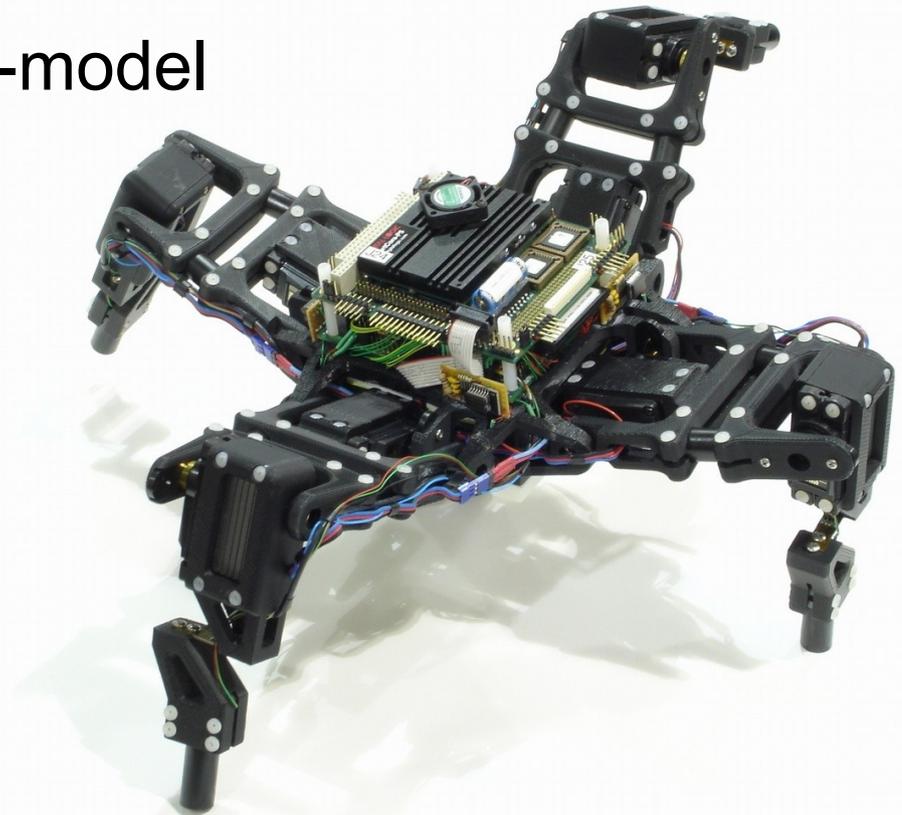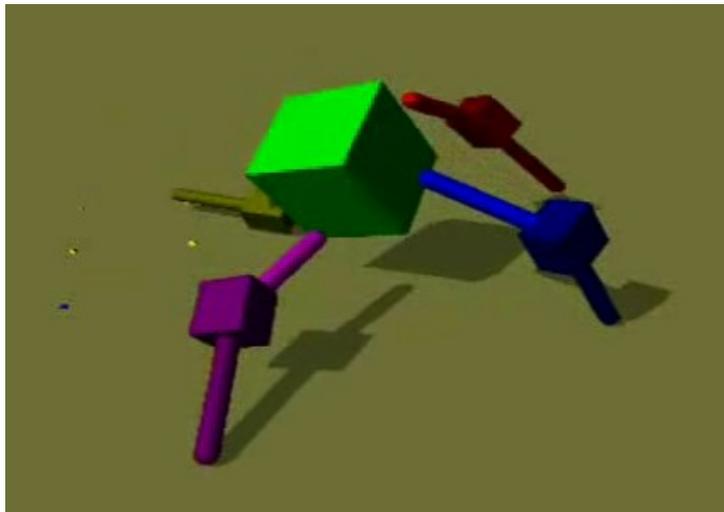# A Self-Modeling Quadruped Robot

- Josh Bongard, Victor Zykov, and Hod Lipson, "Resilient machines through continuous self-modeling", *Science*, 314, pp. 1118-1121, 2006.

- "Starfish" robot autonomously learns about its own body

- Develops internal **self-model** using an evolutionary algorithm

- Learns to walk based on self-model

- Can recover from damage

A Self-Modeling Quadruped Robot

- 9 body parts:
    - central body, 4 legs, 2 parts per leg
- 2 body tilt sensors (left-right, forward-back)
- 8 motorized joints
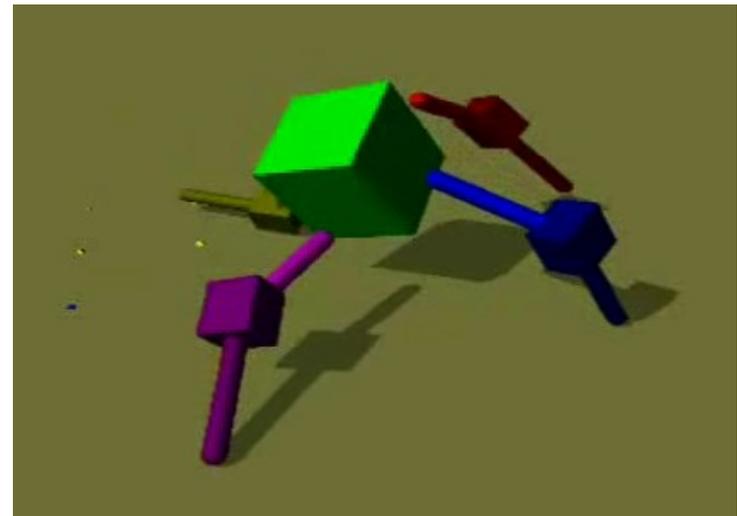- 8 joint angle sensors
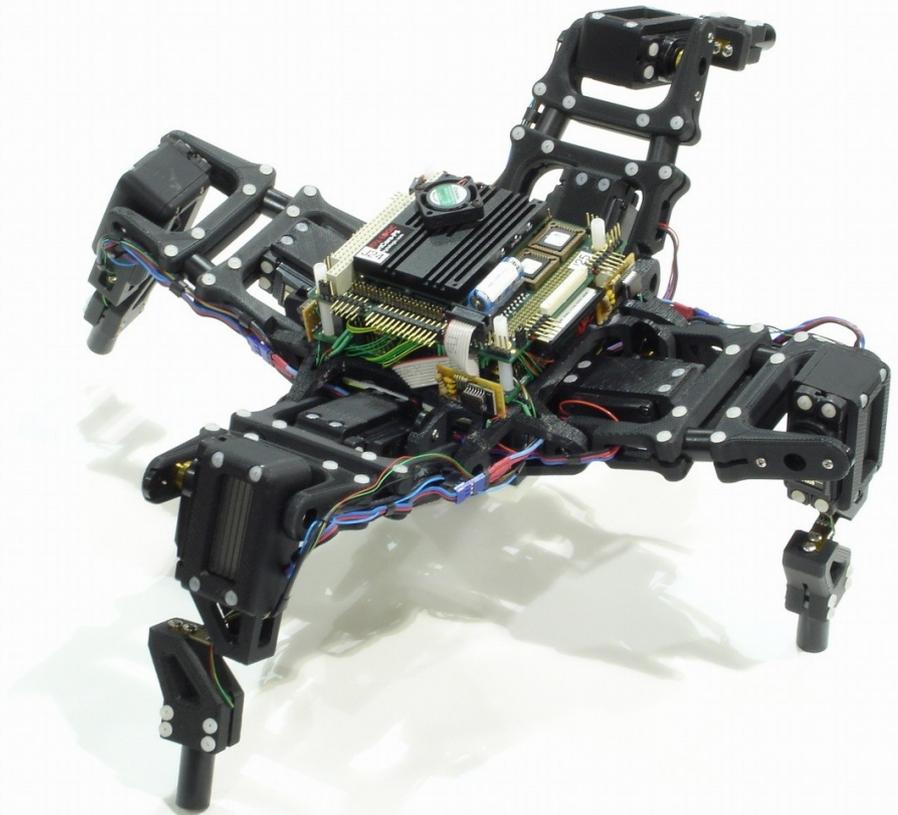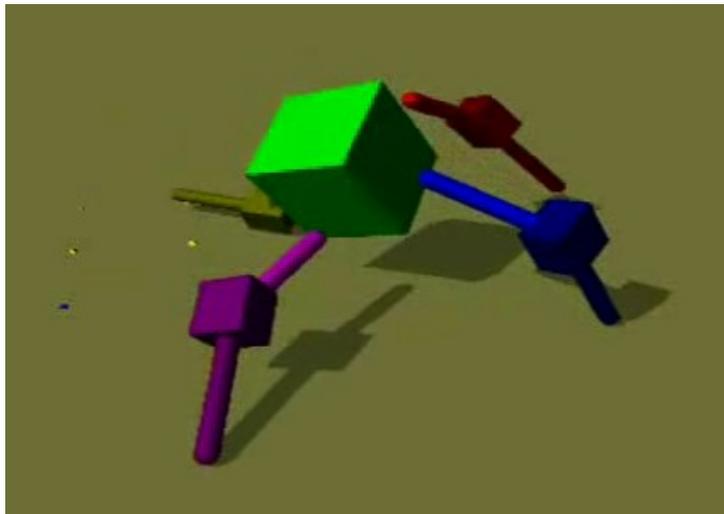
Self-Models

- Robot knows it has 9 body parts, but does NOT know which part is attached to which

- A **self-model** consists of **16 numerical parameters** that specify how the parts are attached to each other

- Robot maintains 15 competing self-models internally
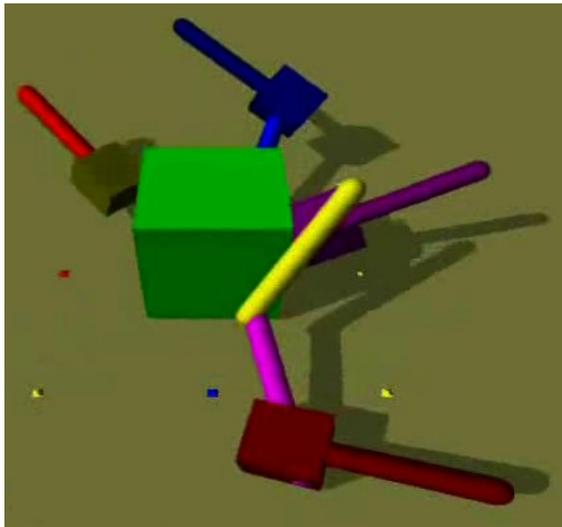
A poor self-model

A good self-model

Stage I: Learning About Itself

(a) 15 internal self-models are randomly created
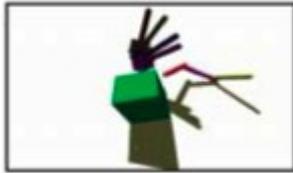
Stage I: Learning About Itself

(a) 15 internal self-models are randomly created

(b) Robot performs a randomly chosen motor action

Physical Action	Observed Result
action 1	result 1

Stage I: Learning About Itself

Physical Action	Observed Result
action 1	*result* 1

(a) 15 internal self-models are randomly created

(b) Robot performs a randomly chosen motor action

(c) Self-models evolve for 200 cycles. Fitness: ability
     to predict observed results of robot's actions so far
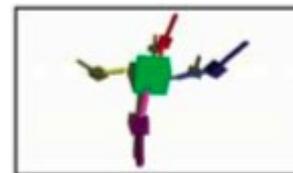
Stage I: Learning About Itself

Physical Action	Observed Result
action 1	*result* 1
action 2	

(a) 15 internal self-models are randomly created

(b) Robot performs a randomly chosen motor action

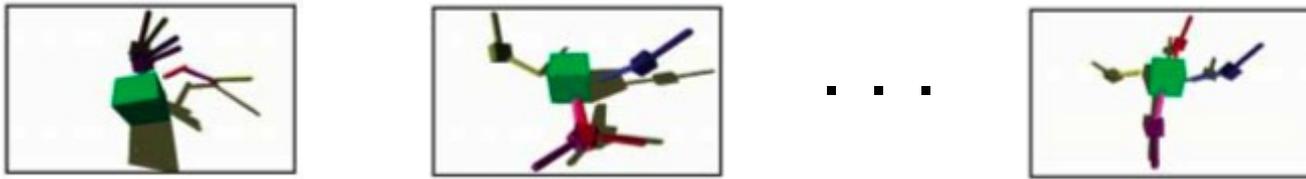(c) Self-models evolve for 200 cycles. Fitness: ability to predict observed results of robot's actions so far

. . .

(d) A new action is chosen that **maximizes disagreement** among the models about that action's predicted effect

Stage I: Learning About Itself

Physical Action	Observed Result
action 1	*result 1*
action 2	*result 2*

(a) 15 internal self-models are randomly created

(b) Robot performs a randomly chosen motor action

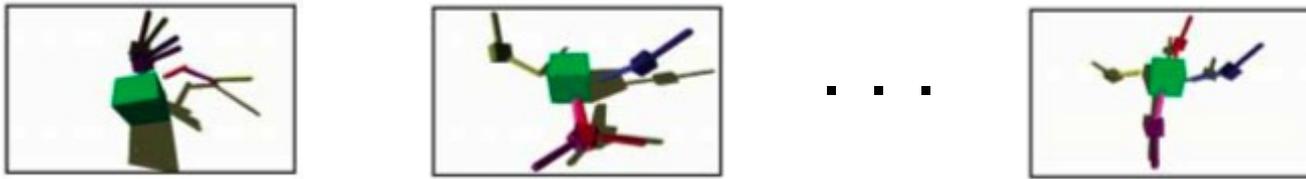(c) Self-models evolve for 200 cycles. Fitness: ability to predict observed results of robot's actions so far

(d) A new action is chosen that **maximizes disagreement** among the models about that action's predicted effect

(e) Robot performs the action and observes the result

Stage I: Learning About Itself

Physical Action	Observed Result
action 1	result 1
action 2	result 2

(a) 15 internal self-models are randomly created

(b) Robot performs a randomly chosen motor action

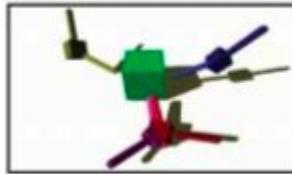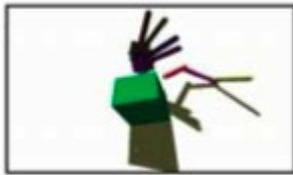(c) Self-models evolve for 200 cycles. Fitness: ability to predict observed results of robot's actions so far

 . . .

Stage I: Learning About Itself

Physical Action	Observed Result
action 1	*result 1*
action 2	*result 2*
action 3	

(a) 15 internal self-models are randomly created

(b) Robot performs a randomly chosen motor action

(c) Self-models evolve for 200 cycles. Fitness: ability to predict observed results of robot's actions so far

 . . .

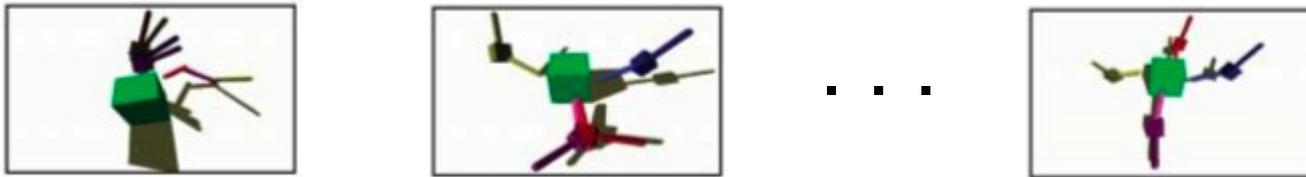(d) A new action is chosen that **maximizes disagreement** among the models about that action's predicted effect

Stage I: Learning About Itself

Physical Action	Observed Result
action 1	result 1
action 2	result 2
action 3	result 3

(a) 15 internal self-models are randomly created

(b) Robot performs a randomly chosen motor action

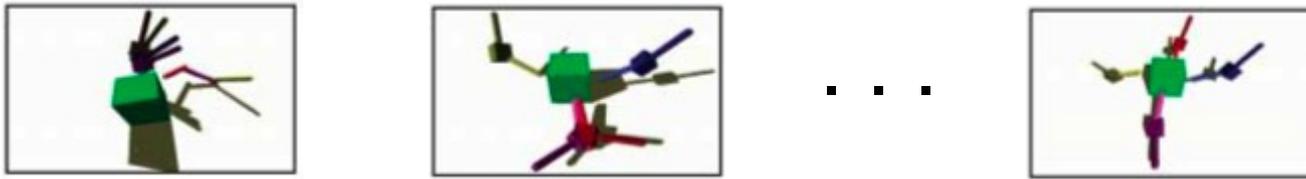(c) Self-models evolve for 200 cycles. Fitness: ability to predict observed results of robot's actions so far

. . .

(d) A new action is chosen that **maximizes disagreement** among the models about that action's predicted effect

(e) Robot performs the action and observes the result

Stage I: Learning About Itself

Physical Action	Observed Result
action 1	*result* 1
action 2	*result* 2
action 3	*result* 3
...	...

(a) 15 internal self-models are randomly created

(b) Robot performs a randomly chosen motor action

(c) Self-models evolve for 200 cycles. Fitness: ability to predict observed results of robot's actions so far

. . .

(d) A new action is chosen that **maximizes disagreement** among the models about that action's predicted effect
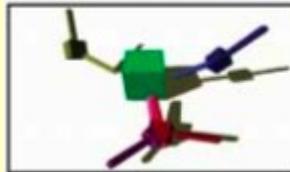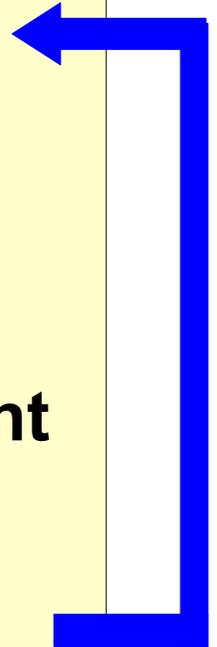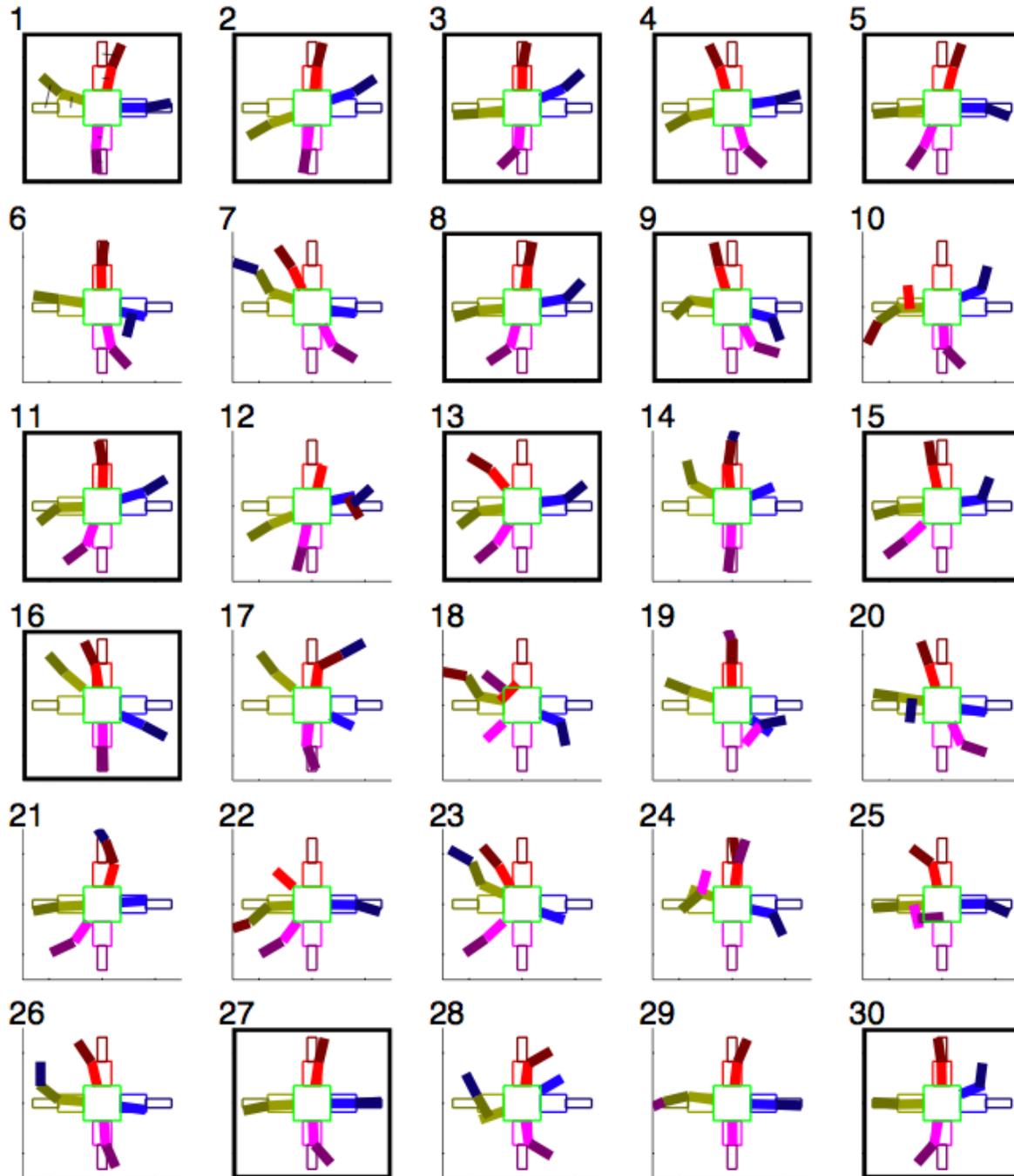
(e) Robot performs the action and observes the result

Steps (c)-(e) are repeated for 16 different exploratory actions

# Stage I: Results

30 independent experiments were performed

13 of them (43%) produced a self-model with the correct topology

Average model inaccuracy: 7.3 cm

Model-Driven Learning Disabled: Baseline 1

(a) 15 internal self-models are randomly created

(b) Robot performs 16 randomly chosen motor actions

(c) Self-models evolve **based on all 16 actions together**.
   Fitness: ability to predict observed results of actions.

. . .
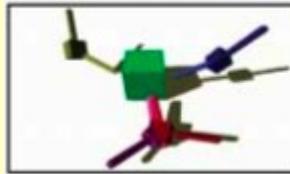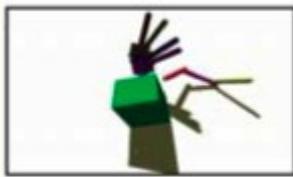
| Physical Action	Observed Result
action 1	result 1
action 2	result 2
action 3	result 3
action 4	result 4
. . .	. . .
action 16	result 16

Results: Baseline 1

- As before, 30 independent experiments were performed

- Only 7 of them (23%) produced a topologically correct self-model, compared to 43% with model-driven learning enabled

- Average model inaccuracy was 9.6 cm, compared to 7.3 cm with model-driven learning enabled

- Interleaving exploratory motor actions chosen on the basis of the evolving self-models significantly improves model synthesis, compared to randomly choosing all exploratory actions first, and then modeling.

Model-Driven Learning Disabled: Baseline 2

(a) 15 internal self-models are randomly created

(b) Robot performs a randomly chosen motor action

(c) Self-models evolve for 200 cycles. Fitness: ability
     to predict observed results of robot's actions so far
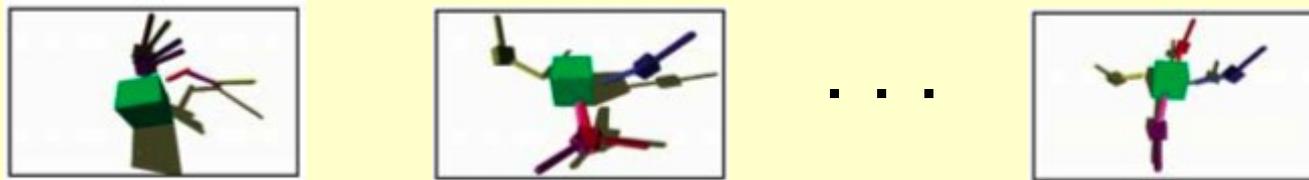
(d) A new action is **chosen randomly**

(e) Robot performs the action and observes the result

Steps (c)-(e) are repeated for 16 different exploratory actions

Results: Baseline 2

- As before, 30 independent experiments were performed

- Only 8 of them (26%) produced a topologically correct self-model, compared to 43% with model-driven learning enabled

- Average model inaccuracy was 9.7 cm, compared to 7.3 cm with model-driven learning enabled

- Interleaving exploratory motor actions chosen on the basis of the evolving self-models, compared to interleaving random exploratory actions, significantly improves model synthesis.

Stage II: Learning To Walk

- Start with the most accurate **self-model** generated in Stage I

- Use an **evolutionary algorithm** to evolve behaviors

  - a behavior is a set of numbers specifying a temporal sequence of joint angles

  - fitness of a behavior: how far does it cause the self-model to move forward in simulation?

- Test the **best evolved behavior** on the physical robot

- Repeat **30 times**, starting from different random populations of behaviors but using **the same self-model** in each case
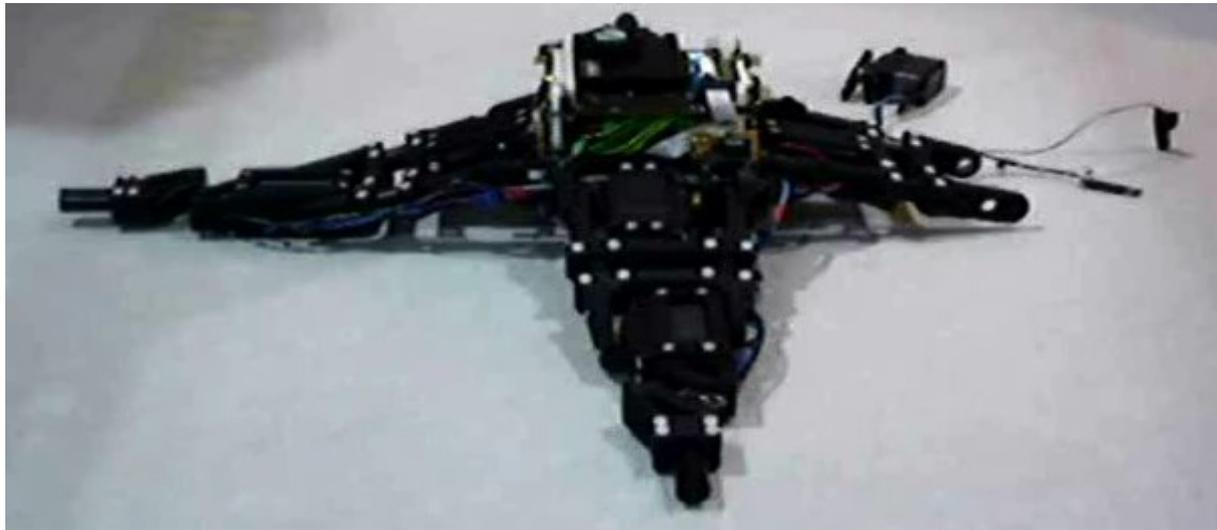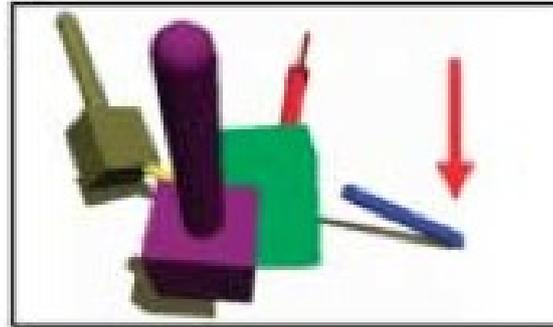
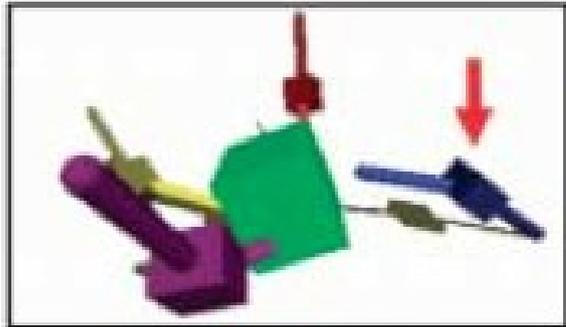# Stage II: Results
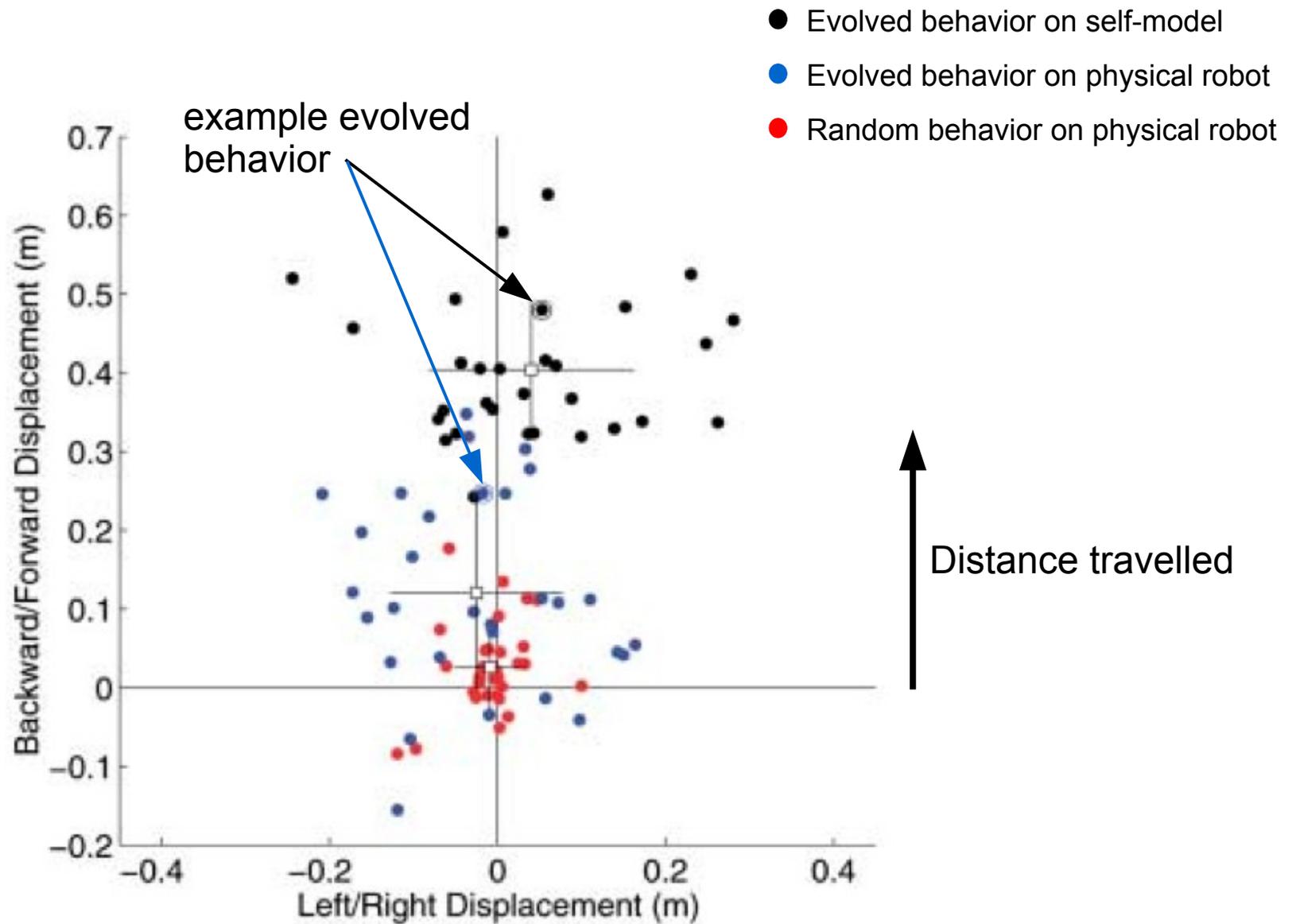
Stage II: Results

Damage Recovery

- If the robot becomes damaged, self-modeling and experimentation recommence with the best model so far

Damage Recovery

- If the robot becomes damaged, self-modeling and experimentation recommence with the best model so far

- The EA varies the relative sizes of leg parts within models until a new, more accurate self-model emerges
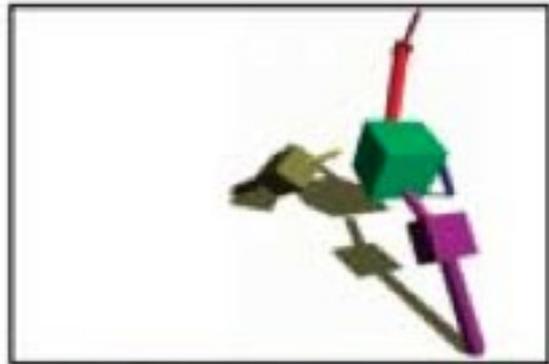
Damage Recovery

- If the robot becomes damaged, self-modeling and experimentation recommence with the best model so far

- The EA varies the relative sizes of leg parts within models until a new, more accurate self-model emerges

- The new, improved self-model is used to synthesize a new walking behavior

Damage Recovery: Results

Feelin' Good!