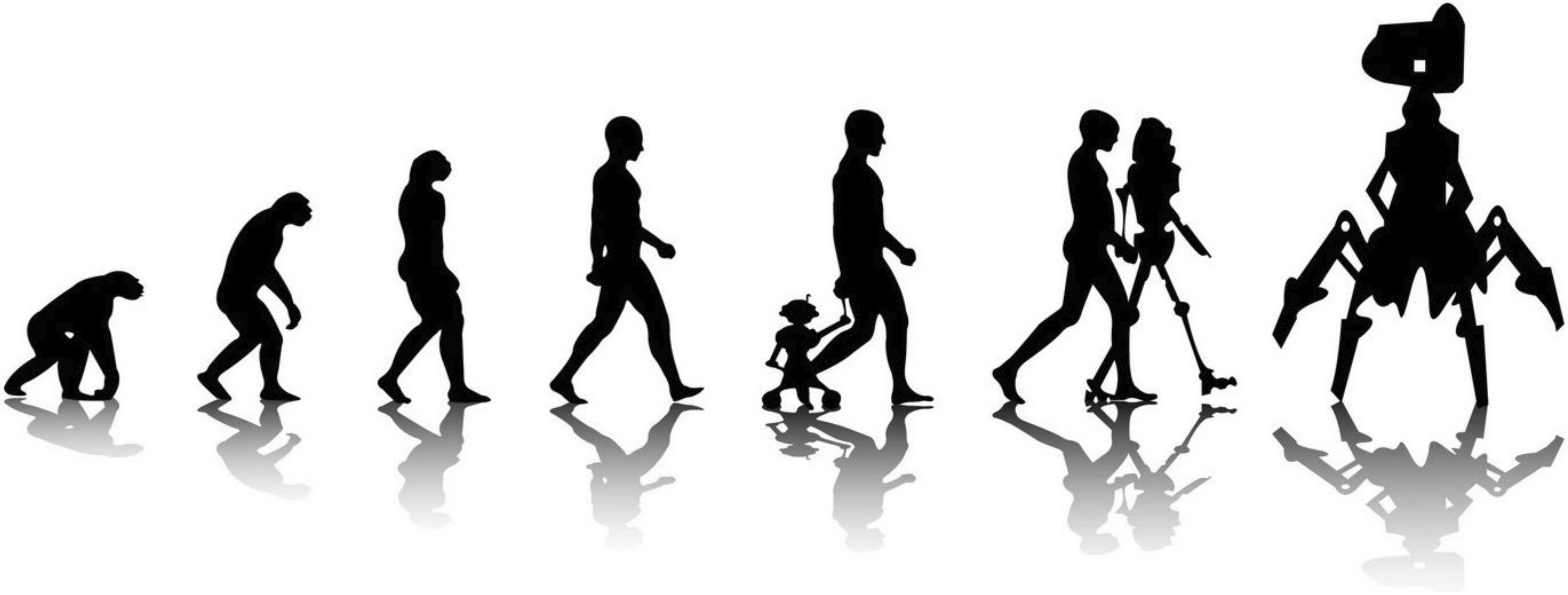


Modern Robotics: Evolutionary Robotics

COSC 4560 / COSC 5560

Professor Cheney

4/13/18

Active Learning
in Self-Modeling

Resilient Machines Through Continuous Self-Modeling

Josh Bongard,[1]*† Victor Zykov,[1] Hod Lipson[1,2]

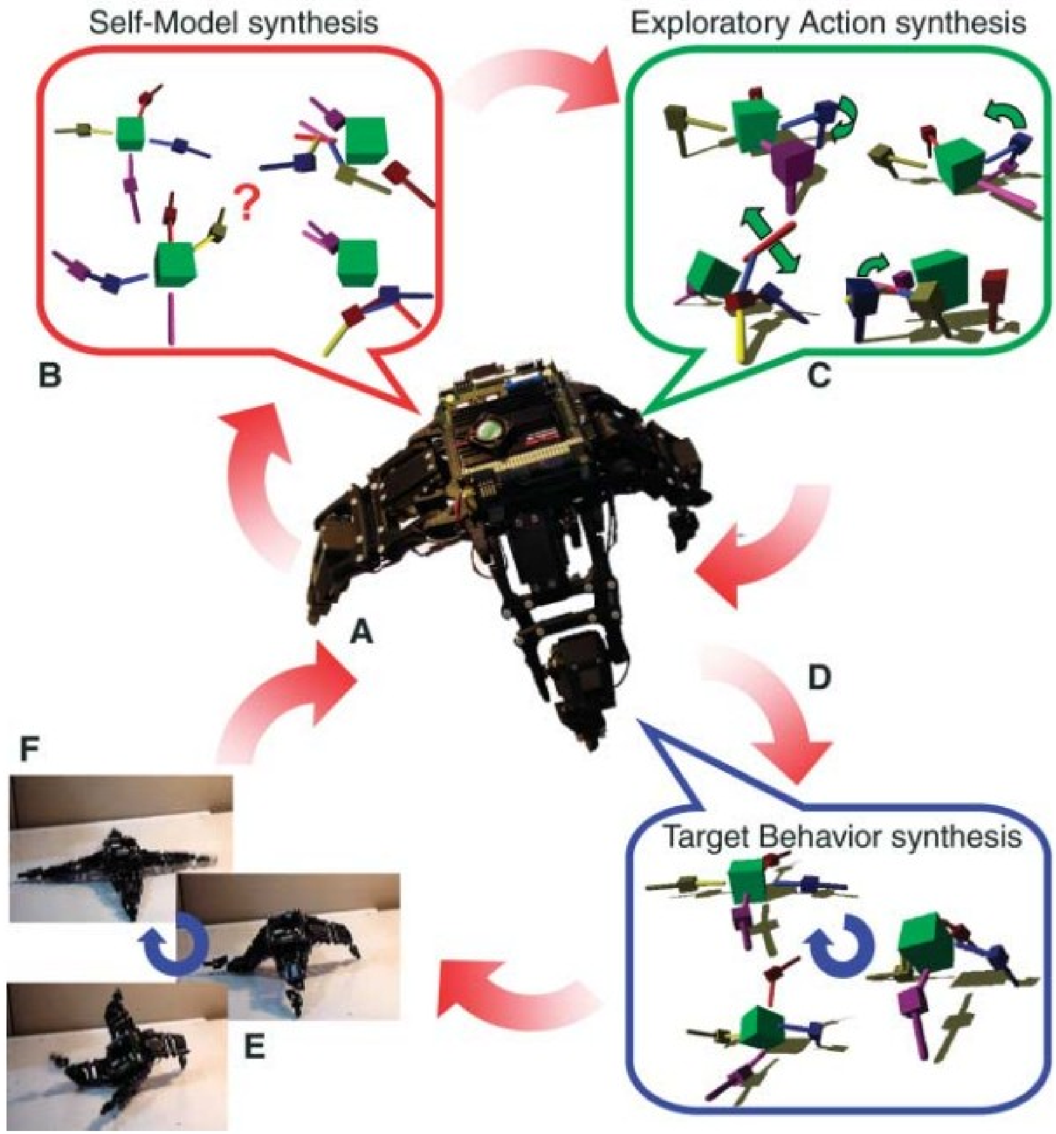


Fig. 1. Outline of the algorithm. The robot continuously cycles through action execution. (**A** and **B**) Self-model synthesis. The robot physically performs an action (A). Initially, this action is random; later, it is the best action found in (C). The robot then generates several self-models to match sensor data collected while performing previous actions (B). It does not know which model is correct. (**C**) Exploratory action synthesis. The robot generates several possible actions that disambiguate competing self-models. (**D**) Target behavior synthesis. After several cycles of (A) to (C), the currently best model is used to generate locomotion sequences through optimization. (**E**) The best locomotion sequence is executed by the physical device. (**F**) The cycle continues at step (B) to further refine models or at step (D) to create new behaviors.

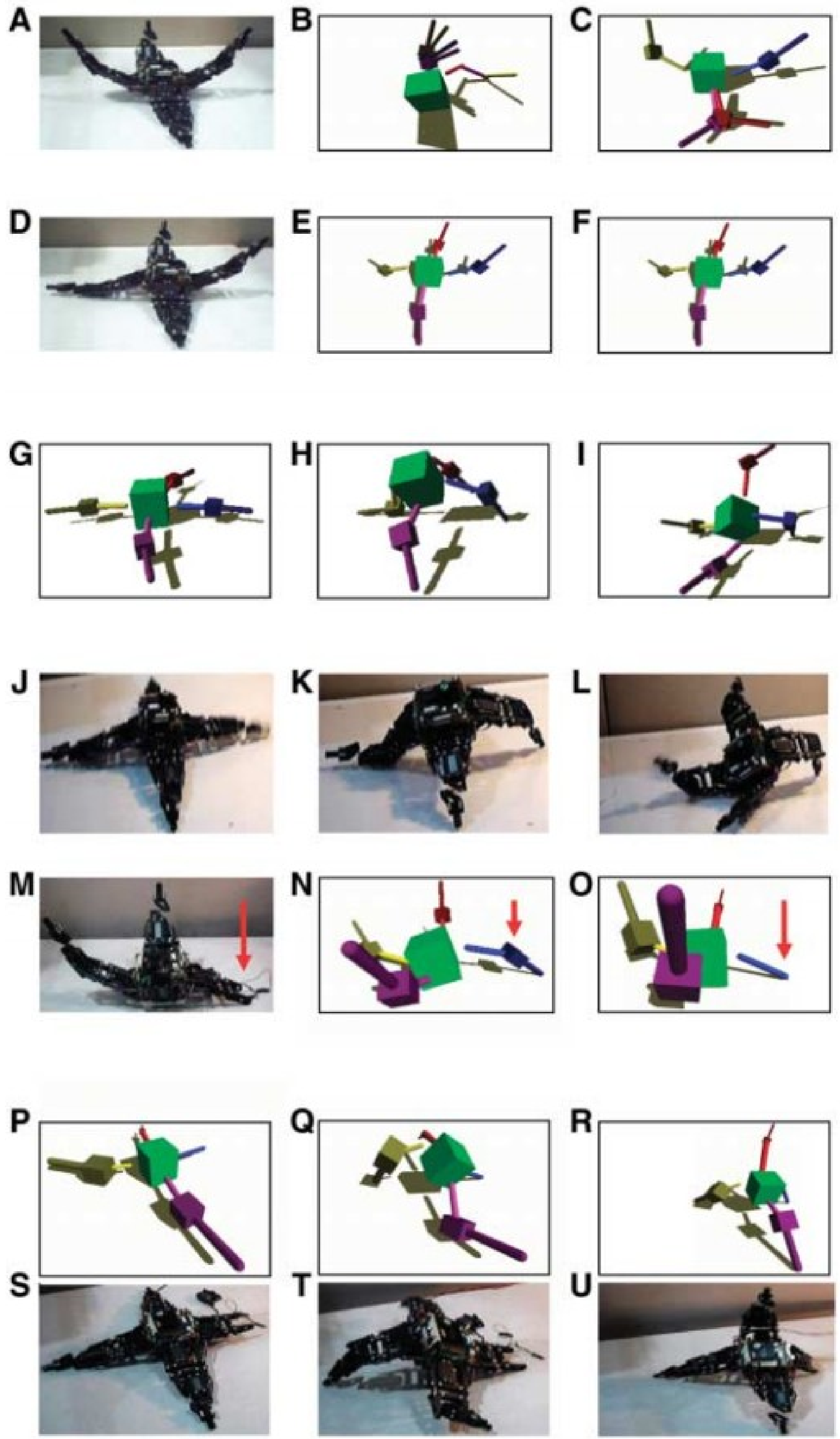


Fig. 2. The robot continually models and behaves. The robot performs a random action (**A**). A set of random models, such as (**B**), is synthesized into approximate models, such as (**C**). A new action is then synthesized to create maximal model disagreement and is performed by the physical robot (**D**), after which further modeling ensues. This cycle continues for a fixed period or until no further model improvement is possible (**E** and **F**). The best model is then used to synthesize a behavior. In this case, the behavior is forward locomotion, the first few movements of which are shown (**G** to **I**). This behavior is then executed by the physical robot (**J** to **L**). Next, the robot suffers damage [the lower part of the right leg breaks off (**M**)]. Modeling recommences with the best model so far (**N**), and using the same process of modeling and experimentation, eventually discovers the damage (**O**). The new model is used to synthesize a new behavior (**P** to **R**), which is executed by the physical robot (**S** to **U**), allowing it to recover functionality despite the unanticipated change.

Self-models. Before damage, candidate morphological models of the robot are encoded as 16 parameters, which are used to construct a simulation of the robot: For each body part the first parameter indicates which other body part it attaches to, and the second parameter indicates where on the perimeter of that part it is attached. It is assumed that the robot initially knows how many body parts it is composed of (nine), the size, weight and mass of each part, and that negative commanded angles rotate body parts downward, and positive angles upward. The numbers are used to construct a virtual robot inside a three-dimensional dynamical simulator (*S6*). The virtual robot is induced to act by supplying it with the same joint angles that were sent to the physical robot, and the resulting two tilt angles are calculated for the main body by the simulator. The quality of a model is estimated by determining how closely the virtual sensor data matches the two tilt angles recorded from the physical device, for all actions that it has performed so far. In previous work (*S2*) we used other sensor types to determine which modalities were most useful for inferring body topology, and it was found that tilt information is sufficient, given the current experimental regime.

Models are optimized using a greedy random-mutation hill climber algorithm: during the first pass through the model synthesis component (Fig. 2B), computation begins with 15 random models, and the quality of each model is assessed. New models are produced by copying the originals, introducing small random modifications, and re-evaluating them. If the new model is more accurate than its original, it replaces it; otherwise, it is discarded. This process is continued for 200 iterations, and the resulting models are then transferred to the action synthesis component. Subsequent passes begin with the best models from the previous pass. Each new pass uses all previous actions performed by the physical robot for assessing model quality. In order to externally determine the success of an experiment after termination, the error of the best model was measured as the mean Euclidean distance between the centroid of each model body part, and where the centroid should be (sample distances are shown for model 1 in Fig. S2); this measure is not available to the robot during inference. Model disagreement was measured as the mean Euclidean distance between body parts across all candidate models at the end of an experiment (the greater the distance, the more disagreement).

Exploratory actions. An action is a set of desired joint angles that are sent to the robot's motors, causing it to move from one static pose to another. An action performed on the physical robot takes 4 to 6 seconds. In the action synthesis component (Fig. 2A), candidate actions are measured internally for how well they might extract new information from the physical robot, which is determined as how much model disagreement they cause, and how reliable they are. Disagreement is measured as the mean square error between the two tilt angles across the optimized models when all perform the same action. It has previously been shown that selecting tests which cause model disagreement accelerates model optimization (*21-24*). Reliability is measured by slightly altering each model, actuating each model pair using the action being considered, and then assessing the degree of similarity between the resulting tilt in the original and altered model pairs. Unreliable actions are those that would cause a similar but not identical model to produce very different sensor data from the physical robot when both perform the same action. These bifurcations provide further evidence for the nonlinear relationship between motor commands and resulting sensor data. The space of possible actions is restricted to actuating only one or two joints downward by 30 degrees and the rest upward by 30 degrees. This ensures that the robot does not obtain information about too many body parts at once, and that it does not assume extreme poses. The action synthesis component selects the action from among these 36 possible actions which induces maximum disagreement and reliability using the current models, but has not yet been performed. We have found that it is possible for the proposed algorithm to uncover test data that is too difficult for the modeling component to digest in its early stages if the algorithm is allowed to perturb all test parameters in parallel. To combat this we have proposed some additional algorithmic components that automatically tune the expected difficulty produced by a candidate test (*S4*,*S5*).

Stage I:
Generating internal models

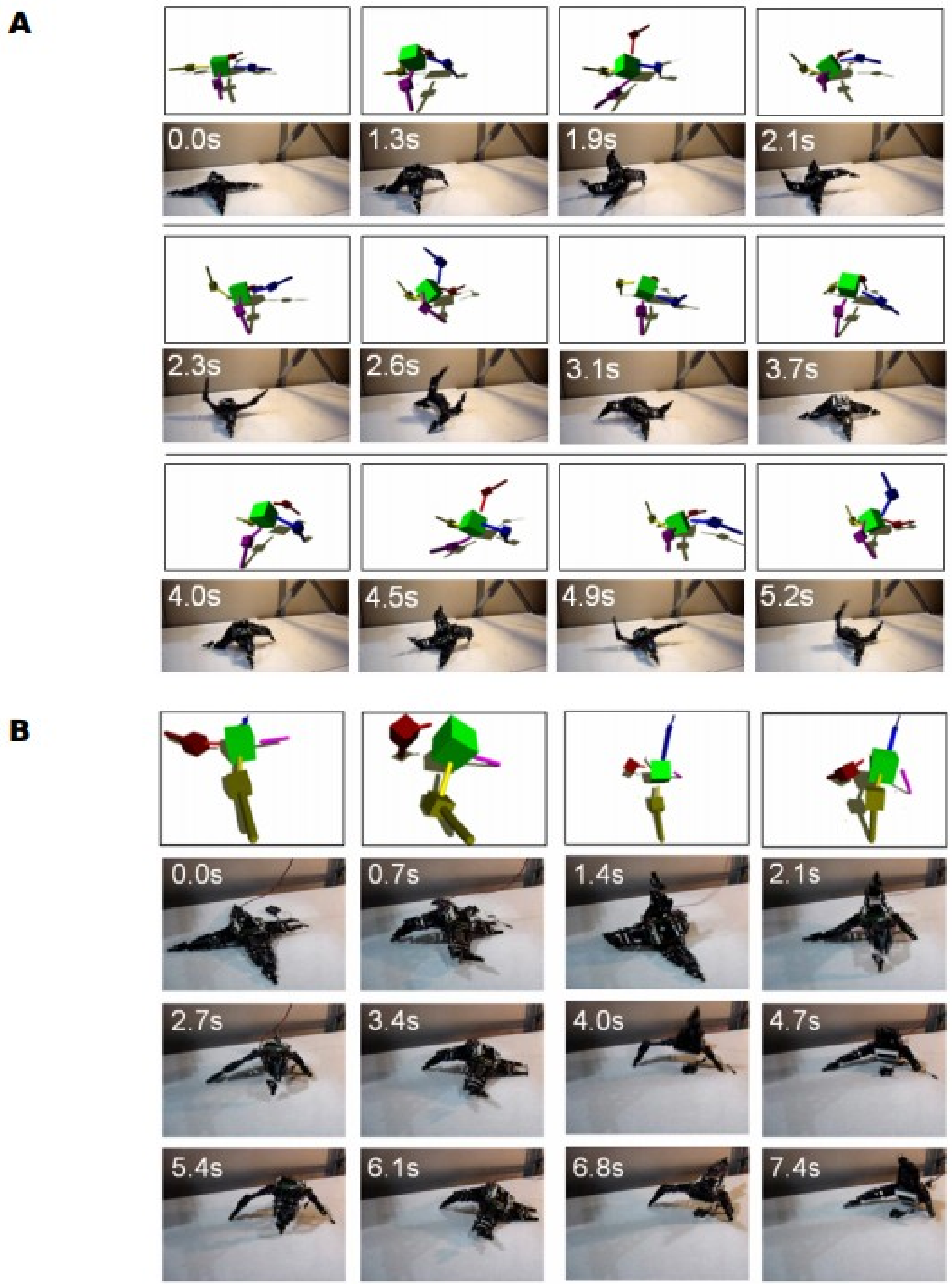


Fig. S1. Behaviors generated using automatically-inferred self-models. A sample gait synthesized by the intact robot (**A**) is shown running on the self-model, and its result on the physical machine. The gait uses the lateral legs to provide support, while the forward and back legs induce an undulatory motion along the direction of travel. After damage, a new self-model is used to create a new gait (**B**). This gait also uses an undulatory motion, but this time along the lateral legs; the asymmetry of the robot causes it to turn such that the undulatory motion again travels along the line of action, as in the intact robot.

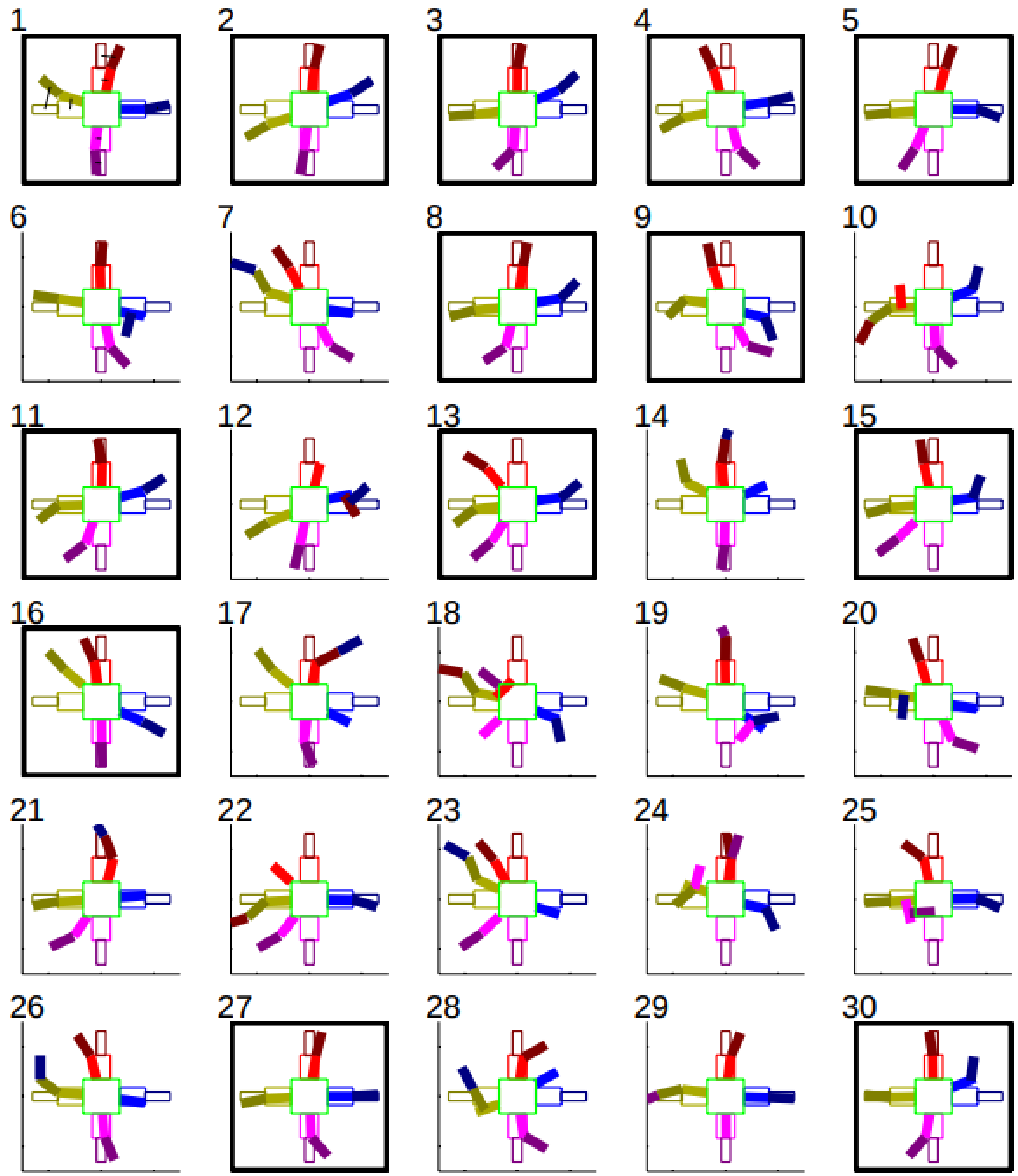


Fig. S2. The best internal morphological models produced by each of the 30 experiments using intelligent action synthesis. Thick colored lines indicate the placement of body parts in the model. Dashed outlines indicate the actual location of the correspondingly-colored body parts on the physical robot. Boxed images indicate experiments in which the topology was successfully inferred (this knowledge is hidden from the robot). Successful inference of morphology is measured by whether all of the body parts are attached to their correct proximal body parts. Thin black lines on model 1 indicate the Euclidean distances between the positions of the model's body parts and where they should be.

‘Managed Challenge’ Alleviates Disengagement in Co-evolutionary System Identification

Josh C. Bongard
Computational Synthesis Laboratory
Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14850 USA
josh.bongard@cornell.edu

Hod Lipson
Computational Synthesis Laboratory
Sibley School of Mechanical and Aerospace Engineering
Cornell University
Ithaca, NY 14850 USA
hod.lipson@cornell.edu

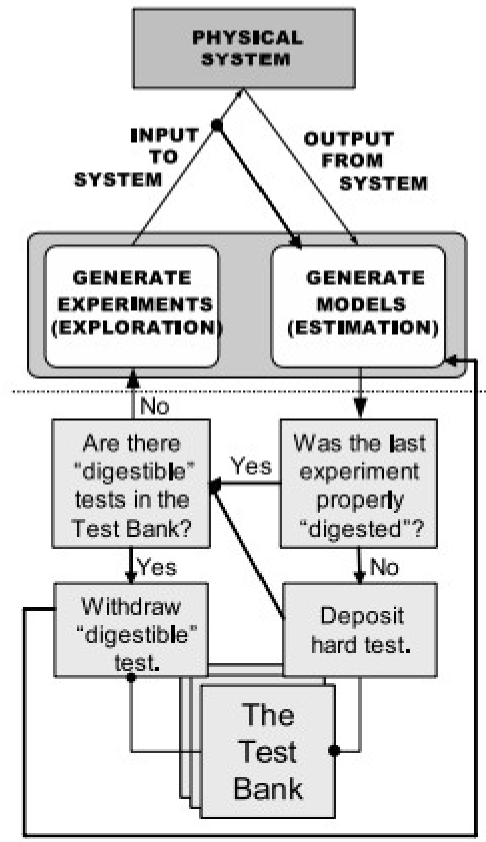
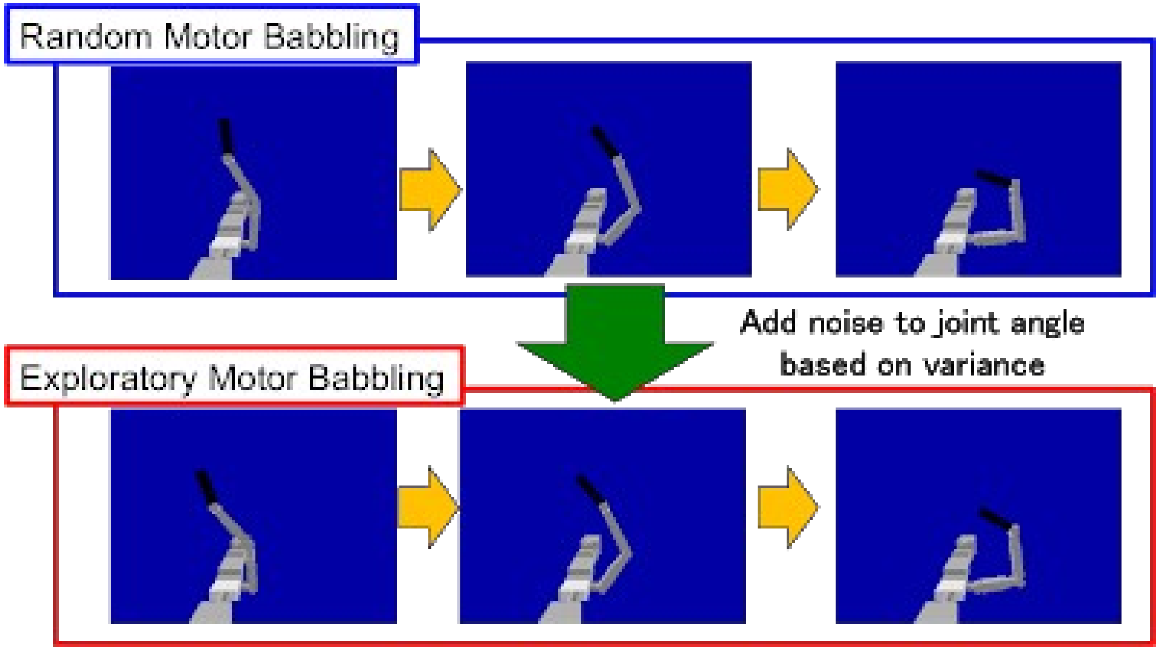


Figure 2: The enhanced estimation-exploration algorithm. The original algorithm is shown above the dotted line; the mechanism for ‘managing challenge’ is shown below the line.


Random Motor Babbling
Add noise to joint angle
based on variance
Exploratory Motor Babbling

COMPUTER SCIENCE

What Do Robots Dream Of?

Robots that create and update internal models of their own structure may be able to navigate the world in a more robust way and provide a test bed for models of self-awareness.

Christoph Adami

Perhaps robots aren't so different from us after all. Like us, they need to constantly ascertain where they are in the world, and like us, they work better if they have an accurate sense of self. On page 1118 of this issue, Bongard *et al.* (*1*) show that robots equipped with an algorithm that infers their own physical structure from stored sensory data—dreams of their prior actions, so to speak—perform better in a simple forward locomotion task than robots whose decisions are not dream-inspired. Furthermore, robots that use these self-models to plan future actions can recover autonomously from injuries, by adapting their gait to compensate for the changed circumstances.

A robot's most formidable enemy is an uncertain and changing environment. Typically, robots depend on internal maps (either provided or learned), and sensory data to orient themselves with respect to that map and to update their location. If the environment is changing or noisy, the robot has to navigate under uncertainty, and constantly update the probabilities that a particular action will achieve a particular result. The situation becomes even worse if the robot's own shape and configuration can change, that is, if its internal model becomes inaccurate. In most cases, such an event constitutes the end of that particular robot's adventure.

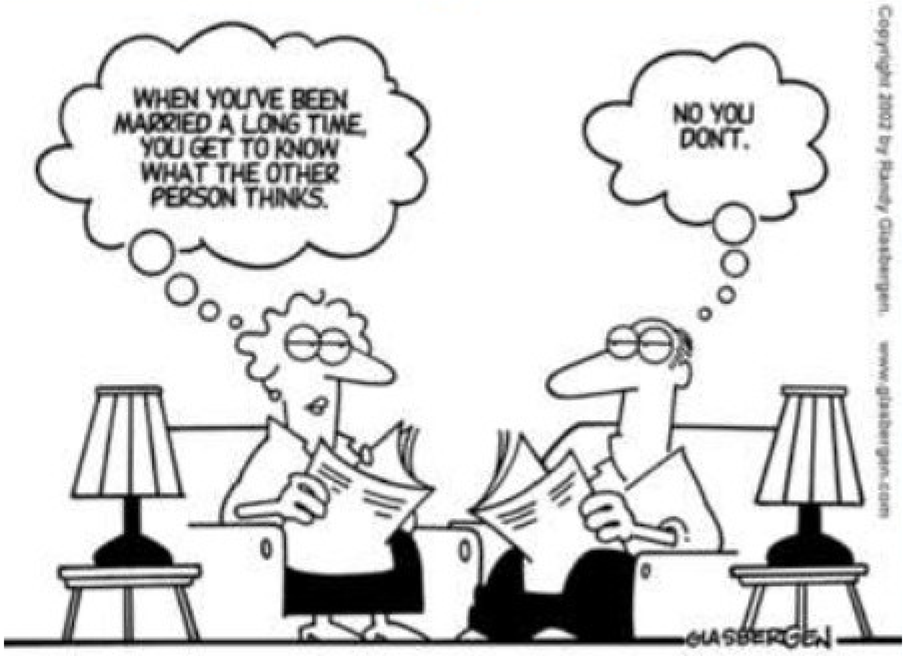
Bongard *et al.* aim to improve a robot's robustness in an environment that may include damage to the robot. At the beginning of a self-modeling cycle, a four-legged robot without an internal model of itself performs actions (while on a flat surface), and records its own response via tilt sensors and angle sensors in its joints. The robot then computationally tests candidate self-models, by re-imagining the actions it just performed and comparing the behavior of the model with its memory of the results—that is, the robot tries to explain the observed relationship between sensory data and leg actuation by making assumptions about its own configuration.

Even though the number of tested models is comparatively small (by only allowing a limited arrangement of limbs and their length), it is easy to imagine that many models can end up explaining the recorded behavior equally well (or equally badly). In the next stage of the cycle, the robot uses these equivalent models to find an action that would serve as the best way to discriminate among them. In other words, we could fancifully imagine the robot thinking: "Well, these three models all seem to work equally well with what I remember, but it seems to me that if I stick what I think is one of my legs out just so, then I can discover if I have a fourth leg or not." To narrow the choice of models, the robot then proceeds to test the action that provides the most information about the model's

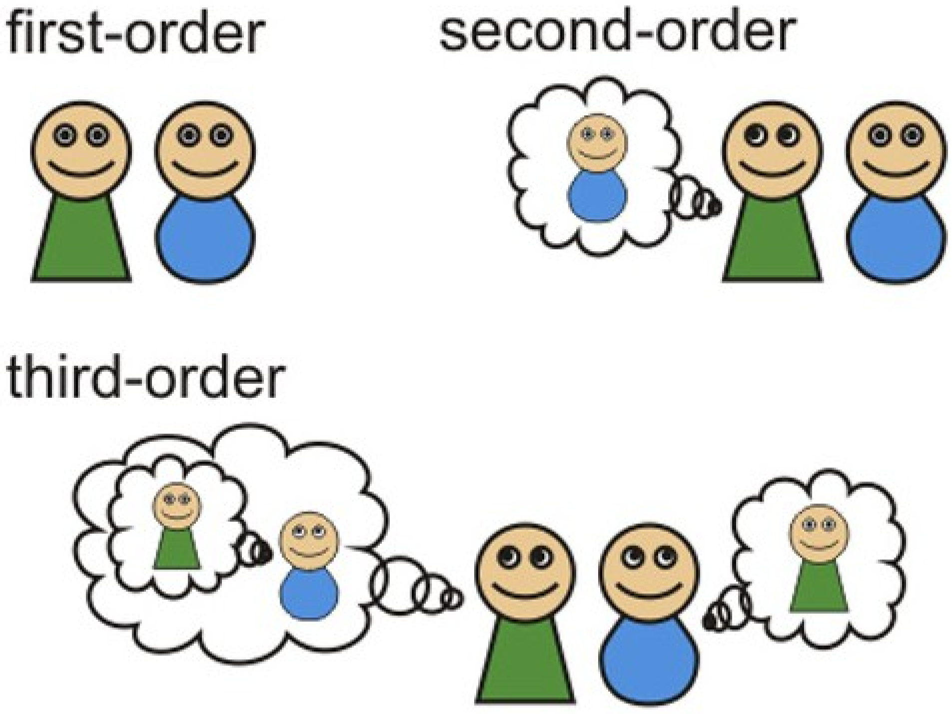
CREDIT: JOE SUTLIFF

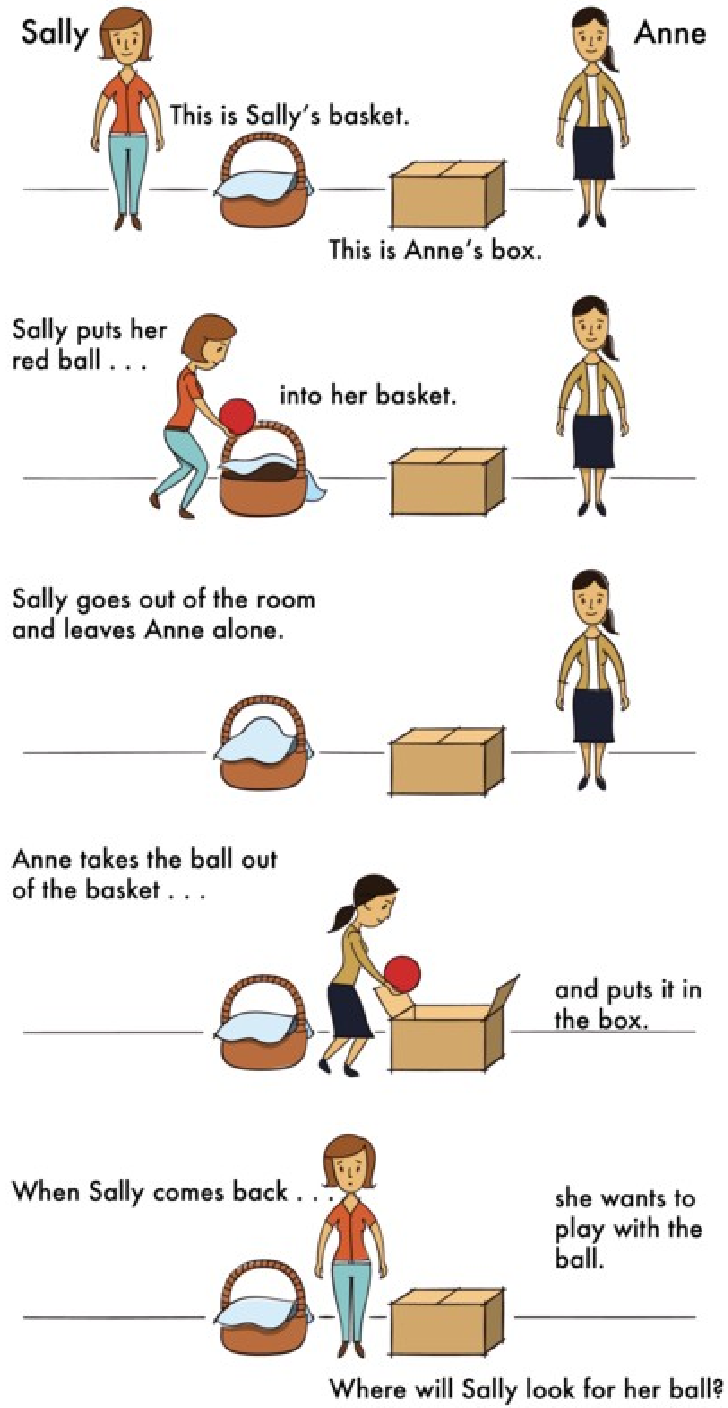
The author is at the Keck Graduate Institute of Applied Life Sciences, Claremont, CA 91711, USA. E-mail: adami@kgi.edu

Theory of Mind



Theory of Mind




Sally
Anne
This is Sally's basket.
This is Anne's box.
Sally puts her
red ball . . .
into her basket.
Sally goes out of the room
and leaves Anne alone.
Anne takes the ball out
of the basket . . .
and puts it in
the box.
When Sally comes back . . .
she wants to
play with the
ball.
Where will Sally look for her ball?