic devices (23). However, ACES and AER are fundamentally dissimilar and serve different niches. First, AER is a point-to-point protocol, simulating a network of interconnected neurons. Events from a neuron are routed to another particular neuron by matching the address with a routing table. It is thus possible to have bidirectional communication in AER. Conversely, ACES is a many-to-one protocol, where there is only one receiver in the network. Therefore, information can only flow from the sensors to the receiver. ACES is thus more suitable for sensing applications, whereas AER may be used in both sensing and computational networks.

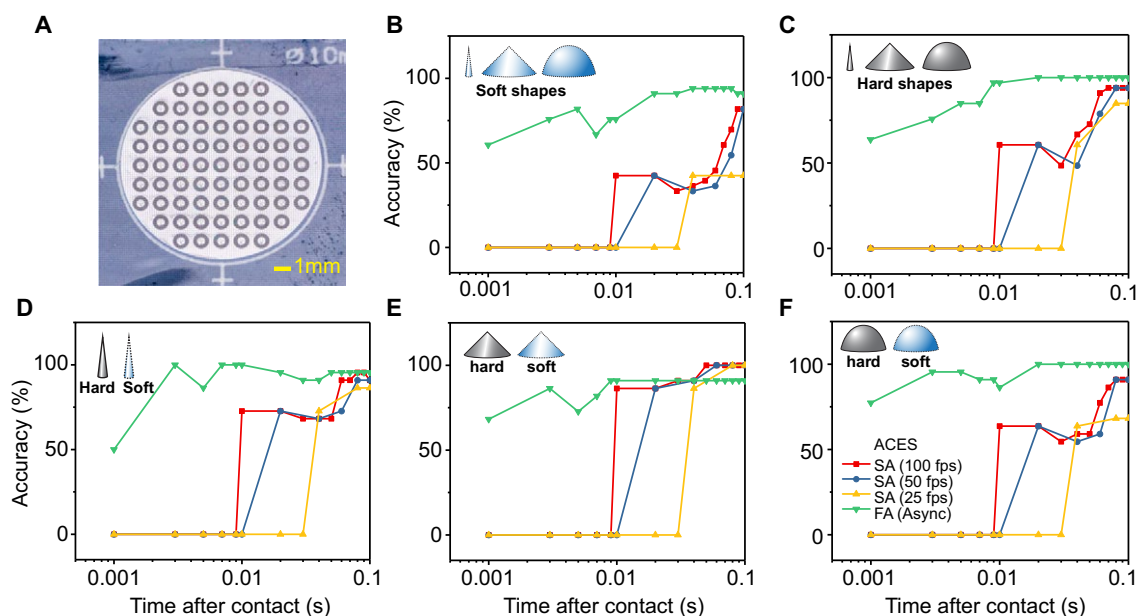


Fig. 6. Classification results from the indentation of objects with various geometry and hardness. (A) A photograph of the sensor array used in the experiment, where classification speed and accuracy were compared using outputs from ACES-FA and ACES-SA receptors, for rapid indentation of (B) three soft shapes, (C) three rigid shapes, (D) two sharp cones of different hardness, (E) two broad cones of different hardness, and (F) two domes of different hardness. Shapes are shown in insets at top left.

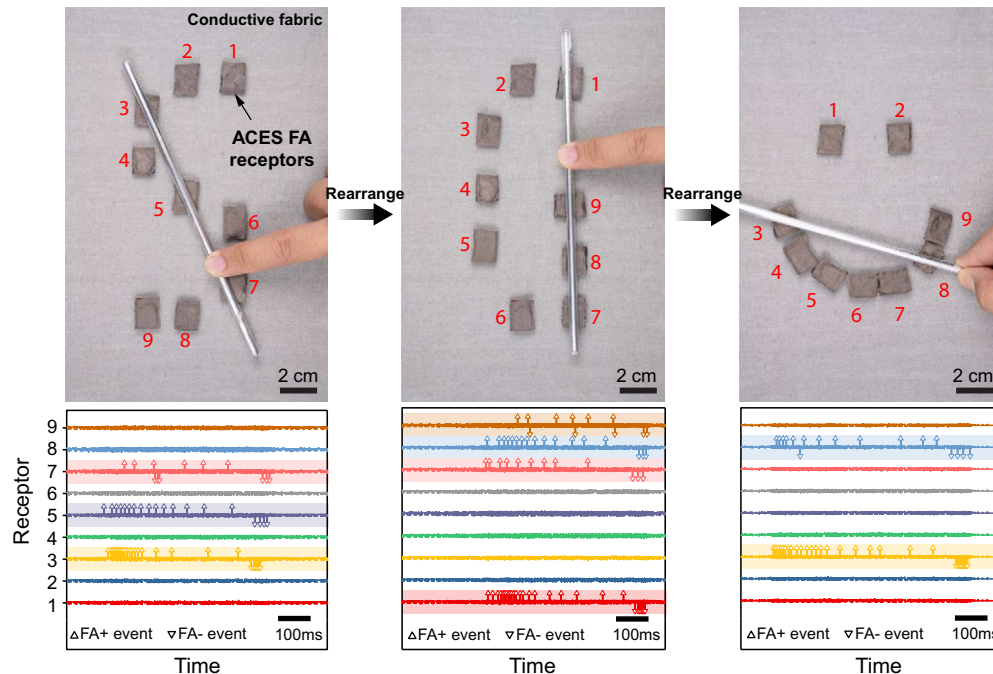


Fig. 7. Flexible and dynamic reconfiguration of ACES receptor placement. Photographs of multiple ACES receptors dynamically arranged into three different formations on a sheet of conductive fabric. The plots below each photograph indicate the corresponding FA events decoded when pressure is applied on each formation. Colored boxes highlight the receptors that responded.

Second, AER is time multiplexed, and arbitration is necessary to ensure proper sharing of the address bus. Most implementations use flow control to ensure that the receiver is ready to receive the next event. AER thus have very low transmission error rates but require additional logic for arbitration and handshaking. Most systems today use field programmable gate arrays (FPGAs) for AER interfaces, which leads to larger and more complex devices. In contrast, ACES can “fire-and-forget” with no need for any arbitration or flow control. However, ACES is not error free, and one potential disadvantage is that it is impossible for the transmitter or receiver to know whether a loss of sensing event has occurred. Nevertheless, the simplicity of ACES enables simple transmitter implementations and exceptional timing precision of stimuli events. Tiny sensor nodes embedded in elastic substrates can be used to implement the ACES platform at relatively high density and sensing speed, which are especially important for e-skin applications where real-time motions and mechanical compliance are critical.

Third, most implementations of AER require numerous wires to realize, from four wires per sensor in serial AER (39) and up to 32 wires in parallel AER (40). Although this enables AER systems to have superior throughput (typically millions of events/s) compared with ACES (hundreds of thousand events/s), it requires high-density interconnects fabricated in silicon. For e-skins, replicating such patterns over large areas on flexible and stretchable substrates remains impractical. Therefore, in e-skin applications, our ACES’s single-shared conductor approach is highly desirable.

Last, our ACES platform can complement AER, rather than being mutually exclusive. For example, we can describe ACES as the nerve bundles from mechanoreceptors (sensors) to the spinal cord (communications bus), where events received can then be transported via AER from the spinal cord to the brain (central microprocessor) of the robot. Such a hierarchy permits compliant and damage robust e-skin patches to be wrapped over the exterior of the device, complemented by high-

throughput AER networks at the robot interior where the use of rigid components are more palatable.

The use of one microcontroller per receptor in current prototypes is currently more complex when compared with $n \times m$ type cross-bar sensor arrays, inevitably imposing upper limits to the network in terms of spatial density and power savings. However, similar limitations apply to any sensor array that uses local computation and communication. For instance, Roboskin (41) on the iCub has 12 elements per 390-mm² triangular tile (32.5 mm² per receptor), whereas Hex-o-skin (42) has 3 elements per 665-mm² module (221 mm² per receptor). With a density of 9 mm² per receptor (fig. S3A), ACES compares favorably with the aforementioned systems in terms of achievable spatial density.

Our prototype consumes about 7 mW per receptor at 3.3 V (see Materials and Methods). Similar e-skin architectures consume from 0.2 to 55 mW per sensor (41–43). It is important to note that our reported power consumption should be seen as an upper bound, because these microcontroller-based prototypes were meant to be an early demonstration of ACES using off-the-shelf components. We anticipate smaller-sized receptors with much lower power consumption to be achievable through the use of application-specific integrated circuitry (ASIC). ASICs each representing several ACES receptors per chip are another viable option for regions of e-skin where extremely high densities of receptors are desirable.

A key benefit of ACES-enabled e-skins is the unique combination of high mechanical robustness and wiring simplicity. The damage robustness of an ACES network stems from the physical redundancy of connections from each receptor to the receiver. When a planar conductor, such as a conductive fabric or stretchable electrodes, is used to link the receptors to the receiver, each receptor has multiple direct connections to the receiver. The ACES sensor network thus retains full functionality as long as there exists at least a physical connection

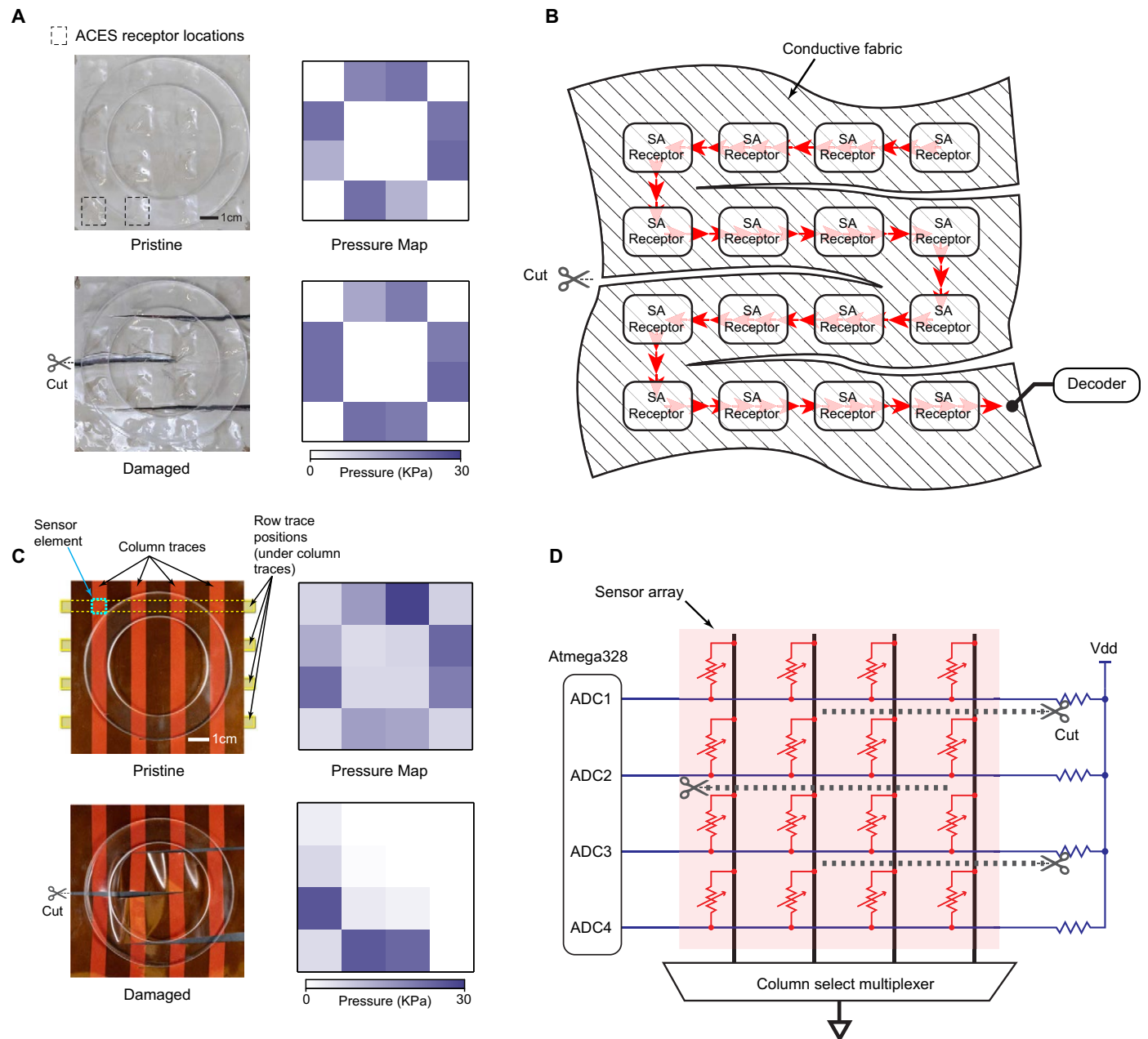


Fig. 8. Damage resilience of ACES architecture compared with a conventional row-column multiplexed array. (A) Photographs show the array before (top left) and after damage (bottom left), where the stretchable polyurethane substrate embedded with ACES-SA receptors was cut at three different locations. The corresponding pressure distributions reconstructed from ACES receptor outputs are shown to the right of each image. (B) Schematic of signal propagation path on damaged substrate of ACES receptor array. (C) Photographs of the four-by-four multiplexed sensor array before and after being cut. The plots to the right of each photograph indicate the sensor readout corresponding to the physical state of the sensor. (D) Schematic representation of the multiplexed sensor array. Each sensor element is represented by a variable resistor. Dashed lines indicate traces affected by the damage.

between each receptor to the receiver. Damage to the conductive substrate can be interpreted as merely a change in the shape of the substrate and will not affect the system functionality (movie S1).

The fire-and-forget nature of ACES pulse signatures also ensures that individual receptors are plug and play. Multiple patches of ACES e-skins are easily combined by connecting the substrates together. Similarly, gross damage that disconnects several receptors from the system will not affect the functionality of the remaining intact con-

nected network. Although the receiver may not immediately realize the loss of the disconnected receptors, additional protocols built on top of ACES can be implemented to identify missing receptors. For instance, disconnected SA receptors may be detected if no signatures from these receptors are received after a specified timeout period. The most severe damage that could occur would be the shorting of the signal carrier to the supply voltage or ground traces, which would render the entire network inoperable. This affects any other wired

communication systems, and the vulnerability could be minimized through proper design of the system encapsulation.

The assembly of thousands of microcontrollers in a network may appear to be complex in comparison with $n \times m$ matrix-type sensor arrays. Although this is true for rectangular and uniformly distributed sensor arrays, the complexity of fabricating matrix-type arrays rapidly increases if stretchable e-skins with nonuniform densities are desired. In contrast, the bulk of the manufacturing complexity for ACES-based networks is the assembly of common electronic components, for which established volume manufacturing processes are available. Hence, in terms of manufacturing complexity, ACES-based networks can be implemented more conveniently for e-skins with nonuniform shapes and densities.

The mechanoreceptor models implemented in this paper are relatively simplistic when compared with the more sophisticated biological models reported (44). However, our goal is not to reproduce biologically accurate tactile responses but to capitalize on ACES's high temporal resolution for efficient information transfer. ACES events, similar to biological action potentials, are slow to propagate individually (1 ms), but these events can be propagated in parallel. If each event were to represent a single bit of a digitally encoded resistance value, it would require eight events (assuming 8-bit resolution) to be transmitted consecutively, resulting in an 8-ms latency. Theoretically, this latency can be reduced to 2.015 ms using ACES-SA encoding, where two events spaced 15.36 μ s (60 ns \times 256) apart effectively encode the same 8-bit resistance value. Moreover, current research efforts suggest that classification tasks can be rapidly performed even without knowledge of actual pressure values by relying on relative time differences of events from a population of receptors (12, 13, 37). ACES-FA encoding is designed to meet these requirements. Nevertheless, the programmable nature of individual ACES receptors ensures that receptor models with increased efficiency and/or biologically relevant mechanoreceptor output can be simulated in the future.

By using our ACES architecture, one potential disadvantage is the increase in computational complexity required in the receiver/decoder. Similar to the human brain, concentrating power-hungry operations at the receiver makes practical sense, because this design allows the sensors (transmitters) to be realized with low-cost and low-power hardware.

In summary, our neuro-inspired ACES platform enables highly scalable and ultra-fast somatosensory perception, with damage robustness and sensor placement flexibility that rival wireless solutions. We anticipate that our ACES architecture will help advance human-machine-environment interactions for autonomous anthropomorphic robots (2), naturalistic embodiment of neuroprostheses (2, 4, 45), high-performance brain-machine interfaces (46, 47), and soft machines (48).

MATERIALS AND METHODS

Objectives and study design

Our objectives were to show that (i) the ACES architecture have the capacity to multiplex asynchronously transmitted events from numerous sensors and reconstruct the output at the receiver and (ii) the high temporal resolution of tactile events afforded by our technique can be uniquely applied to solve various challenges in e-skin applications.

Design of ACES pulse signatures

We designed a set of electrical voltage pulse codes that can be asynchronously added with a low probability of false decoding. Each pulse signature consists of W voltage pulses spaced apart at

specific time instances $\tau = \{\tau_1, \tau_2, \dots, \tau_{w-1}\}$. At the receiving end, pulses are received at time instances $\tau' = \{\tau'_1, \tau'_2, \dots, \tau'_v\}$. The decoder finds the intersection $T = (\tau \cap \tau')$ where the cardinality $|T|$ denotes the correlation strength. If $|T|$ exceeds a predefined decoder threshold, then an event is deemed to have been received (Fig. 1C).

An ideal set of pulse signatures is one that has minimal autocorrelation and cross-correlation. There should also be enough unique signatures in the set to identify all receptors in the array. A family of pulse signatures may be characterized by three parameters: F , the maximum number of voltage pulses that fit within the duration of a signature; W , the weight or the number of pulses per signature; and L , the maximum allowable interference (autocorrelation and cross-correlation) between two signatures.

For a pulse signature with a duration of $T_s = 1$ ms with voltage pulses lasting $T_p = 100$ ns each, the maximum number of voltage pulses that fit within the duration of a signature is calculated as follows

$$F = \frac{T_s}{T_p} = 10,000$$

Parameters W and L are closely related to the error performance of the decoder. Under ideal conditions, all W voltage pulses from the target signature should match the template; thus W is the maximum signal strength. L is the maximum interference (cross-correlation) allowed between signatures of the same family. If N nontarget signatures overlap, the amount of interference could be as high as $N \times L$. Error performance will thus degrade if $N \times L$ is much larger than W .

There needs to be as many unique pulse signatures as there are ACES receptors. To accommodate for the thousands of ACES receptors, the pulse signatures used in this paper have $L = 2$ such that the number of signatures (C) will follow the inequality (49)

$$C \leq \frac{(F-1)(F-2)}{W(W-1)(W-2)}$$

Therefore, with $F = 10,000$ and $W = 10$, the array could accommodate up to 138,847 receptors, which should be sufficient for whole-body robot sensor skins (50). Several techniques for finding these codes have been documented (51).

The voltage pulses in a pulse signature can either be positive or negative, which allows each signature to take on several variants. For instance, ACES-FA receptors use two variants of the same signature to indicate an increase or decrease in pressure.

Implementation of an ACES receptor array

Each receptor consists of a resistive sensor, a microcontroller, and several passive components to perform signal conditioning (Fig. 2A). A potential divider circuit converts sensor resistance to voltage values. The voltage is then sampled at 10 KHz with 10-bit resolution by the analog-to-digital converter (ADC) on the ATtiny20 microcontroller (Microchip Technology, United States). The values are sent to firmware models to mimic the fast (FA) or slow (SA) adaptation behavior of receptors found in the human skin.

The ACES-FA model generates an event whenever a measured voltage has changed more than 50 mV since the last transmitted event. A positive/negative pulse signature is transmitted for a pressure increase/decrease event, respectively. After transmitting the event, a new

voltage baseline is set (fig. S2A). To mimic SA mechanoreceptors, the model generates events at intervals proportional to the 8-point averaged ADC value. For Fig. 3, each bit corresponds to a 100- μ s interval. Together with a 1-ms pulse signature duration, a maximum ADC value of 1023 will correspond to a 103.3-ms interval between events. Receptors with the SA-mimicking model only transmit positive versions of the pulse signature.

The pulse signature is generated by toggling a digital pin at specific time intervals. A capacitive high-pass filter is used, ensuring that only the high-frequency components of the signal are transmitted, in the form of voltage pulses (fig. S1A). An inverted summing circuit, constructed using an OPA354 operational amplifier (Texas Instruments, United States) is used to combine pulses from the multiple receptors. The resultant signal is digitized with 8-bit resolution at 125 MHz using an oscilloscope (PicoScope 3406D). Decoding is performed offline (MATLAB). Voltage thresholds V_+ and V_- (fig. S1A) are heuristically defined in software to be 40% of the amplitude of a single pulse, above and below 0 V, respectively. We found this to reliably distinguish a pulse from background noise.

For the purpose of controlled experimentation, we developed a modular prototype system consisting of a board with 80 ACES receptors. The transducer arrays, namely, a flexible 80-element transducer for slip detection and grating classification (Fig. 3C and fig. S3B) and a rigid 69-element transducer for local curvature and hardness classification (Fig. 6A and fig. S3C), are separate entities that can be attached to the board for data collection. An additional prototype of 25 receptors connected by serpentine interconnects (fig. S3A) was also developed for the purposes of movie S1.

Effects of EMI on signal

We induced EMI of 13.56 MHz using a commercial RFID card reader (RF430FRL15xH by Texas Instruments). We evaluated the condition when the antenna was placed 1 cm above the summing amplifier and when the antenna was placed 1 cm above a 15 cm by 15 cm conductive fabric used for propagating the ACES signatures. We also evaluated the situation when a grounded aluminum foil sheet was placed between the antenna and the system for the aforementioned antenna placements. The signal-to-noise ratio, computed as the ratio of peak-to-peak voltage of the pulse signature to the peak-to-peak voltage of background noise, was measured for all four scenarios. The 2.4-GHz interference was projected from a TG.30 antenna (Taoglas Antenna Solutions) driven by a SMB100A signal generator (Rohde & Schwarz). The antenna was placed 20 cm above the ACES system without shielding.

Determining DSIR and verification of timing precision

The physical hardware test setup consisted of an array of 240 receptors, each programmed with a unique pulse signature. Each trial began with an external digital edge signal that was broadcasted to all receptors, triggering a request for transmission. Upon receiving the trigger, each receptor transmits its pulse signature after a random delay of less than 1 ms, thus ensuring that all the 240 transmitted signatures will overlap at varying temporal offsets between trials. For trials involving less than 240 receptors, the excluded receptors were programmed to ignore the trigger.

Receptors (16 of 240) have recording probes attached to their transmission pins. Signals from these probes served as the ground truth on the actual time of the pulses transmitted. The digital signals from these 16 probes, as well as the combined pulse signatures from the 240 recep-

tors, were digitized at 125 MHz simultaneously by a mixed signal oscilloscope (PicoScope 3406D), thus ensuring that all channels were synchronized in time.

We defined the DSIR metric to characterize the ease of an event detection in the presence of interference. The interference value is taken to be the root mean square value of the decoder correlation $|T|$ during the 1-ms transmission window of the corresponding receptor. The last 100 ns of the decoder output was excluded because it corresponded to the detection of the correct pulse signature and should not be considered as interference. DSIR was then computed as the ratio between the signature weight W and the interference value. The DSIR was computed for the decoders responsible for the 16 probed receptors. For each network size, 1000 trials were conducted, and the reported DSIR was averaged from the 16 decoder outputs across all 1000 trials.

Timing precision was obtained as the difference in time between the start of last pulse transmission, as obtained from the attached probe, and the transmission time, as determined from the decoder output. The measured time difference is thus independent of the mechanical inertia of the transducer and processing time of the receptor model. The reported timing precision in Fig. 2A was obtained as an average across all the 1000 trials for each network size.

Monte Carlo simulations

Monte Carlo simulation was used to obtain estimates of DSIR for network sizes above 240 receptors. For each trial involving a simulated array of N receptors, we additively combined N pulse signatures at random offsets of <1 ms. The resultant signal was then decoded using the same software decoder as the physical experiments. One thousand Monte Carlo trials were performed for each network size.

SPICE simulations

The temporal precision of an ACES receptor array was limited mainly by the duration of a single voltage pulse. As more receptors are added to the array, the capacitance of the electrical conductor in which the pulses propagate also increases. The increased capacitance results in reduced phase margin of the op-amp feedback loop and causes ringing in the output (fig. S4A). The ringing can be reduced by increasing the feedback capacitance (C_F in fig. S4B) to improve stability. However, pulse width also increases as a consequence. Hence, a SPICE simulation (Cadence Spectre) was used to determine how the pulse width changes with increasing number of receptors. The simulated circuit is shown in fig. S4B. The input from each receptor was modeled as a voltage source with a square wave. The edges of the waveform were high-pass-filtered to obtain the waveform of the voltage pulse. A transient simulation was run for $N = 200$ to 16,000. For each N , the value of C_F was swept to find the minimum C_F that has acceptably low levels of ringing (i.e., the overshoot does not exceed quantization threshold set at 40% of pulse amplitude). Last, the resultant pulse width was taken as the length of time in which the voltage remained above quantization threshold. The temporal precision of the system was interpreted as $0.5 \times$ pulse width.

ACES receptor response to pressure and prick

The analog output channel of a load cell (Mark-10) was connected to the ADC pin of an ACES-FA receptor and an ACES-SA receptor, as well as a recording channel on an oscilloscope, to serve as ground truth (PicoScope). The summed signal from both receptors was simultaneously recorded using a separate channel on the same oscilloscope.

Two mechanical stimuli were presented, the first being a finger press, and the second being a prick by a lancet using a lancing device. The load cell reading in Fig. 3B was smoothed using a moving average filter of eight points (OriginLab 2017). Reconstruction traces were computed offline using MATLAB.

Slip detection

An object (an acrylic disk with a diameter of 1 cm in Fig. 4B and a needle with a diameter of 0.8 mm in Fig. 4C) was held vertically between two flat surfaces of a bench vise. A pressure-sensing array with 80 sensors (fig. S3B), interfaced with an array of ACES-FA receptors, was pasted on one of the surfaces between the vice and the object. A thread connected the object to a load cell (Mark-10 5i). A separate piece of thread, connected to the opposite side of the load cell, was pulled to make the object slip out of the vice. The combined signal output from the 80 ACES receptors was sampled together with the analog output of the load cell at 125 MHz using an oscilloscope (PicoScope 3406D). Computation of slip detection was processed offline in MATLAB.

The computation of local movement estimates (Fig. 4A) was generally as follows:

1) For an event from a particular receptor A that occurred at t_0 , look for prior events from receptors within distance D of receptor A that occurred at t_{prior} where $t_0 - \Delta t < t_{\text{prior}} < t_0$.

2) For each prior event, compute movement magnitude

$$\text{Magnitude} = \frac{D}{t_0 - t_{\text{prior}}}$$

3) Movement direction for each prior event

$$\text{Direction} = a \tan 2(d_y, d_x)$$

where d_x and d_y are the x and y components of the distance D .

4) By averaging the magnitude and direction for each prior event, the local movement estimate at receptor A's location was obtained.

The global movement estimate was obtained as the moving average (exponential kernel of 5-ms time constant) for all the local movement estimates. For Fig. 4 (B and C), the time interval (Δt) was 5 ms and the distance (D) was 2 mm. Ground-truth information was obtained using an optical camera recording at 240 fps to track a marked spot on the thread during the experiment.

Grating classification

Rectangular holes of $40 \times X$ mm were laser-cut from 5-mm-thick acrylic sheets, where X denotes the pitch of the grating. A pattern consists of multiple holes were spaced X mm apart. Five grating pitches were used ($X = 2$ to 6 mm in 1-mm increments). Each pattern was at least 25 cm in length. An 80-element transducer array (fig. S3B) was mounted on a 5-mm-thick piece of polyurethane rubber for compliance (Ecoflex 00-30, Smooth-On). To achieve realistic stimulus conditions, we moved the array over the gratings by hand at varying tangential speeds and normal forces. Fifty trials were collected per pattern, each lasting at least 1 s. To extract frequency domain information, we compiled inter-event time intervals for each ACES-FA receptor into a histogram of 100 bins (1 to 100 ms in 1-ms bins). The histograms were computed from moving time windows of 100-ms duration in steps of 10 ms each. Thus, a 1-s trial generates 91 histograms. Speed

estimates were computed using the same technique as the slip detection with $\Delta t = 20$ ms and $D = 2$ mm and averaged within each time window. A secondary speed estimate was computed by multiplying the event frequency corresponding to the highest bin count with the grating pitch used. This second estimate serves as the ground truth (blue dashed lines in Fig. 5, A and B).

We classified individual histograms for grating pitch using a multi-layer perceptron with 40 hidden neurons and 5 output neurons (nntool, MATLAB). Input to the network was a 100×1 vector of bin counts from the inter-event time histogram (magenta box in Fig. 5A). For the case with speed estimates, a 101×1 vector was used, where the first 100 elements were the bin counts and the last element was the estimated speed. In all cases, the scaled conjugate gradient backpropagation (trainscg) algorithm was used for training. One randomly selected trial per grating was excluded for each training instance and used for test. Results reported in Fig. 5C were compiled from 20 training instances. To simulate reduced temporal resolution, we binned time stamps of the events at discrete values that are multiples of the reported temporal resolution.

To compare data rates (Fig. 5D), we assumed a single frame to consume 12 bytes (80 bits for 80 receptors + 1 byte escape character + 1 byte delimiter). An event was assumed to be 1 byte long (7 bits address, 1 bit data). Statistics for the event-based representation were obtained by counting the number of received events in all the 100-ms time windows acquired.

Local curvature and hardness classification

The shapes used for indentation are (i) a spherical dome of 10-mm radius, (ii) a broad right circular cone of 7.5-mm height and 10-mm base radius, and (iii) a sharp right circular cone of 7.5-mm height and 2.5-mm base radius. Each shape is 3D printed twice, once using a rigid RGD810 VeroClear material and a duplicate using the softer FLX980 TangoBlackPlus material (fig. S5A). A 3D printer (Objet260 Connex) was used to produce the shapes. (See PolyJet Materials Data Sheet for the material properties.)

The indentations were performed using a mechanized z -axis stage (Newmark) with force feedback from the force gauge (Mark-10 5i) at a speed of 50 mm/s. The depth of indentation of the softer shapes were fixed at 2800 μm , whereas the hard shapes were indented to depths that produced the same forces as their softer counterparts (~ 9 N for the sphere, ~ 2 N for the broad cone, and ~ 0.3 N for the sharp cone). The shapes were indented onto the center of a circular array of 69 sensors interfaced with an array of ACES-FA receptors covered with a 1-mm-thick elastomer sheet (fig. S5B). Each shape was indented 10 times, with a 5-s interval between indentations to allow the deformable material to recover. Data from the force gauge and the ACES receptor array were recorded simultaneously at 125 MHz using an oscilloscope (PicoScope). All the trials were repeated with the sensors interfaced with ACES-SA receptors to obtain the pressure distribution.

The ACES-SA receptors were programmed to have an inter-event interval of 10 μs per ADC bit. Therefore, each receptor will generate events at a rate of about 100 Hz and will thus be sufficient to model a conventional 100-fps pressure sensor array.

The output from ACES-FA receptors were classified on the basis of the Van Rossum spike distance measure (52), commonly used to quantify the similarity between spatiotemporal spike patterns. Events from each receptor are convolved with a double exponential kernel of 5-ms decay, thus yielding a continuous trace. Traces from the same

receptor can then be compared between trials using the Euclidean distance. The distance between any two trials was then calculated as the sum of the differences between individual traces. Last, a K -nearest neighbor algorithm (53) (with $K = 5$) was used to classify the trials based on the summed distances.

To classify the output from ACES-SA receptors, we first reconstructed traces of pressure intensity over time for each receptor based on the time interval between events. The traces were then decimated at 10-, 20-, and 40-ms intervals to mimic 100-, 50-, and 25-fps sensors, respectively. Similar to ACES-FA receptors, the Euclidean distance was used to compare the pressure traces from the same receptor between trials, and the sum of differences between individual traces defines the distance between any two trials. The K -nearest neighbor algorithm ($K = 5$) was then used to classify the trials based on their distances.

Free-form receptor arrangements

To demonstrate the flexible arrangement of ACES receptors on a single electrical conductor, we developed ACES receptors with their own battery power source. The receptors and the decoding circuit were connected only by a stretchable conductor (knit jersey conductive fabric, Adafruit). Pressure was applied by pressing a conductive rod onto the receptors. The conductive rod provides a charge return path, such that charges from the environment can flow back to the receptors by coupling with the human operating the prototype. The same effect could have been achieved through the use of a grounded conductive encapsulant. However, this approach was omitted to achieve better clarity of the demonstration.

Robustness against severe damage

Battery-powered ACES receptors, connected together with a stretchable conductive fabric (knit jersey conductive fabric, Adafruit), were encapsulated in stretchable silicone rubber (Ecoflex 00-30, Smooth-On). A stretchable coat of silver ink (PE873, DuPont) and encapsulant (PE73, DuPont) was applied over the rubber via screen printing and grounded to provide the charge return path.

To construct the conventional cross-bar multiplexed sensor array used in the comparison, we fabricated two flexible printed circuit boards (PCBs) to form the row and column traces. A piezoresistive layer (Velostat, 3M) was sandwiched between the PCBs. Each intersection between a row and a column formed a pressure-sensitive element. Traces from the PCBs were connected to an ATmega328 microcontroller (Atmel). Software running on the microcontroller polled each sensor element sequentially to obtain the pressure distribution of the array. Figure 8D illustrates the circuitry used. Because of the simplicity of the readout circuit, some cross-talk will be expected (54). Nevertheless, because the array is relatively small, the cross-talk did not affect the results significantly.

A ring-shaped acrylic object was pressed onto the sensor arrays to deliver the stimulus. We cut the sensor arrays using a pair of scissors to cause damage.

Power consumption measurements

Using a source measurement unit (Keithley 2450), we measured the power consumption of a prototype board of 80 ACES receptors at 3.3-V operating voltage. The receptors were programmed with ACES-SA behavior, and results were an average from a 10-s observation. We did not observe significant differences in power consumption when ACES-FA behavior was implemented.

Fabrication of transducers

Details of material fabrication and characterization are available in texts S1 and S2. Figure S2F was obtained using an automated microindenter (FemtoTools).

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/32/eaax2198/DC1

Text S1. Pressure sensor fabrication and characterization

Text S2. Temperature sensor fabrication and characterization

Fig. S1. Additional characterization of ACES signaling.

Fig. S2. Characterization of transducers.

Fig. S3. Example prototypes of ACES sensor arrays.

Fig. S4. SPICE circuit used for simulation.

Fig. S5. Setup for local curvature and hardness classification.

Movie S1. A typical 5×5 cross-bar sensor array subjected to physical damage.

Movie S2. Robustness of an ACES sensor array to physical damage.

REFERENCES AND NOTES

1. T. Asfour, T. Asfour, J. Schill, H. Peters, C. Klas, J. Bückner, C. Sander, S. Schulz, A. Kargov, T. Werner, V. Bartenbach, ARMAR-4: A 63 DOF torque controlled humanoid robot, in *Proceedings of the 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, (2015) pp. 390–396.
2. S. Micera, Neuroprosthetics: Restoring multi-joint motor control. *Nat. Biomed. Eng.* **1**, 0073 (2017).
3. A. Chortos, Z. Bao, Skin-inspired electronic devices. *Mater. Today* **17**, 321–331 (2014).
4. D. W. Tan, M. A. Schiefer, M. W. Keith, J. R. Anderson, J. Tyler, D. J. Tyler, A neural interface provides long-term stable natural touch perception. *Sci. Transl. Med.* **6**, 257ra138 (2014).
5. B. C. K. Tee, J. Ouyang, Soft Electronically Functional Polymeric Composite Materials for a Flexible and Stretchable Digital Future. *Adv. Mater.* **30**, 1802560 (2018).
6. T. Someya, Y. Kato, T. Sekitani, S. Iba, Y. Noguchi, Y. Murase, H. Kawaguchi, T. Sakurai, Conformable, flexible, large-area networks of pressure and thermal sensors with organic transistor active matrices. *Proc. Natl. Acad. Sci. U.S.A.* **102**, 12321–12325 (2005).
7. D. H. Ho, Q. Sun, S. Y. Kim, J. T. Han, D. H. Kim, J. H. Cho, Stretchable and multimodal all graphene electronic skin. *Adv. Mater.* **28**, 2601–2608 (2016).
8. J. Park, Y. Lee, J. Hong, Y. Lee, M. Ha, Y. Jung, H. Lim, S. Y. Kim, H. Ko, Tactile-Direction-Sensitive and Stretchable Electronic Skins Based on Human-Skin-Inspired Interlocked Microstructures. *ACS Nano* **8**, 12020–12029 (2014).
9. J. Kim, M. Lee, H. J. Shim, R. Ghaffari, H. R. Cho, D. Son, Y. H. Jung, M. Soh, C. Choi, S. Jung, K. Chu, D. Jeon, S.-T. Lee, J. H. Kim, S. H. Choi, T. Hyeon, D.-H. Kim, Stretchable silicon nanoribbon electronics for skin prosthesis. *Nat. Commun.* **5**, 5747 (2014).
10. C. Jaramillo, Y. Noguchi, K. Hata, T. Fukushima, T. Aida, T. Someya, A Rubberlike Stretchable Active Matrix Using Elastic Conductors. *Science* **321**, 1468–1473 (2008).
11. M. Kaltenbrunner, T. Sekitani, J. Reeder, T. Yokota, K. Kuribara, T. Tokuhara, M. Drack, R. Schwödlauer, I. Graz, S. Bauer-Gogonea, S. Bauer, T. Someya, An ultra-lightweight design for imperceptible plastic electronics. *Nature* **499**, 458–463 (2013).
12. R. S. Johansson, I. Birnieks, First spikes in ensembles of human tactile afferents code complex spatial fingertip events. *Nat. Neurosci.* **7**, 170–177 (2004).
13. R. S. Johansson, J. R. Flanagan, Coding and use of tactile signals from the fingertips in object manipulation tasks. *Nat. Rev. Neurosci.* **10**, 345–359 (2009).
14. R. S. Dahiya, P. Mittendorf, M. Valle, G. Cheng, V. J. Lumelsky, Directions toward effective utilization of tactile skin: A review. *IEEE Sensors J.* **13**, 4121–4138 (2013).
15. C. Schürmann, M. Schöpfer, R. Haschke, H. Ritter, A high-speed tactile sensor for slip detection, in *Springer Tracts in Advance Robotics* (Springer, 2012), vol. 76, pp. 403–415.
16. W. W. Lee, S. L. Kukreja, N. V. Thakor, A kilohertz kilotaxel tactile sensor array for investigating spatiotemporal features in neuromorphic touch, in *Proceedings of the 2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)* (2015), pp. 1–4.
17. W. Fukui, F. Kobayashi, F. Kojima, H. Nakamoto, N. Imamura, T. Maeda, H. Shirasawa, High-Speed Tactile Sensing for Array-Type Tactile Sensor and Object Manipulation Based on Tactile Information. *J. Robot.* **2011**, 691769 (2011).
18. O. Oballe-Peinado, J. A. Hidalgo-Lopez, J. A. Sanchez-Duran, J. Castellanos-Ramos, F. Vidal-Verdu, Architecture of a tactile sensor suite for artificial hands based on FPGAs, in *Proceedings of the 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)* (2012), pp. 112–117.
19. C. Bartolozzi, C. Bartolozzi, P. M. Ros, F. Diotalevi, N. Jamali, L. Natale, M. Crepaldi, D. Demarchi, Event-driven encoding of off-the-shelf tactile sensors for compression and latency optimisation for robotic skin, in *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2017), pp. 166–173.

20. F. Bergner, P. Mittendorfer, E. Dean-Leon, G. Cheng, Event-based signaling for reducing required data rates and processing power in a large-scale artificial robotic skin, in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2015), 17 December 2015, pp. 2124–2129.
21. U. B. Rongala, A. Mazzoni, C. M. Oddo, Neuromorphic Artificial Touch for Categorization of Naturalistic Textures. *IEEE Trans. Neural. Netw. Learn. Syst.* **28**, 819–829 (2017).
22. A. Vanarse, A. Osseiran, A. Rassau, A review of current neuromorphic approaches for vision, auditory, and olfactory sensors. *Front. Neurosci.* **10**, 115 (2016).
23. K. A. Boahen, Point-to-point connectivity between neuromorphic chips using address events. *IEEE Trans. Circuits Syst. II Express Briefs* **47**, 416–434 (2000).
24. C. Bader, F. Bergner, G. Cheng, A Robust and Efficient Dynamic Network Protocol for a large-scale artificial robotic skin, in *Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2019), pp. 1600–1605.
25. D. Um, V. Lumelsky, Fault tolerance via component redundancy for a modularized sensitive skin, in *Proceedings of the 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)* (1999), pp. 722–727.
26. V. E. Abaira, D. D. Ginty, The sensory neurons of touch. *Neuron* **79**, 618–639 (2013).
27. M. A. Harvey, H. P. Saal, J. F. Dammann III, S. J. Bensmaia, Multiplexing stimulus information through rate and temporal codes in primate somatosensory cortex. *PLOS Biol.* **11**, e1001558 (2013).
28. A. J. Viterbi, Spread spectrum communications: myths and realities. *IEEE Commun. Mag.* **40**, 34–41 (2002).
29. S. C. B. Mannsfeld, B. C.-K. Tee, R. M. Stoltenberg, C. V. H.-H. Chen, S. Barman, B. V. O. Muir, A. N. Sokolov, C. Reese, Z. Bao, Highly sensitive flexible pressure sensors with microstructured rubber dielectric layers. *Nat. Mater.* **9**, 859–864 (2010).
30. E. A. Tansey, C. D. Johnson, Recent advances in thermoregulation. *Adv. Physiol. Educ.* **39**, 139–148 (2015).
31. H. Yousef, M. Boukallel, K. Althoefer, Tactile sensing for dexterous in-hand manipulation in robotics—A review. *Sensors Actuators A Phys.* **167**, 171–187 (2011).
32. R. Benosman, S.-H. Ieng, C. Clercq, C. Bartolozzi, M. Srinivasan, Asynchronous frameless event-based optical flow. *Neural Netw.* **27**, 32–37 (2012).
33. A. Dépeault, E.-M. Meftah, C. E. Chapman, Tactile Speed Scaling: Contributions of Time and Space. *J. Neurophysiol.* **99**, 1422–1434 (2008).
34. J. A. Fishel, G. E. Loeb, Bayesian exploration for intelligent identification of textures. *Front. Neurobot.* **6**, 4 (2012).
35. L. Qin, Z. Yi, Y. Zhang, Enhanced surface roughness discrimination with optimized features from bio-inspired tactile sensor. *Sensors Actuators A Phys.* **264**, 133–140 (2017).
36. S. S. Baishya, B. Bäuml, Robust material classification with a tactile skin using deep learning. *IEEE International Conference on Intelligent Robots and Systems*, pp. 8–15 (2016).
37. W. W. Lee, S. L. Kukreja, N. V. Thakor, Discrimination of dynamic tactile contact by temporally precise event sensing in spiking neuromorphic networks. *Front. Neurosci.* **11**, 5 (2017).
38. J. A. Fishel, G. E. Loeb, Sensing tactile microvibrations with the BioTac—Comparison with human sensitivity, in *Proceedings of the 2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)* (2012), pp. 1122–1127.
39. P. M. Ros, M. Crepaldi, C. Bartolozzi, D. Demarchi, Asynchronous DC-free serial protocol for event-based AER systems, in *Proceedings of the 2015 IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, pp. 248–251.
40. D. B. Fasnacht, A. M. Whatley, G. Indiveri, A serial communication infrastructure for multi-chip address event systems, in *Proceedings of the 2008 IEEE International Symposium on Circuits and Systems* (2008), 648–651.
41. G. Cannata et al., Modular Skin for Humanoid Robot Systems. *CogSys 2010 Conf. Proc.* **231500**, 231500 (2010).
42. P. Mittendorfer, G. Cheng, Humanoid multimodal tactile-sensing modules. *IEEE Trans. Robot.* **27**, 401–410 (2011).
43. A. Schmitz, P. Maiolino, M. Maggiali, L. Natale, G. Cannata, G. Metta, Methods and technologies for the implementation of large-scale robot tactile sensors. *IEEE Trans. Robot.* **27**, 389–400 (2011).
44. H. P. Saal, B. P. Delhay, B. C. Rayhaun, S. J. Bensmaia, Simulating tactile signals from the whole hand with millisecond precision. *Proc. Natl. Acad. Sci. U.S.A.* **114**, E5693–E5702 (2017).
45. E. D'Anna, G. Valle, A. Mazzoni, I. Strauss, F. Iberite, J. Patton, F. M. Petrini, S. Raspopovic, G. Granata, R. D. Iorio, M. Controzzi, C. Cipriani, T. Stieglitz, P. M. Rossini, S. Micera, A closed-loop hand prosthesis with simultaneous intraneural tactile and position feedback. *Sci. Robot.* **4**, eaau8892 (2019).
46. R. Chen, A. Canales, P. Anikeeva, Neural recording and modulation technologies. *Nat. Rev. Mater.* **2**, 1–16 (2017).
47. D. Khodagholy, J. N. Gelinas, T. Thesen, W. Doyle, O. Devinsky, G. G. Malliaras, G. Buzsáki, NeuroGrid: Recording action potentials from the surface of the brain. *Nat. Neurosci.* **18**, 310–315 (2015).
48. C. Larson, B. Peele, S. Li, S. Robinson, M. Totaro, L. Beccai, B. Mazzolai, R. Shepherd, Highly stretchable electroluminescent skin for optical signaling and tactile sensing. *Science* **351**, 1071–1074 (2016).
49. F. R. K. Chung, J. A. Salehi, V. K. Wei, Optical orthogonal codes: design, analysis and applications. *IEEE Trans. Inf. Theory* **35**, 595–604 (1989).
50. W. T. Navaraj, C. García Núñez, D. Shakhiveli, V. Vinciguerra, F. Labeau, D. H. Gregory, R. Dahiya, Nanowire FET based neural element for robotic tactile sensing skin. *Front. Neurosci.* **11**, 501 (2017).
51. R. A. M. Chandra, S. Chauhan, thesis, Uttar Pradesh Technical University, Lucknow, India (2015).
52. M. C. W. Van Rossum, A novel spike distance. *Neural Comput.* **13**, 751–763 (2001).
53. T. Cover, P. Hart, Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* **13**, 21–27 (1967).
54. H. Liu, Y.-F. Zhang, Y.-W. Liu, M.-H. Jin, Measurement errors in the scanning of resistive sensor arrays. *Sensors Actuators A Phys.* **163**, 198–204 (2010).

Acknowledgments: We thank T. Sun for microfabrication assistance, H. Godaba for help in micromechanical testing, and Z. Goh for help with some figure illustrations. **Funding:** B.C.K.T. acknowledges the support from the Singapore National Research Fellowship (NRFF2017-08), NUS Start-up Grant, and the Singapore National Robotics Programme (NRP). B.C.K.T. and J.S.H. acknowledges grant support from Institute for Health Innovation and Technology and the N.I Institute for Health, NUS. **Author contributions:** W.W.L., J.S.H., K.A.N., and B.C.K.T. conceived the idea. B.C.K.T. directed the research activities. W.W.L. and B.C.K.T. designed and led the experiments (with inputs from Y.J.T., H.Y., S.L., M.H., H.H.S., and B.X.). W.W.L., Y.J.T., H.H.S., and B.C.K.T. contributed to data analysis and interpretation. W.W.L. and B.C.K.T. wrote the paper, and all authors provided feedback. **Competing interests:** The National University of Singapore has filed a patent application on this work. **Data and materials availability:** All data needed to evaluate the conclusions in this paper are present in the paper or the Supplementary Materials. Data and software code can be made available by materials transfer agreement upon reasonable request.

Submitted 8 March 2019

Accepted 21 June 2019

Published 17 July 2019

10.1126/scirobotics.aax2198

Citation: W. W. Lee, Y. J. Tan, H. Yao, S. Li, H. H. See, M. Hon, K. A. Ng, B. Xiong, J. S. Ho, B. C. K. Tee, A neuro-inspired artificial peripheral nervous system for scalable electronic skins. *Sci. Robot.* **4**, eaax2198 (2019).

PROSTHETICS

Biomimetic sensory feedback through peripheral nerve stimulation improves dexterous use of a bionic hand

J. A. George^{1*†}, D. T. Kluger^{1†}, T. S. Davis², S. M. Wendelken¹, E. V. Okorokova³, Q. He³, C. C. Duncan⁴, D. T. Hutchinson⁵, Z. C. Thumser⁶, D. T. Beckler⁶, P. D. Marasco⁶, S. J. Bensmaia³, G. A. Clark^{1*}

Copyright © 2019
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

We describe use of a bidirectional neuromyoelectric prosthetic hand that conveys biomimetic sensory feedback. Electromyographic recordings from residual arm muscles were decoded to provide independent and proportional control of a six-DOF prosthetic hand and wrist—the DEKA LUKE arm. Activation of contact sensors on the prosthesis resulted in intraneural microstimulation of residual sensory nerve fibers through chronically implanted Utah Slanted Electrode Arrays, thereby evoking tactile percepts on the phantom hand. With sensory feedback enabled, the participant exhibited greater precision in grip force and was better able to handle fragile objects. With active exploration, the participant was also able to distinguish between small and large objects and between soft and hard ones. When the sensory feedback was biomimetic—designed to mimic natural sensory signals—the participant was able to identify the objects significantly faster than with the use of traditional encoding algorithms that depended on only the present stimulus intensity. Thus, artificial touch can be sculpted by patterning the sensory feedback, and biologically inspired patterns elicit more interpretable and useful percepts.

INTRODUCTION

State-of-the-art upper-limb prostheses have become capable of mimicking many of the movements and grip patterns of endogenous human hands (1–3). Although these devices have the capabilities to replace much of the motor function lost after hand amputation, the methods for controlling and receiving feedback from these prosthetic limbs are still primitive (4, 5). The advent of neuromuscular implant systems capable of recording efferent motor activity and stimulating afferent sensory nerve fibers improves the transfer of sensorimotor information to and from a user's peripheral nervous system, paving the way for more dexterous bionic hands (6–9).

Conveying sensory feedback through an electrical interface with the peripheral nervous system has been shown to confer functional benefits (9–16). However, demonstrations of these improvements are limited, and the sensory encoding algorithms themselves are often unsophisticated. The human hand is innervated by several types of tactile nerve fibers that each respond to different aspects of skin deformations. Manual interactions generally activate all of the fiber types, and tactile percepts are shaped by complex spatiotemporal patterns of activation across the different afferent populations (17, 18). One of the notable features of the aggregate afferent activity is the massive phasic bursts during the onset and offset of contact and the far weaker response during maintained contact (19–22). Most extant sensory encoding mechanisms track sensor output (e.g., the absolute pressure, force, or torque from a prosthetic device) by modulating stimulation intensity and thus disregard this important and salient aspect of natural sensory feedback (9, 10, 12, 23–29). To the extent that artificially induced sensory signals mimic natural ones, they are likely to

elicit more naturalistic percepts and confer greater dexterity to the user (15, 30).

In the present study, we first demonstrate that closed-loop sensory feedback improved performance on dexterous tasks and enabled sensory discrimination during active manipulation of objects. We then show that artificial sensory experiences were enriched when the stimulation regimes were designed to mimic the natural patterns of neuronal activation that are evoked during manual interactions with a native hand. These results constitute an important step toward the development of dexterous bionic hands and have broad implications for neural interfaces and prosthetic devices.

RESULTS

We implanted one Utah Slanted Electrode Array (USEA) in the median nerve and another in the ulnar nerve, plus eight electromyographic recording leads (iEMGs) in the forearm muscles of an individual with a transradial amputation halfway between the wrist and elbow. The participant used this neuromyoelectric interface to control and sense through a state-of-the-art dexterous sensorized prosthetic hand and wrist (LUKE arm, DEKA; Fig. 1). Control signals were obtained using the filtered iEMG recordings as input to a modified Kalman filter (29, 31). The participant was able to control all six DOFs of the prosthesis independently, proportionally, and simultaneously in real time, achieving performance comparable with those of clinically available prosthetics in the modified Box and Blocks test (fig. S1) (32)—a standard test of manual dexterity—and efficiency comparable with that of able-bodied participants in a novel foraging task (fig. S2) (33). Recordings of muscle activation remained reliable over the entire duration of the study (14 months). Using muscle recordings rather than neural ones as control signals eliminates the problem of stimulation artifacts and allows for uncompromised sensory feedback.

Electrical stimulation of the residual nerves evokes sensations on the phantom hand

Electrical stimulation of the residual nerves through the chronically implanted USEAs evoked localized sensations that were experienced

¹Department of Biomedical Engineering, University of Utah, Salt Lake City, UT 84112, USA. ²Department of Neurosurgery, University of Utah, Salt Lake City, UT 84112, USA. ³Department of Organismal Biology and Anatomy, University of Chicago, Chicago, IL 60637, USA. ⁴Department of Physical Medicine and Rehabilitation, University of Utah, Salt Lake City, UT 84112, USA. ⁵Department of Orthopaedics, University of Utah, Salt Lake City, UT 84112, USA. ⁶Department of Biomedical Engineering, Lerner Research Institute, Cleveland Clinic, Cleveland, OH 44195, USA. *Corresponding author. Email: jacob.george@utah.edu (J.A.G.); greg.clark@utah.edu (G.A.C.)

†These authors contributed equally to this work.

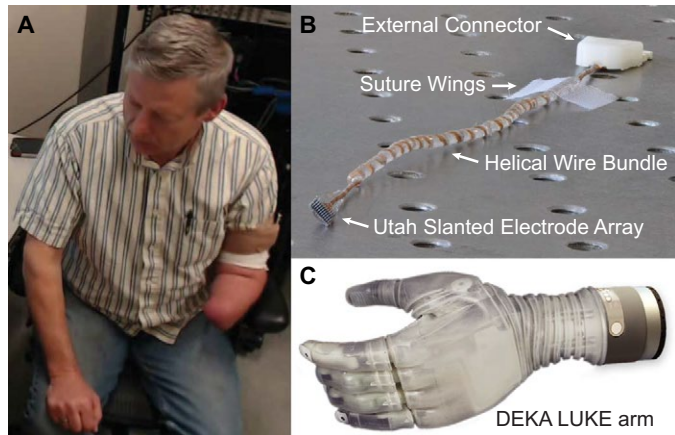


Fig. 1. Participant and sensorized bionic hand. A transradial amputee (A) had two total USEAs (B) implanted, one each, into the residual median and ulnar nerves of the arm. Activation of contact sensors on the DEKA LUKE arm (C) triggered stimulation of individual USEA electrodes or groups of USEA electrodes so that the amputee felt a sensation on his phantom hand at the corresponding location. For example, when contact was made with the index fingertip sensor, current was delivered through USEA electrodes with projection fields on the phantom index fingertip. Thus, when the prosthetic index fingertip made contact with an object, the participant experienced a sensation on the index fingertip.

on the phantom hand. The participant reported up to 119 sensory percepts distributed over the hand and varying in their quality (Fig. 2 and fig. S3). As might be expected given the known patterns of innervation of the skin, a preponderance of percepts originated in the fingers and particularly the fingertips. The quality of the percepts also varied; some were described as “vibration” (36%), “pressure” (29%), or “tapping” (3%), which were likely associated with activation of cutaneous tactile nerve fibers; others were described as pain (16%), presumably reflecting activation of nociceptive fibers; and a few were described as “tightening” (12%) and joint movement (3%), presumably reflecting activation of proprioceptive nerve fibers such as muscle afferents. Activation of contact sensors on the prosthetic hand triggered stimulation of individual USEA electrodes or groups of USEA electrodes with congruent receptive fields. For example, when contact was made with the index fingertip sensor, a current was delivered through USEA electrodes with projection fields on the index fingertip of the phantom; that way, when the prosthetic index fingertip made contact with an object, the participant experienced a sensation on the index fingertip.

Sensory feedback improves grasping performance

The grip force required to grasp an object depends on its mass and on the coefficient of friction between skin and object: Heavy and slippery objects are gripped with more force than are light, high-friction ones (34). With our native hands, we are exquisitely proficient at exerting just enough pressure on an object to grasp it, an ability for which we rely on the sense of touch (34).

Some tests of manual dexterity do not benefit from tactile feedback. For example, performance on the modified Box and Blocks test is only slightly improved with touch because visual feedback is available and no penalty is incurred for exerting too much force on an object. However, other tests of manual dexterity are highly dependent on tactile feedback. In one such test, a participant moves an object from one location to another, as in the modified Box and Blocks test (fig. S1). However, the object is “fragile” and “breaks” if squeezed too hard (fig. S4) (35, 36).

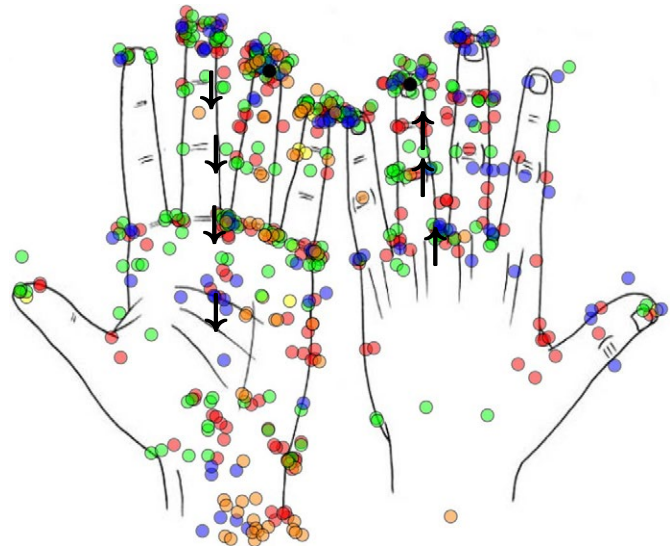


Fig. 2. Centroids of the projected fields for cutaneous percepts (circles) and location of proprioceptive percepts (black arrows) evoked by stimulation through individual USEA electrodes in the residual median or ulnar nerves. A total of 119 sensory percepts were evoked (72% from median nerve) 2 weeks after the implantation surgery. The quality of the evoked percepts varied across electrodes: 37% vibration (red), 29% pressure (green), 16% pain (blue), 12% tightening (orange), 3% movement (arrows), 3% tapping (yellow), and 1% buzzing (black). A map of the complete projected fields can be found in fig. S3.

The participant moved the object without breaking it significantly more often with sensory feedback than without (32 of 40 times versus 22 of 40 times; Pearson’s χ^2 test, $P = 0.017$; Fig. 3) and did so more rapidly (9.13 ± 0.44 s versus 11.14 ± 0.49 s per trial; t test, $P < 0.001$; Fig. 3).

Performance of activities of daily living (ADLs) often involves dividing attention between multiple simultaneous subtasks—e.g., holding a jar while twisting off its lid—so sensory feedback that is attentionally demanding is inappropriate (37). To test whether the sensory feedback conveyed through nerve stimulation was resistant to divided attention, we had participants perform the fragile object test while counting backward. We found that the feedback-induced boost in performance was maintained with divided attention but only the effect on duration remained statistically significant under this condition (5.91 ± 0.20 s versus 7.68 ± 0.42 s; t test, $P < 0.001$; Fig. 3).

Another way to assess the impact of sensory feedback on object interactions is to characterize the degree to which we exert a consistent amount of force on an object upon repeated grasping (38). To test this capability, we had the participant repeatedly grasp a load cell with the prosthetic hand. Sensory feedback was provided on some experimental blocks but not others. The participant’s grip performance was more precise with sensory feedback than without, as evidenced by less variable grip force on six of eight objects (Fig. 4). Furthermore, sensory feedback significantly reduced the coefficient of variation (ratio of grip precision to grip force) across all objects [Fig. 4 and fig. S5 show the standardized Grasping Relative Index of Performance (GRIP) for this test] (38).

Sensory feedback enables haptic perception

When we manipulate objects, we acquire information about their shape, size, and texture through sensory signals from our hands (39, 40). Haptic perception relies on an interplay between exploratory movements

and the sensory consequences of those movements (41). To assess the degree to which the prosthesis could convey object information, we developed a closed-loop sensorimotor task in which the participant actively manipulated one of two objects with the prosthetic index finger (fig. S6). Stimulation was at a fixed frequency and amplitude and was delivered as long as contact with the object was maintained. On each

trial, one of two objects was presented—a golf ball or a (larger) lacrosse ball—and the participant's task was to report the size of the object (small versus large). Alternative sensory cues were reduced or eliminated by having the prosthesis mounted externally on a table (rather than being worn by the participant) and by having the participant wear an eye mask and headphones. The participant was able to perform this task almost perfectly with the sensory feedback, correctly reporting the size on 31 of 32 object presentations (binomial test, $P < 0.0001$).

To further assess haptic perception, we developed a closed-loop sensorimotor task in which the participant actively manipulated one of two objects—a soft foam block or a hard plastic block—and discriminated the compliance (soft versus hard; fig. S7). In this experiment, the amplitude of electrical stimulation increased linearly with the output of the sensor. The participant was able to distinguish between the two objects significantly better than chance (60 of 80 trials; binomial test, $P < 0.0001$) and did so after squeezing the object several times (Fig. 5), highlighting the interplay between motor behavior and sensory feedback.

Biomimetic peripheral nerve stimulation improves object discrimination

In the studies described above, sensory feedback provided either a contact signal or a signal proportional to the contact force. Although both regimes of stimulation led to significant improvements in closed-loop sensorimotor tasks, neither regime is liable to produce naturalistic patterns of activation in the nerve. Interactions with objects are characterized by a strong burst of activation at the onset and offset of contact and much weaker activation during maintained contact (42). This initial onset conveys important information about the shape of the object (40). The aggregate response of tactile nerve fibers is determined not only by the degree to which the skin is indented but also by the rate at which the skin is indented, and the latter component dwarfs the former one.

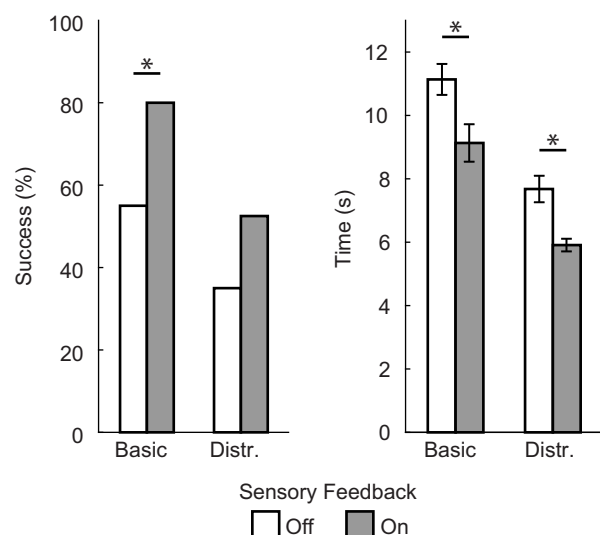


Fig. 3. Sensory feedback improves object manipulation. The participant's task was to move a fragile object that breaks if the grip force is too strong. With sensory feedback, the participant moved the object more often without breaking it and did so more rapidly (basic). With divided attention (distr.), the feedback-induced boost in performance was maintained, but only the effect on duration remained statistically significant. * $P < 0.05$, $n = 80$ for both basic and distr. cases. Data show means \pm SEM.

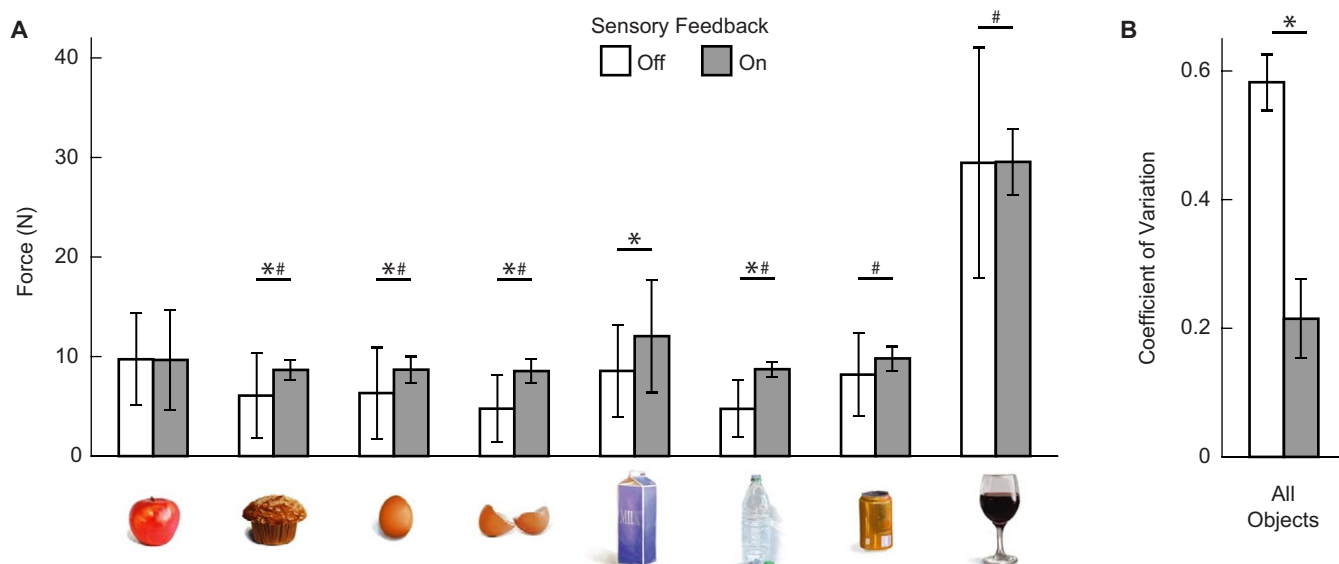


Fig. 4. Sensory feedback improves grip precision. (A) Forces (means \pm SD) generated by the participant when grasping a load cell while viewing one of eight different virtual objects. Sensory feedback improved grip precision, as evidenced by less variable grip force on six of eight objects. Without sensory feedback, the participant erred on the side of caution and underestimated desired grip force for fragile objects (bread, eggs, and open water bottle). (B) Coefficient of variation (means \pm SEM) of grip force across all eight objects. Sensory feedback significantly reduced the coefficient of variation (i.e., the ratio of grip precision to grip force). Asterisk (*) indicates different means ($P < 0.05$), and sharp (#) indicates different SDs ($P < 0.05$); $n = 40$ for each object.

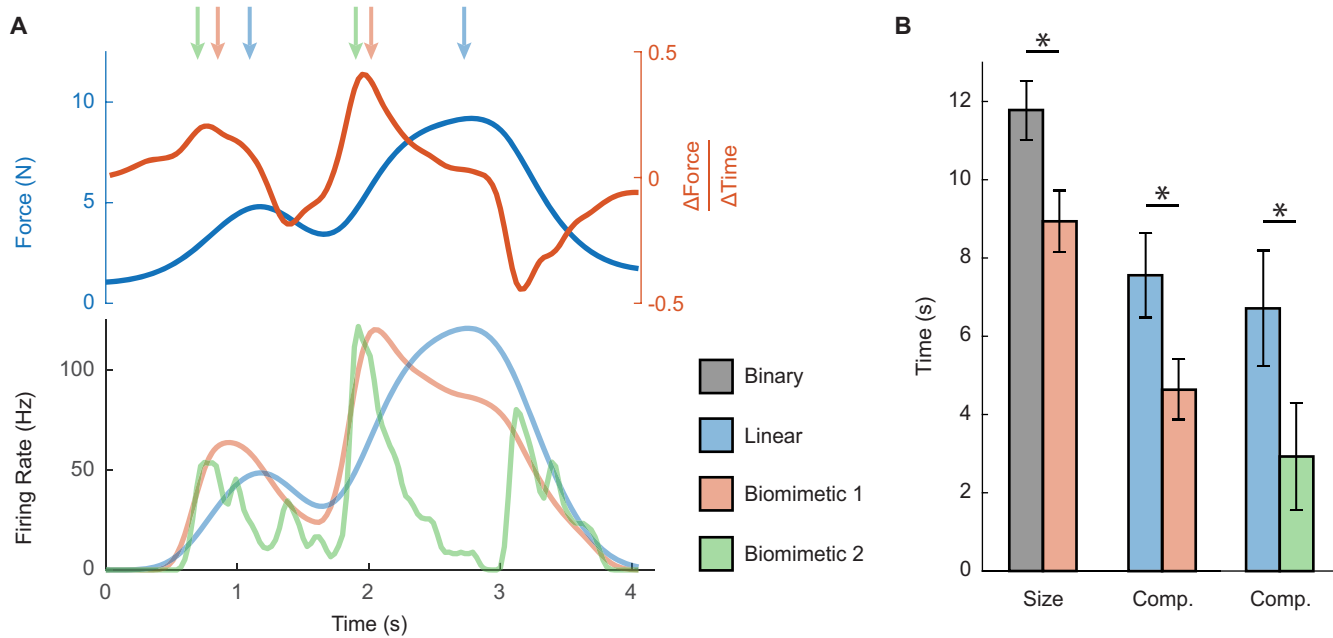


Fig. 5. Biomimetic sensory feedback improves performance on object discrimination tasks. (A) Example force (top; blue) and change in force (top; red) when the participant actively manipulated a soft foam block. Note the repetitive waxes and wanes in force (e.g., at ~2 s), associated with the participant's active exploration of the object. Traditional linear encoding tracks force only (bottom; light blue), whereas the first-order biomimetic encoding incorporates the first derivative of force (bottom; light red) and second-order biomimetic mimics the aggregate responses of tactile nerve fibers (bottom; light green). Linear algorithms were scaled (doubled) such that peak stimulation amplitude and frequency were matched to the biomimetic algorithms; arrows highlight the time when peak stimulation occurs for the different algorithms. (B) Biomimetic sensory feedback improved response time relative to its nonbiomimetic counterpart in size and compliance (comp.) discrimination tasks. Performance across experiments varied because of changes in stimulation parameters, but biomimetic stimulation consistently outperformed nonbiomimetic stimulation. * $P < 0.05$, $n = 32$ for binary versus biomimetic 1, $n = 48$ for linear versus biomimetic 1, and $n = 32$ for binary versus biomimetic 2. Data show means \pm SEM.

We therefore sought to implement a sensory feedback algorithm that incorporates this temporal property of natural tactile signals. As a first-order approximation, we developed a sensory feedback algorithm in which the intensity of stimulation was proportional not only to the contact force but also to its rate of change. This first-order biomimetic algorithm leads to stronger stimulation at the contact onset, when the rate of change is highest to mimic the phasic bursts observed in natural nerve activation during contact transients. To test this simple biomimetic algorithm, we had the participant discriminate the size and compliance of objects, and we compared his performance with that using the standard sensory encoding algorithms (contact tracking and force tracking). We found that the participant was able to perform these tasks significantly faster with the biomimetic feedback than with its nonbiomimetic counterparts. Biomimetic sensory feedback improved response time by 24% for size discrimination (11.78 ± 0.75 s versus 8.94 ± 0.79 s; t test, $P < 0.05$; Fig. 5) and by 44% for compliance discrimination (14.16 ± 1.05 s versus 7.91 ± 0.81 s; t test, $P < 0.005$; fig. S7).

In the above implementation of biomimetic feedback, the peak intensity of stimulation was higher than with nonbiomimetic feedback because the overall charge was approximately matched. One possibility, then, was that the improvement in performance with biomimetic feedback was a consequence of the higher peak stimulation intensity. Although a higher peak firing rate might itself be more biomimetic, improved discrimination would not necessarily depend on differences in temporal firing patterns between the biomimetic and nonbiomimetic encoding schemes. To distinguish between these possibilities, we implemented a version of the biomimetic algorithm such that the peak

stimulation intensity (pulse amplitude and frequency) was matched to that of the nonbiomimetic algorithms. Even with matched peak intensity, the biomimetic feedback led to a 46% improvement in performance (7.56 ± 1.08 s versus 4.64 ± 0.77 s; t test, $P < 0.005$; Fig. 5). Another potential confound is that biomimetic algorithm might peak faster than the nonbiomimetic ones, leading to faster performance. However, the improvement in response time was on a longer time scale than the shift in peak stimulation, so this effect was not a trivial consequence of the timing of stimulation. Rather, it reflects an improvement in the intuitiveness and informativeness of the artificial sensory signals, which capture some of the essential temporal characteristics of natural tactile signals.

The above results suggest that dynamics of the response evoked through electrical stimulation—if it mimics a natural response—can lead to more interpretable and useful sensory feedback. However, the above biomimetic algorithm captured some aspects of the natural tactile feedback—namely, the increase in sensitivity to contact transients—but not others, borne out of the idiosyncratic properties of the different classes of tactile nerve fibers and their respective innervation densities. In light of this, we tested another sensory encoding algorithm that sought to more faithfully mimic natural nerve activations. Briefly, this algorithm is designed to reflect the measured sensitivity of populations of nerve fibers to skin indentation and its two derivatives (rate and acceleration) (19). With this second-order biomimetic feedback, the participant identified object compliance 56% faster than with the traditional linear feedback (6.71 ± 1.47 s versus 2.93 ± 1.37 s; t test, $P < 0.05$; Fig. 5). These results further demonstrate that biomimicry improves the intuitiveness of the artificial sensory feedback.

Participant successfully performed a variety of ADLs

An important concern in laboratory demonstrations of neuroprosthetic control is whether tasks that are used to assess the performance of the prosthesis are ecologically valid. With regard to the present study, will improvements in performance with sensory feedback on laboratory tasks translate to improved performance on ADLs? We evaluated this by having the participant complete several ADLs over 3 days of testing. With just the prosthesis alone or in conjunction with his intact hand, he performed basic ADLs (feeding and dressing) (43), instrumental ADLs (housework, meal preparation, and technology use) (44), and ADLs that he had found challenging without the prosthesis (loading a pillow into a pillowcase, hammering, donning, and doffing a ring; Fig. 6). Improvements are difficult to quantify with ADLs, but the participant noted that sensory feedback was particularly useful when manipulating fragile objects (e.g., eggs and grapes) and spontaneously reported that he enjoyed the sensation of “feeling” objects in his hand.

DISCUSSION

In the present study, we demonstrate that artificial sensory feedback improves fine motor control and confers to the user the ability to sense object properties through a bionic hand. Furthermore, these artificial sensory experiences are enriched when the sensory feedback is designed to mimic the nervous system’s natural language. By capturing some of the essential characteristics of natural tactile signals, biomimetic stimulation improves the intuitiveness and informativeness of the sensory feedback, as evidenced by swifter object discrimination capabilities.

The present results build on previous work, showing that sensory feedback leads to improved grip and handling of fragile objects (10). We extend these previous findings by showing that grasp force is

achieved faster and more accurately and that fragile objects are transferred faster with sensory feedback than without. These improvements are augmented when the sensory feedback is biomimetic. Although previous studies have demonstrated that object properties can be sensed through a prosthetic hand (12), we extend these previous findings to a different sensorimotor task—compliance discrimination—and directly demonstrate the improvement of biomimetic feedback relative to its nonbiomimetic counterpart. In this respect, our work is consistent with a recent study, showing that biomimetic stimulation leads to more naturalistic percepts, leads to greater embodiment, and improves performance on object manipulation tasks (30). In the present study, we extend these previous findings to a new technology and a new task, an important replication of the benefits of sensory feedback and biomimicry, given that the relevant studies thus far have involved a single participant (12, 13, 15, 30, 45).

Amputees have expressed a desire for sensory feedback to reduce their dependence on visual feedback (37). The ability to feel grip force while grasping and holding objects is the most important aspect of sensory feedback for amputees (46). The sensory feedback provided here allowed the participant to perform object discrimination tasks without visual or auditory feedback and enabled the participant to exert grip forces more precisely.

Ideally, sensory peripheral nerve interfaces and encoding algorithms would activate each afferent nerve fiber selectively and independently, so as to replicate the spatiotemporal pattern of neural discharges that would be transmitted from an intact hand. The ability of different USEA electrodes to activate a large number of different percepts (Fig. 2) increases the ability to provide more biomimetic sensory input. The present experiments used relatively simple receptive fields and sensorimotor tasks to study the importance of temporal aspects of sensory encoding at a population level in isolation and hence did not fully explore these capabilities. However, such capabilities may prove increasingly useful with richer sensorimotor tasks and with the advent of prosthetic hands with greater numbers and varieties of sensors.

In addition to sensorimotor functional improvements, closed-loop sensorized prostheses often bring psychological benefits (9, 10, 29, 47–49). The same participant in this study reported decreased phantom pain and increased embodiment of the prosthesis as a result of the sensory feedback (29). After the study, the participant highlighted the emotional impact of artificial touch when he used the bionic hand to shake hands with his wife and felt her touch through it for the first time. The functional and emotional benefits of dexterous motor control and biomimetic sensory feedback are likely to be further enhanced with long-term use, and efforts are underway to develop a portable take-home system (50).

MATERIALS AND METHODS

Study design

We considered the participant for this chronic implant study due to the transradial level of his amputation, his willingness to volunteer, and overall good health. Termination of the study and explantation of the electrodes were voluntary or would occur if the implants were causing a health risk as indicated by a qualified physician or at 14 months after the implant date. Previous studies from this group (6, 51, 52) were limited in duration (less than 5 weeks) for safety considerations; because no health risks emerged from these previous studies, the University of Utah Institutional Review Board and the participant agreed to a 14-month duration for this study.



Fig. 6. Sensory feedback supports ADLs. The participant performed several one- and two-handed ADLs while using the sensorized prosthesis, including moving an egg (A), picking grapes (B), texting on his phone (C), and shaking hands with his wife (D).

The experiments performed in this study were completed in 2- to 3-hour sessions, one to three times a week, across the 14-month duration of the study. The number of replicates per experiment was matched to that of previous studies involving fragile object manipulation (10), object discrimination (12), and the Grasping Relative Index of Performance (38). Data were considered outliers if they fell outside three SDs from the mean (38).

Human participant and implanted devices

A male left transradial amputee, whose amputation occurred 13 years before the onset of the study, underwent surgeries and performed experiments with informed consent and under protocols approved by the University of Utah Institutional Review Board and the Department of the Navy Human Resources Protection Program. Under general anesthesia, two 100-electrode USEAs were implanted in the median, and ulnar nerves of the residual limb, proximal to the elbow, and eight iEMGs, with four electrical contacts each, were implanted in the upper forearm with attempted targeting of each lead to different lower-arm extensor or flexor muscles. Additional information and figures regarding the devices and implantation procedure can be found in the Supplementary Materials and (29), which reports on the same participant in this study.

Decoding motor intent

Motor intent was decoded from residual forearm muscles recorded at 1 kHz, while the participant actively mimicked prosthetic hand movements, as previously reported in (6, 29, 31). Every 33 ms, the mean absolute value (MAV) over a 300-ms window was calculated for the 32 iEMG electrodes and the 496 possible differential pairs. A total of 528 features were generated (MAV for 32 single-ended and 496 differential pairs). To save computational time and reduce potential overfitting, the 528 features were then down-selected to the best 48 features using a Gram-Schmidt channel-section algorithm (53). These 48 features served as an input to a modified Kalman filter (MKF)-based decoder that uses customizable, non-unity thresholds and gains (29, 54). The output of the MKF was used to directly control the position or velocity of the six DOFs of the prosthesis. The ability to proportionally control position or velocity was toggled on a DOF-by-DOF basis. More information regarding the prosthetic control algorithm can be found in (54) and the Supplementary Materials.

Mapping of USEA-evoked percepts

Electrical stimulation was delivered via USEAs using the Ripple Neuro LLC Grapevine System with Micro2+Stim front ends. All stimulation was delivered as biphasic, cathodic-first pulses, with 200- to 320- μ s phase durations and a 100- μ s interphase duration. The stimulation frequency varied between 10 and 500 Hz, and stimulation amplitudes were in the range of 1 to 100 μ A.

USEA stimulation threshold maps were collected roughly every 4 to 8 weeks, during which each electrode of the USEAs was stimulated in isolation at increasing amplitudes. Electrodes that evoked a sensory percept at less than 100 μ A were noted, and the location, quality, and intensity of each percept were documented as well as the threshold amplitude at which the percept was evoked. For these mappings, stimulation was delivered in a pulsed fashion with a 500-ms train of 100-Hz stimulation being delivered every second. Additional descriptions for electrode mapping (6) and the stimulation parameters we used (29) exist elsewhere. Sensory percepts were stable over the course of these experiments and persisted 14 months after the implant

(fig. S8). More information regarding the stability of the USEA-evoked percepts is available in the Supplementary Materials.

Encoding sensory feedback

Stimulation through a single USEA electrode typically evoked a single percept with a distinct receptive field (e.g., sensations were isolated to only the index finger, or only the middle finger, but not both fingers). Occasionally, stimulation of a single USEA electrode would evoke multiple percepts in distinct receptive fields (e.g., stimulation of a single USEA electrode evoked sensations on both the index and middle fingers); these electrodes with multiple distinct percepts were not used for real-time sensory feedback.

The distinctly evoked percepts were then assigned to a single contact (cutaneous) or motor (proprioceptive) sensor on the prosthesis with a corresponding receptive field. For example, if stimulation through USEA electrode X evoked a pressure-like percept on the middle finger and if separately stimulating through USEA electrode Y also evoked a percept on the middle finger, then both electrodes X and Y would be assigned to the middle finger contact sensor on the prosthesis. We stimulated between 1 and 12 USEA electrodes that had overlapping receptive fields with a given sensor on the prosthesis (Table 1). Because of the time-intensive nature of assigning all electrodes, a subset of sensors on the prosthesis were used for each task; the specific sensors used for a particular task are detailed in the corresponding section for that task. Activation of sensors resulted in biphasic, charge-balanced stimulation (200- or 320- μ s phase durations, cathodic first, with a 100- μ s interphase duration). We encoded percept intensity by modulating the frequency or current amplitude of stimulation with either linear or biomimetic encoding algorithms (see next section). For all encoding algorithms, the intensity of the sensation increased with increasing stimulation amplitude and frequency, but there were no reported changes in perceptive field location or sensory modality.

Stimulation parameters were adjusted at the start of each experimental session to maximize the naturalism and perceived intensity range of the stimulation. To the extent possible, the participant's sensory experience (e.g., perceived intensity range, perceptive field, etc.) was kept consistent across days. Stimulation typically produced natural-feeling pressure sensations on the palmar aspects of the hand. The exact parameters (electrodes, encoding algorithm, amplitude, frequency, and pulse duration) used for each task are summarized in Table 1.

Sensory encoding algorithms

For binary sensory encoding, the stimulation was fixed at the specified amplitude (100 μ A, 320 μ S) and frequency (100 Hz) as long as any contact was made. For traditional, linear sensory encoding, the stimulation frequency and amplitude increased solely on the basis of the absolute sensor value. For biomimetic 1 sensory encoding, the stimulation frequency and amplitude increased on the basis of the absolute sensor value and on positive rate of change of the sensor; stimulation tracked the current sensor value plus 10 times any positive finite difference between the current and previous sensor value. For scaled, traditional, linear sensory feedback, the stimulation frequency and amplitude were multiplied by a constant factor (≈ 2) such that the range was comparable with that of the biomimetic stimulation (Fig. 5). Stimulation amplitude and frequency increased together over their respective ranges (see Tables 1 and 2).

The biomimetic 2 sensory encoding algorithm was developed from recordings of nonhuman primate cutaneous afferents in response to physical contact with the fingertip (19). This computationally inexpensive

Table 1. Stimulation parameters used for each task.

Task	Sensory encoding algorithm(s)	USEA electrodes	Amplitude (μ A)	Frequency (Hz)	Duration (μ s)
Fragile object (first set)	Traditional linear	2, 5, 6, 9, 10, 12, 15, 16, 20, 25	80–100	10–100	200
Fragile object (second set)	Traditional linear	2, 5, 6, 9, 10, 12, 15, 16, 20, 23, 25	70–100	10–100	200
GRIP	Biomimetic 1	5, 6, 9, 10, 12, 15, 16, 20, 23	80–95	10–200	320
Size discrimination	Biomimetic 1	2, 5, 6, 9, 10, 12, 15, 16, 20, 23	80–95	10–200	200
Size discrimination	Binary	5, 6, 9, 10, 12, 15, 16, 20, 25, 26	100	100	320
Compliance discrimination (first set)	Biomimetic 1 versus traditional linear	2, 5, 6, 9, 10, 12, 15, 16, 20, 23	90–100	10–200	200
Compliance discrimination (second set)	Biomimetic 1 versus traditional linear	2, 5, 6, 9, 10, 12, 15, 16, 20, 23	80–95	10–200	320
Compliance discrimination (first set)	Biomimetic 1 versus scaled traditional linear	2, 5, 6, 9, 10, 12, 15, 16, 20, 23	80–95	10–200	320
Compliance discrimination (second set)	Biomimetic 1 versus scaled traditional linear	5, 6, 9, 10, 12, 15, 16, 20, 23, 25, 26	80–100	10–200	320
Compliance discrimination	Biomimetic 2 versus scaled traditional linear	5, 6, 9, 10, 12, 15, 16, 20, 23, 25, 26	70	10–400	320
ADL (first set)	Traditional linear	23, 26, 33, 41, 42, 47, 63	70–100	10–100	200
ADL (second set)	Traditional linear	23, 26, 27, 33, 34	60–100	100	200
ADL (third set)	Traditional linear	9, 10, 20	80–100	10–100	200

model describes the instantaneous firing rate (i.e., stimulation frequency) of the afferent population using the contact stimulus position, velocity, and acceleration. Similar to other biomimetic algorithms (30), the biomimetic 2 sensory encoding algorithm leverages TouchSim (55) to simulate the responses of all tactile fibers to any spatiotemporal deformation of the skin of the hand. This model—dubbed TouchMime—provides a more computationally efficient approach to the aggregate response of the nerve to time-varying pressure applied to the fingertip, allowing for high-accuracy biomimetic sensory encoding in real time. In addition, the parameters of the model were tuned for the sampling rate of the DEKA LUKE arm sensors (30 Hz) and for USEA stimulation (i.e., intrafascicular stimulation at 30 Hz) at a fixed, suprathreshold stimulation amplitude, further improving the accuracy of the biomimetic encoding (19). Additional details regarding the model development and validation can be found in (19).

Both models presented here are distinct from those used in (30). The biomimetic 1 algorithm concurrently modulates frequency and amplitude most closely replicating the responses of populations of slowly adapting type 1 (SA1) and rapidly adapting (RA) fibers. The biomimetic 2 algorithm provides a more faithful replication of a complete aggregate nerve response, keeping the population size constant (fixed stimulation amplitude) and mimicking the aggregate firing rate of SA1, RA, and Pacinian fibers within that population of the nerve. Both models are computationally efficient, allowing for real-time biomimetic sensory encoding. Analytic formulations for each encoding algorithm are provided in Table 2.

We did not attempt to measure the intuitiveness or naturalism of the sensory encoding algorithms, nor did we track the participant's ability to interpret this feedback. Experimental sessions were kept under 2 hours, and no learning effects were observed in this time frame.

Fragile object test

The fragile object test [originally introduced in (35)] has been used as a variant of the modified Box and Blocks test (36) to show the benefits of sensory feedback (14, 30, 36). Our implementation of this test differed from its predecessors in that the object was much heavier and the ratio between the weight and breaking force was much smaller, rendering the overall task more difficult. In (36), the fragile object weighed 8 g and broke if a force of 10.7 ± 1 N was applied to it (ratio of 1.34 N/g), and in (14, 30), the object weighed ~ 80 g and broke with a force of 1.23 ± 0.02 N (ratio of 0.15 N/g). In contrast, the object used in this study weighed 630.57 g and broke at 14.79 ± 0.34 N (ratio of 0.02 N/g).

The participant used only thumb abduction/adduction, and artificial sensory feedback was provided on the basis of the thumb contact sensor. Trial failure was defined as “breaking” the object, which occurred when the compression force exceeded 14.79 ± 0.34 N, or an inability to move the object in 30 s. Trial success was defined as a trial in which the participant lifted and placed the unbroken object within an adjacent circle on the table (~ 10 cm away) within 30 s. In half of the sets, the participant was intentionally distracted by having to count backward by twos from a random even number between 50 and 100.

A single trial was performed once every minute. A single experimental block consisted of eight trials with or without artificial sensory feedback. The participant completed five experimental blocks with and without sensory feedback for both the basic and distracted conditions. The experimental blocks were counterbalanced to reduce order effects. Under all conditions, the participant was able to use audiovisual feedback to help locate and grasp the object, as well as to identify when the object broke.

Statistical analyses were run separately for the basic and distracted conditions. A 50% binomial test was used to determine whether

Table 2. Sensory encoding algorithms. F_t , frequency at time t ; A_t , amplitude at time t ; c_t , normalized contact value at time t ; v_t , velocity at time t ; a_t , acceleration at time t ; min, minimum value; max, maximum value. Note that for all algorithms, sensory feedback is off and no stimulation occurs when $c_t = 0$.

Sensory encoding algorithm(s)	Analytic formulation
Binary	$F_t = F_{\min}$ $A_t = A_{\min}$
Traditional linear	$F_t = c_t(F_{\max} - F_{\min}) + F_{\min}$ $A_t = c_t(A_{\max} - A_{\min}) + A_{\min}$
Scaled traditional linear	$F_t = 2c_t(F_{\max} - F_{\min}) + F_{\min}$ $A_t = 2c_t(A_{\max} - A_{\min}) + A_{\min}$
Biomimetic 1	$F_t = \begin{cases} c_t(F_{\max} - F_{\min}) + F_{\min}, & v_t < 0 \\ (5v_t + c_t) * (F_{\max} - F_{\min}) + F_{\min}, & v_t \geq 0 \end{cases}$ $A_t = \begin{cases} c_t(A_{\max} - A_{\min}) + A_{\min}, & v_t < 0 \\ (5v_t + c_t) * (A_{\max} - A_{\min}) + A_{\min}, & v_t \geq 0 \end{cases}$
Biomimetic 2	$F_t = 186c_t - 185c_{t-1} + 1559v_t - 360v_{t-1} - 109v_{t-2} + 364a_t + 170a_{t-1} - 3$ $A_t = A_{\min}$

Table 3. Motor control specifications.

DOF	Range	Precision	Angle at rest
Thumb adduction/abduction	0°–75°	0.08° per bit	22.5°
Thumb reposition/opposition	50°–100°	0.10° per bit	80°
Index extend/flex	0°–90°	0.09° per bit	27°
D3, D4, and D5 extend/flex	0°–90°	0.09° per bit	27°
Wrist supinate/pronate	–120°–175°	0.29° per bit	0°
Wrist extend/flex	–55°–55°	0.11° per bit	0°

performance was significantly greater than chance alone. For comparison of completion time for the successful trials, response times showed no deviations from normality (Anderson-Darling, Jarque-Bera, and Lilliefors tests). Unpaired t tests (unequal sample size due to different success rates) were then used to compare completion times.

Object discrimination tasks

For size discrimination, the participant had to distinguish between a “large” lacrosse ball and a “small” golf ball (fig. S6). The two objects were chosen so that they represented real-world interactions, minimized differences in compliance, and maximized differences in size while still requiring some degree of active flexion to make contact. Relative to the index finger’s full range of motion, the large object required a 19% decrease in joint angle to make contact, and the small object required a 49% decrease. Response time was measured from the start of the trial to when the participant verbally reported the object’s size.

For compliance discrimination, the participant had to distinguish between a “soft” foam block and a “hard” plastic block (fig. S7). The soft block was cut to match the size of the hard block so that stimulation due to initial contact occurred at the same degree of index flexion. Response time was measured from the start of stimulation (i.e., measurable contact was made with the object and the participant felt the object) to the time when the participant verbally reported the object’s compliance.

We did not attempt to quantify how many levels of size and compliance the participant was able to discriminate. With traditional linear feedback, the just-noticeable difference of the neural stimulation would bind the discrimination capabilities. Instead, we focused on quantifying improvements in the intuitiveness of the sensory feedback (measured by response time) when biomimetic stimulation regimes are used.

For both, the output of the modified Kalman filter was used to directly control the position of the index finger. Position control (i.e., postural control) provided improved performance relative to velocity control (fig. S6), which is consistent with the natural encoding schemes of the hand (56). The participant received cutaneous sensory feedback from the index contact sensor; proprioceptive sensory feedback was not provided, although endogenous proprioception of residual forearm muscles and efference copy may have been present. The participant was blindfolded and wore headphones, and the physical prosthesis was detached from his residual limb, so that external cues about the object were eliminated.

A single trial was performed once every minute. For each trial, the participant was given 30 s to complete the task. A single experimental block consisted of eight trials using a single algorithm. The participant completed two experimental blocks for each size discrimination algorithm and two to six experimental blocks for each compliance discrimination algorithm. The order of the objects was pseudorandomized such that equal numbers of both appeared in the experimental block. The experimental blocks were counterbalanced to reduce order effects.

Statistical analyses were run separately for each algorithm comparison. Because of limited time with the participant, direct comparisons were limited to biomimetic 1 versus traditional linear, biomimetic 1 versus scaled traditional linear, and biomimetic 2 versus scaled traditional linear. A 50% binomial test was used to determine whether performance was significantly greater than chance alone. For algorithm comparisons, response times showed no deviations from normality (Anderson-Darling, Jarque-Bera, and Lilliefors tests). Paired t tests were then used for these comparisons on a trial-by-trial basis to control for order effects and sensory adaptation (57). Statistical analysis of response times with biomimetic and nonbiomimetic encoding algorithms was confined to algorithms using the same stimulation parameters on the

Table 4. Sensor specifications.

Sensor	Range	Precision
Thumb adduction/abduction	0°–75°	2.08×10^{-40} per bit
Thumb reposition/opposition	0°–100°	1.56×10^{-40} per bit
Index extend/flex	0°–90°	1.74×10^{-40} per bit
MRP extend/flex	0°–90°	1.74×10^{-40} per bit
Wrist supinate/pronate	–120°–175°	5.30×10^{-50} per bit
Wrist extend/flex	–55°–55°	1.42×10^{-40} per bit
All contact sensors	0 to 25.6 N	0.1 N per bit

same day to avoid any variations in evoked sensations that may have occurred across days.

Grasping Relative Index of Performance test

A detailed description of the GRIP test is reported elsewhere (38). Briefly, a screen was placed between the participant's line of sight to the prosthesis and the load cell to eliminate audiovisual cues from the prosthetic hand. In contrast to the fragile object test, the GRIP test measures the ability to modulate grip force without audiovisual feedback. The participant was presented with pictures of one of eight objects (Fig. 4) and instructed to grab the load cell with a force appropriate for gripping the object shown in the picture. The participant grabbed each of the eight objects 20 times without sensory feedback and 20 times with sensory feedback. Outliers and trials with preemptive grasps were not included in the analysis (38). Peak grasping forces showed no deviations from normality (Anderson-Darling, Jarque-Bera, and Lilliefors tests). Unpaired *t* tests were used to compare means, and Levene's test was used to compare SDs.

DEKA LUKE arm and ADLs

The DEKA LUKE arm, as used in this study, has 6 moveable DOFs (Table 3), 6 position sensors, and 13 contact sensors (Table 4). The prosthetic is interfaced via a controller area network communication protocol with 100-Hz update cycles. The accuracy of the movements is dictated by the precision of the motor commands (Table 3). For more information regarding the accuracy of the control algorithm, see the Supplementary Materials and (54).

The DEKA LUKE arm, in its transradial configuration, weighs about 1.27 kg (58), slightly more than that of an intact human hand. There are no temperature or pain sensors on the DEKA LUKE arm. Furthermore, electrical stimulation of sensory afferents preferentially activates larger diameter fibers first (59), making USEA-evoked pain or temperature percepts uncommon.

The six position sensors correspond to the six moveable DOFs. The 13 contact sensors are made of nine torque sensors for contact applied to the fingers and four force sensors for contact applied to the hand. There is a torque sensor for digits D2 to D5 that detects torque applied to the finger opposing flexion (e.g., during grasping of an object) and a torque sensor for the lateral portion of D2 (e.g., during a key grip). D1 also has four additional torque sensors to detect contact due to adduction, abduction, reposition, or opposition.

ADLs were performed with the DEKA LUKE arm mounted to a custom socket that fit to the residual limb of the participant. With only

the prosthesis or with in conjunction with his intact hand, the participant performed basic ADLs (feeding and dressing) (43), instrumental ADLs (housework, meal preparation, and technology use) (44), and ADLs that he had found challenging without the prosthesis (loading a pillow into a pillowcase, hammering, donning and doffing a ring; Fig. 6). All ADLs were performed with audiovisual feedback to best approximate real-world use. Traditional linear sensory feedback was provided because ADLs were performed before implementing the biomimetic encoding algorithms. Because of limited patient time and an inability to precisely quantify performance, ADLs were not repeated with biomimetic sensory feedback.

Statistical analyses

All statistical analyses were run with significance as $\alpha = 0.05$. Data were checked for normality to ensure that the appropriate parametric analysis or nonparametric equivalent was used. Subsequent pairwise analyses were corrected for multiple comparisons using the Dunn-Šidák approach. All data are shown as means \pm SEM, unless otherwise stated.

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/32/eaax2352/DC1
 Stability of the USEA
 Decoding intended movements with a modified Kalman filter
 Surgical procedure
 Fig. S1. Modified Box and Blocks test.
 Fig. S2. Prosthesis efficiency and profitability task.
 Fig. S3. Projected fields of electrically evoked sensations.
 Fig. S4. Fragile object test.
 Fig. S5. Grasping Relative Index of Performance task.
 Fig. S6. Size discrimination task.
 Fig. S7. Compliance discrimination task.
 Fig. S8. Stability of USEA-evoked sensations.
 References (60–66)

REFERENCES AND NOTES

1. Mobius Bionics, "LUKE arm details—Mobius Bionics"; <http://mobiusbionics.com/luke-arm/>.
2. Ottobock, "Bebionic technical manual"; www.ottobock.com/media/local-media/prosthetics/upper-limb/files/14112_bebionic_user_guide_lo.pdf.
3. Touch Bionics, "User brochure: i-limb quantum"; http://touchbionics.com/sites/default/files/files/MA01374US%20rev.%202%2C%20January%202017%20User%20Brochure_i-limb%20quantum.pdf.
4. E. Biddiss, T. Chau, Upper-limb prosthetics: Critical factors in device abandonment. *Am. J. Phys. Med. Rehabil.* **86**, 977–987 (2007).
5. F. Cordella, A. L. Ciancio, R. Sacchetti, A. Davalli, A. G. Cutti, E. Guglielmelli, L. Zollo, Literature review on needs of upper limb prosthesis users. *Front. Neurosci.* **10**, 209 (2016).
6. S. Wendelken, D. M. Page, T. Davis, H. A. C. Wark, D. T. Kluger, C. Duncan, D. J. Warren, D. T. Hutchinson, G. A. Clark, Restoration of motor control and proprioceptive and cutaneous sensation in humans with prior upper-limb amputation via multiple Utah Slanted Electrode Arrays (USEAs) implanted in residual peripheral arm nerves. *J. Neuroeng. Rehabil.* **14**, 121 (2017).
7. T. A. Kuiken, G. Li, B. A. Lock, R. D. Lipschutz, L. A. Miller, K. A. Stubblefield, K. B. Englehart, Targeted muscle reinnervation for real-time myoelectric control of multifunction artificial arms. *JAMA* **301**, 619–628 (2009).
8. T. A. Kung, R. A. Bueno, G. K. Alkhalefah, N. B. Langhals, M. G. Urbanek, P. S. Cederna, Innovations in prosthetic interfaces for the upper extremity. *Plast. Reconstr. Surg.* **132**, 1515–1523 (2013).
9. M. Schiefer, D. Tan, S. M. Sidek, D. J. Tyler, Sensory feedback by peripheral nerve stimulation improves task performance in individuals with upper limb loss using a myoelectric prosthesis. *J. Neural Eng.* **13**, 016001 (2016).
10. D. W. Tan, M. A. Schiefer, M. W. Keith, J. R. Anderson, J. Tyler, D. J. Tyler, A neural interface provides long-term stable natural touch perception. *Sci. Transl. Med.* **6**, 257ra138 (2014).

11. G. S. Dhillon, K. W. Horsch, Direct neural sensory feedback and control of a prosthetic arm. *IEEE Trans. Neural Syst. Rehabil. Eng.* **13**, 468–472 (2005).
12. S. Raspopovic, M. Capogrosso, F. M. Petrini, M. Bonizzato, J. Rigosa, G. D. Pino, J. Carpaneto, M. Controzzi, T. Boretius, E. Fernandez, G. Granata, C. M. Oddo, L. Citi, A. L. Ciano, C. Cipriani, M. C. Carrozza, W. Jensen, E. Guglielmelli, T. Stieglitz, P. M. Rossini, S. Micera, Restoring natural sensory feedback in real-time bidirectional hand prostheses. *Sci. Transl. Med.* **6**, 222ra19 (2014).
13. C. M. Oddo, S. Raspopovic, F. Artoni, A. Mazzoni, G. Spigler, F. Petrini, F. Giambattistelli, F. Vecchio, F. Miraglia, L. Zollo, G. Di Pino, D. Camboni, M. C. Carrozza, E. Guglielmelli, P. M. Rossini, U. Faraguna, S. Micera, Intraneural stimulation elicits discrimination of textural features by artificial fingertip in intact and amputee humans. *eLife* **5**, e09148 (2016).
14. F. M. Petrini, G. Valle, I. Strauss, G. Granata, R. Di Iorio, E. D'Anna, P. Čvančara, M. Mueller, J. Carpaneto, F. Clemente, M. Controzzi, L. Bioni, C. Carboni, M. Barbaro, F. Iodice, D. Andreu, A. Hiairassary, J.-L. Divoux, C. Cipriani, D. Guiraud, L. Raffo, E. Fernandez, T. Stieglitz, S. Raspopovic, P. M. Rossini, S. Micera, Six-month assessment of a hand prosthesis with intraneural tactile feedback: Hand prosthesis. *Ann. Neurol.* **85**, 137–154 (2019).
15. L. Zollo, G. D. Pino, A. L. Ciano, F. Ranieri, F. Cordella, C. Gentile, E. Noce, R. A. Romeo, A. D. Bellingegni, G. Vadalà, S. Miccinilli, A. Mioli, L. Diaz-Balzani, M. Bravi, K.-P. Hoffmann, A. Schneider, L. Denaro, A. Davalli, E. Gruppioni, R. Sacchetti, S. Castellano, V. D. Lazzaro, S. Sterzi, V. Denaro, E. Guglielmelli, Restoring tactile sensations via neural interfaces for real-time force-and-slippage closed-loop control of bionic hands. *Sci. Robot.* **4**, eaau9924 (2019).
16. E. D'Anna, G. Valle, A. Mazzoni, I. Strauss, F. Iberite, J. Patton, F. M. Petrini, S. Raspopovic, G. Granata, R. D. Iorio, M. Controzzi, C. Cipriani, T. Stieglitz, P. M. Rossini, S. Micera, A closed-loop hand prosthesis with simultaneous intraneural tactile and position feedback. *Sci. Robot.* **4**, eaau8892 (2019).
17. H. P. Saal, S. J. Bensmaia, Touch is a team effort: Interplay of submodalities in cutaneous sensibility. *Trends Neurosci.* **37**, 689–697 (2014).
18. S. Hsiao, Central mechanisms of tactile shape perception. *Curr. Opin. Neurobiol.* **18**, 418–424 (2008).
19. E. V. Okorokova, Q. He, S. J. Bensmaia, Biomimetic encoding model for restoring touch in bionic hands through a nerve interface. *J. Neural Eng.* **15**, 066033 (2018).
20. G. Westling, R. S. Johansson, Responses in glabrous skin mechanoreceptors during precision grip in humans. *Exp. Brain Res.* **66**, 128–140 (1987).
21. R. S. Johansson, G. Westling, Programmed and triggered actions to rapid load changes during precision grip. *Exp. Brain Res.* **71**, 72–86 (1988).
22. G. Westling, R. S. Johansson, Factors influencing the force control during precision grip. *Exp. Brain Res.* **53**, 277–284 (1984).
23. E. D'Anna, F. M. Petrini, F. Artoni, I. Popovic, I. Simanić, S. Raspopovic, S. Micera, A somatotopic bidirectional hand prosthesis with transcutaneous electrical nerve stimulation based sensory feedback. *Sci. Rep.* **7**, 10930 (2017).
24. C. Pylatiuk, A. Kargov, S. Schulz, Design and evaluation of a low-cost force feedback system for myoelectric prosthetic hands. *J. Prosthet. Orthot.* **18**, 57–61 (2006).
25. H. J. B. Witteveen, E. A. Droog, J. S. Rietman, P. H. Veltink, Vibro- and electro tactile user feedback on hand opening for myoelectric forearm prostheses. *IEEE Trans. Biomed. Eng.* **59**, 2219–2226 (2012).
26. E. Mastinu, P. Doguet, Y. Botquin, B. Häkansson, M. Ortiz-Catalan, Embedded system for prosthetic control using implanted neuromuscular interfaces accessed via an osseointegrated implant. *IEEE Trans. Biomed. Circuits Syst.* **11**, 867–877 (2017).
27. M. Ortiz-Catalan, E. Mastinu, R. Brånemark, B. Häkansson, Direct neural sensory feedback and control via osseointegration, in *XVI World Congress of the International Society for Prosthetics and Orthotics* (2017).
28. H. Charkhkar, C. E. Shell, P. D. Marasco, G. J. Pinault, D. J. Tyler, R. J. Triolo, High-density peripheral nerve cuffs restore natural sensation to individuals with lower-limb amputations. *J. Neural Eng.* **15**, 056002 (2018).
29. D. M. Page, J. A. George, D. T. Kluger, C. Duncan, S. Wendelken, T. Davis, D. T. Hutchinson, G. A. Clark, Motor control and sensory feedback enhance prosthesis embodiment and reduce phantom pain after long-term hand amputation. *Front. Hum. Neurosci.* **12**, 352 (2018).
30. G. Valle, A. Mazzoni, F. Iberite, E. D'Anna, I. Strauss, G. Granata, M. Controzzi, F. Clemente, G. Rognini, C. Cipriani, T. Stieglitz, F. M. Petrini, P. M. Rossini, S. Micera, Biomimetic intraneural sensory feedback enhances sensation naturalness, tactile sensitivity, and manual dexterity in a bidirectional prosthesis. *Neuron* **100**, 37–45.e7 (2018).
31. J. A. George, M. R. Brinton, C. C. Duncan, D. T. Hutchinson, G. A. Clark, Improved training paradigms and motor-decode algorithms: Results from intact individuals and a recent transradial amputee with prior complex regional pain syndrome, in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (IEEE, 2018), pp. 3782–3787.
32. J. S. Hebert, J. Lewicke, Case report of modified Box and Blocks test with motion capture to measure prosthetic function. *J. Rehabil. Res. Dev.* **49**, 1163–1174 (2012).
33. D. T. Beckler, Z. C. Thumser, P. D. Marasco, Descriptive outcome metrics of sensorized upper limb performance using optimal foraging theory, in *MEC17 - A Sense of What's to Come, Myoelectric Controls and Upper Limb Prosthetics Symposium* (Univ. of New Brunswick, 2017), no. 52.
34. A.-S. Augurelle, A. M. Smith, T. Lejeune, J.-L. Thonnard, Importance of cutaneous feedback in maintaining a secure grip during manipulation of hand-held objects. *J. Neurophysiol.* **89**, 665–671 (2003).
35. E. D. Engeberg, S. Meek, Improved grasp force sensitivity for prosthetic hands through force-derivative feedback. *IEEE Trans. Biomed. Eng.* **55**, 817–821 (2008).
36. F. Clemente, M. D'Alonzo, M. Controzzi, B. B. Edin, C. Cipriani, Non-invasive, temporally discrete feedback of object contact and release improves grasp control of closed-loop myoelectric transradial prostheses. *IEEE Trans. Neural Syst. Rehabil. Eng.* **24**, 1314–1322 (2016).
37. D. Atkins, D. C. Y. Heard, W. H. Donovan, Epidemiologic overview of individuals with upper-limb loss and their reported research priorities. *J. Prosthet. Dent.* **8**, 2–11 (1996).
38. Z. C. Thumser, A. B. Slifkin, D. T. Beckler, P. D. Marasco, Fitts' law in the control of isometric grip force with naturalistic targets. *Front. Psychol.* **9**, 560 (2018).
39. B. P. Delhay, K. H. Long, S. J. Bensmaia, Neural basis of touch and proprioception in primate cortex. *Compr. Physiol.* **8**, 1575–1602 (2018).
40. R. S. Johansson, I. Birznieks, First spikes in ensembles of human tactile afferents code complex spatial fingertip events. *Nat. Neurosci.* **7**, 170–177 (2004).
41. S. J. Lederman, R. L. Klatzky, Extracting object properties through haptic exploration. *Acta Psychol.* **84**, 29–40 (1993).
42. T. Callier, A. K. Suresh, S. J. Bensmaia, Neural coding of contact events in somatosensory cortex. *Cereb. Cortex* **2019**, bhy337 (2019).
43. S. E. Hardy, Consideration of function and functional decline, in *Current Diagnosis & Treatment: Geriatrics*, B. A. Williams, A. Chang, C. Ahalt, H. Chen, R. Conant, C. S. Landefeld, C. Ritchie, M. Yukawa, Eds. (McGraw-Hill Education, 2014), pp. 3–8.
44. S. S. Roley, J. V. DeLany, C. J. Barrows, S. Brownrigg, D. Honaker, D. I. Sava, V. Talley, K. Voelkerding, D. A. Amini, E. Smith, P. Toto, S. King, D. Lieberman, M. C. Baum, E. S. Cohen, P. A. M. Cleveland, M. J. Youngstrom; American Occupational Therapy Association Commission on Practice, Occupational therapy practice framework: Domain & practice, 2nd edition. *Am. J. Occup. Ther.* **62**, 625–683 (2008).
45. M. Ortiz-Catalan, B. Häkansson, R. Brånemark, An osseointegrated human-machine gateway for long-term sensory feedback and motor control of artificial limbs. *Sci. Transl. Med.* **6**, 257re6 (2014).
46. S. Lewis, M. F. Russold, H. Dietl, E. Kaniusas, User demands for sensory feedback in upper extremity prostheses, in *2012 IEEE International Symposium on Medical Measurements and Applications Proceedings* (IEEE, 2012), pp. 1–4.
47. E. L. Graczyk, L. Resnik, M. A. Schiefer, M. S. Schmitt, D. J. Tyler, Home use of a neural-connected sensory prosthesis provides the functional and psychosocial experience of having a hand again. *Sci. Rep.* **8**, 9866 (2018).
48. B. Rosén, H. H. Ehrsson, C. Antfolk, C. Cipriani, F. Sebelius, G. Lundborg, Referral of sensation to an advanced humanoid robotic hand prosthesis. *Scand. J. Plast. Reconstr. Surg. Hand Surg.* **43**, 260–266 (2009).
49. C. Dietrich, K. Walter-Walsh, S. Preißler, G. O. Hofmann, O. W. Witte, W. H. R. Miltner, T. Weiss, Sensory feedback prosthesis reduces phantom limb pain: Proof of a principle. *Neurosci. Lett.* **507**, 97–100 (2012).
50. M. R. Brinton, E. Barcikowski, M. Paskett, T. Davis, J. Nieveen, D. T. Kluger, J. A. George, D. J. Warren, G. A. Clark, Development of a take-home system for control of advanced prosthetics, paper presented at Neuroscience 2018, San Diego, CA, 3 to 7 November 2018.
51. T. S. Davis, H. A. C. Wark, D. T. Hutchinson, D. J. Warren, K. O'Neill, T. Scheinblum, G. A. Clark, R. A. Normann, B. Greger, Restoring motor control and sensory feedback in people with upper extremity amputations using arrays of 96 microelectrodes implanted in the median and ulnar nerves. *J. Neural Eng.* **13**, 036001 (2016).
52. G. A. Clark, S. Wendelken, D. M. Page, T. Davis, H. A. Wark, R. A. Normann, D. J. Warren, D. T. Hutchinson, Using multiple high-count electrode arrays in human median and ulnar nerves to restore sensorimotor function after previous transradial amputation of the hand, in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (IEEE, 2014), pp. 1977–1980.
53. J. G. Nieveen, Y. Zhang, S. Wendelken, T. S. Davis, D. T. Kluger, J. A. George, D. J. Warren, D. T. Hutchinson, C. Duncan, G. A. Clark, J. V. Mathews, Polynomial kalman filter for myoelectric prosthetics using efficient kernel ridge regression, in *2017 8th International IEEE/EMBS Conference on Neural Engineering (NER)* (IEEE, 2017), pp. 432–435.
54. J. A. George, T. S. Davis, M. R. Brinton, G. Clark, Intuitive neuromyoelectric control of a dexterous bionic arm using a modified Kalman filter. *J. Neurosci. Methods* (In Review).
55. H. P. Saal, B. P. Delhay, B. C. Rayhaun, S. J. Bensmaia, Simulating tactile signals from the whole hand with millisecond precision. *Proc. Natl. Acad. Sci. U.S.A.* **114**, E5693–E5702 (2017).
56. J. M. Goodman, G. A. Tabot, A. S. Lee, A. K. Suresh, A. T. Rajan, N. G. Hatsopoulos, S. J. Bensmaia, Postural representations of the hand in primate sensorimotor cortex. *bioRxiv*, 566539 (2019).

57. E. L. Graczyk, B. P. Delhay, M. A. Schiefer, S. J. Bensmaia, D. J. Tyler, Sensory adaptation to electrical stimulation of the somatosensory nerves. *J. Neural Eng.* **15**, 046002 (2018).
58. L. Resnik, S. L. Klinger, K. Etter, The DEKA Arm: Its features, functionality, and evolution during the Veterans Affairs Study to optimize the DEKA Arm. *Prosthetics Orthot. Int.* **38**, 492–504 (2014).
59. D. R. Merrill, M. Bikson, J. G. R. Jefferys, Electrical stimulation of excitable tissue: Design of efficacious and safe protocols. *J. Neurosci. Methods* **141**, 171–198 (2005).
60. B. Baker, R. Caldwell, D. Crosland, R. Sharma, D. Kluger, J. A. George, A. Harding, L. Reith, Improved long-term performance of Utah slanted arrays in clinical studies, paper presented at Neuroscience 2018, San Diego, CA, 3 to 7 November 2018. <https://www.abstractsonline.com/pp8/#!/4649/presentation/41178>.
61. D. J. Warren, S. Kellis, J. G. Nieveen, S. M. Wendelken, H. Dantas, T. S. Davis, D. T. Hutchinson, R. A. Normann, G. A. Clark, V. J. Mathews, Recording and decoding for neural prostheses. *Proc. IEEE* **104**, 374–391 (2016).
62. W. Wu, M. J. Black, D. Mumford, Y. Gao, E. Bienenstock, J. P. Donoghue, Modeling and decoding motor cortical activity using a switching Kalman filter. *IEEE Trans. Biomed. Eng.* **51**, 933–942 (2004).
63. R. L. Rennaker, J. Miller, H. Tang, D. A. Wilson, Minocycline increases quality and longevity of chronic neural recordings. *J. Neural Eng.* **4**, L1–L5 (2007).
64. P. J. Rousche, R. A. Normann, A method for pneumatically inserting an array of penetrating electrodes into cortical tissue. *Ann. Biomed. Eng.* **20**, 413–422 (1992).
65. L. Spataro, J. Dilgen, S. Retterer, A. J. Spence, M. Isaacson, J. N. Turner, W. Shain, Dexamethasone treatment reduces astroglia responses to inserted neuroprosthetic devices in rat neocortex. *Exp. Neurol.* **194**, 289–300 (2005).
66. Y. Zhong, R. V. Bellamkonda, Dexamethasone coated neural probes elicit attenuated inflammatory response and neuronal loss compared to uncoated neural probes. *Brain Res.* **1148**, 15–27 (2007).

Acknowledgments: We thank the participant in this study who freely donated 14 months of his life for the advancement of knowledge and for a better future for amputees. **Funding:** This work was sponsored by the Hand Proprioception and Touch Interfaces (HAPTIX) program administered by the Biological Technologies Office (BTO) of the Defense Advanced Research Projects Agency (DARPA), through the Space and Naval Warfare Systems Center (contract no. N66001-15-C-4017). Additional sponsorship was provided by the NSF through grant no. NSF ECCS-1533649 and NSF GRFP award no. 1747505. **Author contributions:**

J.A.G. developed and implemented sensory encoding algorithms, designed experiments, collected data, and drafted the manuscript. D.T.K. developed the software for the prosthesis, designed experiments, collected data, and assisted with drafting of the manuscript. T.S.D. developed the software for closed-loop prosthesis control. E.V.O. and Q.H. developed the sensory encoding algorithm based on afferent recordings. C.C.D. provided clinical support and expertise throughout. D.T.H. implanted and explanted the USEAs and iEMGs, provided clinical oversight throughout the study, and oversaw the development of experiments and protocols. Z.C.T., D.T.B., and P.D.M. provided the equipment, analysis, and guidance for GRIP and PEP tests (under DARPA contract no. N66001-15-C-4015). S.J.B. oversaw the development of sensory encoding algorithms, assisted in the experimental design, and aided with drafting of the manuscript. G.A.C. oversaw and led the development of all methods, experiments, and protocols and assisted with experiments and drafting of the manuscript. All authors contributed to the revision of the manuscript. **Competing interests:** D.T.K., T.S.D., S.M.W., C.C.D., and G.A.C. are inventors of the prosthetic control algorithm used in this study (international application no. PCT/US2017/044947). D.T.K., S.M.W., D.T.H., and G.A.C. are inventors of the protective carrier devices used during the implant procedure (international application no. PCT/US2017/044427). G.A.C. holds patents on the implants' signal referencing and antinoise architecture (U.S. patent nos. 8639312 and 8359083). T.S.D. is a former consultant for Ripple Neuro LLC. D.T.K. is now an employee of Blackrock Microsystems, which provided implanted materials for this study. **Data and materials availability:** All data needed to evaluate the conclusions are available in the paper or the supplementary materials. Data and materials requests should be sent to G.A.C. (greg.clark@utah.edu). Requestors may need to be approved by the human-subjects research committees (e.g., local institutional review board and Department of the Navy Human Resources Protection Program) to comply with Health Insurance Portability and Accountability Act requirements.

Submitted 5 March 2019

Accepted 21 June 2019

Published 24 July 2019

10.1126/scirobotics.aax2352

Citation: J. A. George, D. T. Kluger, T. S. Davis, S. M. Wendelken, E. V. Okorokova, Q. He, C. C. Duncan, D. T. Hutchinson, Z. C. Thumser, D. T. Beckler, P. D. Marasco, S. J. Bensmaia, G. A. Clark, Biomimetic sensory feedback through peripheral nerve stimulation improves dexterous use of a bionic hand. *Sci. Robot.* **4**, eaax2352 (2019).

BIOMIMETICS

Insect-scale fast moving and ultrarobust soft robot

Yichuan Wu^{1,2,3}, Justin K. Yim⁴, Jiaming Liang^{1,2,3}, Zhichun Shao^{2,3}, Mingjing Qi^{2,3,5}, Junwen Zhong^{2,3*}, Zihao Luo², Xiaojun Yan⁵, Min Zhang^{6*}, Xiaohao Wang^{1,6}, Ronald S. Fearing⁴, Robert J. Full^{4,7}, Liwei Lin^{1,2,3*}

Mobility and robustness are two important features for practical applications of robots. Soft robots made of polymeric materials have the potential to achieve both attributes simultaneously. Inspired by nature, this research presents soft robots based on a curved unimorph piezoelectric structure whose relative speed of 20 body lengths per second is the fastest measured among published artificial insect-scale robots. The soft robot uses several principles of animal locomotion, can carry loads, climb slopes, and has the sturdiness of cockroaches. After withstanding the weight of an adult footstep, which is about 1 million times heavier than that of the robot, the system survived and continued to move afterward. The relatively fast locomotion and robustness are attributed to the curved unimorph piezoelectric structure with large amplitude vibration, which advances beyond other methods. The design principle, driving mechanism, and operating characteristics can be further optimized and extended for improved performances, as well as used for other flexible devices.

INTRODUCTION

Mobility and robustness are two engineering challenges for robots. Unlike large-scale robots based on materials of high stiffness/density and powered by bulky actuators/motors, small-scale soft robots are often restricted to small actuators with low output power based on materials of low stiffness/density. Hence, insect-scale soft robots are known to be easily damaged, exhibit poor control of locomotion, or are slow moving due to the nature of their small structures. Improving the mobility, efficiency, and robustness of soft robots made of a deformable body with the capability to carry extra weights to perform various functions has been challenging (1–4). Researchers have tried to develop soft robots that negotiate complex environments by taking advantage of soft matter physics in the interdisciplinary field termed “robophysics” (5, 6). Recent advances include micro/millimeter-scale robots with good mobility, such as crawling robots (7–11), hopping robots (12, 13), and multi-legged robots (14–18). However, these robots are made of rigid or partially rigid parts, resulting in poor robustness and low adaptability to shape changes and/or external perturbations. On the other hand, soft robots actuated by humidity (19–21), light (22–24), heat (25), and magnetic force (26–28) have been demonstrated but have slow responses, whereas others require bulky setups to generate the external power sources such as magnetic fields. Robots using thin film-based actuators based on lead zirconate titanate (PZT) have been successfully developed (17, 18, 29–31), but PZT is a brittle material containing poisonous lead. Polyvinylidene difluoride (PVDF) is soft, flexible, and lightweight, making it suitable for potential soft robot applications (32, 33), but one key challenge has been to generate fast, effective movement and even locomotion (34).

The locomotion mechanisms of animals continue to inspire the design of soft robots (4, 35). In particular, arthropods show how rapid, cyclic locomotion at high frequencies at this scale is possible without compromising robustness and survivability in harsh conditions (36, 37). Flying mosquitos can oscillate or vibrate their wings at more than 800 Hz (38), and 1-mm mites attain relative ground speeds exceeding 200 body lengths per second (BL/s) (39, 40). In this work, we introduce fast and robust insect-scale soft robots based on a curved piezoelectric PVDF unimorph structure to achieve several key advancements: (i) Under an alternating current (AC) driving power near the resonant frequency (850 Hz) of the structure, a prototype 10-mm-long robot (0.024 g) attained a relative speed of 20 BL/s—the fastest among published reports of insect-scale soft ground robots; (ii) after stepping on the robot with an adult human’s full body weight (59.5 kg, about 1 million times heavier than the robot), the robot could still move afterward, demonstrating exceptional robustness; (iii) the robot could move smoothly carrying a load weighing 0.406 g, which is six times heavier than that of the robot; (iv) further enhancement of agility was demonstrated by designing the moving mechanism to emulate features of galloping-like gaits using a two-leg prototype robot.

RESULTS

Structure and working mechanism

A prototype 3 cm-by-1.5 cm robot, consisting of a curved body and a leg-like structure at the front, is pictured alongside a U.S. quarter in Fig. 1A. A cross-sectional view scanning electron microscopy (SEM) image shows the unimorph structure made of an 18-μm-thick PVDF layer, two 50-nm-thick palladium (Pd)/gold (Au) electrodes (top and bottom of the PVDF film), a 25-μm-thick adhesive silicone, and a 25-μm-thick polyethylene terephthalate (PET) substrate at the bottom. The PVDF layer can produce periodic extension and contraction by the piezoelectric effect under an AC driving voltage to change the shape of the robot; the details of the actuation mechanism are explained in section S1 and fig. S1. This results in an oscillatory center of mass (COM) trajectory pattern (see movie S1) similar to many running animals (36). Figure 1B compares the COM motion of a cockroach and our prototype robot. Although the robot

Copyright © 2019
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

¹Tsinghua-Berkeley Shenzhen Institute, Tsinghua University, Shenzhen 518055, China.

²Berkeley Sensor and Actuator Center, University of California at Berkeley, Berkeley, CA 94720, USA. ³Department of Mechanical Engineering, University of California at Berkeley, Berkeley, CA 94720, USA. ⁴Department of Electrical Engineering and Computer Science, University of California at Berkeley, Berkeley, CA 94720, USA. ⁵School of Energy and Power Engineering, Beihang University, Beijing 100191, China. ⁶Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China. ⁷Department of Integrative Biology, University of California at Berkeley, Berkeley, CA 94720, USA.

*Corresponding author: junwenzhong@berkeley.edu (J.Z.); zhang.min@sz.tsinghua.edu.cn (M.Z.); lwlin@berkeley.edu (L.L.)

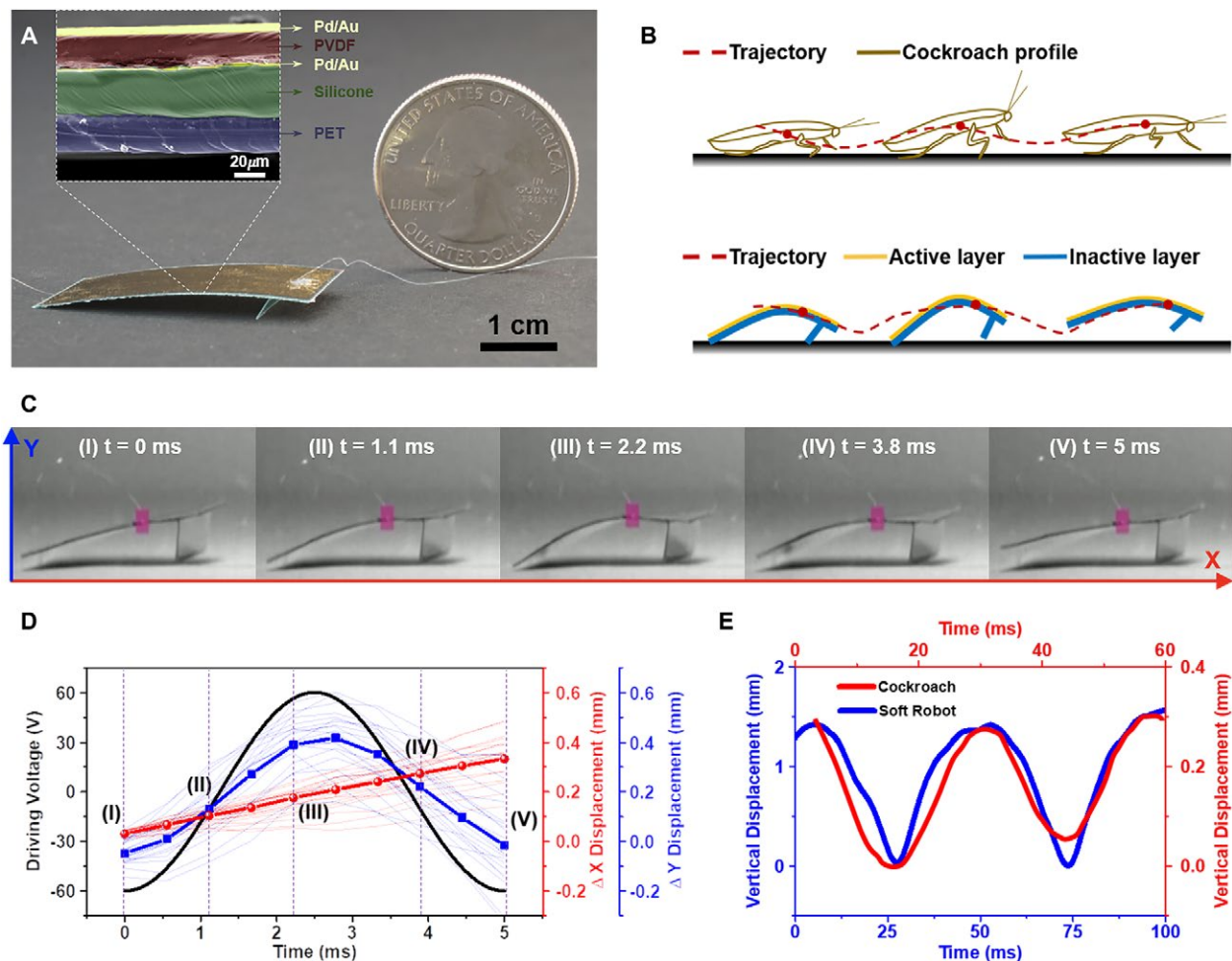


Fig. 1. The prototype and working mechanism. (A) Optical photo showing a robot connected with two electrical wires to the top and bottom electrodes, respectively, alongside a U.S. quarter coin. The inset SEM image shows the cross-sectional view of the prototype robot with different layers of materials. (B) Comparison of the wavelike running paths showing the movements of the COM of a cockroach (41) and a prototype robot (from movie S1). (C) Series of optical images recording the movements of a prototype robot in one driving cycle. (D) Applied driving signal (black line) and vertical (blue lines) and lateral displacements (red lines) of a prototype robot, where the bold solid lines are the average movements for 20 cycles. (E) Two-step cycles of the vertical displacement of the COM during cockroach running [red line for a period of 60 ms, (41)] compared with a prototype soft robot (blue line for a period of 100 ms).

has a unique morphology compared with many animals, it also showed a wavelike path. We developed a two-segment mass-spring model to best predict the robot's dynamic movements.

High-speed videos (sampling rate of 2000 frames per second) with a sequence of optical images seen in Fig. 1C have been used to record the postures and positions of the prototype robots (under -60 to 60 V of sinusoidal driving voltage at 200 Hz, which generates nonmaximal running performance; movie S2) running on a standard printing paper substrate. Within one cycle of the applied sinusoidal voltage, one set of the corresponding successive postures is depicted as states (I) to (V) in Fig. 1C as an example. In state (I) under -60 V, the body is extended and the front leg of the robot is in the ground-touching posture, whereas the abdomen is in the aerial posture. After 1.1 ms at state (II) under an applied voltage close to 0 V, the body recovers its initial shape, whereas the front leg of the robot is still in the ground-touching posture and the abdomen is in the aerial posture with a shorter distance to the ground as compared with that in state (I). In state (III) under $+60$ V, the body is contracted, and both the front leg and abdomen of

the robot are in the ground-touching posture. In the first-half driving cycle from state (I) to state (III), the body transitions from extended near-flat shape, to the initially curved shape, and then to the contracted shape. These shape changes cause the front leg to strike against the ground and produce a forward-pushing ground reaction force. In the second-half driving cycle from state (III) to state (V), the body goes through the similar shape changes, with the reverse order from the contracted shape to near-flat shape, which could cause the front leg to produce a backward-pushing ground reaction force to slow down the forward moving speed of the robot. Hence, we designed the bending angle of the front leg to be less than 90° to enhance the forward movements and reduce backward movements. Driven under high-frequency actuation coupled with various ground impact conditions and manufacturing variations, the exact shape changes and movements of the robot are rather complex. However, by varying design and operation parameters, the forward moving speed of the robot could be optimized. For example, Fig. 1D shows the experimental results of the lateral displacement (red lines) and vertical displacement

(blue lines) of a prototype robot under a driving voltage between -60 and 60 V (black line). In this analysis, the lateral/vertical displacement is defined as the lateral/vertical movements of the COM of the robot with respect to the original position. The randomness of the individual cycle is clearly observed in both displacements, although the average vertical COM positions follow the driving patterns and the average lateral COM positions show incremental forward movement. Figure 1E compares two-step cycles of the vertical movement of a cockroach (41) and the prototype soft robot (movie S1) with respect to time.

Animals appear to use resonant frequencies to oscillate their muscles and segments (42), with the flight muscles and thorax of flying insects serving as an example in the higher frequency range (43). We found that it was also desirable to drive the prototype robots near their resonant frequencies for largest deformations. To constrain the running direction of the robot so as to characterize their relative running speed, we used a transparent quartz tube with an inner diameter of 1 inch, as shown in fig. S2 (A and B). In this case, a 10-mm-long prototype (0.024 g) robot was used to achieve a relative running speed up to 20 BL/s driven near its resonant frequency at 850 Hz. In comparison, under driving frequencies of 800 and 900 Hz, lower relative running speeds of 13 and 3.6 BL/s were recorded, respectively (movie S3).

Animals appear to use resonant frequencies to oscillate their muscles and segments (42), with the flight muscles and thorax of flying insects serving as an example in the higher frequency range (43). We found that it was also desirable to drive the prototype robots near their resonant frequencies for largest deformations. To constrain the running direction of the robot so as to characterize their relative running speed, we used a transparent quartz tube with an inner diameter of 1 inch, as shown in fig. S2 (A and B). In this case, a 10-mm-long prototype (0.024 g) robot was used to achieve a relative running speed up to 20 BL/s driven near its resonant frequency at 850 Hz. In comparison, under driving frequencies of 800 and 900 Hz, lower relative running speeds of 13 and 3.6 BL/s were recorded, respectively (movie S3).

Locomotion analysis

We observed four main postures during the operation of the robot: aerial, front-touching, back-touching, and both-touching. In each posture, the robot's body can be either expanding or contracting depending on the applied driving signal at that instant. Hence, there are eight possible configurations, as shown in Fig. 2 (A to D), where gray dashed lines indicate the previous shapes and red solid lines are the immediate current shapes. In this illustration, G , F_b , and F_a are the gravitational force, ground reaction force at the

front leg, and ground reaction force at the end of the abdomen, respectively. The shape of the robot changes periodically based on the PVDF actuation force to excite elastic oscillations for the curved unimorph structure. The directions of the ground reaction force at the front leg and abdomen (blue arrows in the figure) will change depending on the posture and shape changes of the robot. The front leg construction of the robot is important because it produces

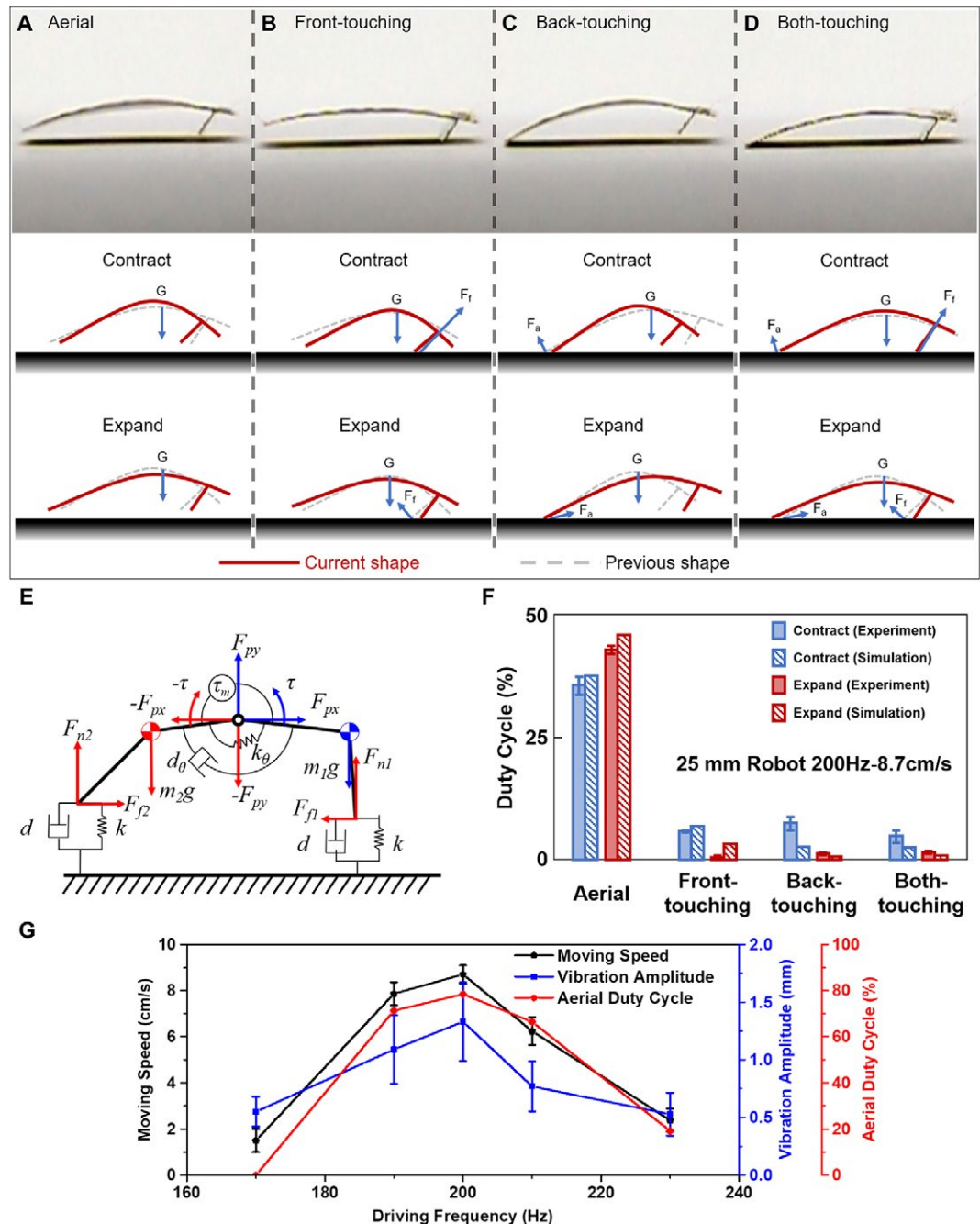


Fig. 2. Locomotion gait analysis. (A to D) Optical photos from the high-speed camera (top), corresponding contracted configurations (middle), and corresponding expanded configurations (bottom) of a prototype robot showing different gaits in the cross-sectional views. (E) Simplified dynamic model based on two rigid bodies joined by a pin joint (both-touching posture as an example) with a torsional spring-damper system. (F) Duty cycles in different gaits of both experimental and simulation results for a 25-mm-long prototype robot driven at its fastest speed at resonance of 200 Hz. (G) Relationships between the vibration amplitude and moving speed as well as aerial duty cycle for driving frequencies of 170, 190, 200, 210, and 230 Hz. Error bars indicate mean \pm 1 SD.

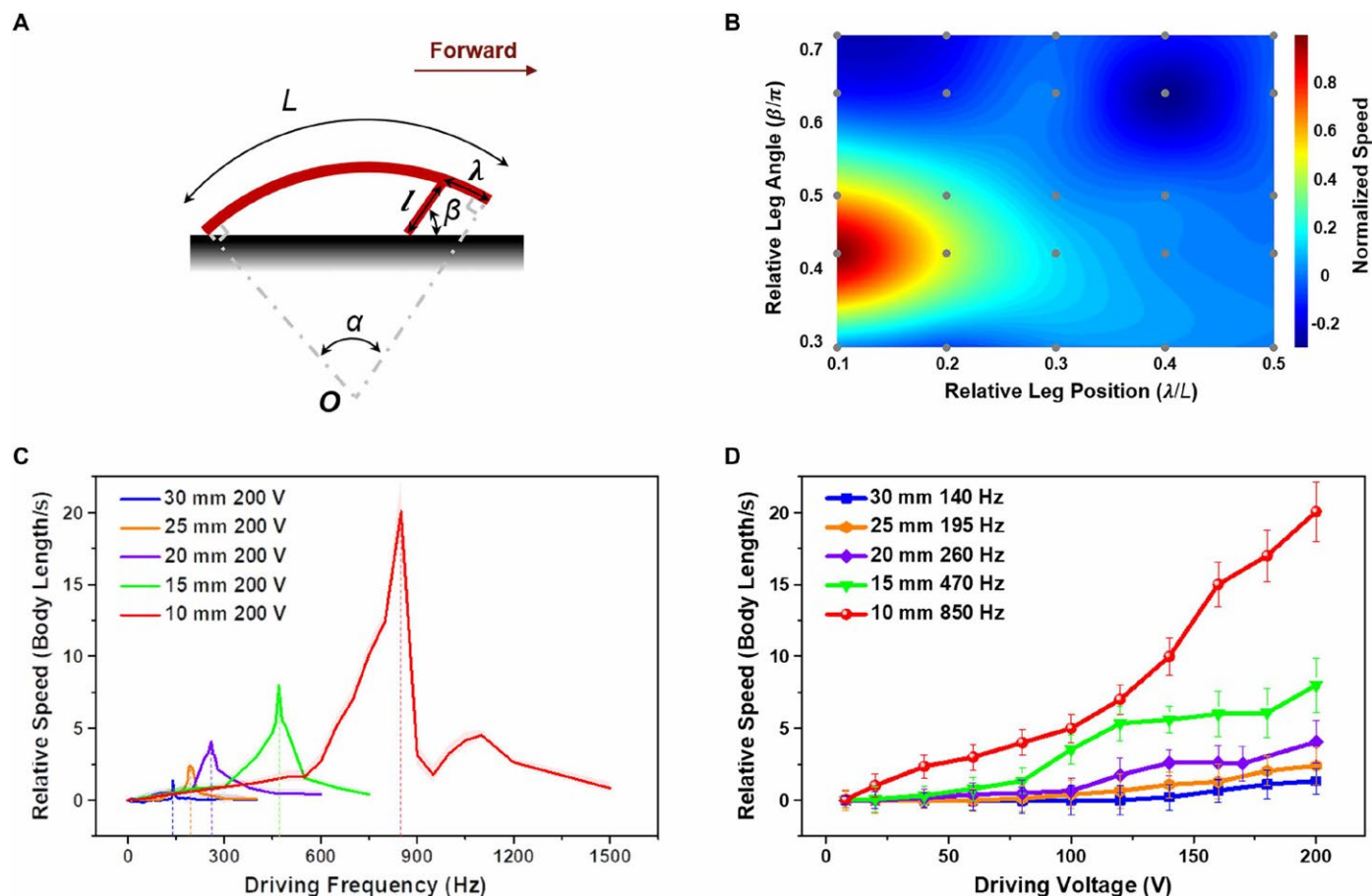


Fig. 3. Geometric parameter optimization and performance characterization. (A) Side view of a robot with the definitions of geometric parameters. (B) Experimental results (gray dots) for normalized running speeds from a 10-mm-long robot with different geometric combinations used to plot a color map as a function of relative leg position (λ/L) and relative leg angle (β/π). The color map is interpolated by the thin-plate spline interpolation scheme for surface fitting. Original data can be found in table S2. (C) Relative running speeds of robots versus the driving frequencies for robots with lengths of 10, 15, 20, 25, and 30 mm. Shaded bands represent 90% confidence limits. (D) Relative running speeds (under the resonant frequency) of robots versus the driving voltages for robots with lengths of 10, 15, 20, 25, and 30 mm. Error bars indicate mean ± 1 SD.

anisotropic lateral forces to generate forward locomotion. The most effective configuration for forward motion, a contracting touchdown of the front leg (contraction of the front-touching posture), greatly increased the lateral component of the ground reaction force and the vertical component for taking off. The abdomen functions to keep the balance of the dynamic system by adjusting the pitch of the body when it taps the ground, preventing the robot from flipping over backward. The details for qualitative introduction of the locomotion mechanism are discussed in section S2 and figs. S3 and S4.

Using a high-speed camera, we found that the first vibrational mode is easily excited and dominates the shape change, whereas higher-order modes can be neglected. A dynamic mass-spring model consisting of two rigid bodies (m_1 , m_2) joined by a pin joint was analyzed in MATLAB to model the curved robot, as shown in Fig. 2E (both-touching posture as an example). A torsional spring-damper (k_θ - d_θ) at the pin joint is excited by a sinusoidally varying torque source (τ_m) to represent the mechanical motions of the PVDF layer under the AC excitation. We modeled the ground contact at the front and back as a vertical spring-damper (k - d) with a normal force in the vertical direction (F_n) and a friction force in the lateral direction (F_t). The values of material parameters used in the model

and simulation can be found in table S1, and the simulation details are discussed in section S3 and fig. S5.

A 25-mm-long prototype robot was driven on a paper substrate and recorded under different driving parameters. The fastest running speed was 8.7 cm/s (movie S1) under 200 V at 200 Hz. The running speed reduced under the same applied voltage of 200 V at varying frequencies of 170, 190, 210, and 230 Hz, respectively (movie S4). To further study the operation of the robot, we statistically characterized the duty cycles for the eight configurations (Fig. 2, A to D). We plotted the results in Fig. 2F (shaded columns) for the trial with the fastest speed at 8.7 cm/s at 200 Hz (see fig. S6 for other frequencies with slower speeds). We also compared these results with the simulation data from the mass-spring model running at 200 Hz (Fig. 2F, hatched columns, and movie S5). We observed that a large percentage of aerial duty cycles were required to generate fast running speeds for the robot. For example, in this trial, the aerial cycles for the contracted and expanded configurations are about 36 and 43%, respectively, whereas all the other configurations have the duty cycles of less than 10%. In principle, large-amplitude oscillation driven at the resonant frequency should result in large deformation and greater forces to induce longer aerial duty cycles and higher foot velocities for faster speeds. Figure 2G compares the

average vibration amplitudes (measured when the robot is in the aerial posture) under the driving frequencies of 170, 190, 200, 210, and 230 Hz and their corresponding running speeds and aerial duty cycles. As expected, the large vibration amplitude due to the structural resonance resulted in faster running speed as well as longer aerial duty cycles (the combination of contracted and expended configurations). Again, we note that although the morphology and motion of our robot do not mimic any specific animal, small runners, such as cockroaches (41) and desert ants (44), also use aerial phases to attain their fastest speeds.

Parameter optimization and performance characterization

Geometric parameters play an important role in the performance of the robot. To simplify the structure and identify appropriate configurations, we defined the geometric parameters as shown schematically in Fig. 3A, where L is the body length of a robot, α is the body curvature, l is the length of the front leg, λ is the distance between the front leg and the head, and β is the contact angle between the front leg and the ground. Using a prototype robot of 10 mm (length) by 15 mm (width) by 3 mm (height) as an example, we first selected 25 combinations (Fig. 3B, gray dots) of the above geometric parameters to fabricate prototypes and conducted experiments to plot the normalized running speed map as a function of relative leg position (λ/L) and relative leg angle (β/π) in Fig. 3B. The color bar shows the magnitudes and directions of the normalized speed of the robot, with the red color areas representing the fastest running speed. We found that the values of λ/L and β/π near 0.1 and 0.4, respectively, resulted in robots with the fastest running speeds.

We then fabricated prototype robots with different lengths ranging from 10 to 30 mm at an interval of 5 mm using the map of λ/L and β/π of 0.1 and 0.4 for guidance. The resonant frequencies of robots with different lengths were approximately evaluated both analytically and experimentally and discussed in section S4 and figs. S7 and S8. In general, robots with smaller lengths have higher resonant frequencies and faster relative speeds. The relationships between driving frequency and relative speed for robots with lengths of 10, 15, 20, 25, and 30 mm are shown in Fig. 3C driven by a peak-to-peak voltage of 200 V to achieve measured maximum speeds of 20 BL/s (20 cm/s), 8 BL/s (12 cm/s), 4.05 BL/s (8.1 cm/s), 2.4 BL/s (6 cm/s), and 1.33 BL/s (4 cm/s), respectively. In Fig. 3D, the amplitude of the driving voltage versus the relative speed of robots with different lengths was measured near their resonant frequencies. As expected, larger driving voltages result in faster running speeds. For a 10-mm-long robot, as shown in fig. S9 (A and B), we observed noticeable motion even under an AC drive voltage as low as 8 V peak to peak (movie S6), which is a relatively low voltage requirement among insect-scale piezoelectric actuators

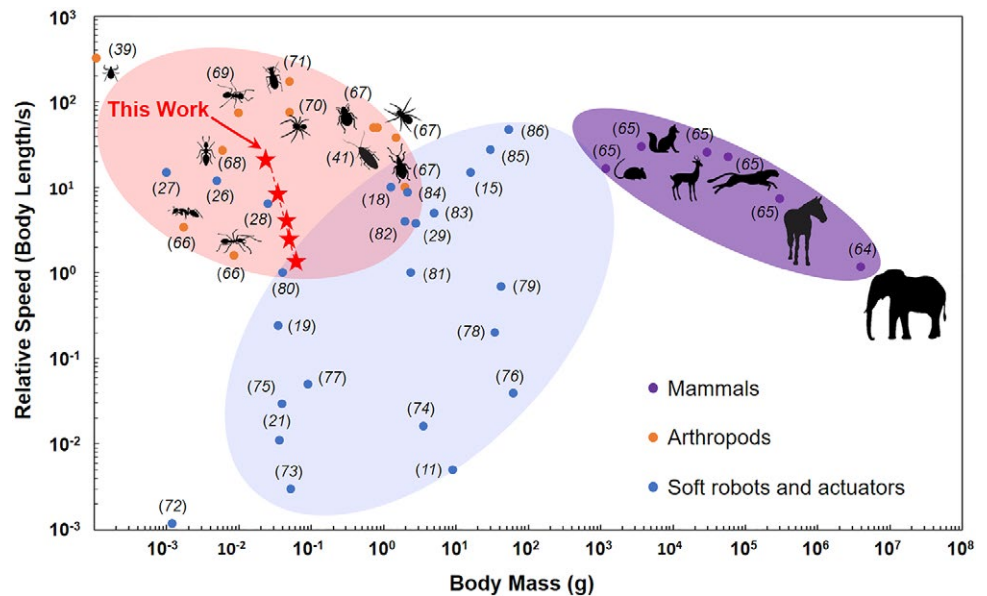


Fig. 4. Relative running speeds of some mammals, arthropods, soft robots, and actuators versus body mass. For animals including both mammals (purple) and arthropods (orange), relative speeds show a strong negative scaling law with respect to the body mass, showing that relative running speeds increase as body masses decrease. However, for soft robots, the relationship appears to be the opposite, where the relative running speeds decrease as the body mass decrease. The performances of the prototype robots (red stars with body lengths from 30 to 10 mm) follow a scaling law similar to that of living animals: Higher relative running speed was attained as the body mass decreased, with the fastest measured running speed at 20 BL/s among reported insect-scale robots and actuators (blue). For data, see table S3.

(45). Using a prototype 30-mm-long robot operating at 140 Hz as an example, we measured the voltage and current simultaneously using a data acquisition system for five cycles (fig. S10). The power consumption could be estimated as 0.343 mW by the integral of the voltage-current measurement results. When the robot operated near its resonant frequency, the cost of transport (COT) of the robot was about 14 (section S5), the lowest reported COT for robots and insects below 1 g (fig. S11) (46–59) with a relative speed of 4 BL/s under a power of 0.343 mW.

The comparison of relative moving speed

The relative moving speed is very important for animals because they often depend on fast locomotion to hunt for food, escape from predators, and/or compete for mating partners (60, 61). Researchers have shown that animals with high relative speeds are less likely to be caught and that relative speed may be more “ecologically relevant” than the absolute speed in various performance characterizations (61, 62). Figure 4 shows a comparison of relative running speeds with respect to body weights including our robots (red stars) and living animals, such as terrestrial mammals (purple) and running arthropods (orange), along with reported artificial soft robots or actuators (blue). For mammals, the trend in the elliptical and purple color shaded area indicates that the relative speed decreases as the body mass increases due to the scaling of mechanical constraints on the locomotive performance (61). However, small-size arthropods outperform larger animals in terms of their relative moving speeds. For example, a small mite, *Paratarsotomus macropalpis*, is now the world’s fastest known running animal, with a relative speed at several hundred body lengths per second (39). An opposite trend exists for soft robots, as shown in the elliptical and blue color shaded area, which

suggests that the relative speed increases as the body mass increases (19) except for recent robots driven by an external magnetic force (26–28). The robots presented in this work (five red stars with the body lengths from 30 to 10 mm; Fig. 4) have sizes similar to those of arthropods with a similar performance trend where the relative speed increases as the body mass decreases. As discussed in section S4, the relative running speed of our prototype robots had a positive correlation with the resonant frequency, so our smaller robots operated at higher resonant frequencies to achieve faster relative running speeds. The working efficiency of our prototype robot is high because the simple structure contains no redundant energy-consuming components. Although some soft robots driven by magnetic fields, humidity, or heat or light sources can have fast instantaneous running speeds, slow responses and a bulky setup to generate the external power, such as the magnetic field, are among the limitations.

Robustness

Robustness is essential for the survival of animals displaying both fail-safe and fault-tolerant behavior. For example, a cockroach can withstand a load 900 times its own body weight without injury because of its soft and shape-changing exoskeleton (37). The robot presented here also has exceptional robustness characteristics resulting from the assembly of soft materials with simple structures. Experimentally, the robustness of the prototype soft robot was demonstrated by applying a 100-g mass (1500 times its own body weight) with little change in its speed after the mass was removed, as shown in movie S7. Moreover, the soft robot could continue to function (one-half of the original speed) after being stepped on by an adult human (59.5 kg), a load about 1 million times its own body weight (Fig. 5, A to C, and movie S7). We systematically tested the robust performance of a 3 cm-by-1.5 cm prototype robot (fig. S12). The prototype robot was driven with the same condition (200 V and 140 Hz) before and after the applied load. In fig. S12, we report the speeds of a prototype robot after applying and removing different loads with magnitudes ranging from 10 to 59,500 g. We define the compressibility as the ratio of the vertical height change to the original height of the robot. We observed that if the applied load is below 100 g, then the robot can recover back to the original shape and maintain greater than 88% of its original speed. As the applied load increased, the moving speed decreased. When the applied load was above 10 kg, the compressibility of the robot increased and saturated at about 0.95, while the speed reduced and saturated at near 50% of the original speed. Even under heavy applied loads that flatten the robot initially, the robot could still partially recover after the removal of the applied load.

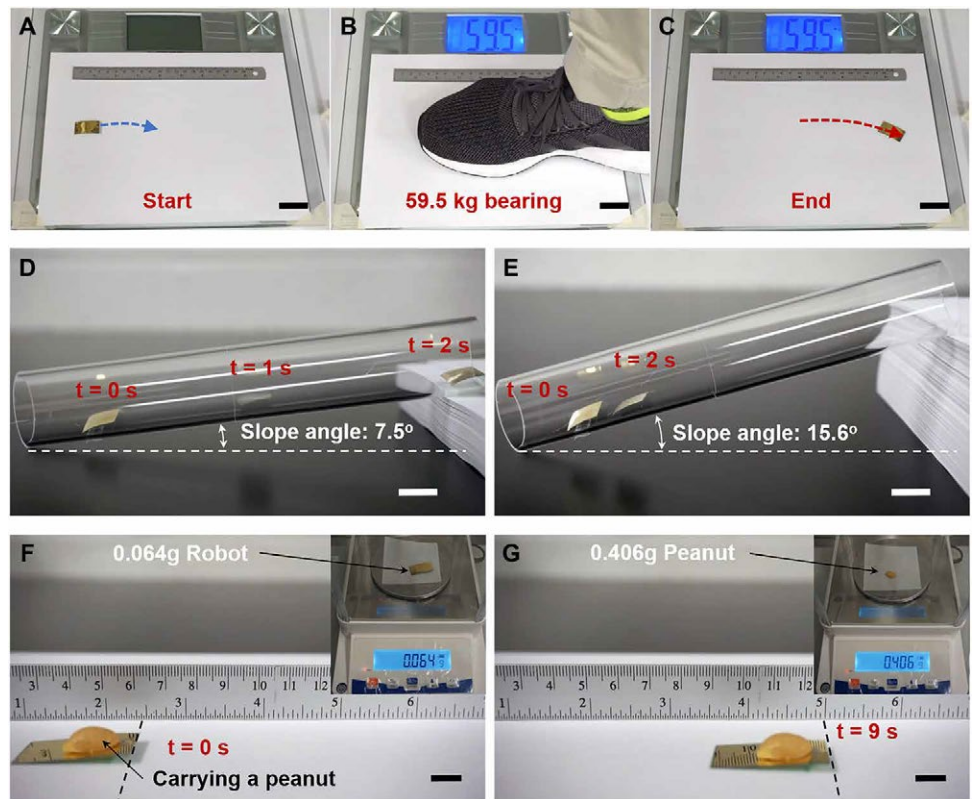


Fig. 5. Weight-bearing, slope-climbing, and load-carrying capabilities. (A to C) Soft robot can continue to function (one-half of the original speed) after being stepped on by an adult human (59.5 kg), a load about 1 million times its own body weight. Scale bars, 3 cm. A robot climbs a slope (D) of 7.5° with a relative speed of 7 BL/s and a slope (E) of 15.6° with a relative speed of 1 BL/s. Scale bars, 1 cm. (F and G) A robot (0.064 g) carries a peanut (0.406 g), which is six times its own body weight, to show the load-carrying capability. The speed with the peanut on top is about one-sixth of the original speed without the peanut. Scale bars, 1 cm.

Climbing and carrying loads

Animals and robots often need to do work such as climbing and carrying loads. The slope climbing capability of the robot is demonstrated in movie S7, in which the robot reached 7 BL/s while climbing a slope with an angle of 7.5° (Fig. 5D) and 1 BL/s while climbing a slope with an angle of 15.6° (Fig. 5E). Our soft prototype robot could also carry loads equal to the weight of a peanut (0.406 g) (Fig. 5, F and G). The robot was able to move smoothly while carrying a load that is six times its own weight at about one-sixth of the original speed (movie S7).

Speed enhancement by galloping-like gait

To further increase the running speed, we added and attached a back leg to a 3 cm-by-5 cm prototype robot to emulate galloping-like gaits (movie S8). Galloping is used by some rapid running mammals, where back bending increases stride length and allows the recovery of stored elastic energy (63). Specifically shown in (i) to (xi) of Fig. 6A, successive stages in a galloping stride and their corresponding schematic diagrams illustrate the operation of the galloping gait. With the more effective galloping-like gait mechanism, a two-legged robot achieved a running speed about three times that of a one-legged 3 cm-by-1.5 cm robot under similar driving conditions, as shown in movie S9. To investigate quantitative details, we show (Fig. 6B) the statistical duty cycles of various postures

between the one-legged and two-legged robots. We found that the prototype two-legged robot had longer aerial duty cycles (75% versus 51%) to boost the running speed.

CONCLUSION

By generalizing several solutions found in animals, we introduce a fast and ultrarobust insect-scale soft robot for potential applications in environmental exploration, structural inspection, information reconnaissance, and disaster relief. Our robot uses the large vibration amplitude and a bouncing gait mechanism to generate a wavelike locomotion near its resonant frequency. Our prototype robot achieved a maximum relative speed of 20 BL/s, which is comparable with those of fast-moving arthropods and is faster than those of currently reported insect-scale robots. Furthermore, the robot can function with a low voltage supply of only 8 V, which demonstrates promise for the further integration of onboard circuits for future untethered operations. The scaling trend from the tested robots shows that miniaturization with higher resonant frequencies could further increase the relative speeds, but precision fabrication, the requirement of powering wires, and untethered operations could be the key challenges in pursuing smaller-scale robots. The working mechanism and structure of the robots presented here also show exceptional robustness in weight-bearing, slope-climbing, and load-carrying performances. The control of the robot's movement direction is another important next step. One simple way to turn would be to assemble two separated electrical domains, as shown in figs. S13 and S14 and movie S10. Driving signals (frequency, amplitude, or phase) of the two domains are controlled independently so that each of them allows different ground reaction forces to turn in the desired direction. By assembling domains with different sizes or shapes, a robot could add further maneuverability. Hence, we hope the proposed insect-scale robot paves a way to pursue fast and robust robots for practical applications.

MATERIALS AND METHODS

The prototype robot consists of a curved unimorph structure and a folded leg assembled with a simple process. The unimorph structure uses an 18- μm -thick PVDF film (PolyK Technologies, LLC) on top as the active layer and a 50- μm -thick PET tape (Gizmo Dorks) at the bottom as the inactive layer (25- μm -thick adhesive silicone and 25- μm -thick PET). The fabrication and assembly processes of a prototype robot are shown in fig. S15. To pattern electrodes on both sides of a PVDF film, we used a 2-mm-thick acrylic board (Soto Laser Cutting) patterned by laser as a shadow mask. The electrode consists of 10-nm-thick Pd and 40-nm-thick Au. The PVDF film was precisely cut with margins along the patterned electrodes by a pro-

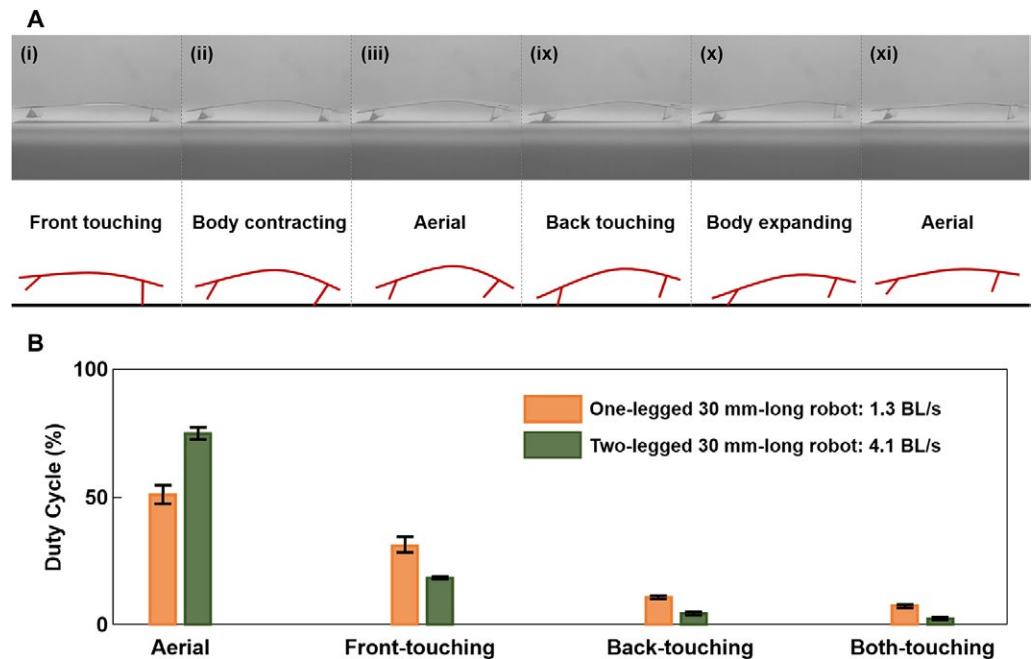


Fig. 6. Galloping-like gait with the design of a two-legged robot. (A) Series of optical images (top) from the high-speed camera to show the galloping strides and their corresponding schematic diagrams (bottom). (B) Comparison of one-legged and two-legged robots in duty cycles in different operation postures. Error bars indicate mean \pm 1 SD. See movie S9.

grammable paper-cutting machine (Silhouette America). Then, the PET tape with the same dimensions as the PVDF film was cut the same way. Two wires (Aluminium/Silicon Alloy, Custom Chip Connections) with a diameter of 25.4 μm were attached by aluminum tape on each side of the electrodes, serving as connections between the robot and external power supply. Next, the PVDF film and the PET tape were laminated together on top of a three-dimensional printed curved mold to form a curved (45°) unimorph structure. Last, a PET tape was adhered to the bottom layer of the robot to serve as the front leg.

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/32/eaax1594/DC1

Section S1. Actuation mechanism of PVDF film and curved unimorph structure

Section S2. Qualitative analysis of the locomotion mechanism

Section S3. Simplified dynamic model for the robot's locomotion

Section S4. Resonant frequency evaluation

Section S5. COT calculation

Fig. S1. Actuating mechanism of PVDF film and curved unimorph structure.

Fig. S2. Locomotion performances inside a tube.

Fig. S3. Conceptual image of the free body diagram: A robot at a both-touching posture.

Fig. S4. Velocity and force analysis for front-leg touchdown.

Fig. S5. System configurations of the simplified dynamic model.

Fig. S6. Gait statistics near fast speed.

Fig. S7. The relationship between robot length and resonant frequency for FEM simulation

results under different boundary conditions compared with that of experimental results.

Fig. S8. Dynamic tests when the robot is clamped at one end.

Fig. S9. Locomotion of a robot under low driving voltage.

Fig. S10. Measurement of electrical parameters.

Fig. S11. COT of select robots (circles) and insects (squares) plotted against their body masses.

Fig. S12. Performance of a 3 cm-by-1.5 cm prototype robot after applying and removing different loads.

Fig. S13. Fabrication processes of a prototype robot with the turning ability.

Fig. S14. Direction control.

Fig. S15. Main fabrication and assembly processes of a prototype robot.

Table S1. Material parameters.

Table S2. Data of 25 combinations of λ/L and β/π as well as their normalized speeds.

Table S3. Data of relative speed versus mass of some animals as well as soft robots and actuators.

Movie S1. Locomotion observation of prototype robot running at the fastest speed.

Movie S2. Posture and position observation of prototype robot.

Movie S3. Locomotion inside tube with different driving frequencies.

Movie S4. Locomotion observation of prototype robot running at slower speeds.

Movie S5. Locomotion of the simplified dynamic model in MATLAB simulation.

Movie S6. Locomotion of prototype robot under low driving voltage.

Movie S7. Robustness, climbing, and carrying loads.

Movie S8. Galloping-like gaits of two-legged robot.

Movie S9. Comparison of locomotion of one-legged robot and two-legged robot.

Movie S10. Robot with two separate electrical domains for turning.

References (64–87)

REFERENCES AND NOTES

1. M. Calisti, G. Picardi, C. Laschi, Fundamentals of soft robot locomotion. *J. R. Soc. Interface* **14**, 20170101 (2017).
2. D. Rus, M. T. Tolley, Design, fabrication and control of soft robots. *Nature* **521**, 467–475 (2015).
3. S. I. Rich, R. J. Wood, C. Majidi, Untethered soft robotics. *Nat. Electron.* **1**, 102–112 (2018).
4. S. Kim, C. Laschi, B. Trimmer, Soft robotics: A bioinspired evolution in robotics. *Trends Biotechnol.* **31**, 287–294 (2013).
5. J. Aguilar, T. Zhang, F. Qian, M. Kingsbury, B. McInroe, N. Mazouchova, C. Li, R. Maladen, C. Gong, M. Travers, R. L. Hatten, H. Choset, P. B. Umbanhowar, D. I. Goldman, A review on locomotion robophysics: The study of movement at the intersection of robotics, soft matter and dynamical systems. *Rep. Prog. Phys.* **79**, 110001 (2016).
6. Y. O. Aydin, J. M. Rieser, C. M. Hubicki, W. Savoie, D. I. Goldman, Physics approaches to natural locomotion: Every robot is an experiment, in *Robotic Systems and Autonomous Platforms* (Woodhead Publishing, 2019), pp. 109–127.
7. A. Rafsanjani, Y. Zhang, B. Liu, S. M. Rubinstein, K. Bertoldi, Kirigami skins make a simple soft actuator crawl. *Sci. Robot.* **3**, eaar7555 (2018).
8. W. Wang, J.-Y. Lee, H. Rodrigue, S.-H. Song, W.-S. Chu, S.-H. Ahn, Locomotion of inchworm-inspired robot made of smart soft composite (SSC). *Bioinspir. Biomim.* **9**, 046006 (2014).
9. K. Jung, J. C. Koo, J.-d. Nam, Y. K. Lee, H. R. Choi, Artificial annelid robot driven by soft actuators. *Bioinspir. Biomim.* **2**, S42–S49 (2007).
10. S. A. Rios, A. J. Fleming, Y. K. Yong, Miniature resonant ambulatory robot. *IEEE Robot. Autom. Lett.* **2**, 337–343 (2017).
11. B. Kim, M. G. Lee, Y. P. Lee, Y. Kim, G. Lee, An earthworm-like micro robot using shape memory alloy actuator. *Sens. Actuators A Phys.* **125**, 429–437 (2006).
12. D. W. Haldane, M. M. Plecnik, J. K. Yim, R. S. Fearing, Robotic vertical jumping agility via series-elastic power modulation. *Sci. Robot.* **1**, eaag2048 (2016).
13. N. Kagawa, H. Kazerooni, Biomimetic small walking machine, in *Proceedings of the 2001 IEEE/IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Como, Italy, 8 to 12 July 2001 (IEEE, 2001).
14. K. L. Hoffman, R. J. Wood, Turning gaits and optimal undulatory gaits for a modular centipede-inspired millirobot, in *Proceedings of the 4th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechanics*, Roma, Italy, 24 to 27 June 2012 (IEEE, 2012).
15. P. Birkmeyer, K. Peterson, R. S. Fearing, DASH: A dynamic 16g hexapedal robot, in *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, 11 to 15 October 2009 (IEEE, 2009).
16. U. Saranlı, M. Buehler, D. E. Koditschek, RHEx: A simple and highly mobile hexapod robot. *Int. J. Rob. Res.* **20**, 616–631 (2016).
17. S. Kim, J. E. Clark, M. R. Cutkosky, iSprawl: Design and tuning for high-speed autonomous open-loop running. *Int. J. Rob. Res.* **25**, 903–912 (2006).
18. A. T. Baisch, O. Ozcan, B. Goldberg, D. Ithier, R. J. Wood, High speed locomotion for a quadrupedal microrobot. *Int. J. Rob. Res.* **33**, 1063–1082 (2014).
19. B. Shin, J. Ha, M. Lee, K. Park, G. H. Park, T. H. Choi, K.-J. Cho, H.-Y. Kim, Hygrobot: A self-locomotive ratcheted actuator powered by environmental humidity. *Sci. Robot.* **3**, eaar2629 (2018).
20. S.-W. Lee, J. H. Prosser, P. K. Purohit, D. Lee, Bioinspired hygromorphic actuator exhibiting controlled locomotion. *ACS Macro Lett.* **2**, 960–965 (2013).
21. Y. Ma, Y. Zhang, B. Wu, W. Sun, Z. Li, J. Sun, Polyelectrolyte multilayer films for building energetic walking devices. *Angew. Chem. Int. Ed.* **123**, 6378–6381 (2011).
22. E. Wang, M. S. Desai, S.-W. Lee, Light-controlled graphene-elastin composite hydrogel actuators. *Nano Lett.* **13**, 2826–2830 (2013).
23. M. Rogóż, H. Zeng, C. Xuan, D. S. Wiersma, P. Wasylczyk, Light-driven soft robot mimics caterpillar locomotion in natural scale. *Adv. Optic. Mater.* **4**, 1689–1694 (2016).
24. S.-J. Park, M. Gazzola, K. S. Park, S. Park, V. Di Santo, E. L. Blevins, J. U. Lind, P. H. Campbell, S. Dauth, A. K. Capulli, F. S. Pasqualini, S. Ahn, A. Cho, H. Yuan, B. M. Maoz, R. Vijaykumar, J.-W. Choi, K. Deisseroth, G. V. Lauder, L. Mahadevan, K. K. Parker, Phototactic guidance of a tissue-engineered soft-robotic ray. *Science* **353**, 158–162 (2016).
25. N. Cheng, G. Ishigami, S. Hawthorne, H. Chen, M. Hansen, M. Telleria, R. Playter, K. Iagnemma, Design and analysis of a soft mobile robot composed of multiple thermally activated joints driven by a single actuator, in *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*, Anchorage, AK, 3 to 8 May 2010 (IEEE, 2010).
26. W. Hu, G. Z. Lum, M. Mastrangeli, M. Sitti, Small-scale soft-bodied robot with multimodal locomotion. *Nature* **554**, 81–85 (2018).
27. R. St. Pierre, W. Gosrich, S. Bergbreiter, A 3D-printed 1mg legged microrobot running at 15 body lengths per second, paper presented at Solid-State Sensors, Actuators, and Microsystems Workshop, Hilton Head, SC, 3 to 7 June 2018.
28. D. Vogtmann, R. S. Pierre, S. Bergbreiter, A 25mg magnetically actuated microrobot walking at > 5 body lengths/sec, *Proceedings of the 2018 IEEE International Conference on Micro Electro Mechanical Systems (MEMS)*, Las Vegas, NV, 22 to 26 January 2017 (IEEE, 2017).
29. B. Goldberg, R. Zufferey, N. Doshi, E. F. Helbling, G. Whittredge, M. Kovac, R. J. Wood, Power and control autonomy for high-speed locomotion with an insect-scale legged robot. *IEEE Trans. Robot. Autom.* **3**, 987–993 (2018).
30. N. Lobontiu, M. Goldfarb, E. Garcia, A piezoelectric-driven inchworm locomotion device. *Mech. Mach. Theory* **36**, 425–443 (2001).
31. H. H. Harii, L. A. Prasetya, S. Foong, G. S. Soh, K. N. Otto, K. L. Wood, A tether-less legged piezoelectric miniature robot using bounding gait locomotion for bidirectional motion, in *Proceedings of the 2016 IEEE International Conference on Robotics and Automation*, Stockholm, Sweden, 16 to 21 May 2016 (IEEE, 2016).
32. Y. Wu, K. Y. Ho, K. Kariya, R. Xu, W. Cai, J. Zhong, Y. Ma, M. Zhang, X. Wang, L. Lin, Pre-curved PVDF/PI unimorph structures for biomimic soft crawling actuators, *Proceedings of the 2018 IEEE International Conference on Micro Electro Mechanical Systems (MEMS)*, Belfast, UK, 21 to 25 January 2018 (IEEE, 2018).
33. P. Xiao, N. Yi, T. Zhang, Y. Huang, H. Chang, Y. Yang, Y. Zhou, Y. Chen, Construction of a fish-like robot based on high performance graphene/PVDF bimorph actuation materials. *Adv. Sci.* **3**, 1500438 (2016).
34. L. Hines, K. Petersen, G. Z. Lum, M. Sitti, Soft actuators for small-scale robotics. *Adv. Mater.* **29**, 1603483 (2017).
35. 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).
36. M. H. Dickinson, C. T. Farley, R. J. Full, M. A. R. Koehl, R. Kram, S. Lehman, How animals move: An integrative view. *Science* **288**, 100–106 (2000).
37. 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).
38. R. J. Bomphrey, T. Nakata, N. Phillips, S. M. Walker, Smart wing rotation and trailing-edge vortices enable high frequency mosquito flight. *Nature* **544**, 92–95 (2017).
39. S. Rubin, M. H.-Y. Young, J. C. Wright, D. L. Whitaker, A. N. Ahn, Exceptional running and turning performance in a mite. *J. Exp. Biol.* **219**, 676–685 (2016).
40. G. C. Wu, J. C. Wright, D. L. Whitaker, A. N. Ahn, Kinematic evidence for superfast locomotory muscle in two species of tenebrionid mites. *J. Exp. Biol.* **213**, 2551–2556 (2010).
41. R. J. Full, M. S. Tu, Mechanics of a rapid running insect: Two-, four- and six-legged locomotion. *J. Exp. Biol.* **156**, 215–231 (1991).
42. B. K. Ahlborn, R. W. Blake, W. Megill, Frequency tuning in animal locomotion. *Fortschr. Zool.* **109**, 43–53 (2006).
43. M. H. Dickinson, M. S. Tu, The function of dipteran flight muscle. *Comp. Biochem. Physiol.* **116**, 223–238 (1997).
44. V. Wahl, S. E. Pfeffer, M. Wittlinger, Walking and running in the desert ant *Cataglyphis fortis*. *J. Comp. Physiol. A* **201**, 645–656 (2015).
45. G.-Y. Gu, J. Zhu, L.-M. Zhu, X. Zhu, A survey on dielectric elastomer actuators for soft robots. *Bioinspir. Biomim.* **12**, 011003 (2017).
46. R. St. Pierre, S. Bergbreiter, Toward autonomy in sub-gram terrestrial robots. *Annu. Rev. Control. Robot. Auton. Syst.* **2**, 231–252 (2019).
47. E. Y. Erdem, Y.-M. Chen, M. Mohebbi, J. W. Suh, G. T. A. Kovacs, R. B. Darling, K. F. Böhringer, Thermally actuated omnidirectional walking microrobot. *J. Microelectromech. Syst.* **19**, 433–442 (2010).
48. T. Ebefor, J. U. Mattsson, E. Kälvesten, G. Stemme, A walking silicon micro-robot, in *Proceedings of the 10th International Conference on Solid-State Sensors and Actuators: Transducers* (IEEE, 1999).
49. W. A. Churaman, L. J. Currano, C. J. Morris, J. E. Rajkowski, S. Bergbreiter, The first launch of an autonomous thrust-driven microrobot using nanoporous energetic silicon. *J. Microelectromech. Syst.* **21**, 198–205 (2012).

50. M. Qi, Y. Zhu, Z. Liu, X. Zhang, X. Yan, L. Lin, A fast-moving electrostatic crawling insect, in *Proceedings of the 2018 IEEE International Conference on Micro Electro Mechanical Systems (MEMS)*, Las Vegas, NV, 22 to 26 January 2017 (IEEE, 2017).
51. S. Hollar, A. Flynn, C. Bellew, K. Pister, Solar powered 10 mg silicon robot, in *Sixteenth Annual International Conference on Micro Electro Mechanical Systems* (IEEE, 2003).
52. J. T. Greenspun, K. Pister, First leaps of an electrostatic inchwormmotor-driven jumping microrobot, in *Hilton Head Solid-State Sensors, Actuators, and Microsystems Workshop*, Hilton Head Island, SC, 3 to 7 June 2018.
53. D. Berrigan, J. R. Lighton, Bioenergetic and kinematic consequences of limblessness in larval Diptera. *J. Exp. Biol.* **179**, 245–259 (1993).
54. T. F. Jensen, I. Holm-Jensen, Energetic cost of running in workers of three ant species, *Formica fusca* L., *Formica rufa* L., and *Camponotus herculeanus* L. (Hymenoptera, Formicidae). *J. Comp. Physiol.* **137**, 151–156 (1980).
55. J. R. B. Lighton, G. A. Bartholomew, D. H. Feener, Energetics of locomotion and load carriage and a model of the energy cost of foraging in the leaf-cutting Ant *Atta colombica* Guer. *Physiol. Zool.* **60**, 524–537 (1987).
56. W. Driesen, A. Rida, J. Breguet, R. Clavel, Friction based locomotion module for mobile MEMS robots, in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2007).
57. L. Reinhardt, R. Blickhan, Level locomotion in wood ants: Evidence for grounded running. *J. Exp. Biol.* **217**, 2358–2370 (2014).
58. D. S. Contreras, D. S. Drew, K. S. J. Pister, First steps of a millimeter-scale walking silicon robot, in *IEEE 19th International Conference on Solid-State Sensors, Actuators and Microsystems (Transducers'17)* (IEEE, 2017).
59. J. H. Fewell, J. F. Harrison, J. R. B. Lighton, M. D. Breed, Foraging energetics of the ant, *Paraponera clavata*. *Oecologia* **105**, 419–427 (1996).
60. M. R. Hirt, W. Jetz, B. C. Rall, U. Brose, A general scaling law reveals why the largest animals are not the fastest. *Nat. Ecol. Evol.* **1**, 1116–1122 (2017).
61. J. Iriarte-Diaz, Differential scaling of locomotor performance in small and large terrestrial mammals. *J. Exp. Biol.* **205**, 2897–2908 (2002).
62. R. Van Damme, T. J. M. Van Dooren, Absolute versus per unit body length speed of prey as an estimator of vulnerability to predation. *Anim. Behav.* **57**, 347–352 (1999).
63. R. M. Alexander, N. J. Dimery, R. F. Ker, Elastic structures in the back and their role in galloping in some mammals. *J. Zool.* **207**, 467–482 (1985).
64. R. M. Alexander, Allometry of the limbs of antelopes (Bovidae). *J. Zool.* **183**, 125–146 (1977).
65. R. M. Alexander, V. A. Langman, A. S. Jayes, Fast locomotion of some African ungulates. *J. Zool.* **183**, 291–300 (1977).
66. F. D. Duncan, R. M. Crewe, A comparison of the energetics of foraging of three species of *Leptogenys* (Hymenoptera, Formicidae). *Physiol. Entomol.* **18**, 372–378 (1993).
67. G. A. Bartholomew, J. R. B. Lighton, G. N. Louw, Energetics of locomotion and patterns of respiration in tenebrionid beetles from the Namib Desert. *J. Comp. Physiol. B* **155**, 155–162 (1985).
68. A. H. Hurlbert, F. Ballantyne, S. Powell, Shaking a leg and hot to trot: The effects of body size and temperature on running speed in ants. *Ecol. Entomol.* **33**, 144–154 (2008).
69. M. Wittlinger, R. Wehner, H. Wolf, The desert ant odometer: A stride integrator that accounts for stride length and walking speed. *J. Exp. Biol.* **210**, 198–207 (2007).
70. C. C. Amaya, P. D. Klawinski, D. R. Formanowicz Jr., The effects of leg autotomy on running speed and foraging ability in two species of wolf spider, (Lycosidae). *Am. Midl. Nat.* **145**, 201–205 (2001).
71. S. Kamoun, S. A. Hogenhout, Flightlessness and rapid terrestrial locomotion in tiger beetles of the Cicindela L. Subgenus Rivacindela van Nidek from saline habitats of Australia (Coleoptera: Cicindelidae). *Coleopt. Bull.* **50**, 221–230 (1996).
72. S. Maeda, Y. Hara, T. Sakai, R. Yoshida, S. Hashimoto, Self-walking gel. *Adv. Mater.* **19**, 3480–3484 (2007).
73. D. Morales, E. Palleau, M. D. Dickey, O. D. Velev, Electro-actuated hydrogel walkers with dual responsive legs. *Soft Matter* **10**, 1337–1348 (2014).
74. N. Tomita, K. Takagi, K. Asaka, Development of a quadruped soft robot with fully IPMC body, in *Proceedings of the 2011 SICE Annual Conference*, Tokyo, Japan, 13 to 18 September 2011 (SICE, 2011).
75. H. Lu, M. Zhang, Y. Yang, Q. Huang, T. Fukuda, Z. Wang, Y. Shen, A bioinspired multilegged soft millirobot that functions in both dry and wet conditions. *Nat. Commun.* **9**, 3944–3950 (2018).
76. S. W. Kwok, S. A. Morin, B. Mosadegh, J.-H. So, R. F. Shepherd, R. V. Martinez, B. Smith, F. C. Simeone, A. A. Stokes, G. M. Whitesides, Magnetic assembly of soft robots with hard components. *Adv. Funct. Mater.* **24**, 2180–2187 (2014).
77. L. Xu, H.-Q. Chen, J. Zou, W.-T. Dong, G.-Y. Gu, L.-M. Zhu, X.-Y. Zhu, Bio-inspired annelid robot: A dielectric elastomer actuated soft robot. *Bioinspir. Biomim.* **12**, 025003 (2017).
78. C. T. Nguyen, H. Phung, T. D. Nguyen, H. Jung, H. R. Choi, Multiple-degrees-of-freedom dielectric elastomer actuators for soft printable hexapod robot. *Sens. Actuators A Phys.* **267**, 505–516 (2017).
79. T. Li, G. Li, Y. Liang, T. Cheng, J. Dai, X. Yang, B. Liu, Z. Zeng, Z. Huang, Y. Luo, T. Xie, W. Yang, Fast-moving soft electronic fish. *Sci. Adv.* **3**, e1602045 (2017).
80. M. Duduta, D. R. Clarke, R. J. Wood, A high speed soft robot based on dielectric elastomer actuators, in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation* (IEEE, 2017), Singapore, 29 May to 3 June 2017.
81. A. M. Hoover, E. Steltz, R. S. Fearing, RoACH: An autonomous 2.4g crawling hexapod robot, in *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2008), Nice, France, 22 to 26 September 2008.
82. A. T. Baisch, P. S. Sreetharan, R. J. Wood, Biologically-inspired locomotion of a 2g hexapod robot, in *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2010), Taipei International Convention Center, Taipei, Taiwan, 18 to 22 October 2010.
83. H.-T. Lin, G. G. Leisk, B. Trimmer, GoQBot: A caterpillar-inspired soft-bodied rolling robot. *Bioinspir. Biomim.* **6**, 026007 (2011).
84. S.-i. Aoshima, T. Tsujimura, T. Yabuta, A miniature mobile robot using piezo vibration for mobility in a thin tube. *J. Dyn. Syst. Meas. Control* **115**, 270–278 (1993).
85. D. W. Haldane, K. C. Peterson, F. L. Garcia Bermudez, R. S. Fearing, Animal-inspired design and aerodynamic stabilization of a hexapedal millirobot, in *Proceedings of the 2013 IEEE International Conference on Robotics and Automation* (IEEE, 2013), Karlsruhe, Germany, 6 to 10 May 2013.
86. D. W. Haldane, R. S. Fearing, Running beyond the bio-inspired regime, in *Proceedings of the 2015 IEEE International Conference on Robotics and Automation* (IEEE, 2015), Washington, USA, 26 to 30 May 2015.
87. C.-H. Hsueh, Modeling of elastic deformation of multilayers due to residual stresses and external bending. *J. Appl. Phys.* **91**, 9652–9656 (2002).

Acknowledgments: We thank the University of California at Berkeley Marvell Nanofabrication Laboratory for the deposition of electrodes. We thank the insightful suggestion from N. Ramirez and the gaits statistics from F. Sui. **Funding:** This work was supported by the Berkeley Sensor and Actuator Center (BSAC), an NSF/Industry/University Research Cooperation Center. Y.W. is supported by a scholarship from China Scholarship Council (CSC) and Tsinghua-Berkeley Shenzhen Institute (TBSI), Tsinghua University. **Author contributions:** Y.W. conceived, designed, and fabricated the prototype robots and experimental setup; performed the experiments; analyzed the data; and wrote the paper. J.K.Y. built dynamic simulations for robot model, performed high-speed video recordings and friction tests, analyzed the results, and wrote the associated method. J.L. assisted the experiment, built the setups, tracked the COM from movies, and analyzed the results. Z.S. built the FEM simulations, assisted the experiment, and analyzed the results. M.Q., Z.L., and X.Y. assisted the experiments and interpreted the data. M.Z. and X.W. reviewed and commented on the paper. R.S.F. commented on the introduction and edited the paper and figures. R.J.F. suggested the two-legged robot, analyzed arthropods' maximum speed, and edited the associated section. J.Z., R.J.F., and L.L. directed the research and revised the paper. **Competing interests:** M.Z., Y.W., and X.W. are inventors on patent application (201910211869.3) submitted by Graduate School at Shenzhen, Tsinghua University, that covers a soft microrobot. **Data and materials availability:** All data needed to evaluate the conclusions of the paper are available in the paper or the Supplementary Materials. Source code is available on GitHub (<https://github.com/justinyim/ISFMURSR-model>).

Submitted 6 March 2019

Accepted 21 June 2019

Published 31 July 2019

10.1126/scirobotics.aax1594

Citation: Y. Wu, J. K. Yim, J. Liang, Z. Shao, M. Qi, J. Zhong, Z. Luo, X. Yan, M. Zhang, X. Wang, R. S. Fearing, R. J. Full, L. Lin, Insect-scale fast moving and ultrarobust soft robot. *Sci. Robot.* **4**, eaax1594 (2019).

COLLECTIVE BEHAVIOR

A robot made of robots: Emergent transport and control of a smarticle ensemble

William Savoie¹, Thomas A. Berrueta², Zachary Jackson¹, Ana Pervan², Ross Warkentin¹, Shengkai Li¹, Todd D. Murphey², Kurt Wiesenfeld¹, Daniel I. Goldman^{1*}

Copyright © 2019
The Authors, some
rights reserved;
exclusive licensee
American Association
for the Advancement
of Science. No claim
to original U.S.
Government Works

Robot locomotion is typically generated by coordinated integration of single-purpose components, like actuators, sensors, body segments, and limbs. We posit that certain future robots could self-propel using systems in which a delineation of components and their interactions is not so clear, becoming robust and flexible entities composed of functional components that are redundant and generic and can interact stochastically. Control of such a collective becomes a challenge because synthesis techniques typically assume known input-output relationships. To discover principles by which such future robots can be built and controlled, we study a model robophysical system: planar ensembles of periodically deforming smart, active particles—smarticles. When enclosed, these individually immotile robots could collectively diffuse via stochastic mechanical interactions. We show experimentally and theoretically that directed drift of such a supersmarticle could be achieved via inactivation of individual smarticles and used this phenomenon to generate endogenous phototaxis. By numerically modeling the relationship between smarticle activity and transport, we elucidated the role of smarticle deactivation on supersmarticle dynamics from little data—a single experimental trial. From this mapping, we demonstrate that the supersmarticle could be exogenously steered anywhere in the plane, expanding supersmarticle capabilities while simultaneously enabling decentralized closed-loop control. We suggest that the smarticle model system may aid discovery of principles by which a class of future “stochastic” robots can rely on collective internal mechanical interactions to perform tasks.

INTRODUCTION

Self-propulsion (1) is a feature of living and artificial systems across scales—from crawling cells to swimming spermatozoa (2), micro- (3) and nanoswimmers (4), running cockroaches (5), and robots (6, 7). It is generally assumed that self-propelling systems require carefully orchestrated integration of many diverse components to perform the seemingly simple behavior of spatial translation. Thus, artificial locomoting systems typically consist of a central controller, a set of actuators and sensors to perform feedback control, and an objective function written in terms of individual system states; such designs have led to progress in machines that robustly and nearly autonomously roll (8), fly (9), and walk (10, 11) in relatively predictable environments.

In contrast to such “deterministically” designed robots, future more “stochastically” designed robots could generate self-propulsion using systems in which a delineation of components is not so clear, such that many redundant and generic elements fluidly interact and collaborate to achieve complex tasks (Fig. 1). Although such designs are potentially advantageous due to wide system reconfigurability and robustness to component damage, it is not yet clear how to build such a system to operate in natural environments. There are several reasons for this, some of which have been anticipated by insights from modular and swarm robotics (12–15), physics of active matter (13, 16–21), amorphous computing (22), and engineering of reliable systems from unreliable components (23).

For one, future stochastically designed robots (and collectives/swarms) may contain so many components (members) (24, 25) that it might be infeasible to carefully arrange and couple the elements to generate coordinated translation or rotation. Further, like in crawling cells where locomotion is generated through cytoskeletal reconfigurations via shape-changing proteins, individual elements

may be task-incapable [e.g., unable to move on their own, unlike in collective robot locomotion via mechanical rectification of individual bristlebots in (26, 27)]. In such situations, the robot’s objective should not depend on deterministic interactions between components but instead on emergent ensemble-level behaviors (25, 28). Thus, it becomes

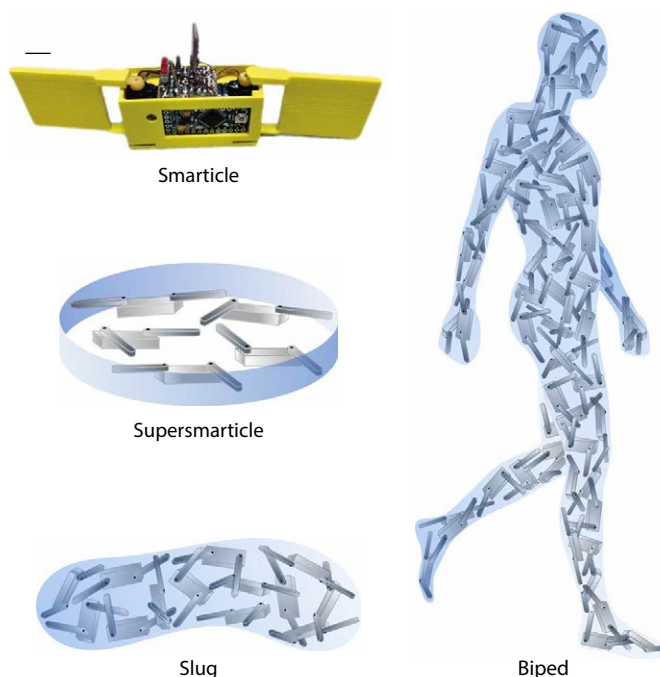


Fig. 1. Stochastic robotic collectives. Future robots may be composed of components whose delineation is neither clear nor deterministic, yet are capable of self-propulsion via the expression of ensemble-level behaviors leading to collective locomotion. In such a robot, groups of largely generic agents may be able to achieve complex goals, as routinely observed in biological collectives.

¹School of Physics, Georgia Institute of Technology, Atlanta, GA 30332, USA. ²Department of Mechanical Engineering, Northwestern University, Evanston, IL 60208, USA. *Corresponding author. Email: daniel.goldman@physics.gatech.edu

a question of leveraging or mitigating the inherent uncertainty of internal component interactions to develop reliable control schemes of the ensemble.

Traditional control synthesis techniques determine which inputs best, and most robustly, enable a system to achieve an objective, such as self-propulsion. It can be challenging enough to find control inputs that realize a well-defined objective in a deterministic system. In the case of a robot composed of robots with highly complex interactions between the system and the environment, and no single configuration of individual components necessary for the robot to achieve locomotion, control synthesis using traditional methods is infeasible. The notion that an ensemble may be able to accomplish a goal independently of the specification of its individual states is incompatible with typical theories of control that assume a central control architecture with full state information.

Here, we sought to discover principles by which a collective can overcome individual locomotor limitations via opportunistic but stochastic mechanical interactions among individuals. Specifically, we studied a simplified robophysical (29) model of controllable, smart, active particles—smarticles—that are immotile but have mutable shape. An enclosed smarticle ensemble—a supersmarticle—however, can self-propel diffusively using interactions arising from the shape modulation of smarticles. Despite stochastic interactions between elements, a supersmarticle is capable of directed motion by selectively inactivating its constituents, which we demonstrate by achieving endogenously steered phototaxis. To understand the supersmarticle diffusion and its dependence on internal mechanical interactions, we developed a model based on kinetic theory. To further explore the ensemble's abilities, we introduce a data-driven algorithm that enables decentralized control synthesis with respect to ensemble properties. Using this algorithm, we modeled supersmarticle dynamics and demonstrate that the ensemble is capable of rich locomotion by taking advantage of more complex control strategies. We validated our algorithm by leading the supersmarticle on a simple path, demonstrating that collective locomotors may be reliably controlled through their ensemble properties despite being composed of stochastically interacting unreliable elements.

RESULTS

Smarticle dynamics

The smarticle's form (Figs. 1 and 2A and Materials and Methods) was inspired by insights from a previous study of rigid, non-active “u-particles” (30), which demonstrated how material properties of an entangling collective could vary via changes in the shape of its constituents. Smarticles, when active, perform a “square gait,” inspired by the dynamics of Purcell's three-link swimmer (31, 32), and depicted in Fig. 2B. Outside of a frictional medium [e.g., (31, 32)], when resting in an orientation where the links' axis of rotation is parallel to the normal of the surface it rests on, smarticles are incapable of translation or rotation (Fig. 2C) over hundreds of oscillation cycles. The moving links rest above the central link and never interact with the surface (movie S1).

Despite their inability to significantly self-propel, an individual robotic smarticle's position and orientation can change as a result of a collision, as shown in Fig. 2D. When viewed as an ensemble, a “cloud” of self-deforming smarticles may display weak cohesion on short time scales, forming a rudimentary collective flocking unit (Fig. 3, A and B, and movie S2). That is, unlike single smarticle ex-

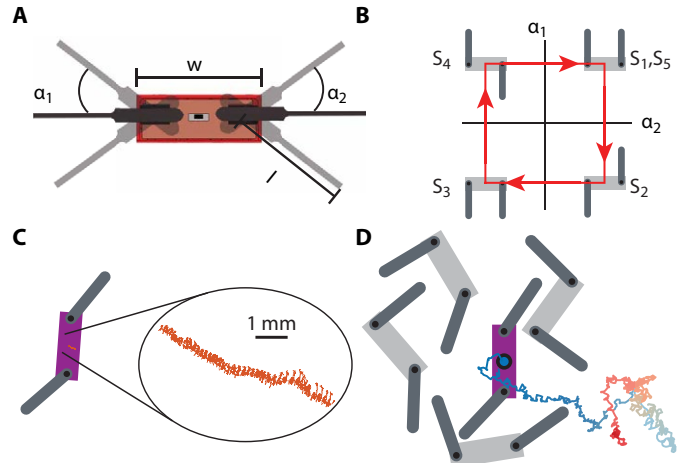


Fig. 2. Smarticle robot dynamics. (A) Top view schematic $w = 5.3$ cm and $l = 4.9$ cm. (B) Clockwise (CW) square gait, with key configurations enumerated. (C) Drift of a single smarticle on a flat surface, executing a square gait over 38τ . (D) Tracked trajectory of a smarticle within an ensemble of other self-deforming smarticles; color gradient (blue to red) represents passage of time 47τ , with $\tau = 1.6$ s.

periments, we found that the center of mass (CoM) of the cloud could diffuse over scales comparable with the size of a smarticle (see fig. S1).

Because of interactions between smarticles, the area fraction ϕ typically decreased over time as in Fig. 3C. Here, $\phi = nA_p/A_c$, where A_c is the area of convex hull of the smarticles (bodies and arms) in the cloud (Fig. 3A), n is the number of smarticles in the system, and A_p is the area of a single smarticle. The decrease in ϕ was not always monotonic; in certain trials, increases in ϕ occurred (Fig. 3C). Despite purely repulsive interactions at surfaces, smarticles could both repel and attract their neighbors (see Fig. 3B). This emerges from the particle geometry: Collisions between particles in concave configurations can generate attraction via arm entanglement (30).

After sufficient time, the cloud's mobility slowed as smarticles separated and no longer interacted strongly. We quantified collective mobility using the cloud's “granular temperature,” defined as $\langle V^2 \rangle = 1/3 \langle \langle v^2 \rangle_n - \langle \bar{v} \rangle_n^2 \rangle_N$, where $v = \sqrt{\dot{x}^2 + \dot{y}^2} + (2l + w)\sqrt{\dot{\theta}^2}$ sums the translational plus rotational velocity of n smarticles of length l and width w and averages over N experiments (33, 34). On long time scales, a single experiment's V^2 may approach the noise floor (seen in Fig. 3D) (35), thereby limiting the flocking ability. For this study, we determined the noise floor empirically by measuring the granular temperature of non-interacting smarticles.

Supersmarticle dynamics

Given the correlation between ϕ (Fig. 3C) and $\langle V^2 \rangle$ (Fig. 3D), we hypothesized that we could sustain locomotion on longer time scales by constraining ϕ of the collective. To achieve this, we confined five smarticles within a ring, creating what we call a supersmarticle. Each smarticle in the supersmarticle starts at a random phase in the square gait and continuously performs a square gait inside an unanchored rigid ring of radius $R = 9.6$ cm and variable mass $m \in [9.8 \text{ g}, 207 \text{ g}]$ (Fig. 4A). It takes $t = 225\tau$ (where $\tau = 1.6$ s) before two smarticles are $>\pi$ out of phase. The ring diameter was chosen such that ϕ and $\langle V^2 \rangle$ remained high, yet there was enough area that jamming was rare and self-resolvable.

The ring confinement maintained ϕ at approximately the value observed at the initiation of the cloud trials (see Fig. 3A). Similarly,

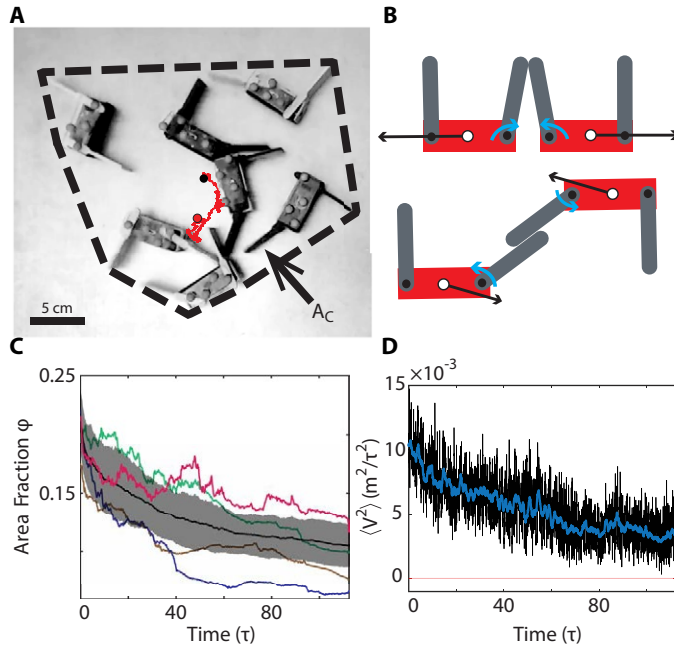


Fig. 3. Smarticle cloud dynamics. (A) Snapshot of experimental trial, with the dashed line indicating the boundary of the convex hull area A_C . The cloud's CoM trajectory is illustrated in red, beginning at the black dot and ending at the red dot. Experiment ran for 113τ . (B) Center link trajectory of geometrically repulsive (top) and attractive interactions (bottom). (C) Evolution of ϕ averaged over 20 trials (black, with gray shaded region representing a single standard deviation); four individual trials are shown in blue, red, green, and brown lines. (D) $\langle V^2 \rangle$ averaged over 20 cloud trials. Raw data are in black; the blue line is moving mean with a window size of 1τ . Red line and area surrounding it represent mean value and single standard deviation of $\langle V^2 \rangle$ noise of an experiment lasting 10τ with seven moving, but non-interacting, smarticles. Here, gait period $\tau = 1.6$ s.

$\langle V^2 \rangle$ of the supersmarticle system (Fig. 4B) remained at approximately the value found at the highest ϕ in cloud trials (Fig. 3, C and D). This led to persistent diffusive transport of the supersmarticle (Fig. 4B). Within the ring, individual smarticles displayed complex interactions, often displacing an amount comparable with, or greater than, the displacement of the ring itself, as shown in Fig. 4C.

Tracking the supersmarticle's motion for a ring of mass $m = 68$ g (movie S3) revealed no correlation between final angular position between trials (e.g., Fig. 4D). We used $\sigma^2(t)$, the mean square displacement (MSD) of the ring, to characterize the motion: $\sigma^2(t) = \langle x^2(t) \rangle - \langle x(t) \rangle^2$, where $\sigma^2(t) \propto t^\gamma$ and γ specifies the type of diffusion the system undergoes. The supersmarticle exhibited different types of diffusion—normal ($0 < \gamma \leq 1$), superdiffusive ($1 < \gamma < 2$), and even approximately ballistic ($\gamma \geq 2$)—depending on the time scale observed (36). The short time scale regime was consistent with $\gamma = 1$ (Fig. 4E), indicating normal diffusive motion. The long time scale regimes were best fit with $\gamma \approx 1.45$ representing directionally invariant superdiffusive motion.

We found that, if a smarticle near the boundary maintained a fixed straight shape or became “inactive” (Fig. 5A), the supersmarticle displayed directed drift on short time scales (movie S4). Because the angular position of the inactive smarticle around the ring was not fixed, drift in a constant direction was not observed on longer time scales in the laboratory frame (see fig. S2). When trajectories were examined in the frame of the inactive smarticle (Fig. 5B), the bias in

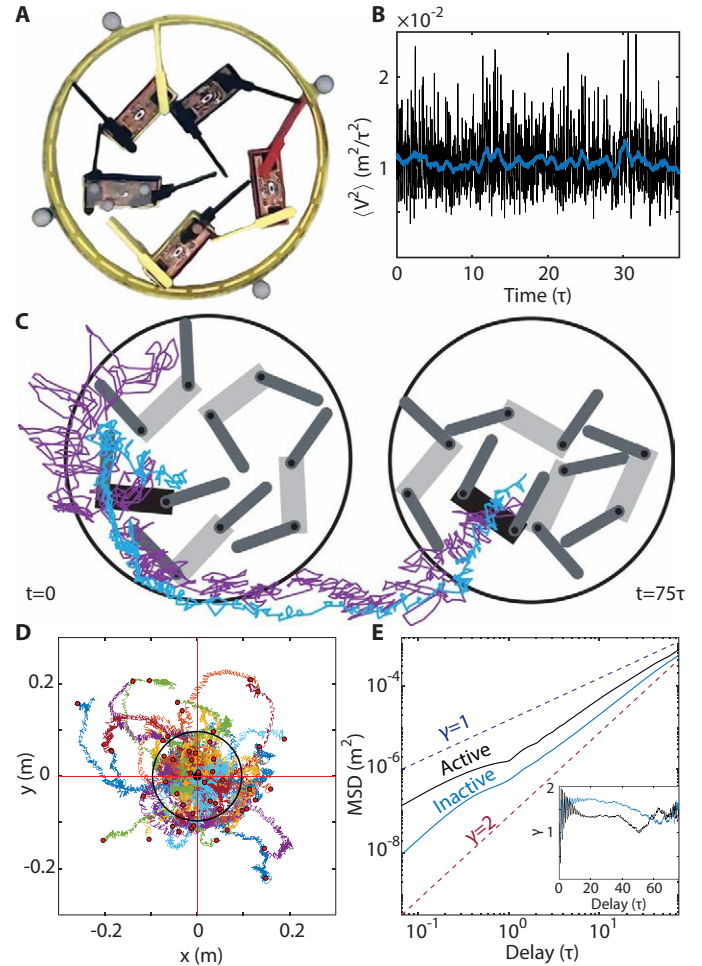


Fig. 4. Collective confined diffusion. (A) Supersmarticle top view; ring inner radius is 9.6 cm. The four gray spheres were used to track the motion of the ring. (B) Granular temperature of five active smarticles confined in a ring; black line is raw data over 10 trials, and blue is a moving window mean with a window size of 1τ . (C) Trajectories, from an experiment, of a smarticle inside the ring (purple), and the ring's center of geometry (blue). (D) Experimental tracks of ring trajectory for 50 trials; $m_{\text{ring}} = 68$ g. The black circle represents the size and initial position of the ring. (E) MSD averaged over 50 and 80 trials, for the active and inactive systems, respectively, all lasting 75τ . The inset shows the average change of γ for active (black) and inactive (blue) systems. The oscillation seen in both the MSD and γ is related to the gait period τ (where $\tau = 1.6$ s).

drift toward the inactive smarticle became clear. In Fig. 5C, the cumulative displacements are shown in the continuously rotating frame attached to the center link of the inactive smarticle such that

$$S_{\parallel}(t) = \sum_{i=0}^t \Delta \vec{s}^i \cdot \hat{R}_{\parallel}^i \quad \text{and} \quad S_{\perp}(t) = \sum_{i=0}^t \Delta \vec{s}^i \cdot \hat{R}_{\perp}^i.$$

Here, $\Delta \vec{s}^i$ denotes the vector connecting the center of the ring at consecutive instants in time, and $\hat{R}_{\parallel}^i, \hat{R}_{\perp}^i$ are the unit vectors specifying the local frame (Fig. 5B). As with the fully active supersmarticle, the dynamics of the supersmarticle containing an inactive smarticle were superdiffusive and, at short time scales, approximately ballistic, as indicated by $\gamma \approx 2$.

Statistical model

To understand the supersmarticle diffusion and its dependence on internal mechanical interactions, we developed a model based on kinetic theory. Formally, the average displacement of the ring would be

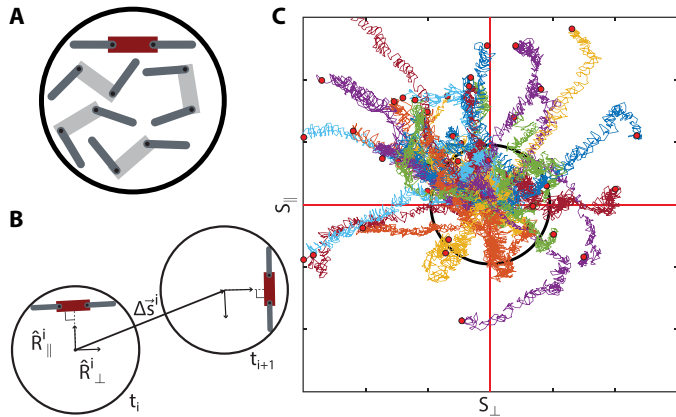


Fig. 5. Biasing supersmarticle transport. (A) Supersmarticle schematic, with the inactive smarticle in red. (B) Supersmarticle trajectory frame transformation from laboratory to inactive smarticle frame. (C) Supersmarticle trajectories rotated into the laboratory frame, where axes are now the perpendicular and parallel components to the frame of the inactive particle.

given by $\langle \Delta \vec{R} \rangle = \int P(\vec{\Omega}) \Delta \vec{R}(\vec{\Omega}) d\vec{\Omega}$, where $\vec{\Omega}$ represents the microstate (i.e., position, orientation, and heading) of the supersmarticle constituents immediately before a collision, and $\Delta \vec{R}$ is the ring displacement due to an individual collision. The resulting mean ring displacement, $\langle \Delta \vec{R} \rangle$, is then computed by integrating displacements due to individual collisions over the microstate probability distribution. However, because we do not have access to the detailed relation of $\vec{\Omega}$ to the complicated smarticle-smarticle and smarticle-ring collisions, this calculation is intractable, demanding the development of a simplified ensemble model.

We imagine active smarticles rattling inside the ring and colliding against the ring and the inactive smarticle. The role of the active smarticle in the supersmarticle is simplified as simple contacts around the ring. The contacts are abstracted as nudges (Fig. 6A). Each nudge has a uniform probability to act in any direction. As a result of symmetries in the system geometry, we may partition the space of $\vec{\Omega}$ into six distinct types of collisions. Six collisions arise from two independent factors: whether or not the inactive smarticle is in contact with the ring and which of three regions of the ring the active smarticle contacts (denoted by roman numerals in Fig. 6B). Each of these individual collision types generates a unique response on the collective system.

This simplification leads to a model with two random variables. The first is an angle Θ that represents the direction of an individual nudge and takes a value between 0 and 2π . The second is a binary variable β that represents whether or not the inactive robot is in contact with the ring. These variables together represent the various microstates Ω of the system. Depending on the value of the random variables, an individual nudge can move either the ring or both the ring and the inactive particle. Then, focusing on the movement of the ring, we must determine R_A and R_B , which are the distances that a nudge will displace the ring when moving only the ring, and when moving both ring and inactive smarticle, respectively. The possibilities for ring movement are summarized in table S1.

With R_A and R_B , we can describe the simplified model of the supersmarticle ensemble in expectation, which we decompose into parallel and perpendicular components (see Fig. 5B). Denoting the

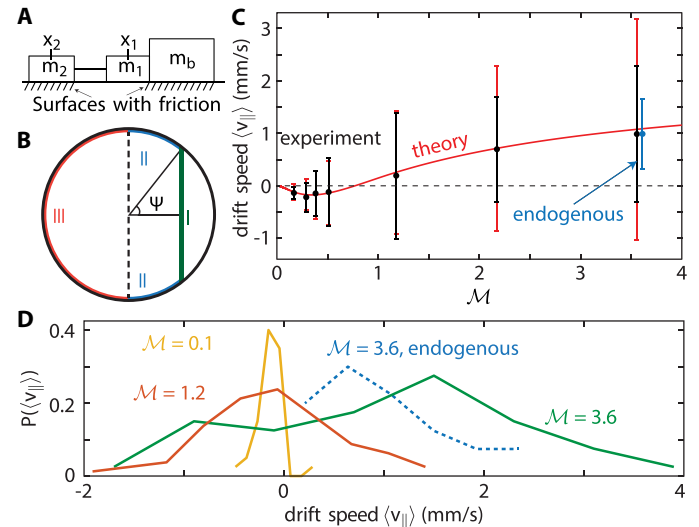


Fig. 6. Statistical model of supersmarticle transport. (A) Schematic of the theoretical collision model. (B) Three regions with distinct collision types for the theory as described in the text. (C) Theoretical (red) and experimental (black and blue) data for velocity versus mass ratio \mathcal{M} , showing mean and standard deviation. The blue data point is offset in \mathcal{M} for visibility and represents an experiment where the inactive particle was endogenously chosen by light (see text) for 40 trials. (D) Distributions of drift speed probabilities for \mathcal{M} regimes.

proportion of time that the inactive smarticle is in contact with the ring as λ , the frequency of nudges as f , the amount of time the supersmarticle has been moving as T , the inactive smarticle's angular diameter as Ψ (see Fig. 6B), and treating each nudge as an independent event, the expected component of the velocity of the ring along \hat{R}_{\parallel} of the inactive smarticle is

$$\langle v_{\parallel} \rangle = (f/\pi) [\lambda(R_A - R_B)(1 - \sin(\Psi)) - (1 - \lambda)R_A \sin(\Psi)] \quad (1)$$

The perpendicular component of the ring velocity is simplified substantially because we know by symmetry that $\langle \Delta \vec{R} \rangle = \sum_{i=1}^6 \langle \Delta \vec{R}_i \rangle = \langle \Delta \hat{R}_{\parallel} \rangle$. This is to say that the mean displacement of the ring averaged over all distinct collisions is in the parallel direction; hence

$$\langle v_{\perp} \rangle = 0 \quad (2)$$

However, the variance along this direction is non-zero. The corresponding variances to the expected parallel and perpendicular velocities, $\text{Var}[v_{\parallel}]$ and $\text{Var}[v_{\perp}]$, are detailed fully in Materials and Methods.

Last, to completely specify this model, we must calculate R_A and R_B . To this end, we determine the relationship between the mass of the ring and the distance it moves from a nudge by modeling the active smarticles as pistons pushing on a sliding mass (see Materials and Methods). The predictions resulting from this model are plotted in Fig. 6C and fig. S3.

The theory correctly predicts the supersmarticle's drift speed relative to that calculated experimentally as $\langle v_{\parallel} \rangle = S_{\parallel}(T)/T$ with $T = 75\tau$. The theory predicts that the direction of $\langle v_{\parallel} \rangle$ will reverse for a large enough ring mass. Directionality depends on the mass ratio $\mathcal{M} = m_{\text{smarticle}}/m_{\text{ring}}$ between the inactive smarticle and the ring, with reversal at a critical value of $\mathcal{M} \approx 0.8$. To test this prediction, we conducted experiments for a series of different ring masses. The

results are summarized in Fig. 6C. The theory closely predicts the mean velocity, including direction reversal (movie S5). Although the theory predicts $\langle v_{\perp} \rangle = 0$ (fig. S3), we observed slight discrepancies, particularly at larger \mathcal{M} . We attribute these discrepancies in variance to correlations between collisions, whereas the theory assumes that collisions are independent.

The model elucidates the physics governing the dependence of $\langle v_{\parallel} \rangle$ on \mathcal{M} , as a function of the ensemble's internal mechanical interactions. Consider first the high- \mathcal{M} limit. The three collision types involving the (light) ring but not the (heavy) inactive smarticle dominate the net motion (see Fig. 6B). Both of the forward collisions (region 2) are of this type, as is one rearward (region 3) collision, resulting in a relatively large positive $\langle \Delta \hat{R}_{\parallel} \rangle$. Conversely, in the low- \mathcal{M} limit, five of six collision types give rise to nearly equal magnitude ring displacements, the exception being the forward collision (region 1) of the active smarticle with the inactive smarticle when the latter is not in contact with the ring, in which case the ring displacement is exactly zero. This deficit in the forward-directed ring displacement results in a (small) negative value for the net displacement.

The close agreement between theory and experiment for the drift speed velocity is perhaps unexpected: With only $N = 4$ active smarticles, it is not clear that a purely statistical kinetic theory approach should work. The theory overestimates the observed fluctuations in Fig. 6C, an indication of substantial correlations in the smarticle swarm collisions, which will be the focus of future work. Yet, despite this incongruity with the variance, the derived theoretical model is still capable of generating the directed motion of the ensemble observed experimentally in the frame of the inactive smarticle.

Directing a phototaxing supersmarticle

On the basis of the intuition gained from the kinetic model, we programmed smarticles to inactivate when light detected from its photo-sensor exceeded a threshold. When illuminated at low angles (i.e., in the plane of the smarticle light sensors), photoinactivated smarticles occlude light from neighbors further from the source (Fig. 7A, inset), creating a situation similar to that analyzed in the previous section. The inactivated smarticle occludes the light from its neighbors: The straightening and resulting occlusion of light serves as a decentralized and stigmergic directive. The inactive smarticle is affecting the motion of the ring by affecting the motion of the remaining smarticles. This decentralized strategy has been used in previous swarm robotic collectives to generate group movement and transport without requiring explicit communication between agents (37, 38).

However, we found that rather than regulate the angular location of an individual inactive smarticle, the static light source induced a switching sequence of inactive smarticles, leading to supersmarticle phototaxis. Because collisions in the ring can cause an inactive smarticle's position to shift, when an inactive smarticle was dislocated from its lighted position, it switched to the active state. Consequently, an active smarticle could then be nudged into a position to receive enough light to become inactive. Hence, the supersmarticle phototactic drift was via endogenous steering, that is, where smarticle immobilization was spontaneously selected for without external feedback (see Fig. 7A and movie S6).

The endogenously forced system drifted in a preferred direction in the laboratory frame with a similar $\langle v_{\parallel} \rangle$ to that of the non-light-driven system (Fig. 6C), whose drift was only observable in the frame of the inactive smarticle. This is remarkable given the complex switching dynamics of the inactive smarticle: For example, depending on distance and orientation relative to the light, it was possible for multiple

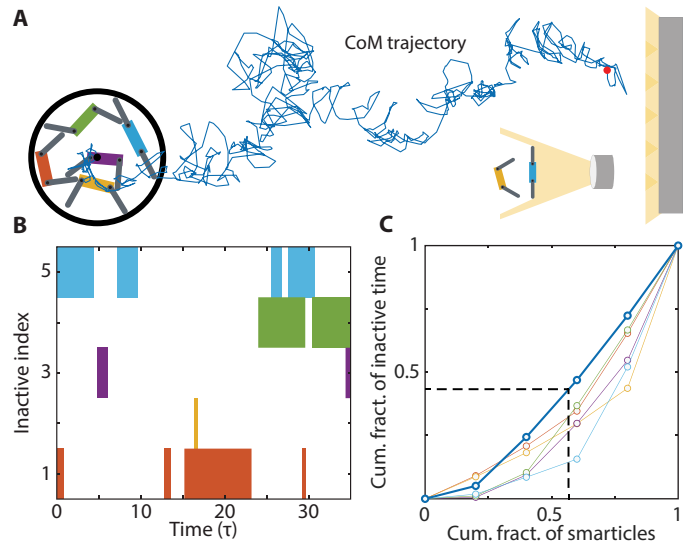


Fig. 7. Endogenous supersmarticle phototaxis. (A) Trajectory from an experiment of a self-directed (endogenously forced) photophilic supersmarticle tracking a static light source (movie S6). Inset: Schematic showing how a smarticle in the straight configuration can occlude light from smarticle behind it. (B) Map depicting when and which smarticles endogenously inactivate. (C) Lorenz plots detailing general equality of smarticle inactivity over a 25-min endogenous trial consisting of five separate excursions in different directions (see movie S6). Over the complete trial, we found $G = 0.21$, as shown in the bolded line. The unbolded lines are the Lorenz curves for the five separate excursions, where we found $G = [0.28, 0.4, 0.42, 0.34, 0.49]$.

smarticles to be simultaneously inactive as depicted in Fig. 7B. Moreover, the rotational symmetry of the supersmarticle allows one to infer that if the supersmarticle can translate in one direction, it should be able to translate in another direction by selecting different inactive smarticles.

To further highlight the supersmarticle systems indifference toward which smarticle is inactive, we plotted the cumulative distribution of total inactivity time in the form of a Lorenz curve (Fig. 7C). The curve presents the share of inactive time covered by the smarticle spending the least time being inactive (39). The shape of the Lorenz curve reflects the inequality in the distribution of the inactivity times of the smarticles: The more concave the curve, the more unequal the distribution. To characterize the Lorenz curve, we introduce the Gini coefficient, G , defined as the ratio of the area between the Lorenz curve and a line representing equality to the total area under the line of equality (39, 40). A value of 0 represents equality, and a value of 1 is perfect inequality. In a single endogenous experiment lasting 25 min where the light changed directions five times (see movie S6), we found that $G = 0.21$ (see bolded line in Fig. 7C). The Lorenz curve using data from all trials shows that 57% of the inactivity time was accounted for by 43% of the smarticles.

By considering each of the five excursions independently, the Gini coefficient and Lorenz curve can change markedly (see unbolded lines in Fig. 7C). For singular excursions, certain smarticles may remain in the inactive position for extended periods of time with a static light source, thus giving the Lorenz curves high values of inequality. This is a result of aforementioned correlations that can happen in smarticle collisions. On shorter time scales, the correlations may incorrectly lead one to believe a smarticle hierarchy exists; however, on long time scales, it becomes apparent that the smarticles are indeed commutable.

Thus, although crude, the endogenously drifting supersmarticle result demonstrates that the collective can perform a task/behavior (40) such that locomotor control of the system is decentralized and offloaded completely to mechanical interactions (41) in response to highly structured environmental signals (i.e., smarticle inactivity patterns).

Discovering emergent control authority

Most control synthesis techniques from the past six decades rely on a deterministic understanding of actuation and its effect on system states (42). But to create an organized system out of disorganized components, it is necessary to understand what the collective can accomplish as a function of uncertain subsystem interactions (43). To enable the discovery of control strategies for collective locomotors, control must be synthesized with respect to ensemble properties rather than individual states.

We expect an ensemble's control authority to be an emergent property rather than intrinsic. To address this, we introduce the notion of a candidate control signal to hypothesize actuation mechanisms based on broken symmetries in the system (44). Using control signals, we can take a system with symmetry—and associated conserved quantities—and apply control to break the symmetry, thereby asserting authority over otherwise conserved quantities. When actuation mechanisms are unknown, symmetry breaking can be used as a way to hypothesize candidate control signals contributing to a system's emergent control authority.

Given a candidate control signal, we apply a nonparametric, unsupervised learning algorithm, dynamical system segmentation (DSS), to discover ensemble-level behaviors in relation to the signal. DSS extracts distinct system dynamics from the interactions of internal states and, when present, the effect of candidate control signals on states (45). Initially, the algorithm constructs a set of system models over sequential windows in time—each locally capturing the net effect of interactions between internal states and candidate control signals on the ensemble dynamics.

In constructing the set of models, we represent elements of the set using the Koopman operator, \mathcal{K} —an infinite-dimensional linear operator describing measure-preserving nonlinear dynamical systems through the evolution of observables (46). This choice of model is important because the Koopman operator does not explicitly require state information to describe the evolution of the system. Instead, the operator depends on observables, which may be any time-varying sensor measurement or property of the system such as mass, volume, or temperature. Formally, observables g are real-valued functions drawn from an infinite-dimensional Hilbert space, \mathbb{H} , that take measurements as their argument. The evolution of an observable through the infinite-dimensional Koopman operator is

$$\mathcal{K}g(x_k) = g(F(x_k)) = g(x_{k+1}) \quad (3)$$

where $\mathcal{K}: \mathbb{H} \rightarrow \mathbb{H}$ acts directly on the observables in the function space. We approximate \mathcal{K} in finite dimensions with a data-driven operator $K: \mathcal{H} \rightarrow \mathcal{H}$ by choosing a basis for some subspace $\mathcal{H} \subset \mathbb{H}$ and applying least-squares optimization to compute K . The finite-dimensional operator, K , is then an $N \times N$ matrix for a given choice of N -dimensional basis.

The algorithm then condenses the set of Koopman operators into a set of non-redundant exemplars by applying nonparametric clustering (47) directly onto the set of models—where each element is itself a matrix. The resulting compressed set contains all unique

dynamical system behaviors observed in the dataset of sequential measurements. DSS achieves this without assuming how many behaviors the system exhibits—an important property when the cardinality is generally unknown a priori.

The output of DSS is a set of distinct, yet related, ensemble behaviors represented by a probabilistic graphical model $\mathcal{G} = (\mathbb{K}, \mathbb{E})$. The graph's node set is specified by the compressed set of system behaviors, and its edge set is determined empirically by the transitions observed in the training dataset. In this model, the ensemble behaviors, each of which is a deterministic description of the ensemble dynamics at a given configuration, are random variables whose joint probabilities are in \mathcal{G} . We refer to the information encoded by \mathcal{G} as the system's behavioral patterns.

Although the literature of learning control is evolving rapidly, existing methods are not immediately well suited for a problem as ill-posed as discovering emergent control authority. For one, reliably designing payoffs to reward emergence may not be possible, making it difficult to directly apply most reinforcement learning approaches (48–50). Moreover, techniques in inverse reinforcement learning, such as learning from demonstration (51) and imitation learning (52), typically suffer from a lack of generalizability, limiting the use of learned behaviors. DSS avoids these pitfalls by directly analyzing distinct system behaviors and constructing a predictive model from these subsystem interactions, leading to a generalizable model. In addition, DSS is extremely data efficient, which is critical given that emergence is typically a rare phenomenon.

Decentralized control of supersmarticles

On the basis of observations made in previous sections, we know the switching sequence of inactive smarticles (Fig. 7B) is causally related to system behavior via the breaking of symmetries in the internal collision distribution of the supersmarticle. We used this sequence as a candidate control signal and modeled its effect on supersmarticle dynamics with DSS. By taking data from a single endogenous phototaxis demonstration (such as Fig. 7A), we instantiated two separate models of the supersmarticle dynamics with DSS—one with candidate control information and one without—and studied their respective behavioral patterns. The basis functions used in DSS were selected on the basis of their ability to represent information about the relative locations of inactive smarticles within the ring and their effect on the motion of the supersmarticle (see the Supplementary Materials for details).

The resulting graphical models are shown in Fig. 8 (A to C) along with a graph constructed from the observed smarticle switching sequence. We refer to the switching sequence of inactive smarticles observed in the given experimental trial as the experimental inactivity patterns, shown in Fig. 8A as a graph. Each node in the graph of Fig. 8A represents a unique combination of smarticles that were experimentally observed to be simultaneously inactive, and the number in each node refers to the unique label of the respective inactive smarticles. For example, in Fig. 8A, the green node labeled 4 indicates that, at some point in the experimental trial, smarticle number 4 was inactive, and subsequently smarticle 1 also became inactive, changing the supersmarticle to the dark blue node labeled 1 and 4. We used the complexity (i.e., graph complexity) of the experimental inactivity patterns graph to represent an estimated baseline complexity of supersmarticle behaviors. If the candidate control signal was causally related to the ensemble dynamics, the system should have responded to actuation leading to behaviors identifiable by DSS.

Without candidate control information, DSS is unable to identify a set of behaviors explaining the observed drift, as seen in the nominal

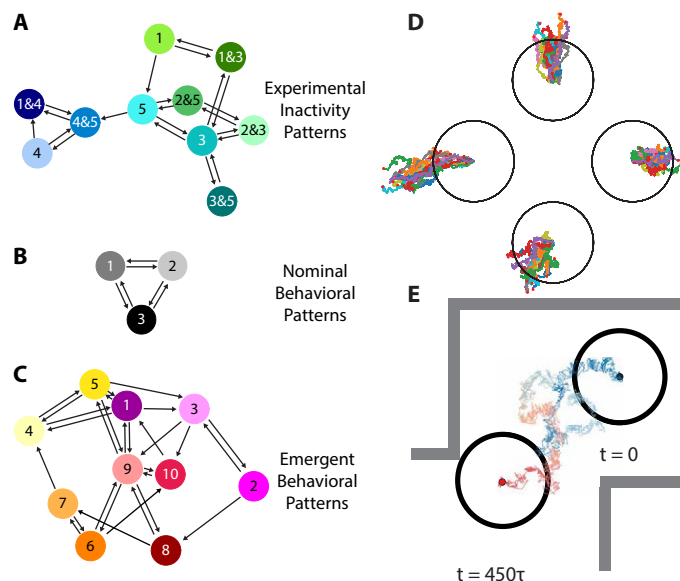


Fig. 8. Comparison between observed and numerically generated behavioral patterns. (A) Model shown is a graphical representation of a single inactive smarticle switching sequence observed in a phototactic experiment (such as Fig. 7A). We extracted graphical models using the DSS algorithm from the same experiment with and without candidate control information, shown in (C) and (B), respectively. (D) Simulated supersmarticle trajectories predict that the ensemble is capable of movement anywhere in the plane while receiving exogenous feedback from an external controller. (E) Experimental trajectory of a photophilic supersmarticle in which the system was exogenously steered through a maze by an experimenter (movie S7), validating the simulation's predictions. The trajectory evolves in time from blue to red, and the black circles represent the initial and final ring configurations.

behavioral patterns in Fig. 8B. However, when candidate control information is incorporated, DSS extracts a set of emergent behavioral patterns of equal cardinality to the inactive smarticle switching sequence (Fig. 8C), where numbered nodes correspond to distinct behaviors identified by DSS. We note that there is no one-to-one correspondence between identified behaviors and elements of the candidate control signal. This is to be expected because the ensemble dynamics are not exclusively determined by the candidate control signal and are also driven by uncertainty, which is captured by the probabilistic transitions in the generated graphs.

Through endogenous steering, we showed that the supersmarticle is capable of directed transport toward a fixed objective. To improve on the controllability of the ensemble shown in previous sections, we looked for alternative locomotion strategies in simulation by synthesizing control to directly manipulate the algorithmically extracted ensemble behaviors. Previous work in control of interconnected stochastic systems has shown that integral control can often be a simple and robust strategy (53). However, due to the discontinuous nature of supersmarticle inactivations, model-based control is necessary to directly optimize system actions.

We designed a simple decentralized model-predictive controller that greedily searches for inactive smarticle switching sequences to alter the ensemble's behavior—as determined by its DSS model—to achieve a collective objective. The collective objective was expressed as a quadratic cost on the position of the supersmarticle centroid with respect to a desired goal location in the world frame. The supersmarticle centroid was calculated via distributed consensus in a fully

connected topology (see the Supplementary Materials for additional details) (54). We conducted four sets of Monte Carlo simulations (40 trials each), over distinct goal locations—left, right, up, and down—with randomized initial conditions for a duration of $T = 75\tau$ per trial (where $\tau = 1.6\text{s}$). The objectives were always equidistant and located directly vertically or horizontally from the initial conditions.

The resulting trajectories shown in Fig. 8D were exogenously steered in the world frame by the independent decision-making of individual smarticles. The simulation results confirm the symmetry-based theoretical predictions: The ensemble should be capable of locomotion anywhere in the plane via exogenously selected smarticle inactivations—even when we train the model using only a single trajectory moving in a single direction. The supersmarticle provides a test case for whether DSS can detect emergent behavior and whether DSS (or related algorithms) should be used in more general settings where symmetry-based inference about control authority is not possible. By allowing an external source of feedback to inactivate smarticles, the decentralized controller manipulated ensemble behaviors to achieve more complicated goals than the model trained on, thereby predicting entirely emergent behavior. As a result of the generalizability of our machine learning model, we are able to make predictions and control the supersmarticle in entirely new settings, thereby harnessing the system's emergent control authority to accomplish brand new tasks. We note that extending the smarticle hardware to accommodate for the proposed control algorithm can be done in a practical and computationally efficient way and will be explored in future work.

On the basis of the simulation results, we experimentally validate the exogenous controllability predictions by guiding the supersmarticle through a simple maze using external feedback from an experimenter with a light source (Fig. 8E). Here, the experimenter is capable of directing the supersmarticle by freely shining a light source onto the ensemble, thereby using more complex inactivity sequences to achieve locomotion anywhere in the plane, just as the proposed decentralized control scheme did. Although the supersmarticle was provided with external guidance, it was able to achieve directed transport without state information or specifying individual objectives for its constituents. All movement was directly emergent from morphological computations in response to environmental signals (55). Hence, by framing the discovery of emergent control authority as a learning problem, we were able to hypothesize and model unconventional actuation leading to expressive controllable motion.

CONCLUSION

Inspired by a future in which a class of task-capable robots could be formed from myriad redundant and task-incapable components, we have created a primitive “robot made of robots” that can perform rudimentary phototaxis, despite none of its components—smarticles—having locomotor capabilities. A generic statistical model accurately captures the fundamental drift dynamics, rationalizing how the supersmarticle can sense an aspect of its environment—light—and use this to endogenously steer itself via asymmetric inactivation of individuals. Further, through the introduction of novel machine learning techniques, we constructed a data-driven model of the ensemble, which enabled discovery and proof-of-control alternatives for generating exogenous steering when agents are capable of computation.

We emphasize that, unlike other mobile robots, the supersmarticle displays phototaxis without a central processor or dedicated motor

components. The key ingredient, and what differentiates our collective from other robot swarms and locomoting collectives, is that our system is made of components that have very low control authority—they cannot locomote individually—and are highly unpredictable—they create emergent behavior from the highly complex interactions of their internal degrees of freedom. Although such a system might seem idiosyncratic, we note that it bears similarities to cascades of conformational changes in the nanomachines that regulate many cellular processes: proteins (56). Given the ubiquity of such processes in these tiny machines, we posit that our model smarticle system could provide inspiration for the generation of substantially more complex task-capable ensembles like those pictured in Fig. 1, including perhaps three-dimensional (3D) collective locomotors and manipulators. Enabling robots to flexibly reconfigure to collectively perform tasks in the presence of environmental noise and individual component malfunction or degradation (23) could enhance robustness in robot swarms across scales, from intravenous delivery (57–59) to search and rescue (60). Further, insights from collective robophysical systems (15, 61, 62) could elucidate principles by which biological collectives [like slime molds (63)] perform tasks in complex natural environments.

MATERIALS AND METHODS

Smarticle robots

Each smarticle's outer shell and arms, or outer links, are 3D printed. The arms are controlled by HD-1440A servomotors to a precision of $<1^\circ$ and with an accuracy of $\pm 6^\circ$. All processing and servomotor control is handled by an Arduino Pro Mini 328 (3.3 V/8 MHz model), which allows smarticles to be programmed to deform to specific configurations and gaits, where we define gaits as periodic trajectories in the configuration space (see Fig. 2B). When assembled, each single smarticle has a mass $m = 34.8 \pm 0.5$ g. The system is powered by a 3.7-V, 150-mAh, 30-C LiPo battery (Venom; Rathdrum, ID) enabling hours of testing. Smarticle positions and orientation were tracked using an infrared video recording hardware/software suite (OptiTrack; Corvallis, OR). All experiments were conducted on a 60 cm-by-60 cm aluminum plate leveled flat to $<0.1^\circ$.

Smarticle experiments

The Gini coefficient (G) is a statistical measure derived from the shape of the Lorenz curve. A value of $G = 0$ represents a situation of perfect equality, or in the case of the supersmarticle, all smarticles spent an equal amount of time being inactive. Conversely, a value of $G = 1$ is a maximally unequal trial, or one where only a single smarticle was inactive over the course of the experiment.

Statistical model

Below, we detail the full form of the variance of v_{\parallel}

$$\begin{aligned} \text{Var}[v_{\parallel}] = & -(f/4\pi^3 T) [\pi^2 \sin(2\Psi) (R_A^2 - R_B^2 \lambda) \\ & + R_A^2 ((-4\lambda^3 + 2\lambda^2 + 2)\Psi + \pi^3 ((4\lambda^2 + 2)/(\pi^2) + \lambda - 2 + (2\Psi)/\pi)) \\ & - 4R_A R_B \lambda (2\lambda + 1) (-\lambda\Psi + \Psi + \pi) \\ & - R_B^2 \lambda (6(\lambda - 1)\lambda\Psi - 6\pi\lambda + 2\pi^2\Psi + \pi^3) \\ & - 2(-\lambda\Psi + \Psi\pi) (R_A - R_B \lambda) (4\lambda(R_A - R_B) \sin(\Psi) \\ & + \cos(2\Psi) (R_A - R_B \lambda))] \end{aligned}$$

Furthermore, the full form of variance of v_{\perp} is shown below

$$\begin{aligned} \text{Var}[v_{\perp}] = & -(f/4\pi T) [R_B^2 \lambda (\pi + 2\Psi) - R_A^2 (\pi(\lambda - 2) + 2\Psi) \\ & + (R_A^2 - R_B^2 \lambda) \sin(2\Psi)] \end{aligned}$$

To calculate the values of R_A and R_B , we must start with masses m_1 and m_2 such that the relative distance between them, $x_1 - x_2$, is specified by the actuation of the smarticles. The first mass, m_1 , represents the arm of a smarticle, and m_2 represents the body. The mass of the boundary they push on is m_b . Both m_2 and m_b have friction between them and the surface they are sitting on. This is shown in Fig. 6A. On the basis of this model, we arrive at the following equations of motion

$$F_2 - f_s = m_2 \ddot{x}_2$$

$$f_s - F_b = (m_b + m_1) \ddot{x}_1 \quad (4)$$

where F_2 and F_b are the friction force on m_2 and m_b , respectively, and f_s is the force between m_1 and m_2 . By specifying $x_1 - x_2 = A_0 \sin(\omega t + \gamma)$, $F_2 = (m_2 + 2m_1)g\mu$, and $F_b = m_b g\mu$, these equations can be integrated to find how far m_b moves. Then, by plugging in for m_b —the mass of just the ring—and the mass of the ring and the inactive smarticle, we can find R_A and R_B , respectively, as well as $\langle v_{\parallel, \perp} \rangle$ and $\text{Var}[v_{\parallel, \perp}]$.

Dynamical system segmentation

The DSS algorithm is composed of three primary subroutines: (i) the calculation of Koopman operators over sequential windows of time via least-squares optimization, (ii) nonparametric clustering over the space of Koopman operators to determine unique system behaviors, and (iii) training a supervised learning model [e.g., support vector machine (SVM)] to learn relationships between system behaviors and construct the complete probabilistic graphical model. In the following sections, we expand on these subroutines, and a full outline of the algorithm can be found in the Supplementary Materials.

Koopman operators

The DSS algorithm first requires calculating finite-dimensional Koopman operators over sequential windows of the dataset. Although there are many ways to frame Koopman operator synthesis, we implement it as a least-squares optimization (64). Given a choice of nonlinear basis function $\psi(x)$ and a data sample $X = \{x_1, \dots, x_M\}$, we can formulate the Koopman operator synthesis problem as solving

$$\min_K \frac{1}{2} \sum_{k=1}^{M-1} \|\psi(x_{k+1}) - K\psi(x_k)\|^2$$

This optimization has a closed form solution of the following form

$$K = AG^\dagger$$

where \dagger denotes the Moore-Penrose pseudoinverse, and the individual matrix components are

$$G = \frac{1}{M} \sum_{k=1}^{M-1} \psi(x_k) \psi(x_k)^T$$

$$A = \frac{1}{M} \sum_{k=1}^{M-1} \psi(x_{k+1}) \psi(x_k)^T$$

Nonparametric clustering

Given a set of Koopman operators, DSS looks for distinct dynamical behaviors by applying nonparametric clustering directly onto the set of operators. In particular, we apply hierarchical density-based spatial clustering of applications with noise (HDBSCAN) (47), which is a nonparametric clustering algorithm that specializes in problems subject to noisy and sparse measurements. By using HDBSCAN, we were able to discern distinct behaviors from the set of Koopman operators. From these clustered classes, we constructed class exemplars as a means of creating a set of distinct Koopman operators corresponding to observed system behaviors.

Supervised model

Once DSS has compiled a condensed set of exemplar behaviors, the algorithm must then determine the dependencies between each behavior and the states of the system. To this end, we trained an SVM. We did this by using the clustered class labels from HDBSCAN to label the state-space data. Then, using this newly labeled dataset, we trained a soft-margin SVM that assigned discerned behaviors to state observations. The SVM, in conjunction with the condensed set of exemplar behaviors, gave rise to the probabilistic graphical model, where the dynamics of the system are described by stochastically shifting Koopman operators.

SUPPLEMENTARY MATERIALS

robotics.sciencemag.org/cgi/content/full/4/34/eaax4316/DC1

Text

Fig. S1. Unrotated center of mass trajectory of the smarticle cloud.

Fig. S2. Unrotated trajectories of the supersmarticle.

Fig. S3. Theoretical and experimental data for the perpendicular component of the supersmarticle drift speed.

Table S1. List of all six different types of collisions in the theoretical model.

Table S2. List of all parameters used in the theoretical model.

Algorithm S1. Dynamical system segmentation.

Movie S1. Individual smarticle performing square gait.

Movie S2. Smarticle cloud: Seven active smarticles.

Movie S3. Supersmarticle: $M=0.51$, five active smarticles.

Movie S4. Supersmarticle: $M=0.51$, one inactive, four active smarticles.

Movie S5. Supersmarticle: $M=3.6$, one inactive, four active smarticles.

Movie S6. Supersmarticle: $M=3.6$, endogenous phototaxing.

Movie S7. Supersmarticle: $M=3.6$, exogenous phototaxing.

REFERENCES AND NOTES

1. A. Shapere, F. Wilczek, Geometry of self-propulsion at low Reynolds number. *J. Fluid Mech.* **198**, 557–585 (1989).
2. J. Teran, L. Fauci, M. Shelley, Viscoelastic fluid response can increase the speed and efficiency of a free swimmer. *Phys. Rev. Lett.* **104**, 038101 (2010).
3. E. Lauga, T. R. Powers, The hydrodynamics of swimming microorganisms. *Rep. Prog. Phys.* **72**, 096601 (2009).
4. J. Yu, B. Wang, X. Du, Q. Wang, L. Zhang, Ultra-extensible ribbon-like magnetic microswarm. *Nat. Commun.* **9**, 3260 (2018).
5. D. I. Goldman, T. S. Chen, D. M. Dudek, R. J. Full, Dynamics of rapid vertical climbing in cockroaches reveals a template. *J. Exp. Biol.* **209**, 2990–3000 (2006).
6. J. Dai, H. Faraji, C. Gong, R. L. Hatton, D. I. Goldman, H. Choset, Geometric swimming on a granular surface. *Robot. Sci. Syst.* (2016).
7. R. D. Maladen, Y. Ding, P. B. Umbanhowar, A. Kamor, D. I. Goldman, Mechanical models of sandfish locomotion reveal principles of high performance subsurface sand-swimming. *J. R. Soc. Interface* **8**, 1332–1345 (2011).
8. E. Frazzoli, M. A. Dahleh, E. Feron, Real-time motion planning for agile autonomous vehicles. *J. Guid. Control Dynam.* **25**, 116–129 (2002).
9. R. Mahony, V. Kumar, P. Corke, Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *IEEE Robot. Autom. Mag.* **19**, 20–32 (2012).
10. M. Raibert, K. Blankespoor, G. Nelson, R. Playter, BigDog, the rough-terrain quadruped robot. *IFAC Proc. Vol.* **41**, 10822–10825 (2008).
11. S. Kuindersma, R. Deits, M. Fallon, A. Valenzuela, H. Dai, F. Permenter, T. Koolen, P. Marion, R. Tedrake, Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot. *Autonom. Robots* **40**, 429–455 (2016).
12. M. Yim, W.-m. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, G. S. Chirikjian, Modular self-reconfigurable robot systems [Grand challenges of robotics]. *IEEE Robot. Autom. Mag.* **14**, 43–52 (2007).
13. M. Rubenstein, A. Cornejo, R. Nagpal, Programmable self-assembly in a thousand-robot swarm. *Science* **345**, 795–799 (2014).
14. W. J. Butera, V. M. Bove Jr., “Programming a paintable computer,” thesis, Massachusetts Institute of Technology, School of Architecture and Planning (2002).
15. S. Li, R. Batra, D. Brown, H.-D. Chang, N. Ranganathan, C. Hoberman, D. Rus, H. Lipson, Particle robotics based on statistical mechanics of loosely coupled components. *Nature* **567**, 361–365 (2019).
16. E. Pinçe, S. K. P. Velu, A. Callegari, P. Elahi, S. Gigan, G. Volpe, G. Volpe, Disorder-mediated crowd control in an active matter system. *Nat. Commun.* **7**, 10907 (2016).
17. J. K. Parrish, W. M. Hammer, *Animal Groups in Three Dimensions: How Species Aggregate (Psychiatry and Medicine)* (Cambridge Univ. Press, 1997).
18. F. J. Ndele, T. Surrey, A. C. Maggs, S. Leibler, Self-organization of microtubules and motors. *Nature* **389**, 305–308 (1997).
19. J. Palacci, S. Sacanna, A. P. Steinberg, D. J. Pine, P. M. Chaikin, Living crystals of light-activated colloidal surfers. *Science* **339**, 936–940 (2013).
20. A. P. Petroff, X.-L. Wu, A. Libchaber, Fast-moving bacteria self-organize into active two-dimensional crystals of rotating cells. *Phys. Rev. Lett.* **114**, 158102 (2015).
21. L. Gioni, M. Hawley-Weld, L. Mahadevan, Swarming, swirling and stasis in sequestered bristle-bots. *Proc. R. Soc. Lond. A Math. Phys. Eng. Sci.* **469**, 20120637 (2013).
22. H. Abelson, D. Allen, D. Coore, C. Hanson, G. Homsy, T. F. Knight Jr., R. Nagpal, E. Rauch, G. J. Sussman, R. Weiss, Amorphous computing. *Commun. ACM* **43**, 74–82 (2000).
23. E. F. Moore, C. E. Shannon, Reliable circuits using less reliable relays. *J. Franklin Inst.* **262**, 191–208 (1956).
24. M. Z. Miskin, K. J. Dorsey, B. Bircan, Y. Han, D. A. Muller, P. L. McEuen, I. Cohen, Graphene-based bimorphs for micron-sized, autonomous origami machines. *Proc. Natl. Acad. Sci. U.S.A.* **115**, 466–470 (2018).
25. S. C. Goldstein, J. D. Campbell, T. C. Mowry, Programmable matter. *Computer* **38**, 99–101 (2005).
26. Z. Gao, H. Li, X. Chen, H. P. Zhang, Using confined bacteria as building blocks to generate fluid flow. *Lab Chip* **15**, 4555–4562 (2015).
27. A. Deblais, T. Barois, T. Guerin, P. H. Delville, R. Vaudaine, J. S. Lintuvuori, J. F. Boudet, J. C. Baret, H. Kellay, Boundaries control collective dynamics of inertial self-propelled robots. *Phys. Rev. Lett.* **120**, 188002 (2018).
28. J. Bourgeois, S. C. Goldstein, Distributed intelligent MEMS: Progresses and perspectives. *IEEE Syst. J.* **9**, 1057–1068 (2015).
29. J. Aguilar, D. I. Goldman, Robophysical study of jumping dynamics on granular media. *Nat. Phys.* **12**, 278 (2016).
30. E. M. Purcell, Life at low Reynolds number. *Am. J. Phys.* **45**, 3–11 (1977).
31. R. L. Hatton, Y. Ding, H. Choset, D. I. Goldman, Geometric visualization of self-propulsion in a complex medium. *Phys. Rev. Lett.* **110**, 078101 (2013).
32. N. Gravish, S. V. Franklin, D. L. Hu, D. I. Goldman, Entangled granular media. *Phys. Rev. Lett.* **108**, 208001 (2012).
33. G. D. Cody, D. J. Goldfarb, G. V. Storch Jr., A. N. Norris, Particle granular temperature in gas fluidized beds. *Powder Technol.* **87**, 211–232 (1996).
34. J. Bougie, S. J. Moon, J. B. Swift, H. L. Swinney, Shocks in vertically oscillated granular layers. *Phys. Rev. E* **66**, 051301 (2002).
35. Because experiments were terminated at a specific time, rather than when all interactions stopped, the cloud granular temperature in Fig. 3D does not reach the average noise floor.
36. E. R. Weeks, H. L. Swinney, Anomalous diffusion resulting from strongly asymmetric random walks. *Phys. Rev. E* **57**, 4915–4920 (1998).
37. J. Chen, M. Gauci, W. Li, A. Kolling, R. Groß, Occlusion-based cooperative transport with a swarm of miniature mobile robots. *IEEE Trans. Robot.* **31**, 307–321 (2015).
38. R. Gross, M. Dorigo, Towards group transport by swarms of robots. *Int. J. Bio-Inspired Comput.* **1**, 1–13 (2009).
39. P. Tenczar, C. C. Lutz, V. D. Rao, N. Goldenfeld, G. E. Robinson, Automated monitoring reveals extreme interindividual variation and plasticity in honeybee foraging activity levels. *Anim. Behav.* **95**, 41–48 (2014).
40. J. Aguilar, D. Monastkova, V. Linevich, W. Savoie, B. Dutta, H.-S. Kuan, M. D. Bettegton, M. A. D. Goodisman, D. I. Goldman, Collective clog control: Optimizing traffic flow in confined biological and robophysical excavation. *Science* **361**, 672–677 (2018).
41. K. Zahedi, N. Ay, Quantifying morphological computation. *Entropy* **15**, 1887–1915 (2013).
42. K. J. Astrom, R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers* (Princeton Univ. Press, 2008).
43. S. Basu, Y. Gerchman, C. H. Collins, F. H. Arnold, R. Weiss, A synthetic multicellular system for programmed pattern formation. *Nature* **434**, 1130–1134 (2005).

44. J. E. Marsden, T. S. Ratiu, *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems* (Springer Publishing Company, Incorporated, 2010).
45. T. A. Berrueta, A. Pervan, K. Fitzsimons, T. D. Murphey, Dynamical system segmentation for information measures in motion. *IEEE Robot. Autom. Lett.* **4**, 169–176 (2019).
46. B. O. Koopman, Hamiltonian systems and transformation in Hilbert space. *Proc. Natl. Acad. Sci. U.S.A.* **17**, 315–318 (1931).
47. L. McInnes, J. Healy, S. Astels, hdbSCAN: Hierarchical density based clustering. *J. Open Source Softw.* **2**, (2017).
48. D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, D. Hassabis, A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* **362**, 1140–1144 (2018).
49. V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, D. Hassabis, Human-level control through deep reinforcement learning. *Nature* **518**, 529–533 (2015).
50. L. P. Kaelbling, M. L. Littman, A. W. Moore, Reinforcement learning: A survey. *J. Artif. Intell. Res.* **4**, 237–285 (1996).
51. B. D. Argall, S. Chernova, M. Veloso, B. Browning, A survey of robot learning from demonstration. *Robot. Auton. Syst.* **57**, 469–483 (2009).
52. P. Abbeel, A. Y. Ng, Apprenticeship learning via inverse reinforcement learning, in *Proceedings of the International Conference on Machine Learning*, Banff, Alberta, Canada, 4 to 8 July 2004 (ACM, 2004).
53. N. Napp, S. Burden, E. Klavins, Setpoint regulation for stochastically interacting robots. *Autonom. Robots* **30**, 57–71 (2011).
54. R. Olfati-Saber, J. A. Fax, R. M. Murray, Consensus and cooperation in networked multi-agent systems. *Proc. IEEE* **95**, 215–233 (2007).
55. R. M. Fuchslin, A. Dzyakanchuk, D. Flumini, H. Hauser, K. J. Hunt, R. H. Luchsinger, B. Reller, S. Scheidegger, R. Walker, Morphological computation and morphological control: Steps toward a formal theory and applications. *Artif. Life* **19**, 9–34 (2013).
56. J. S. Parkinson, G. L. Hazelbauer, J. J. Falke, Signaling and sensory adaptation in *Escherichia coli* chemoreceptors: 2015 update. *Trends Microbiol.* **23**, 257–266 (2015).
57. J. Li, B. E.-F. de Ávila, W. Gao, L. Zhang, J. Wang, Micro/nanorobots for biomedicine: Delivery, surgery, sensing, and detoxification. *Sci. Robot.* **2**, eaam6431 (2017).
58. S. Li, Q. Jiang, S. Liu, Y. Zhang, Y. Tian, C. Song, J. Wang, Y. Zou, G. J. Anderson, J. Y. Han, Y. Chang, Y. Liu, C. Zhang, L. Chen, G. Zhou, G. Nie, H. Yan, B. Ding, Y. Zhao, A DNA nanorobot functions as a cancer therapeutic in response to a molecular trigger in vivo. *Nat. Biotechnol.* **36**, 258–264 (2018).
59. S. Yim, M. Sitti, Shape-programmable soft capsule robots for semi-implantable drug delivery. *IEEE Trans. Robot.* **28**, 1198–1202 (2012).
60. C. Li, A. O. Pullin, D. W. Haldane, H. K. Lam, R. S. Fearing, R. J. Full, Terradynamically streamlined shapes in animals and robots enhance traversability through densely cluttered terrain. *Bioinspir. Biomim.* **10**, 046003 (2015).
61. J. Aguilar, T. Zhang, F. Qian, M. Kingsbury, B. McInroe, N. Mazouchova, C. Li, R. Maladen, C. Gong, M. Travers, R. L. Hatton, H. Choset, P. B. Umbanhowar, D. I. Goldman, A review on locomotion robophysics: The study of movement at the intersection of robotics, soft matter and dynamical systems. *Rep. Prog. Phys.* **79**, 110001 (2016).
62. Y. O. Aydin, J. M. Rieser, C. M. Hubicki, W. Savoie, D. I. Goldman, *Robotic Systems and Autonomous Platforms* (Elsevier, 2019), pp. 109–127.
63. J. T. Bonner, *Cellular Slime Molds* (Princeton Univ. Press, 2015), vol. 2127.
64. M. Williams, I. Kevrekidis, C. Rowley, A data-driven approximation of the Koopman operator: Extending dynamic mode decomposition. *J. Nonlinear Sci.* **25**, 1307–1346 (2015).

Acknowledgments: We thank M. Caveney for experimental assistance and D. Randall, A. W. Richa, A. Zangwill, J. M. Rieser, and P. Umbanhowar for helpful discussions. **Funding:** Support for W.S., S.L., R.W., and D.I.G. was provided by NSF PoLS-0957659, PHY-1205878, DMR-1551095, and ARO W911NF-13-1-0347. Funding support for Z.J. and K.W. was provided by PHY-1205878. Funding for T.A.B. and T.D.M. was provided by NSF CBET-1637764, and funding for A.P. was provided by an NSF GSRF. **Author contributions:** W.S. designed and performed all experiments with the help of R.W. and S.L.; Z.J. designed the theoretical model; and T.A.B. and A.P. designed the modeling algorithm and controller, and performed numerical simulations. W.S., Z.J., T.A.B., K.W., T.D.M., and D.I.G. designed the research and wrote the paper; D.I.G. guided overall research program. **Data and materials availability:** All files needed for constructing smarticles can be found in <https://github.com/wsavoie/ArduinoSmarticle>, and all files needed for applying the DSS algorithm to the supersmarticle, along with representative data samples, can be found in https://github.com/MurpheyLab/Smarticles_SR19.

Submitted 22 March 2019

Accepted 31 July 2019

Published 18 September 2019

10.1126/scirobotics.aax4316

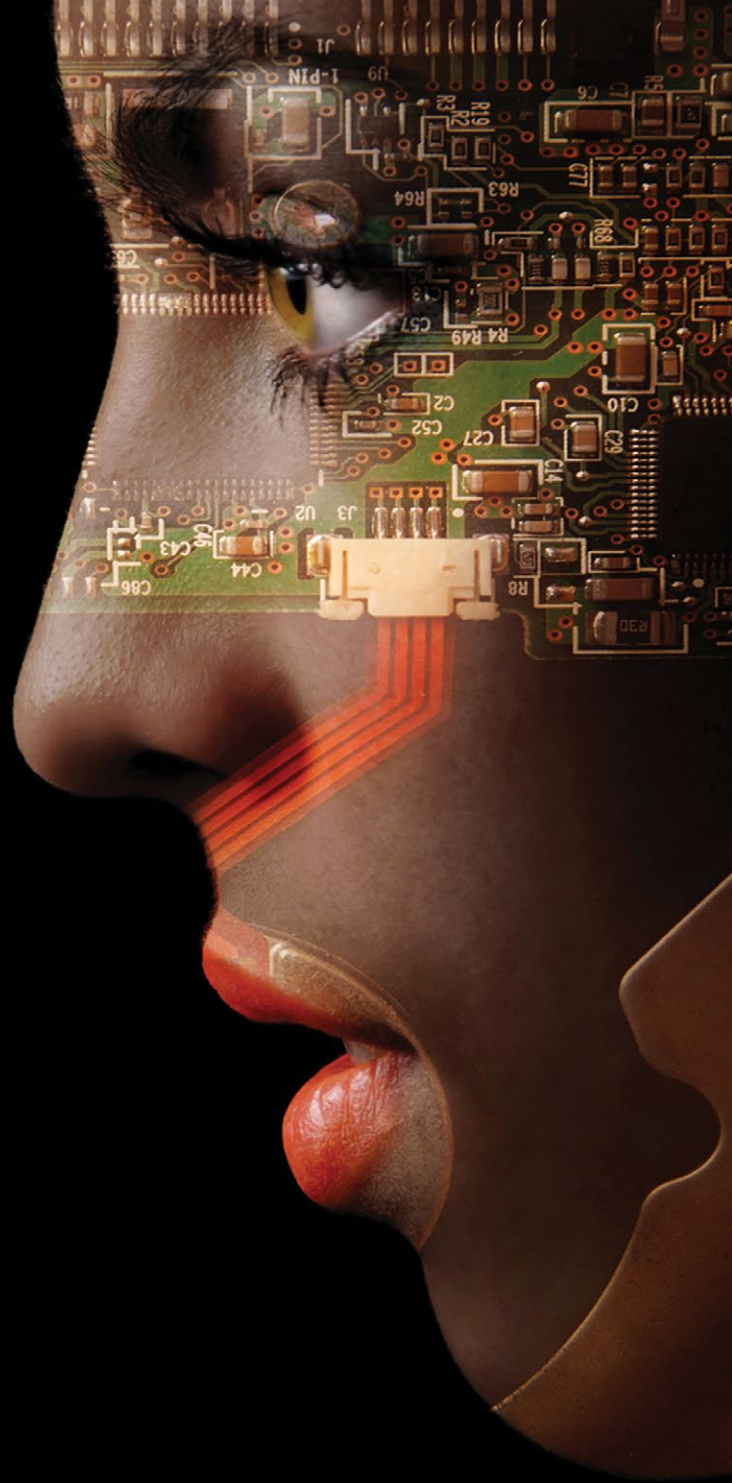
Citation: W. Savoie, T. A. Berrueta, Z. Jackson, A. Pervan, R. Warkentin, S. Li, T. D. Murphey, K. Wiesenfeld, D. I. Goldman, A robot made of robots: Emergent transport and control of a smarticle ensemble. *Sci. Robot.* **4**, eaax4316 (2019).

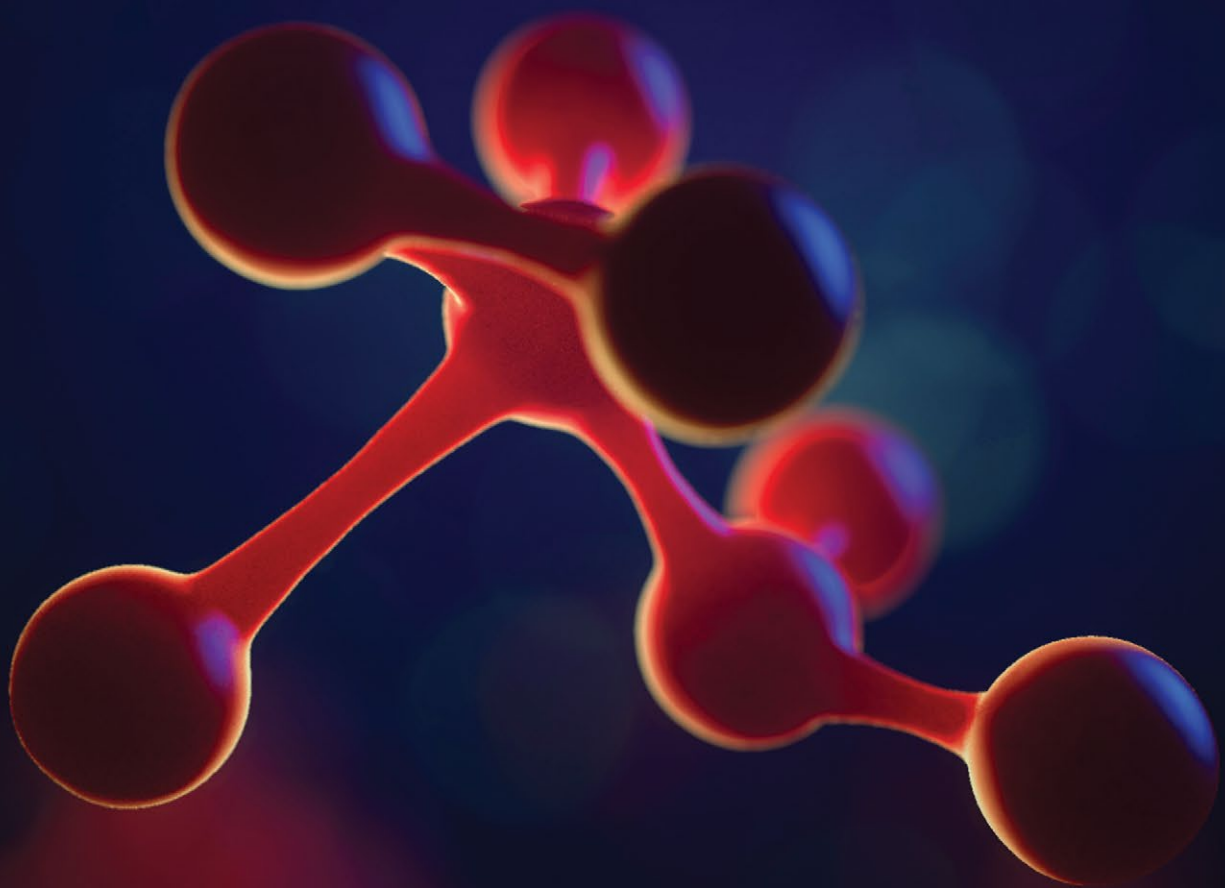
ScienceRobotics.org

DOESN'T YOUR RESEARCH DESERVE THE BEST READERS?

Submit your research:
cts.ScienceMag.org

Science
Robotics
AAAS





Publish your research in the *Science* family of journals

The *Science* family of journals (*Science*, *Science Advances*, *Science Immunology*, *Science Robotics*, *Science Signaling*, and *Science Translational Medicine*) are among the most highly-regarded journals in the world for quality and selectivity. Our peer-reviewed journals are committed to publishing cutting-edge research, incisive scientific commentary, and insights on what's important to the scientific world at the highest standards.

Submit your research today!

Learn more at **[ScienceMag.org/journals](https://www.sciencemag.org/journals)**

Science
JOURNALS AAAS