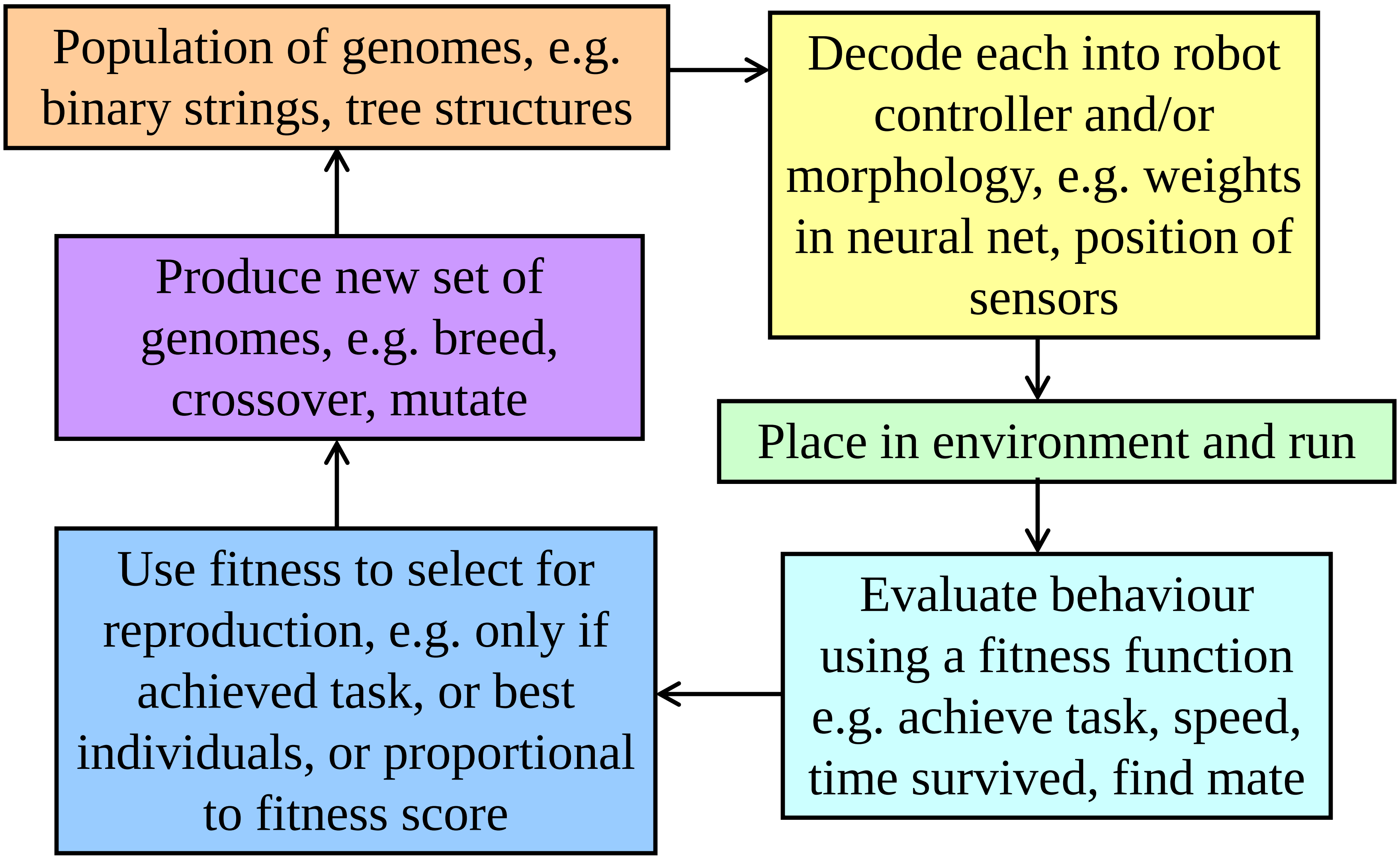
Evolutionary Robotics

IAR Lecture 13

Barbara Webb

Basic process

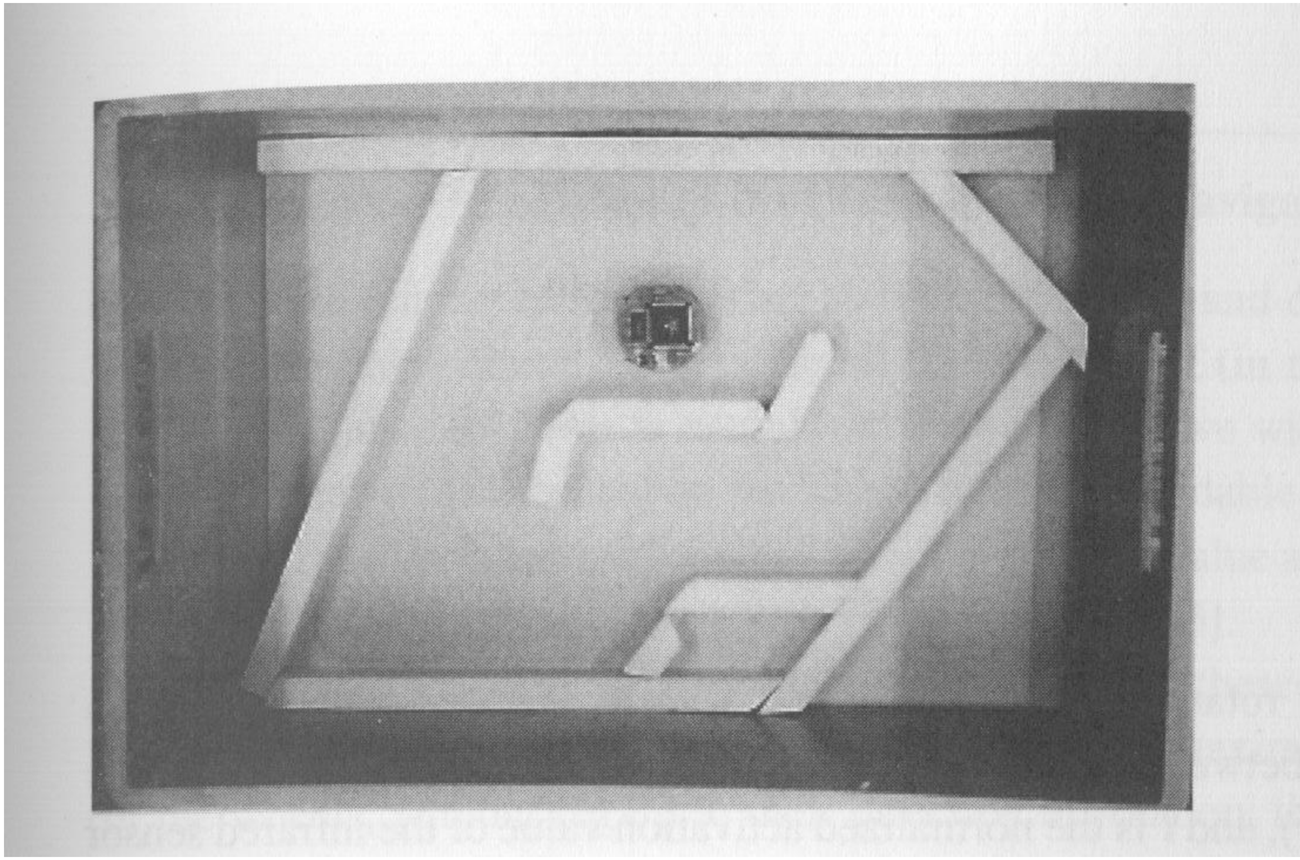


Motivation

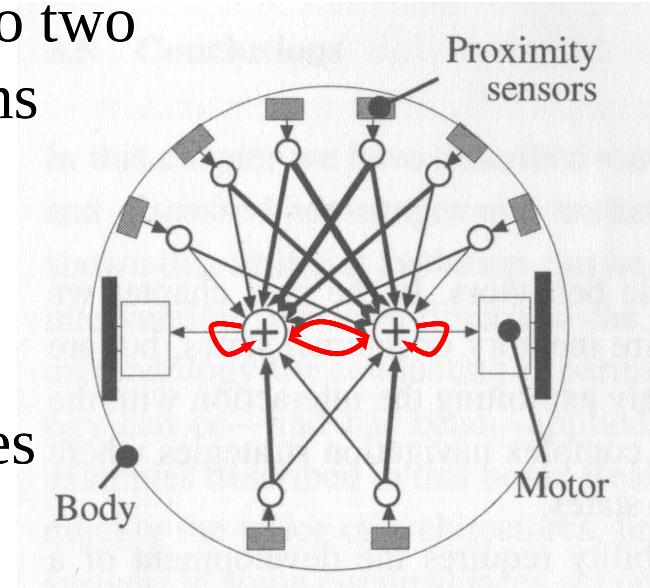
- Lack of design methods that will ensure the right dynamics emerge from the environment-robot-task interaction
- Automate the trial-and-error approach
- Avoid preconceptions in design
- Allow self-organising processes to discover novel and efficient solutions
- Good enough for biology (and might help us understand biology)

‘Typical’ example

Floreano & Mondada (1996): evolving Braitenberg-type control for a Khepera robot to move around maze



- Eight IR sensor input units, feed-forward to two motor output units with recurrent connections



- Standard sigmoidal ANN

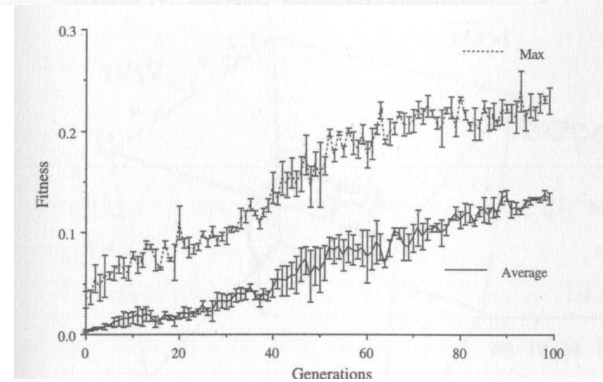
$$y_i = f\left(\sum_j^n w_{ij} x_j\right), where \; f(x) = \frac{1}{1+e^{-kx}}$$

- Genome – bit string encoding weight values

- Fitness function: $\Phi = V(1-\sqrt{\Delta v})(1-i)$

where i is highest IR value, $V = |v_{left}| + |v_{right}|$

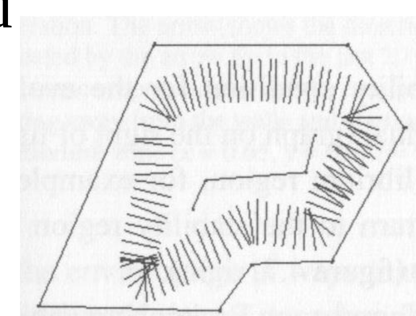
$$\Delta v = |v_{left} - v_{right}|$$



- Population of 80, each tested for approx 30s

- Copied proportional to fitness, then random paired single point crossover and mutation (prob.=0.2)

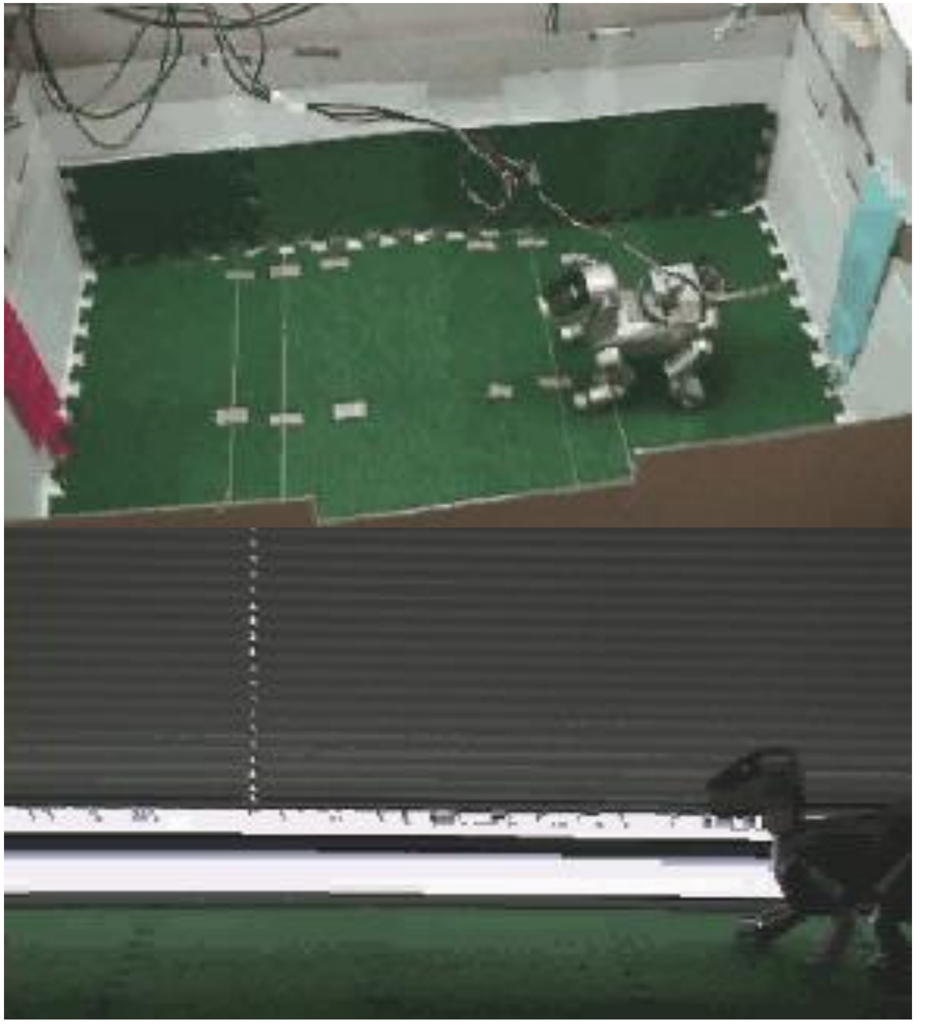
- 100 generations, get smooth travel round maze



Similar approach has been used to evolve controllers for more complex robots

AIBO (Hornby et al 2000)

Blimp (Zufferey et al, 2002)



Issues for the basic process

- How to represent the robot controller
- How to determine to fitness
- How large a population
- How strongly to select
- How to introduce variation, and how much
- How to decide when to stop (fitness threshold, convergence, plateau, time…)

Extensions to the basic process

- Incremental evolution
- Co-evolution
- More powerful or flexible genetic encoding schemes
- Better use of simulation to speed process without compromising transfer to real world
- Evolving morphology

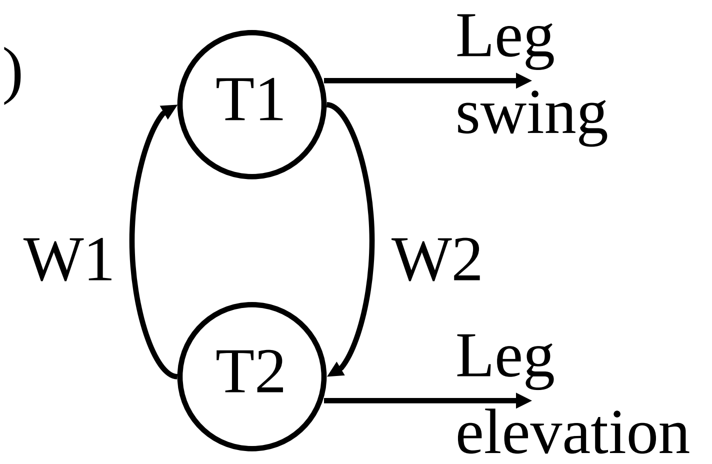
Incremental Evolution

- For complex tasks, early generations may have zero fitness – and slope is too steep to hill-climb
- Two approaches:
  - Start with simpler fitness function, and increase difficulty in several stages
    - N.B. this could include evolving different parts of the controller separately, then combining
  - Start with simpler environment, and gradually increase complexity
    - N.B. this could include starting in simulation and later transferring to robot

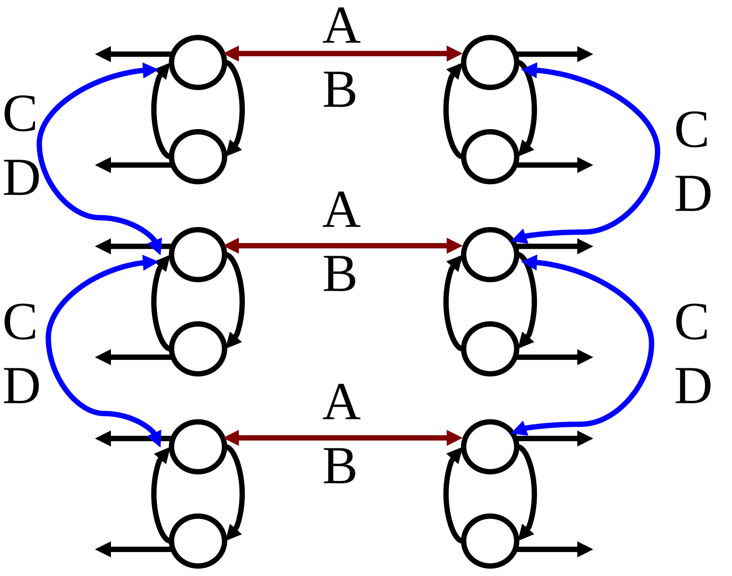
Example: Lewis (1992) evolving six-legged walking

Stage one: evolving two weights (W1,W2) and two thresholds (T1,T2) for co-ordinated single leg motion.

1a: neuron states are non-zero
1b: neurons in opposite states
1c: at least one neuron changes state
1d: damped oscillations
1e: non-damping oscillations
1f: increased oscillation magnitude
1g: oscillation over entire range



Stage two: evolve four weights (A,B,C,D) for inter-leg co-ordination.

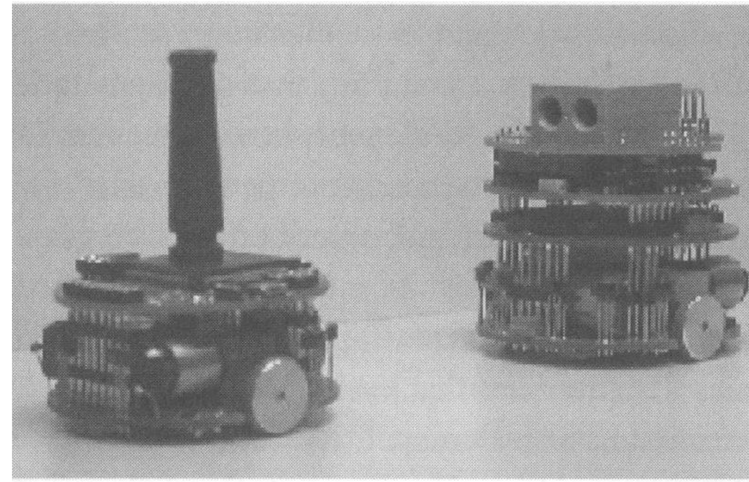


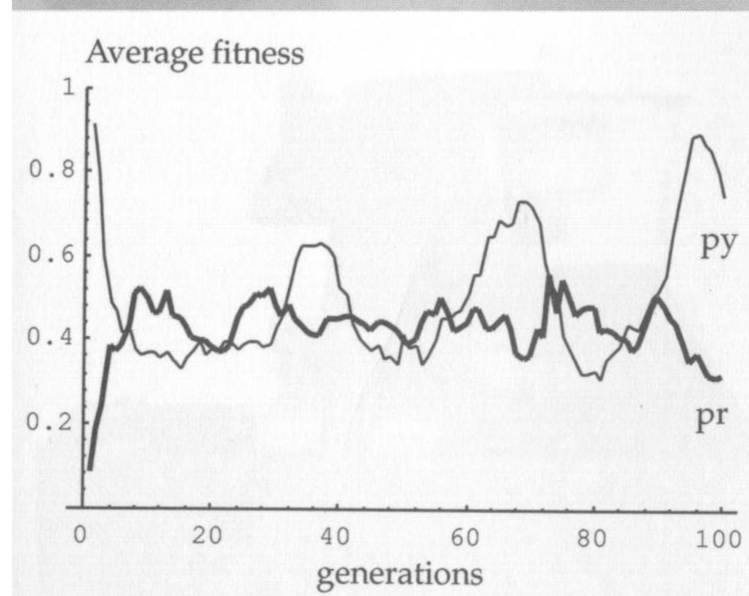
Fitness = aO + bL –cT
Where O is oscillation
L is length of travel
T is degrees turned

- Using small population (10), evolved oscillation in 10-17 generations, and walking in another 10-35.
- Sometimes population split between tripod and wave gaits, but tripod would eventually ‘win’
- Evolved to walk backwards due to robot mechanics

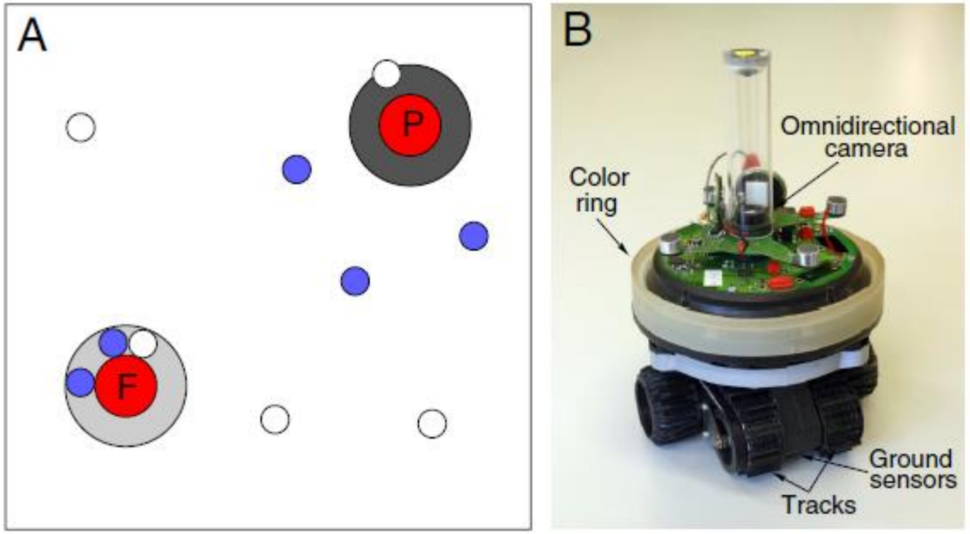
Co-evolution

- Have two or more 'species' competing in one environment
  - E.g. Floreano et al (1998) 'predator vs. prey'
- Each species thus has to evolve in a changing environment
- Potential for unsupervised incremental evolution
- However can also result in cycling





Evolution in collective robots



- Mitria et al 2009
- Fitness: positive for staying at food, negative for being near poison, can only recognise in near vicinity.
- Robots evaluated in groups of 10, 100 groups per generation.
- ‘Inadvertent’ signal of food location by robot’s own light leads to evolution of light approach in others, potential overcrowding. If then allow evolution of signalling some robots evolve to ‘lie’ by turning off their light on food; but this reduces evolutionary pressure to approach light.
- Result is complex balance with mixed strategies.

Alternative encodings

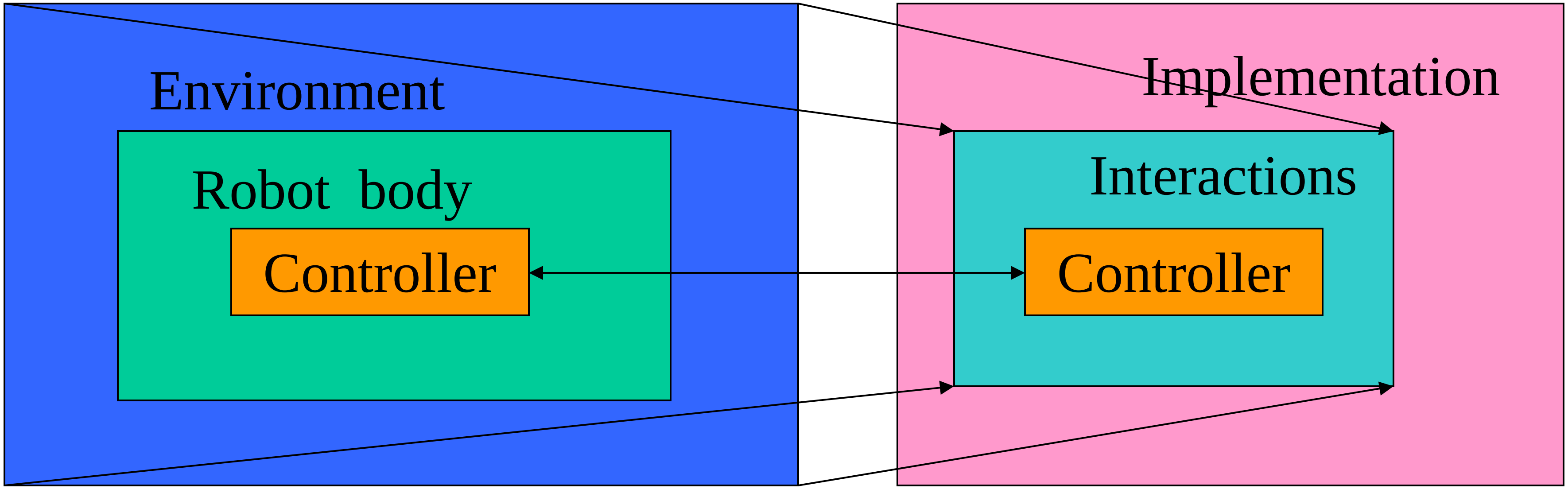
- Use modular networks
  - Reduces risk of disruptive crossover
- Allow changes in genome length
  - Often useful to enforce network symmetry or to allow sections to repeat
  - Can have genome specify growth process (developmental robotics)
- Evolve structured programs rather than networks (e.g. trees, graphs, L-systems)

Better use of simulation

- Evaluating every member of the population on a real robot severely limits population sizes, generations, and evaluation time - and requires robust rechargeable robots.
- Robot controllers developed in simulation often fail when tested in the real world.
- Effective transfer seems to require realistic, hard-to-build, and probably slow simulations.
- Jakobi (1997) proposed “radical envelope of noise” hypothesis to get around these constraints

“Simulations cannot accurately model everything”

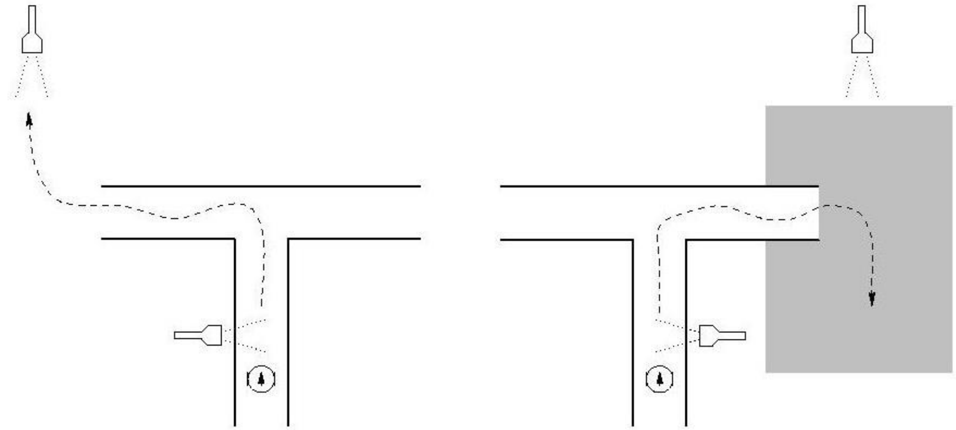
“Simulations cannot accurately model anything”



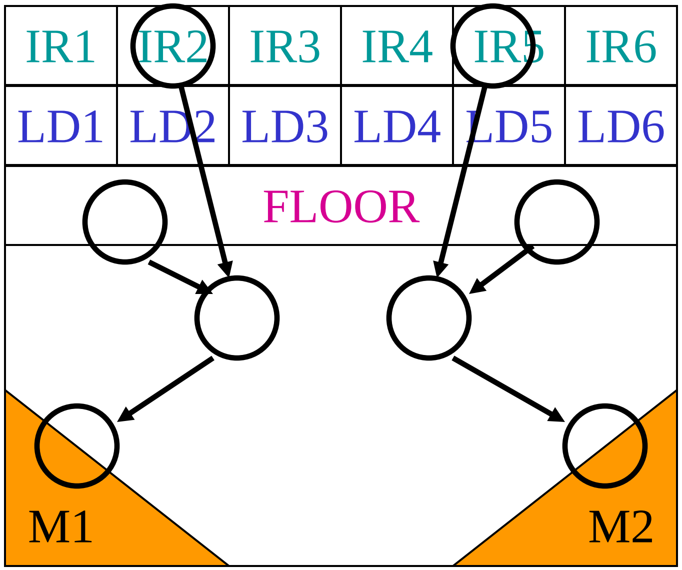
• Behaviour is determined by limited number of interactions – the ‘base set’ which can be modelled simply (with some inaccuracy)

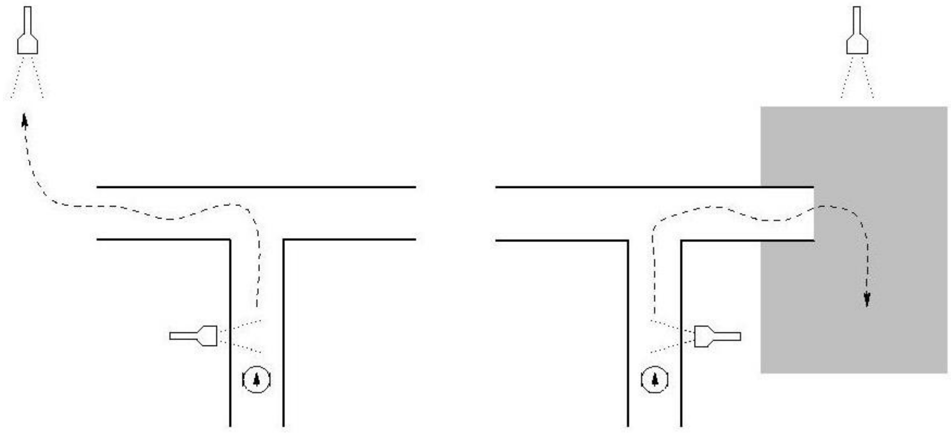
• Ensure the evolved controller is ‘base set exclusive’ and ‘base set robust’ by randomly varying everything else during evolution

- E.g. Jakobi & Quinn (1998)
- Task:



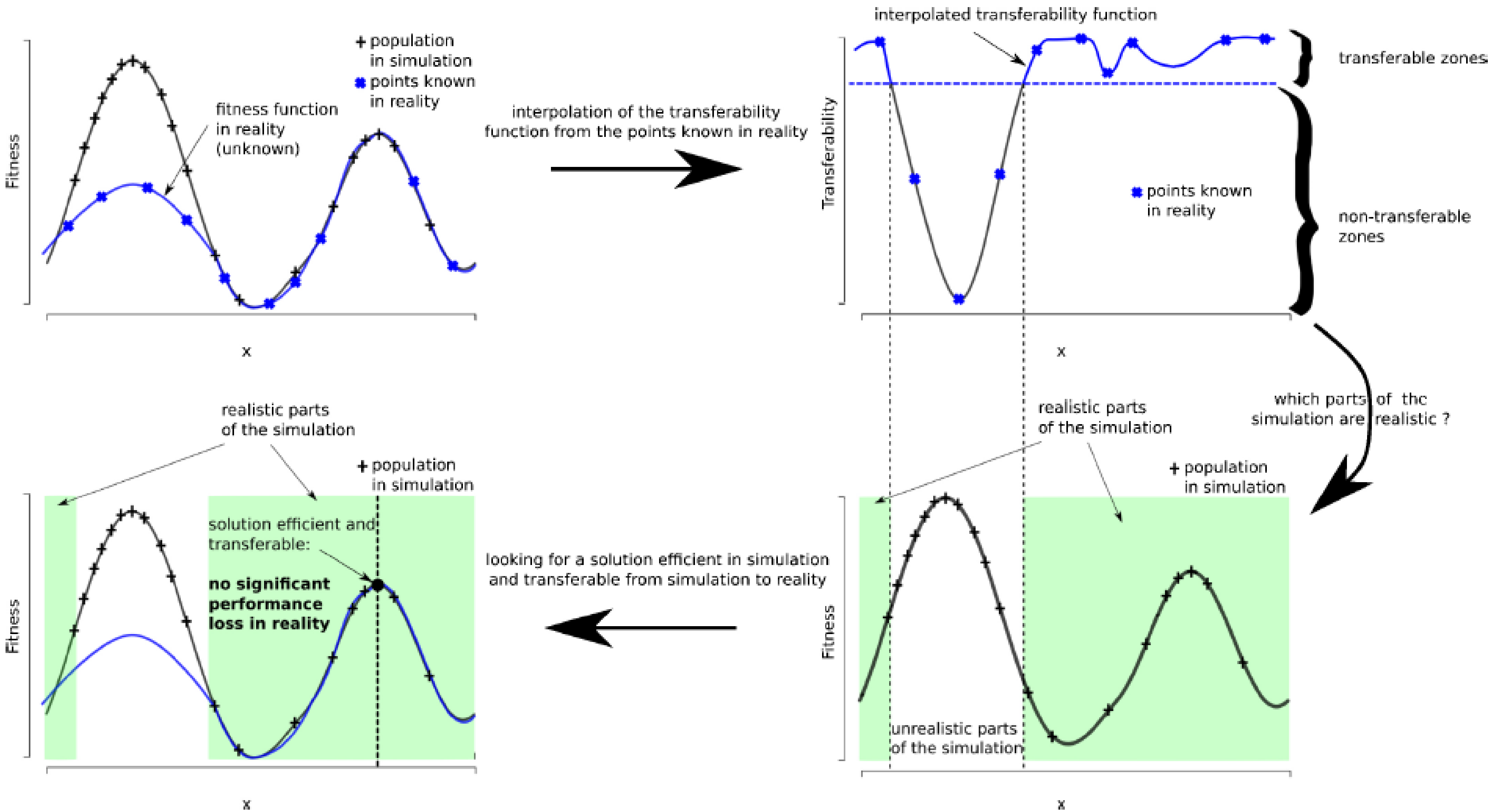
- Using spatially determined encoding: genome specifies position of neurons and their connections in development space, with symmetry
- Using staged evolution





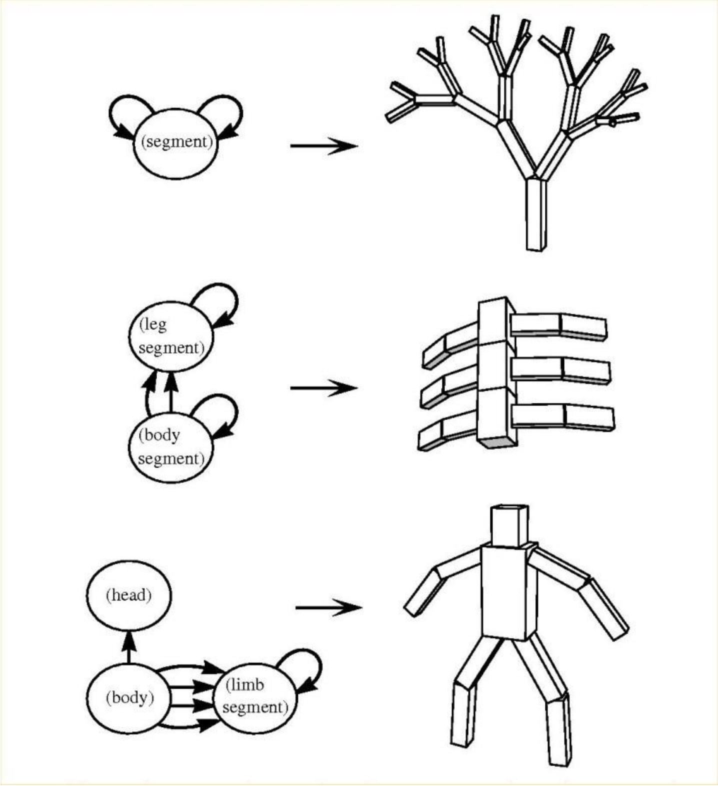
- Simulation uses simple look-up tables for:
  - Movement in response to motor commands
  - IR values for walls
  - Light sensor response to bright vs. normal light
- Introduces substantial random variation e.g.
  - Wheel offsets of ±1cm/s
  - Corridor length 40-60cm, width 13-23cm
- After 6000 generations, successful in completing task, and transferred successfully to real robot.

‘Transferability’ approach (Koos et al. 2013): optimize for task fitness *and* transferability

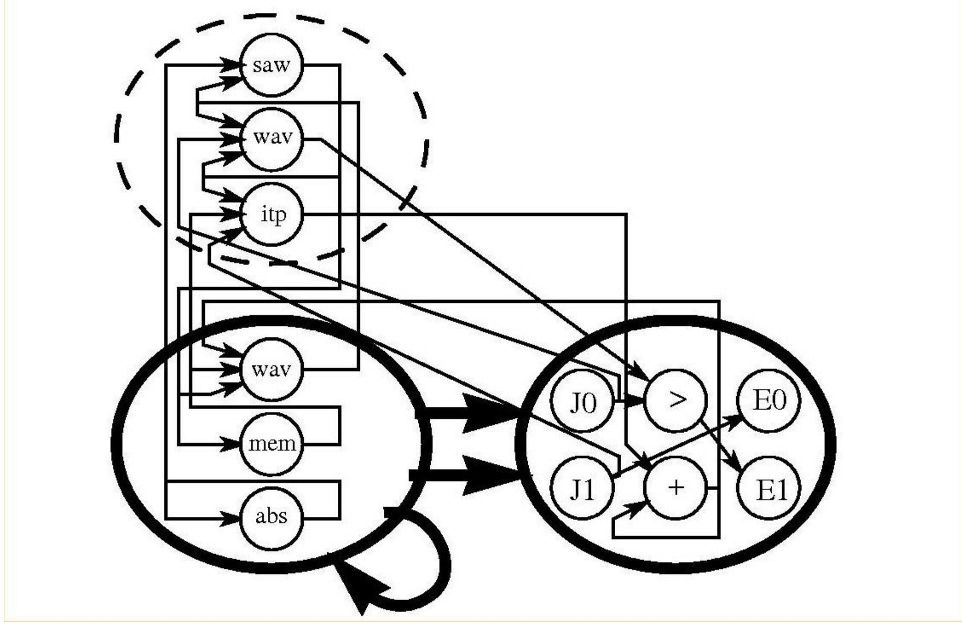


Evolving morphology

- Usually in simulation, e.g. Sims (1994)
- Directed graph representation of bodies and controllers

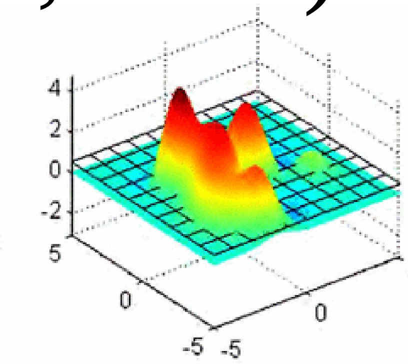


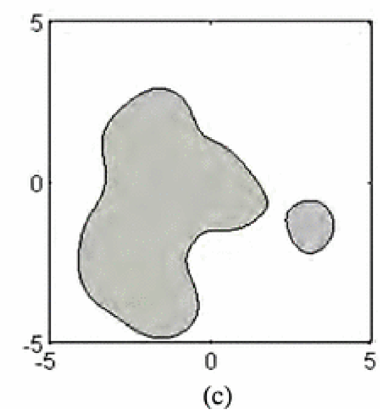
Segments contain sensors, effectors and simple processor nodes, which can pass scalar values in network



Using 3-D printing with mixed materials (Hiller & Lipson, 2012)

- Shape description is a thresholded mixture of 3D gausssians, each representing a different material
- Genome is set of points, each with density, falloff distance, and material index; one material can be actuated, changing its volume by 20%.
- Fitness is distance moved in 10 actuation cycles

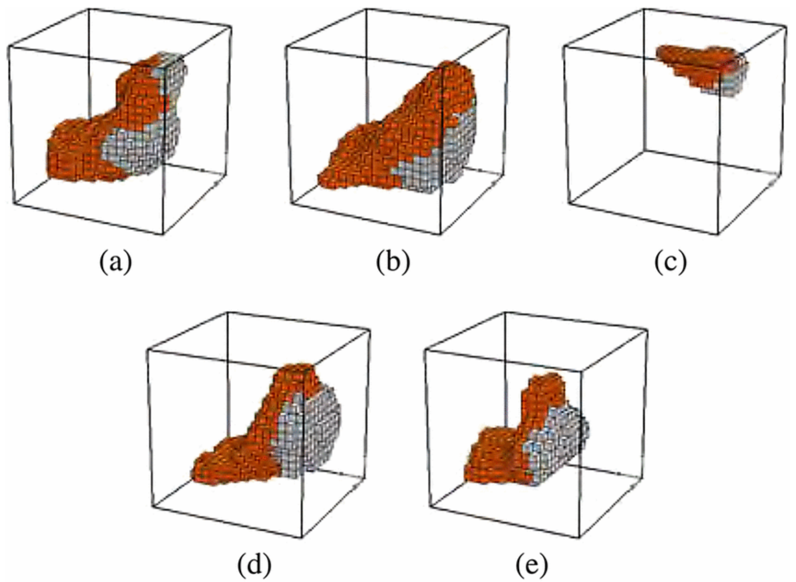


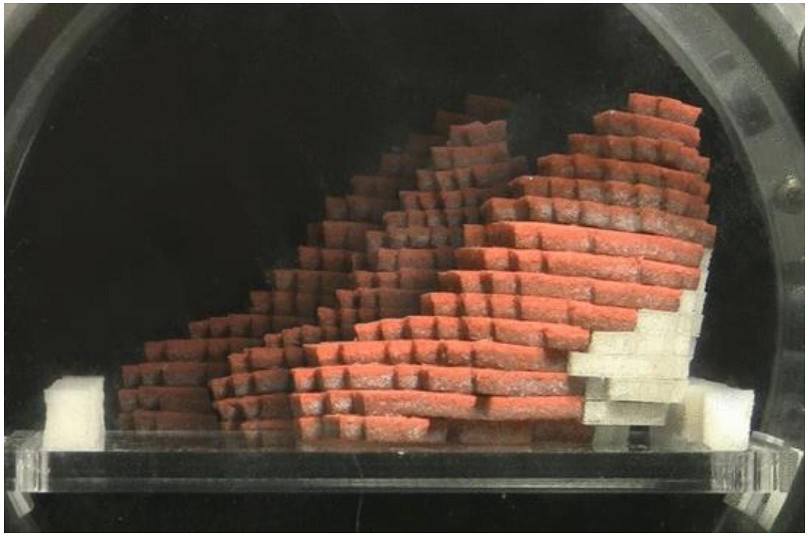


2D illustration of thresholded gaussians

Using 3-D printing with mixed materials (Hiller & Lipson, 2012)

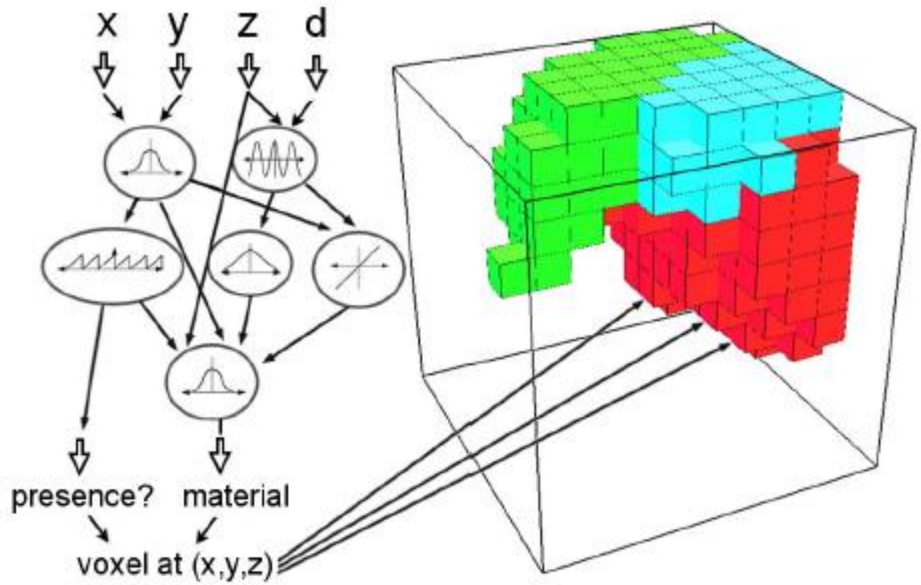
- All solutions found are similar: 'scoot' by expanding forward, tipping weight onto static material (white), contracting rear, and tipping back





Using 3-D printing with mixed materials (Cheney et al, 2014)

- Evolve using richer structural description: composite pattern producing network (CPPN)
- Different material types: actuation in opposite phase; passive soft or stiff
- Evolve with additional constraints: minimize size, or internal volume, or minimize actuation (energy costs)



Remaining Issues

- Resulting robots are often very hard to analyse – not necessarily any gain in understanding of the problem or its solution.
- Assumptions are not completely avoided, but instead built into the fitness function, the architecture, or the simulation variables.
- Not yet a convincing demonstration of greater efficiency than designing by hand.
- Still not clear that can evolve complex control in a reasonable time span.
- May be best seen as one of many tools for meta-heuristic optimisation.

References

Floreano, D.; Mondada, F., Evolution of homing navigation in a real mobile robot, *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on* , vol.26, no.3, pp.396,407, Jun 1996

M. Anthony Lewis , Andrew H. Fagg , Alan Solidum (1992) Genetic Programming Approach to the Construction of a Neural Network for Control of a Walking Robot *IEEE International Conference on Robotics and Automation*

Floreano, Dario, Stefano Nolfi, and Francesco Mondada. "Competitive co-evolutionary robotics: From theory to practice." *From Animals to Animats* 5 (1998): 515-524.

Nick Jakobi (1997) Evolutionary Robotics and the Radical Envelope-of-Noise Hypothesis *Adaptive Behavior 6: 325-368*

Sara Mitria, Dario Floreano and Laurent Keller (2009) The evolution of information suppression in communicating robots with conflicting interests. *PNAS* 106, 15786–15790

References

Koos, S. Mouret, J-B & Doncieux, S. (2013) The transferability approach: crossing the reality gap in evolutionary robotics. *IEEE Transactions on Evolutionary Computation*, 122-145

Karl Sims (July 1994). Evolving Virtual Creatures. *SIGGRAPH '94 Proceedings*: 15–22.

H. Lipson & J. Pollack (2000) Automatic design and manufacture of robotic lifeforms *Nature* 406, 974-978

Nick Cheney, Robert MacCurdy, Jeff Clune, and Hod Lipson. 2014. Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding. *SIGEVOlution* 7, 1 (August 2014), 11-23.

Hiller, J.; Lipson, H., "Automatic Design and Manufacture of Soft Robots," in *Robotics*, *IEEE Transactions on* , vol.28, no.2, pp.457-466, April 2012