

# Robots that can adapt like animals

Antoine Cully<sup>1,2</sup>, Jeff Clune<sup>3</sup>, Danesh Tarapore<sup>1,2,†</sup> & Jean-Baptiste Mouret<sup>1,2,4,5,6,‡</sup>

Robots have transformed many industries, most notably manufacturing<sup>1</sup>, and have the power to deliver tremendous benefits to society, such as in search and rescue<sup>2</sup>, disaster response<sup>3</sup>, health care<sup>4</sup> and transportation<sup>5</sup>. They are also invaluable tools for scientific exploration in environments inaccessible to humans, from distant planets<sup>6</sup> to deep oceans<sup>7</sup>. A major obstacle to their widespread adoption in more complex environments outside factories is their fragility<sup>6,8</sup>. Whereas animals can quickly adapt to injuries, current robots cannot ‘think outside the box’ to find a compensatory behaviour when they are damaged: they are limited to their pre-specified self-sensing abilities, can diagnose only anticipated failure modes<sup>9</sup>, and require a pre-programmed contingency plan for every type of potential damage, an impracticality for complex robots<sup>6,8</sup>. A promising approach to reducing robot fragility involves having robots learn appropriate behaviours in response to damage<sup>10,11</sup>, but current techniques are slow even with small, constrained search spaces<sup>12</sup>. Here we introduce an intelligent trial-and-error algorithm that allows robots to adapt to damage in less than two minutes in large search spaces without requiring self-diagnosis or pre-specified contingency plans. Before the robot is deployed, it uses a novel technique to create a detailed map of the space of high-performing behaviours. This map represents the robot’s prior knowledge about what behaviours it can perform and their value. When the robot is damaged, it uses this prior knowledge to guide a trial-and-error learning algorithm that conducts intelligent experiments to rapidly discover a behaviour that compensates for the damage. Experiments reveal successful adaptations for a legged robot injured in five different ways, including damaged, broken, and missing legs, and for a robotic arm with joints broken in 14 different ways. This new algorithm will enable more robust, effective, autonomous robots, and may shed light on the principles that animals use to adapt to injury.

Current damage recovery in deployed robots typically involves two phases: self-diagnosis, followed by selection of the best pre-designed contingency plan<sup>9,13–15</sup>. Such self-diagnosing robots are expensive, because self-monitoring sensors are expensive, and are difficult to design, because robot engineers cannot foresee every possible situation: this approach often fails either because the diagnosis is incorrect<sup>13,14</sup> or because an appropriate contingency plan is not provided<sup>15</sup>.

Injured animals respond differently (see Fig. 1a): they learn by trial and error how to compensate for damage (for example, learning which limp minimizes pain)<sup>16,17</sup>. Similarly, trial-and-error learning algorithms could allow robots to creatively discover compensatory behaviours without being limited to their designers’ assumptions about how damage may occur and how to compensate for each type of damage. However, state-of-the-art learning algorithms are impractical because of the “curse of dimensionality”<sup>12</sup>: the fastest algorithms, from fields like reinforcement learning<sup>12</sup> and modular robotics<sup>10,11</sup>, depend on human demonstrations<sup>12</sup> or take 15 min or more even for relatively small search spaces (for example, 6–12 parameters, requiring 15–30 min<sup>10,11</sup>). Algorithms without these limitations take several hours<sup>12</sup>. Damage recovery would be much more practical and effective if robots

adapted as creatively and quickly as animals do (for example, in less than 2 min) in larger search spaces and without expensive, self-diagnosing sensors.

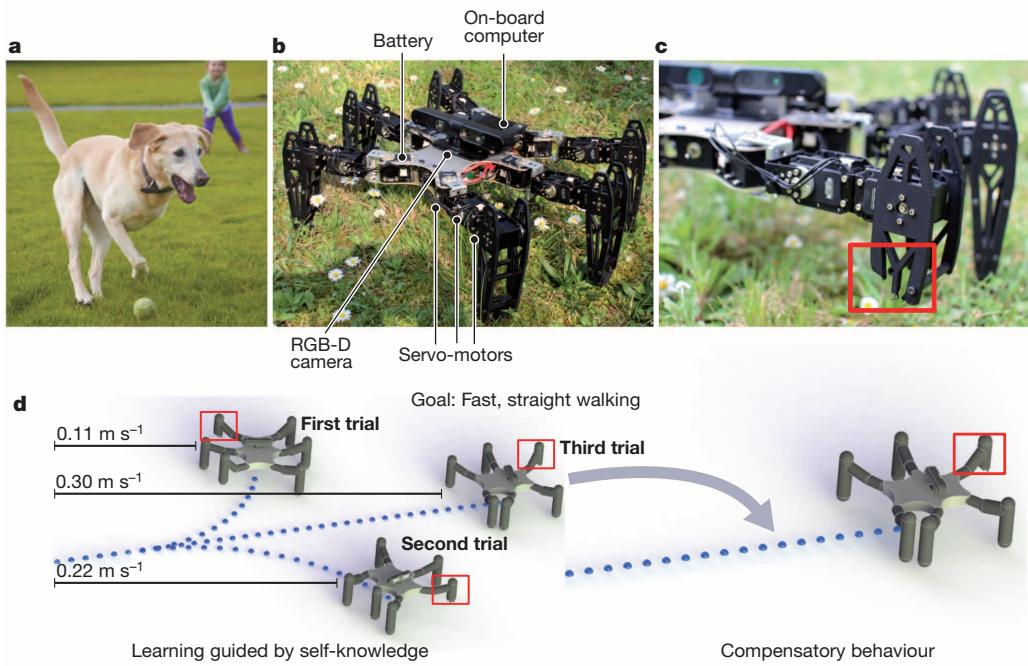
Here we show that rapid adaptation can be achieved by guiding an intelligent trial-and-error learning algorithm with an automatically generated, pre-computed behaviour–performance map that predicts the performance of thousands of different behaviours (Supplementary Video 1). Current learning algorithms either start with no knowledge of the search space<sup>12</sup> or with minimal knowledge from a few human demonstrations<sup>12,18</sup>. Our hypothesis is that animals understand the space of possible behaviours and their value from previous experience<sup>19</sup>, and that animals adapt by intelligently selecting tests that validate or invalidate whole families of promising compensatory behaviours. The key insight here is that robots could do the same.

Our robots store knowledge from previous experience in the form of a map of the behaviour–performance space. Guided by this map, a damaged robot tries different types of behaviours that are predicted to perform well and, as tests are conducted, updates its estimates of the performance of those types of behaviours. The process ends when the robot predicts that the most effective behaviour has already been discovered. A key assumption is that information about many different behaviours of the undamaged robot will still be useful after damage, because some of these behaviours will still be functional despite the damage. The results of our experiments support this assumption for all the types of damage we tested, revealing that a robot can quickly discover a way to compensate for damage (for example, see Fig. 1c) without a detailed mechanistic understanding of its cause, as occurs with animals. We call this approach ‘Intelligent Trial and Error’ (IT&E) (see Fig. 1d).

The behaviour–performance map is created using a novel algorithm and a simulation of the robot, which can be either a standard physics simulator or automatically discovered<sup>14</sup>. The robot’s designers need only describe the dimensions of the space of possible behaviours and a performance measure. For instance, walking gaits could be described by how much each leg touches the ground (a behavioural measure) and speed (a performance measure). An alternative gait behavioural measure could be the percentage of time a robot’s torso has positive pitch, roll, and yaw angles. For grasping, performance could be measured by the amount of surface contact, and it has been demonstrated that 90% of effective poses for the 21-degree-of-freedom human hand can be captured by a three-dimensional behavioural space describing the principal components of ways in which hand-poses commonly vary<sup>20</sup>. To fill in the behaviour–performance map, an optimization algorithm simultaneously searches for a high-performing solution at each point in the behavioural space (Fig. 2a, b and Extended Data Fig. 1). This step requires simulating millions of behaviours, but needs to be performed only once per robot design before deployment (Supplementary Methods).

A low confidence is assigned to the predicted performance of behaviours stored in this behaviour–performance map because they have not been tried in reality (Fig. 2b and Extended Data Fig. 1). During the robot’s mission, if performance drops below a user-defined threshold (owing either to damage or to a different environment), the robot

<sup>1</sup>Sorbonne Universités, Université Pierre et Marie Curie (UPMC), Paris 06, UMR 7222, Institut des Systèmes Intelligents et de Robotique (ISIR), F-75005, Paris, France. <sup>2</sup>CNRS, UMR 7222, Institut des Systèmes Intelligents et de Robotique (ISIR), F-75005, Paris, France. <sup>3</sup>Department of Computer Science, University of Wyoming, Laramie, Wyoming 82071, USA. <sup>4</sup>Inria, Team Larsen, Villers-lès-Nancy, F-54600, France. <sup>5</sup>CNRS, Loria, UMR 7503, Vandœuvre-lès-Nancy, F-54500, France. <sup>6</sup>Université de Lorraine, Loria, UMR 7503, Vandœuvre-lès-Nancy, F-54500, France. <sup>†</sup>Present addresses: Department of Electronics, University of York, York YO10 5DD, UK (D.T.); Inria, Villers-lès-Nancy, F-54600, France (J.-B.M.).



**Figure 1 | Using the Intelligent Trial and Error (IT&E) algorithm, robots, like animals, can quickly adapt to recover from damage.** **a**, Most animals can find a compensatory behaviour after an injury. Without relying on predefined compensatory behaviours, they learn how to avoid behaviours that are painful or no longer effective. Photo credit: Michael Lloyd/*The Oregonian* (2012). **b**, An undamaged, hexapod robot. RGB-D stands for red, green, blue, depth. **c**, One type of damage the hexapod may have to cope with

selects the most promising behaviour from the behaviour–performance map, tests it, and measures its performance. The robot subsequently updates its prediction for that behaviour and nearby behaviours, assigns high confidence to these predictions (Fig. 2c, d and Extended Data Fig. 1), and continues the select–test–update process until it finds a satisfactory compensatory behaviour (Fig. 2e and Extended Data Fig. 1).

All of these ideas are technically captured via a Gaussian process model<sup>21</sup>, which approximates the performance function with already acquired data, and a Bayesian optimization procedure<sup>22,23</sup>, which exploits this model to search for the maximum of the performance function (see Supplementary Methods). The robot selects which behaviours to test by maximizing an information acquisition function that balances exploration (selecting points whose performance is uncertain) and exploitation (selecting points whose performance is expected to be high) (see Supplementary Methods). The selected behaviour is tested on the physical robot and the actual performance is recorded. The algorithm updates the expected performance of the tested behaviour and lowers its uncertainty. These updates are propagated to neighbouring solutions in the behavioural space by updating the Gaussian process (Supplementary Methods). These updated performance and confidence distributions affect which behaviour is tested next. This select–test–update loop repeats until the robot finds a behaviour whose measured performance is greater than some user-defined percentage (here, 90%) of the best performance predicted for any behaviour in the behaviour–performance map (Supplementary Methods).

We first tested our algorithm on a hexapod robot that needs to walk as fast as possible (Fig. 1b, d). The robot has 18 motors, an onboard computer, and a depth camera that allows the robot to estimate its walking speed (Supplementary Methods). The gait is parameterized by 36 real-valued parameters (Supplementary Methods) that describe, for each joint, the amplitude of oscillation, phase shift, and duty cycle (the fraction of the period that the joint angle is on one side of the midline).

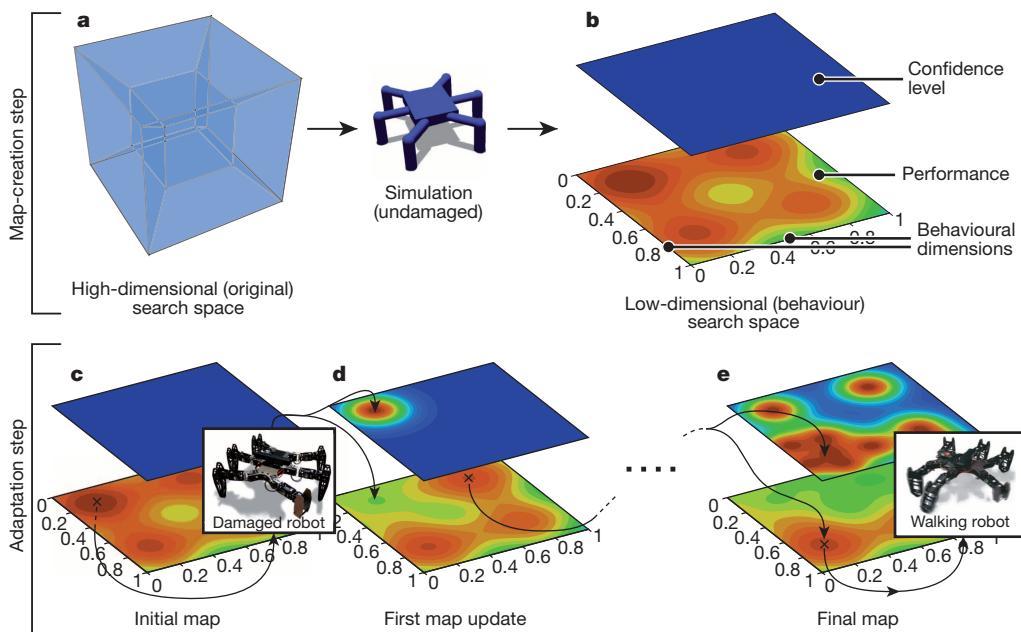
(broken leg). **d**, After damage occurs, in this case making the robot unable to walk fast and in a straight line, damage recovery via IT&E begins. The robot tests different types of behaviours from an automatically generated map of the behaviour–performance space. After each test, the robot updates its predictions of which behaviours will perform well despite the damage. In this way, the robot rapidly discovers an effective compensatory behaviour.

The behaviour space is six-dimensional, where each dimension is the proportion of time the *i*th leg spends in contact with the ground (that is, the duty factor)<sup>1</sup> (Supplementary Methods).

The behaviour–performance map created contains approximately 13,000 different gaits (Supplementary Video 2 shows examples). We tested our robot under six different conditions: undamaged (C1 in Fig. 3a), four different structural failures (C2–C5 in Fig. 3a), and a temporary leg repair (C6 in Fig. 3a). We compare the walking speed of resultant gaits with a widely used, classic, hand-designed tripod gait<sup>1</sup> (Supplementary Methods). For each of the six damage conditions, we ran our adaptation step five times for each of eight independently generated behaviour–performance maps (with the default ‘duty factor’ behavioural description), leading to  $6 \times 5 \times 8 = 240$  experiments in total. We also ran our adaptation step five times on eight independently generated behaviour–performance maps defined by an alternative behavioural description (‘body orientation’, see Supplementary Methods) on two damage conditions (Fig. 3b, c), leading to  $2 \times 5 \times 8 = 80$  additional experiments.

When the robot is undamaged (C1 in Fig. 3a), our approach yields dynamic gaits that are 30% faster than the classic reference gait (Fig. 3b, median  $0.32 \text{ m s}^{-1}$ , 5th and 95th percentiles  $[0.26 \text{ m s}^{-1}; 0.36 \text{ m s}^{-1}]$  versus  $0.24 \text{ m s}^{-1}$ ), suggesting that IT&E is a good search algorithm for automatically producing successful robot behaviours, putting aside damage recovery. In all the damage scenarios, the reference gait is no longer effective (around  $0.04 \text{ m s}^{-1}$  for the four damage conditions, C2–C5 in Fig. 3b). After IT&E, the compensatory gaits achieve a reasonably fast speed ( $>0.15 \text{ m s}^{-1}$ ) and are between three and seven times more efficient than the reference gait for that damage condition (gaits given in metres per second:  $0.24 [0.18; 0.31]$  versus  $0.04$  for C2;  $0.22 [0.18; 0.26]$  versus  $0.03$  for C3;  $0.21 [0.17; 0.26]$  versus  $0.04$  for C4;  $0.17 [0.12; 0.24]$  versus  $0.05$  for C5; and  $0.3 [0.21; 0.33]$  versus  $0.12$  for C6).

These experiments demonstrate that IT&E allows the robot both to initially learn fast gaits and to recover reliably after physical damage.



**Figure 2 | The two steps of IT&E.** **a, b**, Creating the behaviour–performance map. A user reduces a high-dimensional search space to a low-dimensional behaviour space by defining dimensions along which behaviours vary. In simulation, the high-dimensional space is then automatically searched to find a high-performing behaviour at each point in the low-dimensional behaviour space, creating a behaviour–performance map of the performance potential of each location in the low-dimensional space. In our hexapod robot experiments, the behaviour space is six-dimensional, with each dimension representing the portion of time that one leg is in contact with the ground. The confidence regarding the accuracy of the predicted performance for each behaviour in the behaviour–performance map is initially low because no tests on the physical robot have been conducted. **c–e**, Adaptation step: after damage, the robot selects a promising behaviour, tests it, updates the predicted performance of

that behaviour in the behaviour–performance map, and allocates a high confidence to this performance prediction. The predicted performances of nearby behaviours—and confidence in those predictions—are likely to be similar to the tested behaviour and are thus updated accordingly. This select–test–update loop is repeated until a tested behaviour on the physical robot performs better than 90% of the best predicted performance in the behaviour–performance map, a value that can decrease with each test (Extended Data Fig. 1). The algorithm that selects which behaviour to test next balances between choosing the behaviour with the highest predicted performance and behaviours that are different from those tested so far. Overall, the IT&E approach presented here rapidly locates which types of behaviours are least affected by the damage to find an effective, compensatory behaviour.

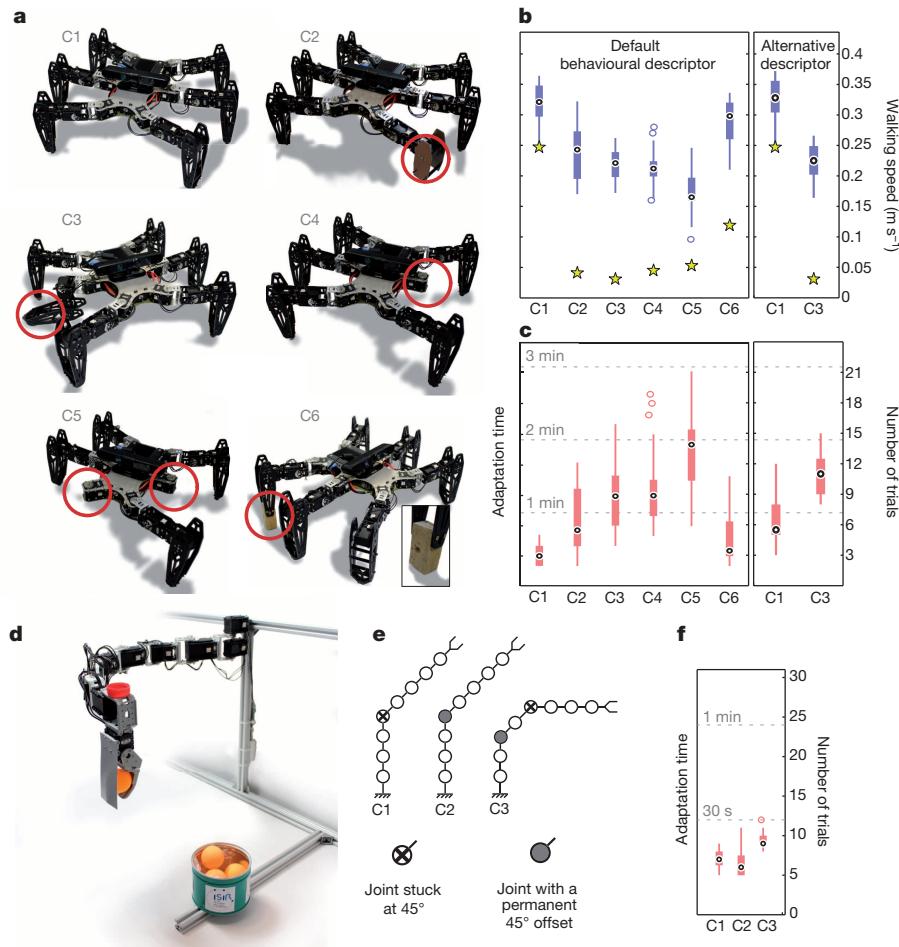
Additional experiments reveal that these capabilities are substantially faster than state-of-the-art algorithms (Bayesian optimization and policy gradient, Extended Data Fig. 2), and that IT&E can help with another major challenge in robotics: adapting to new environments (such as differently sloped terrain, Extended Data Fig. 3). On the undamaged or repaired robot (C6 in Fig. 3a), IT&E learns a walking gait in less than 30 s (Fig. 3c, the undamaged robot takes 24 s [16 s; 41 s] in 3 [2; 5] physical trials and the repaired robot takes 29 s [16 s; 82 s] in 3.5 [2; 10] trials). For the five damage scenarios, the robot adapts in approximately one minute (66 s [24 s; 134 s] in 8 [3; 16] trials). It is possible that for certain types of damage the prior information from the undamaged robot does not help, and could even hurt, in learning a compensatory behaviour (for example, if the map does not contain a compensatory behaviour). We did not find such a case in our experiments with the hexapod robot, but we did find a case on another robot in which the prior information provided little benefit (experiment 1 in the Supplementary Information).

Our results are qualitatively unchanged when using different behavioural characterizations, including randomly choosing six descriptors among 63 possibilities (Fig. 3b, c and Extended Data Fig. 4). Additional experiments show that reducing the high-dimensional parameter space to a low-dimensional behaviour space via the behaviour–performance map is the key component for IT&E: standard Bayesian optimization in the original parameter space does not find working controllers (gaits) (Extended Data Fig. 2). We investigated how the behaviour–performance map is updated when the robot loses a leg (C4 in Fig. 3a). Initially, the map predicts large areas of high performance. During adaptation, these areas disappear because the behaviours do not work well on the damaged robot. IT&E quickly identifies one of the

few remaining high-performance behaviours (Fig. 4 and Extended Data Figs 5 and 6).

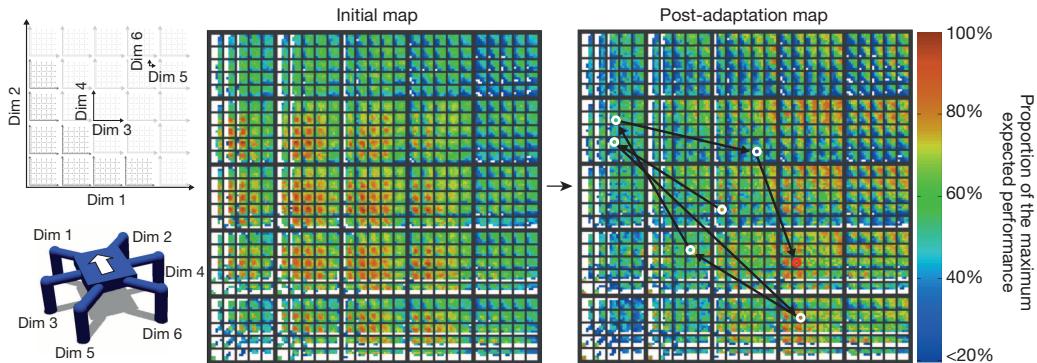
The same damage recovery approach can be applied to any robot, such as a robotic arm. We tested 14 different damage conditions with a planar, eight-jointed robotic arm (Fig. 3d–f and Extended Data Fig. 7). The behaviour–performance map’s behavioural dimensions are the  $x$ ,  $y$  position of the hand at the end of the arm. To show that the map-generating performance measure can be different from the ultimate performance measure, and to encourage smooth arm movements, the performance measure during map-generation is minimizing the variance of the eight specified motor angles (Supplementary Methods). During adaptation, performance is measured as distance to the target. As with the hexapod robot, our approach discovers a compensatory behaviour in less than 2 min, usually in less than 30 s, and with fewer than ten trials (Fig. 3f and Extended Data Fig. 7).

Although animals do not use the specific algorithm we present, there are parallels between IT&E and animal learning. Damage recovery in animals may occur without learning—for instance, due to the built-in robustness of evolved control loops<sup>24</sup>—but if such pre-programmed robustness fails, many animals turn to learning<sup>19</sup>. As in our robot, such learning probably exploits an animal’s intuitions about how its intact body works to experiment with different behaviours to find what works best. Also like animals<sup>25</sup>, IT&E allows the quick identification of working behaviours with a few, diverse tests instead of trying behaviours at random or trying small modifications to the best behaviour found so far. Additionally, the Bayesian optimization procedure followed by our robot appears similar to the technique employed by humans when they optimize an unknown function<sup>23</sup>, and there is strong evidence that animal brains learn probability



**Figure 3 | Main experiments and results.**

**a**, Conditions tested on the physical hexapod robot. C1, The undamaged robot. C2, One leg is shortened by half. C3, One leg is unpowered. C4, One leg is missing. C5, Two legs are missing. C6, An imperfect, makeshift repair to the tip of one leg (by a human operator). **b**, Performance after adaptation. Box plots represent IT&E. The central circle is the median, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme data points that are not considered outliers, and outliers are plotted individually. Yellow stars represent the performance of the handmade reference tripod gait (Supplementary Methods). Conditions C1–C6 are tested five times each for eight independently created behaviour–performance maps with the ‘duty factor’ behaviour description (that is, 40 experiments per damage condition, Supplementary Methods). Damage conditions C1 and C3 are also tested five times each for eight independently created behaviour–performance maps with the ‘body orientation’ behaviour description (Supplementary Methods). **c**, Time and number of trials required to adapt. Box plots represent IT&E. **d**, Robotic arm experiment. The eight-jointed, planar robot arm has to drop a ball into a bin. **e**, Example conditions tested on the physical robotic arm. C1, One joint is stuck at 45 degrees. C2, One joint has a permanent 45° offset. C3, One broken and one offset joint. Each of these damage conditions was replicated with 15 independent maps. A total of 14 conditions were tested (Extended Data Fig. 7). **f**, Time and number of trials required to reach within 5 cm of the bin centre. Each condition is tested with 15 independently created behaviour–performance maps.



**Figure 4 | An example behaviour–performance map.** This map stores high-performing behaviours at each point in a six-dimensional behaviour space. Each dimension is the portion of time that each leg is in contact with the ground. The behavioural space is discretized at five values for each dimension (0; 0.25; 0.5; 0.75 and 1). Each coloured pixel represents the highest-performing behaviour discovered during map creation at that point in the behaviour space. The matrices visualize the six-dimensional behavioural space in two dimensions according to the legend at the top left. The behaviour–performance map is created with a simulated robot (bottom left) in the Open Dynamics Engine physics simulator (<http://www.ode.org>). The left matrix is a

pre-adaptation map produced by the map creation algorithm. During adaptation, the map is updated as tests are conducted (in this case, for the damage condition where the robot is missing one leg: C4 in Fig. 3a). The right matrix shows the state of the map after a compensatory behaviour is discovered. The arrows and white circles represent the order in which behaviours were tested on the physical robot. The red circle is the final, discovered, compensatory behaviour. Among other areas, high-performing behaviours can be found for the damaged robot in the first two columns of the third dimension. These columns represent behaviours that use the central-left leg least, which is the leg that is missing.

distributions, combine them with prior knowledge, and act as Bayesian optimizers<sup>26,27</sup>.

An additional parallel is that IT&E primes the robot for creativity during a motionless period, after which the generated ideas are tested.

This process is reminiscent of the finding that some animals start the day with new ideas that they may quickly disregard after experimenting with them<sup>28</sup>, and more generally, that sleep improves creativity on cognitive tasks<sup>29</sup>. A final parallel is that the simulator and Gaussian

process components of IT&E are two forms of predictive models, which are known to exist in animals<sup>14,30</sup>. Overall, we have shown that IT&E has parallels in biology and makes robots behave more like animals by endowing them with the ability to adapt rapidly to unforeseen circumstances.

Received 24 December 2014; accepted 17 March 2015.

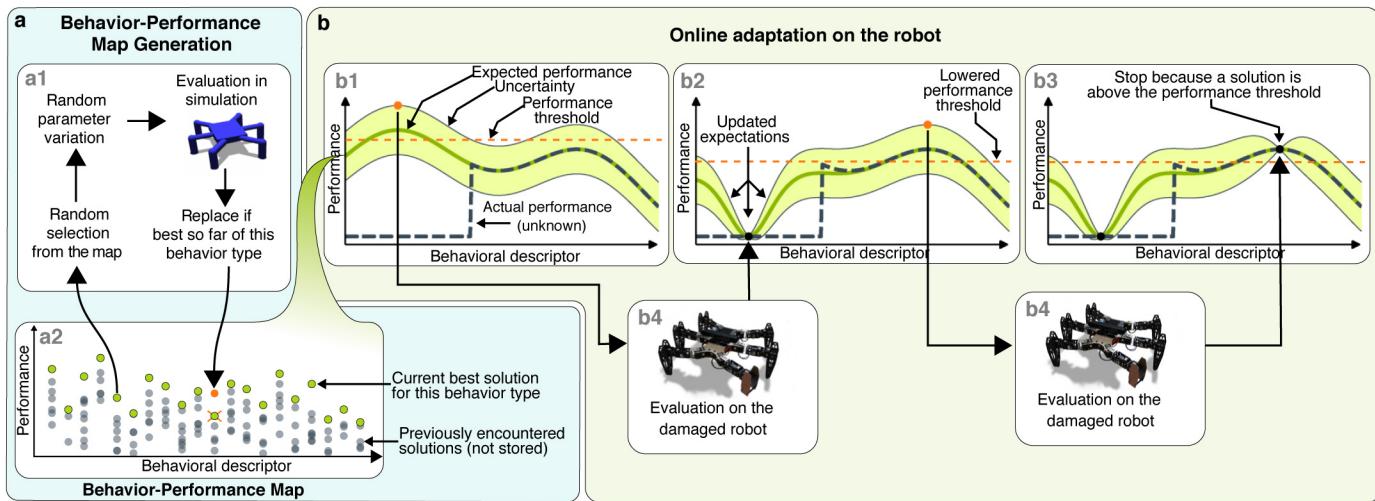
1. Siciliano, B. & Khatib, O. *Springer Handbook of Robotics* (Springer, 2008).
2. Murphy, R. R. Trial by fire. *Robot. Automat. Mag.* **11**, 50–61 (2004).
3. Nagatani, K. *et al.* Emergency response to the nuclear accident at the Fukushima Daiichi nuclear power plants using mobile rescue robots. *J. Field Robot.* **30**, 44–63 (2013).
4. Broadbent, E., Stafford, R. & MacDonald, B. Acceptance of healthcare robots for the older population: review and future directions. *Int. J. Social Robot.* **1**, 319–330 (2009).
5. Thrun, S. *et al.* Stanley: the robot that won the DARPA grand challenge. *J. Field Robot.* **23**, 661–692 (2006).
6. Sanderson, K. Mars rover Spirit (2003–10). *Nature* **463**, 600 (2010).
7. Antonelli, G., Fossen, T. I. & Yoerger, D. R. In *Springer Handbook of Robotics* (eds Siciliano, B. & Khatib, O.) 987–1008 (Springer, 2008).
8. Carlson, J. & Murphy, R. R. How UGVs physically fail in the field. *IEEE Trans. Robot.* **21**, 423–437 (2005).
9. Blanke, M., Kinnaert, M., Lunze, J. & Staroswiecki, M. *Diagnosis and Fault-Tolerant Control* (Springer, 2006).
10. Sproewitz, A., Moeckel, R., Maye, J. & Ijspeert, A. Learning to move in modular robots using central pattern generators and online optimization. *Int. J. Robot. Res.* **27**, 423–443 (2008).
11. Christensen, D. J., Schultz, U. P. & Stoy, K. A distributed and morphology-independent strategy for adaptive locomotion in self-reconfigurable modular robots. *Robot. Auton. Syst.* **61**, 1021–1035 (2013).
12. Kober, J., Bagnell, J. A. & Peters, J. Reinforcement learning in robotics: a survey. *Int. J. Robot. Res.* **32**, 1238–1274 (2013).
13. Verma, V., Gordon, G., Simmons, R. & Thrun, S. Real-time fault diagnosis. *Robot. Automat. Mag.* **11**, 56–66 (2004).
14. Bongard, J., Zykov, V. & Lipson, H. Resilient machines through continuous self-modeling. *Science* **314**, 1118–1121 (2006).
15. Kluger, J. & Lovell, J. *Apollo 13* (Mariner Books, 2006).
16. Jarvis, S. L. *et al.* Kinematic and kinetic analysis of dogs during trotting after amputation of a thoracic limb. *Am. J. Vet. Res.* **74**, 1155–1163 (2013).
17. Fuchs, A., Goldner, B., Nolte, I. & Schilling, N. Ground reaction force adaptations to tripodal locomotion in dogs. *Vet. J.* **201**, 307–315 (2014).
18. Argall, B. D., Chernova, S., Veloso, M. & Browning, B. A survey of robot learning from demonstration. *Robot. Auton. Syst.* **57**, 469–483 (2009).
19. Wolpert, D. M., Ghahramani, Z. & Flanagan, J. R. Perspective and problems in motor learning. *Trends Cogn. Sci.* **5**, 487–494 (2001).
20. Santello, M. Postural hand synergies for tool use. *J. Neurosci.* **18**, 10105–10115 (1998).
21. Rasmussen, C. E. & Williams, C. K. I. *Gaussian Processes for Machine Learning* (MIT Press, 2006).
22. Mockus, J. *Bayesian Approach to Global Optimization: Theory and Applications* (Kluwer Academic, 2013).
23. Borji, A. & Itti, L. Bayesian optimization explains human active search. *Adv. Neural Inform. Process. Syst.* **26**, 55–63 (2013).
24. Grillner, S. The motor infrastructure: from ion channels to neuronal networks. *Nature Rev. Neurosci.* **4**, 573–586 (2003).
25. Benson-Amram, S. & Holekamp, K. E. Innovative problem solving by wild spotted hyenas. *Proc. R. Soc. Lond. B* **279**, 4087–4095 (2012).
26. Pouget, A., Beck, J. M., Ma, W. J. & Latham, P. E. Probabilistic brains: knowns and unknowns. *Nature Neurosci.* **16**, 1170–1178 (2013).
27. Kording, K. P. & Wolpert, D. M. Bayesian integration in sensorimotor learning. *Nature* **427**, 244–247 (2004).
28. Derégnaucourt, S., Mitra, P. P., Fehér, O., Pytte, C. & Tchernichovski, O. How sleep affects the developmental learning of bird song. *Nature* **433**, 710–716 (2005).
29. Wagner, U., Gais, S., Haider, H., Verleger, R. & Born, J. Sleep inspires insight. *Nature* **427**, 352–355 (2004).
30. Ito, M. Control of mental activities by internal models in the cerebellum. *Nature Rev. Neurosci.* **9**, 304–313 (2008).

**Supplementary Information** is available in the online version of the paper.

**Acknowledgements** We thank L. Tedesco, S. Doncieux, N. Bredeche, S. Whiteson, R. Calandri, J. Droulez, P. Bessière, F. Lesaint, C. Thurat, S. Ivaldi, C. Lan Sun Luk, J. Li, J. Huizinga, R. Velez, H. Mengistu, M. Norouzzadeh, T. Clune, and A. Nguyen for feedback and discussions. This work has been funded by the ANR Creadapt project (ANR-12-JS03-0009), the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement number 637972), and a Direction Générale de l'Armement (DGA) scholarship to A.C.

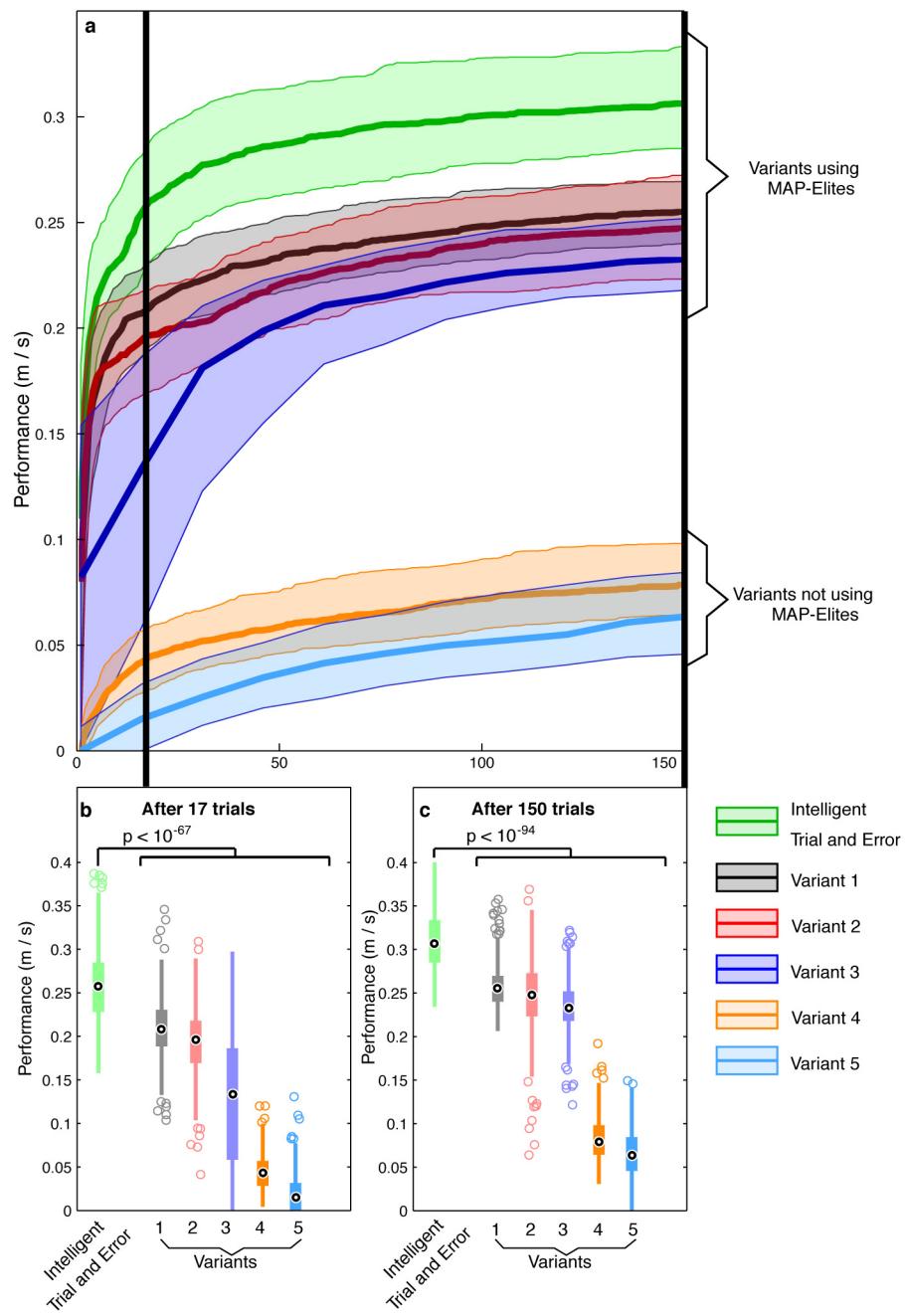
**Author Contributions** A.C. and J.-B. M. designed the study. A.C. and D.T. performed the experiments. A.C., J.-B. M., D.T. and J.C. analysed the results, discussed additional experiments, and wrote the paper.

**Author Information** Reprints and permissions information is available at [www.nature.com/reprints](http://www.nature.com/reprints). The authors declare no competing financial interests. Readers are welcome to comment on the online version of the paper. Correspondence and requests for materials should be addressed to J.-B. M. (jean-baptiste.mouret@inria.fr).



**Extended Data Figure 1 | An overview of the IT&E algorithm.** **a**, Behaviour-performance map creation. After being initialized with random controllers, the behavioural map (a2), which stores the highest-performing controller found so far of each behaviour type, is improved by repeating the process depicted in a1 until newly generated controllers are rarely good enough to be added to the map (here, after 40 million evaluations). This step, which occurs in simulation (and required roughly two weeks on one multi-core computer; see Supplementary Methods, section ‘Running time’), is computationally expensive, but only needs to be performed once per robot (or robot design) before the robot is deployed. **b**, Adaptation. In b1, each behaviour from the behaviour–performance map has an expected performance based on its performance in simulation (dark green line) and an estimate of uncertainty regarding this predicted performance (light green band). The actual performance on the now-damaged robot (black dashed line) is unknown to the algorithm. A behaviour is selected for the damaged robot to try. This selection is made by balancing exploitation—trying

behaviours expected to perform well—and exploration—trying behaviours whose performance is uncertain (Supplementary Methods, section ‘Acquisition function’). Because all points initially have equal, maximal uncertainty, the first point chosen is that with the highest expected performance. Once this behaviour is tested on the physical robot (b4), the performance predicted for that behaviour is set to its actual performance, the uncertainty regarding that prediction is lowered, and the predictions for, and uncertainties about, nearby controllers are also updated (according to a Gaussian process model; see Supplementary Methods, section ‘Kernel function’), the results of which can be seen in b2. The process is then repeated until performance on the damaged robot is 90% or greater of the maximum expected performance for any behaviour (b3). This performance threshold (orange dashed line) lowers as the maximum expected performance (the highest point on the dark green line) is lowered, which occurs when physical tests on the robot underperform expectations, as occurred in b2.

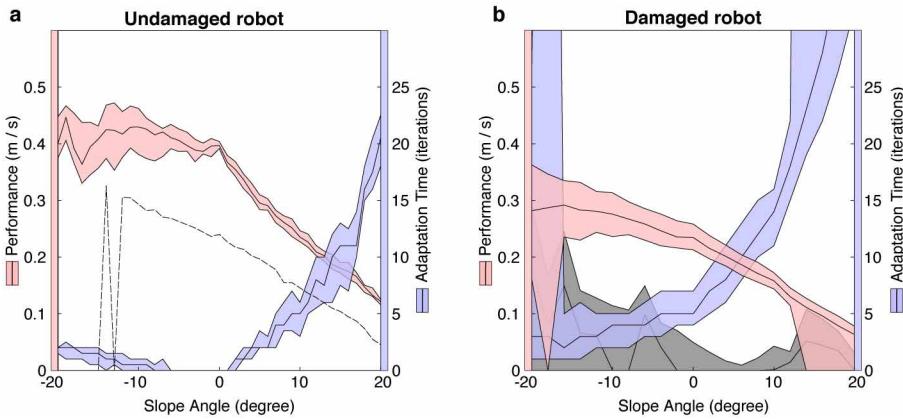


Variant	Behavior-performance map creation	Priors on performance	Search algorithm	equivalent approach
Intelligent Trial and Error	MAP-Elites	yes	Bayesian Optimization	-
Variant 1	MAP-Elites	none	Random Search	-
Variant 2	MAP-Elites	none	Bayesian Optimization	-
Variant 3	MAP-Elites	none	Policy Gradient	-
Variant 4	none	none	Bayesian Optimization	Lizotte et al. (2007)
Variant 5	none	none	Policy Gradient	Kohl et al. (2004)

**Extended Data Figure 2 | The contribution of each subcomponent of the IT&E algorithm.** **a**, Adaptation progress versus the number of robot trials. The walking speed achieved with IT&E and several ‘knockout’ variants that are missing one of the algorithm’s key components (defined in the table below the plots). Some variants (4 and 5) correspond to state-of-the-art learning algorithms (policy gradient<sup>31</sup> and Bayesian optimization<sup>32,33,34</sup>). MAP-Elites (written by J.-B.M. and J.C.; ref. 35) is the algorithm that IT&E uses to create the behaviour–performance map (see Supplementary Methods). The bold lines represent the medians and the coloured areas extend to the 25th and 75th percentiles. **b, c**, Adaptation performance after 17 and 150 trials. Shown is the

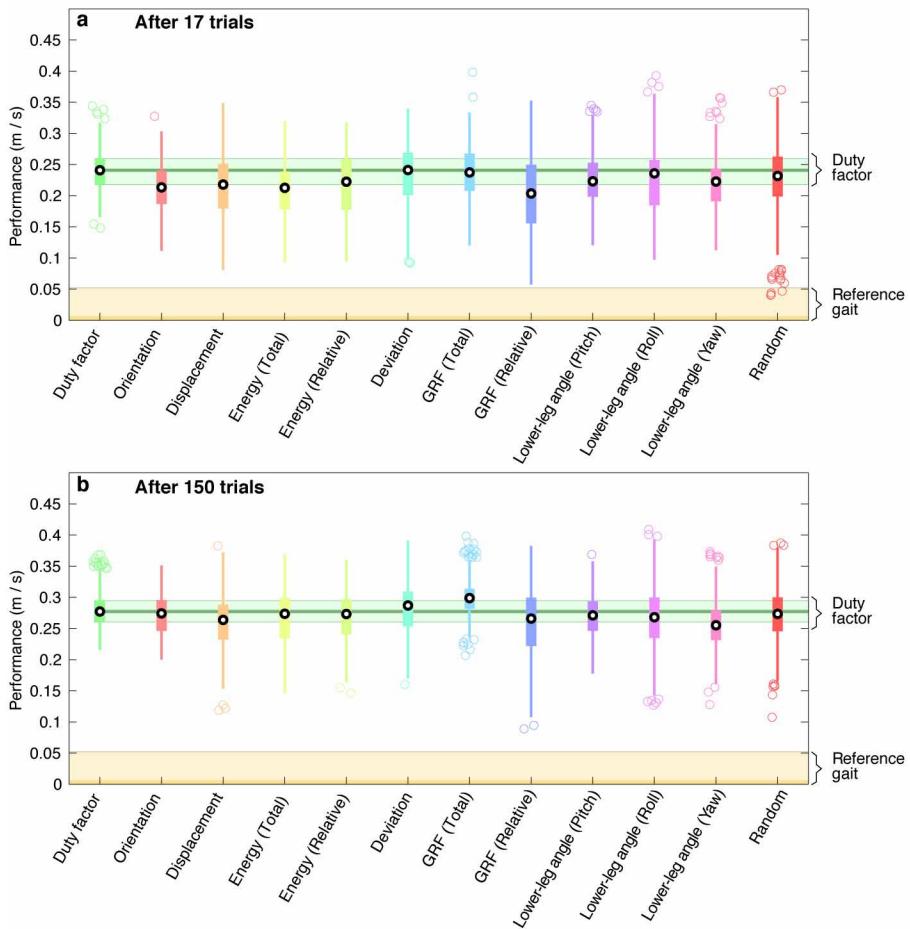
speed of the compensatory behaviour discovered by each algorithm after 17 and 150 evaluations on the simulated robot, respectively. For all panels, data are pooled across six damage conditions (the removal of each of the six legs in turn) for eight independently generated maps and replicated ten times (for a total of 480 replicates). *P* values are computed via a Wilcoxon ranksum test. The central circle is the median, the edges of the box show the 25th and 75th percentiles, the whiskers extend to the most extreme data points that are not considered outliers, and outliers are plotted individually. See experiment 2 in the Supplementary Information for methods and analysis.

31. Kohl, N. & Stone, P. Policy gradient reinforcement learning for fast quadrupedal locomotion. In *Proc. IEEE Int. Conf. on ‘Robotics and Automation’ (ICRA)* 2619–2624 (IEEE, 2004).
32. Lizotte, D. J., Wang, T., Bowling, M. H. & Schuurmans, D. Automatic gait optimization with Gaussian process regression. In *Proc. Int. Joint Conf. on ‘Artificial Intelligence’ (IJCAI)* 944–949 (2007).
33. Tesch, M., Schneider, J. & Choset, H. Using response surfaces and expected improvement to optimize snake robot gait parameters. In *Proc. IEEE/RSJ Int. Conf. on ‘Intelligent Robots and Systems (IROS)’* 1069–1074 (IEEE, 2011).
34. Calandra, R., Seyfarth, A., Peters, J. & Deisenroth, M. P. An experimental comparison of bayesian optimization for bipedal locomotion. In *Proc. IEEE Int. Conf. on ‘Robotics and Automation’ (ICRA)* 1951–1958 (IEEE, 2014).
35. Mouret, J.-B. & Clune, J. Illuminating search spaces by mapping elites. Preprint at <http://arxiv.org/abs/1504.04909> (2015).



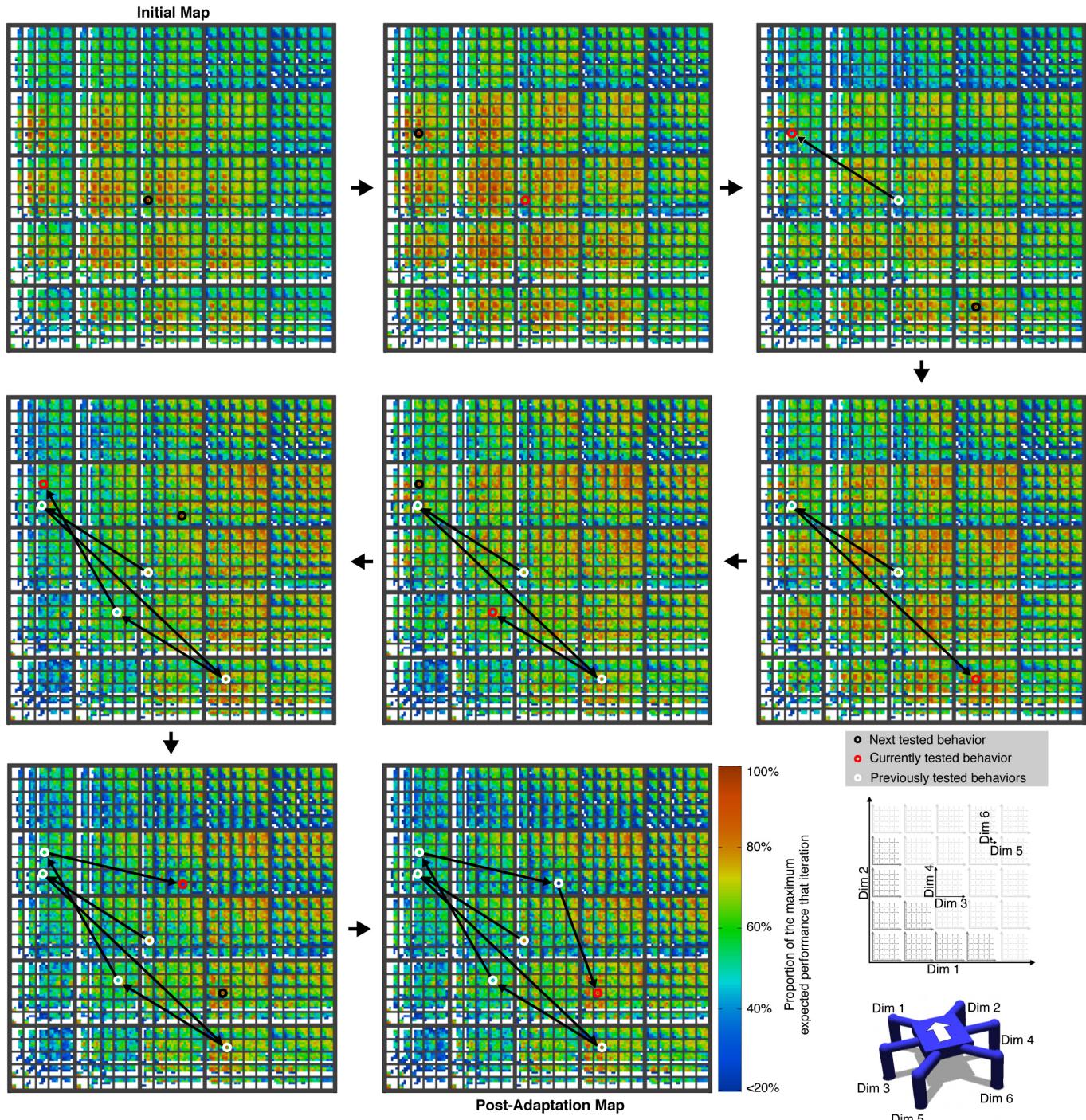
**Extended Data Figure 3 | The IT&E algorithm is robust to environmental changes.** Each plot shows both the performance and required adaptation time for IT&E when the robot must adapt to walk on terrains of different slopes. **a**, Adaptation performance on an undamaged, simulated robot. On all slope angles, with very few trials, the IT&E algorithm (pink shaded region) finds fast gaits that outperform the reference gait (black dashed line). We performed ten replicates for each of the eight maps and each one-degree increment between  $-20^\circ$  and  $+20^\circ$  degrees (a total of 3,280 experiments). **b**, Adaptation performance on a damaged, simulated robot. The robot is damaged by having each of its six legs removed in six different damage scenarios. Data

are pooled from all six of these damage conditions. For each damage condition, we performed ten replicates for each of the eight maps and each two-degree increment between  $-20^\circ$  and  $+20^\circ$  degrees (a total of 10,080 experiments). The median compensatory behaviour found via IT&E outperforms the median reference controller on all slope angles. The middle, black lines represent medians, while the colored areas extend to the 25th and 75th percentiles. In **a**, the black dashed line is the performance of a classic tripod gait for reference. In **b**, the reference gait is tried in all six damage conditions and its median (black line) and 25th and 75th percentiles (black coloured area) are shown. See experiment 3 in the Supplementary Information for methods and analysis.



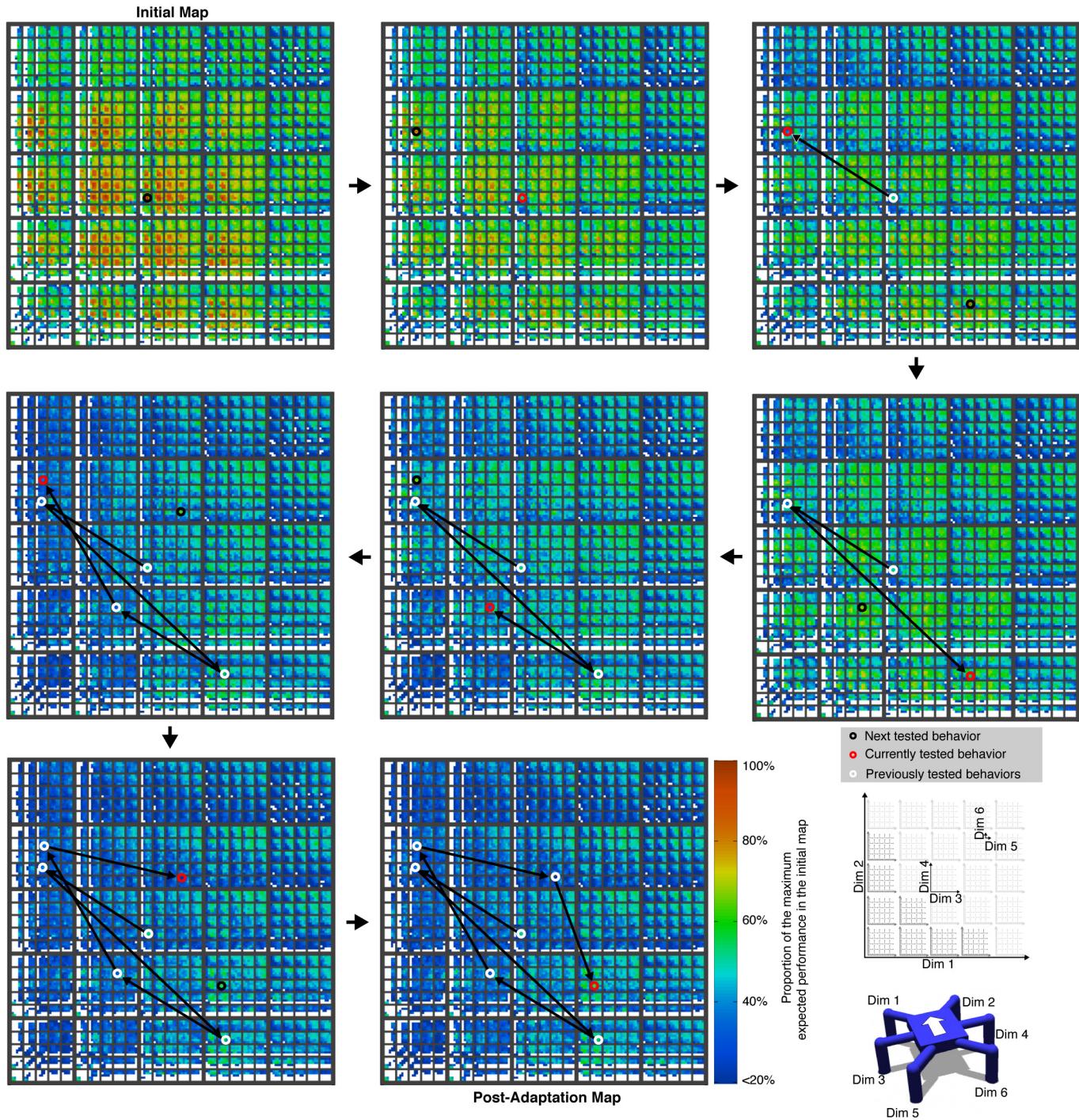
**Extended Data Figure 4 | The IT&E algorithm is largely robust to alternative choices of behaviour descriptors.** **a, b,** The speed of the compensatory behaviour discovered by IT&E for various choices of behaviour descriptors. Performance is plotted after 17 and 150 evaluations in **a** and **b**, respectively. Experiments were performed on a simulated, damaged hexapod. The damaged robot has each of its six legs removed in six different damage scenarios. Data are pooled across all six damage conditions. As described in experiment 5 of the Supplementary Information, the evaluated behaviour descriptors characterize the following: (1) time each leg is in contact with the ground ('Duty factor'); (2) orientation of the robot frame ('Orientation'); (3) instantaneous velocity of the robot ('Displacement'); (4) energy expended by the robot in walking ('Energy (Total)', 'Energy (Relative)'); (5) deviation from a straight line ('Deviation'); (6) ground reaction force on each leg ('GRF (Total)', 'GRF (Relative)'); (7) the angle of each leg when it touches the ground ('Lower-leg angle (Pitch)', 'Lower-leg angle (Roll)', 'Lower-leg angle (Yaw)'); and (8) a random selection without replacement from subcomponents of all the available behaviour descriptors (1)–(7) ('Random').

We created eight independently generated maps for each of 11 intentionally chosen behavioural descriptors (88 in total). We created one independently generated map for each of 20 randomly chosen behavioural descriptors (20 in total). For each of these 108 maps, we performed ten replicates of IT&E for each of the six damage conditions. In total, there were thus  $8 \times 11 \times 10 \times 6 = 5,280$  experiments for the intentionally chosen behavioural descriptors and  $1 \times 20 \times 10 \times 6 = 1,200$  experiments for the randomly chosen behavioural descriptors. For the hand-designed reference gait (yellow) and the compensatory gaits found by the default duty factor behaviour descriptor (green), the bold lines represent the medians and the coloured areas extend to the 25th and 75th percentiles of the data. For the other treatments, including the duty factor treatment, black circles represent the median, the edges of the boxes show the 25th and 75th percentiles, the whiskers extend to the most extreme data points that are not considered outliers, and outliers are plotted individually. See experiment 5 in the Supplementary Information for methods and analysis.



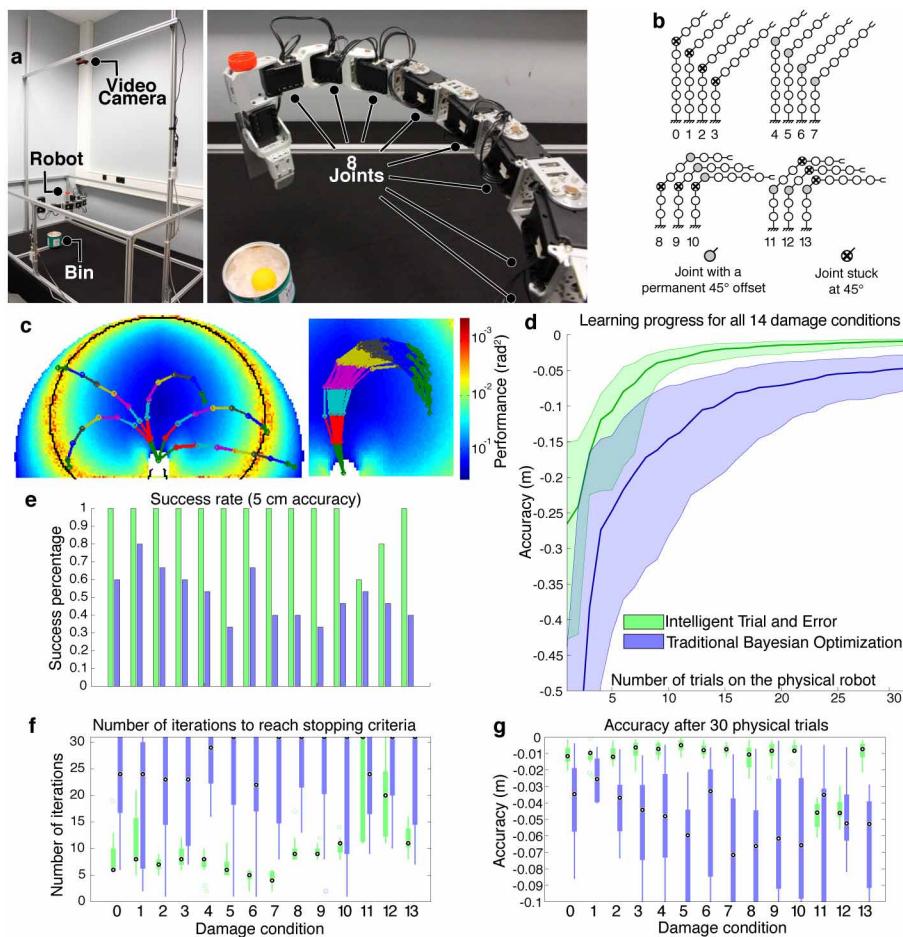
**Extended Data Figure 5 | How the behaviour performance map (data normalized) is explored to discover a compensatory behaviour.** Data are normalized each iteration to highlight the range of remaining performance predictions. The robot is adapting to damage condition C4 (see Fig. 3a). Colours represent the performance prediction for each point in the map relative to the highest performing prediction in the map at that step of the process. A

black circle indicates the next behaviour to be tested on the physical robot. A red circle indicates the behaviour that was just tested (note that the performance predictions surrounding it have changed versus the previous panel). Arrows reveal the order that points have been explored. The red circle in the last map is the final, selected, compensatory behaviour. In this scenario, the robot loses leg number 3. The six-dimensional space is visualized according to the inset legend.



**Extended Data Figure 6 | How the behaviour performance map (data not normalized) is explored to discover a compensatory behaviour.** The data are not normalized in order to highlight that performance predictions decrease as it is discovered that predictions from the simulated, undamaged robot do not work well on the damaged robot. The robot is adapting to damage condition C4 (see Fig. 3a). Colours represent the performance prediction for each point in the map relative to the highest performing prediction in the first map. A black circle indicates the next behaviour to be tested on the physical robot. A red

circle indicates the behaviour that was just tested (note that the performance predictions surrounding it have changed versus the previous panel). Arrows reveal the order that points have been explored. The red circle in the last map in the sequence is the final, selected, compensatory behaviour. In this scenario, the robot loses leg number 3. The six-dimensional space is visualized according to the inset legend. The data visualized in this figure are identical to those in the previous figure: the difference is simply whether the data are renormalized for each new map in the sequence.



**Extended Data Figure 7 | IT&E works on a completely different type of robot (the robotic arm experiment).** **a.** The robotic arm experimental setup. **b.** Tested damage conditions. **c.** Example of behaviour performance maps (colour maps) and behaviours (overlaid arm configurations) obtained with MAP-Elites. Left: A typical behaviour–performance map produced by MAP-Elites with five example behaviours, where a behaviour is described by the angle of each of the eight joints. The colour of each point is a function of its performance (in radians squared), which is defined as having low variance in the joint angles (that is, a zigzag arm has lower performance than a straighter arm that reaches the same point). Right: Neighbouring points in the map tend to have similar behaviours, thanks to the performance function, which would penalize more jagged ways of reaching those points. That neighbours have similar behaviours justifies updating predictions about the performance of nearby behaviours after testing a single behaviour on the real (damaged) robot. **d.** Accuracy (in metres) versus trial number for IT&E and traditional

Bayesian optimization. The experiment was conducted on the physical robot, with 15 independent replications for each of the 14 damage conditions. Accuracy is pooled from all of these  $14 \times 15 = 210$  experiments for each algorithm. The middle lines represent medians, while the coloured areas extend to the 25th and 75th percentiles. **e.** Success for each damage condition. Shown is the success rate for the 15 replications for each damage condition, defined as the percentage of replicates in which the robot reaches within 5 cm of the bin centre. **f.** Trials required to adapt. Shown is the number of iterations required to reach within 5 cm of the bin centre. **g.** Accuracy after 30 physical trials for each damage condition (with the stopping criterion disabled). For **f** and **g**, the central circle is the median, the edges of the box show the 25th and 75th percentiles, the whiskers extend to the most extreme data points that are not considered outliers, and outliers are plotted individually. See experiment 1 in the Supplementary Information for methods and analysis.