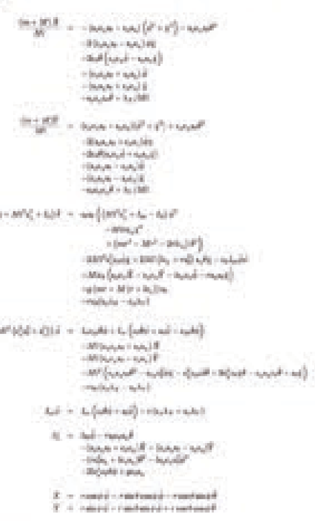
Evolved neurocontrol of a simulated unicycle

What's so hard about riding a unicycle?

Or, why is riding a bicycle easier? Because moving bicycles are self-stable: even without a rider, they'll stay upright. Unicycles are **inherently unstable**, which means the rider can never stop concentrating. There's more to it than this: in control engineering terms, a unicycle is **unstable**, **underactuated**, **highly nonlinear**, and **non-holonomic**.

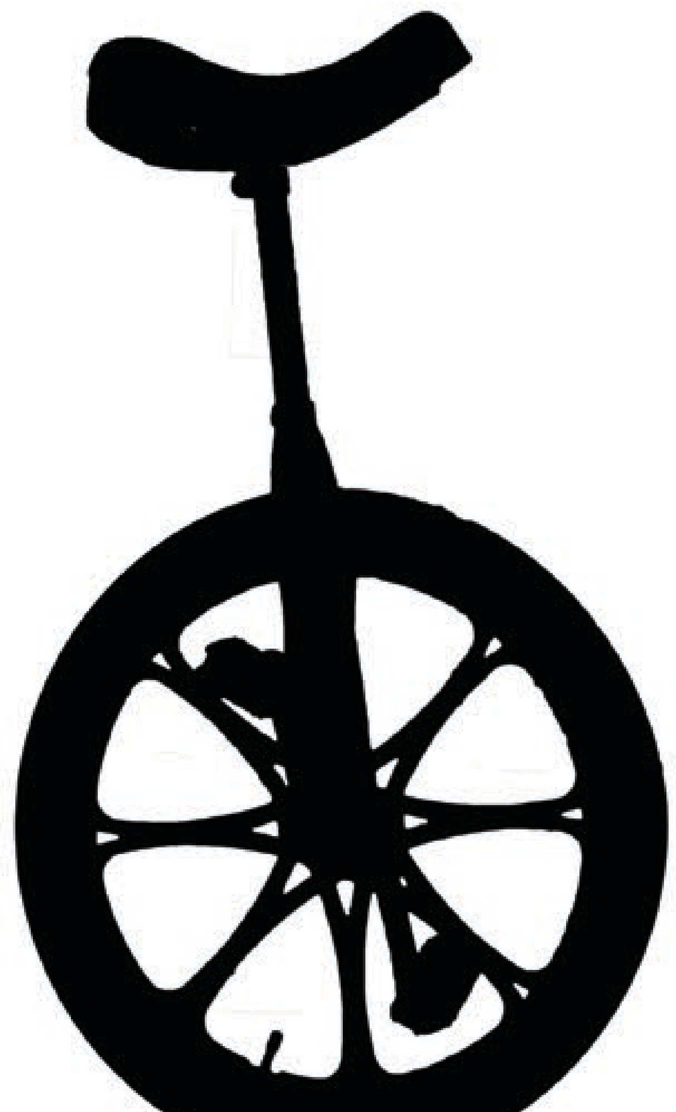


Equations of motion

These features, non-holonomy in particular, make derivation of equations of motion for a unicycle remarkably difficult. For a machine of such simplicity, an accurate physical model is surprisingly ugly.

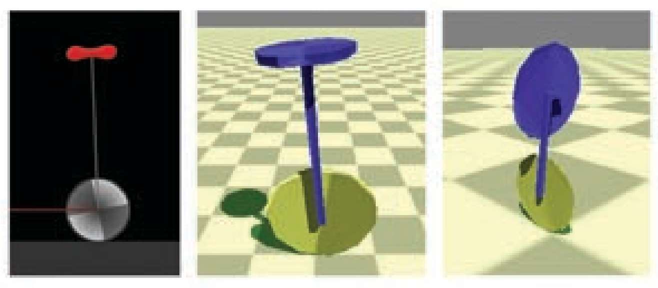
Conventional approaches

Traditional control systems require equations of motion, so previous work often relies on extreme linearisations of unicycle physics, or "piecewise" linearisation, in which the model is treated as behaving approximately linearly in a number of distinct configurations. The success of these approaches is limited, and it seems that no yaw-actuated unicycle robots have ever been successfully tested.



2D and 3D physical models

Both an analytic 2D ("Segway") simulation, and a 3D model, created using the Bullet constraint-solving physics engine, were used to test potential control approaches. The 2D case is a classic control problem, but sadly there is no useful way to modify its solutions for the 3D regime.



Friction modelling

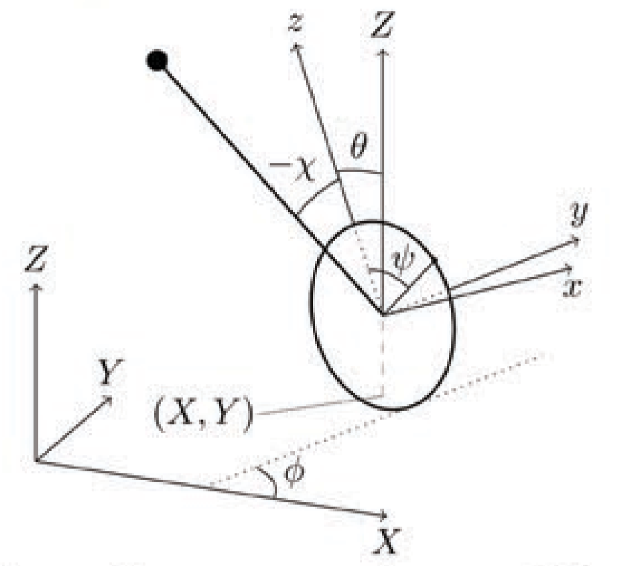
In modelling the unicycle, friction is not a second-order effect. Real unicyclists rely on the stick-slip interaction between the wheel and ground, using it to dump excess angular momentum. The numerical simulation includes a detailed and realistic friction model.

References

Kooijman et al., Science 332, 339–342 (2011).
Paris & Zhiang, Mathematical and Computer Modelling 36, 855-860 (2002).
Naveh et al., Dynamics and Control 9, 279–296 (1999).
Vos & von Flotow, Proceedings of the 29th IEEE Conference on Decision and Control vol. 1, 182–187 (1990).
Schoonwinkel, Ph.D. thesis, Stanford University, California (1988).

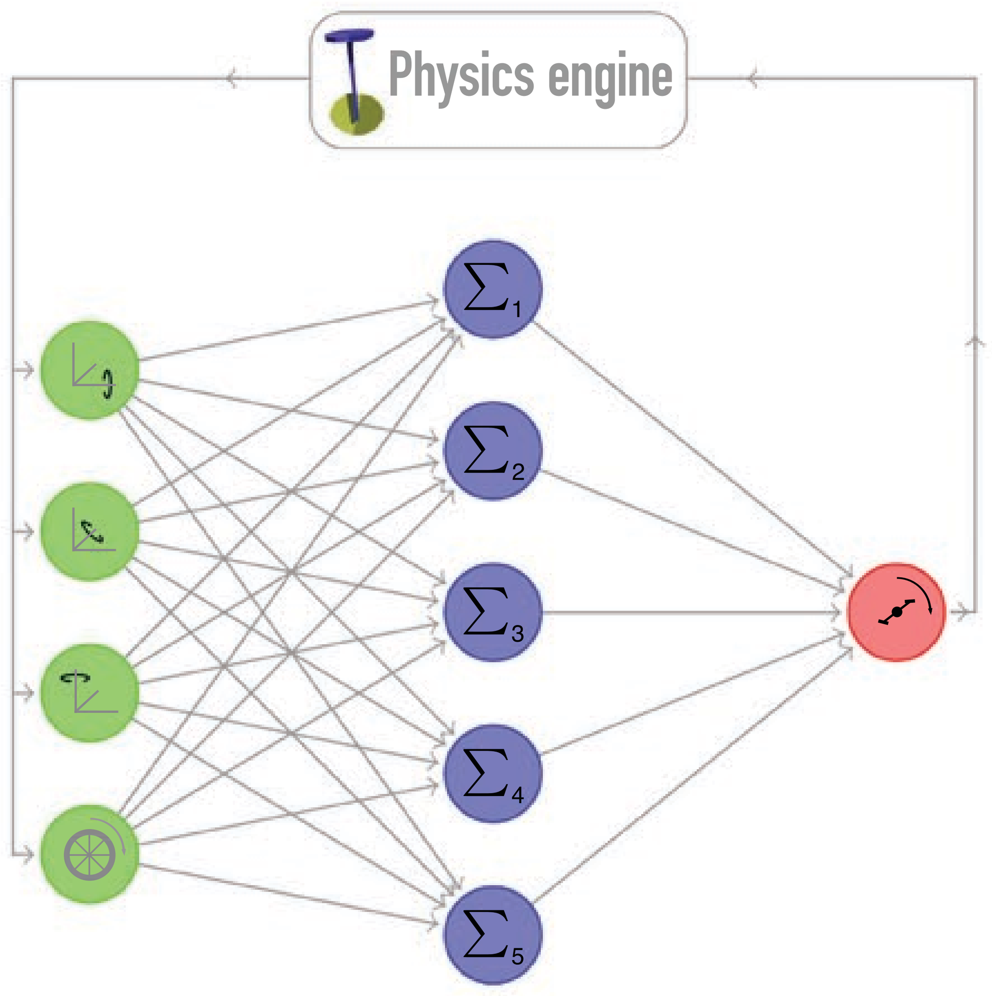
First principles approaches to control

If we can fully parameterise a system in terms of a set of state variables, then a control function is **a mapping between the state variables and the available control inputs**, such that over time the state variables approach some set of target values.



The coordinate systems and state variables of a simple unicycle model.

The mathematical machinery of control engineering exists to help find such a function. What can we do when this apparatus fails us?



The inner loop of our control system. An evolved neural network receives unicycle state variables as inputs, and generates a control response as its output. The weights of the network can encode a vast number of possible control functions.

Neural networks as universal function approximators

Even the simplest feed-forward **neural networks can serve as universal function approximators**. Our control approach exploits this property, using a neural network to map unicycle variables of state to the available actuators: the legs and torso of our simulated rider.

Genetic algorithms

Conventional approaches to training neural networks aren't particularly useful in control applications. How can we know if the decision to kick down on the pedals 35.6 seconds into the simulation was responsible for the eventual failure of the unicycle at 38.2 seconds? This is the kind of information we would need to use backpropagation techniques.

Instead, we use a genetic algorithm to **evolve a population of neural networks**, rewarding those that keep the unicycle upright for longest.

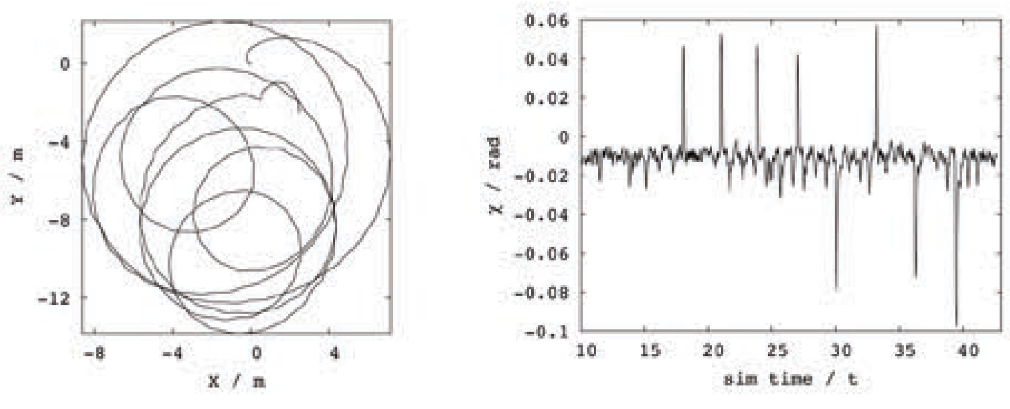
References

Cybenko, Mathematics of Control, Signals, and Systems 2, 303–314 (1989).
Montana & Davis, Proc. of the International Joint Conf. on Artificial Intelligence, 762–767 (1989).
Reil & Massey, Theory in Biosciences 120, 327-339 (2001).

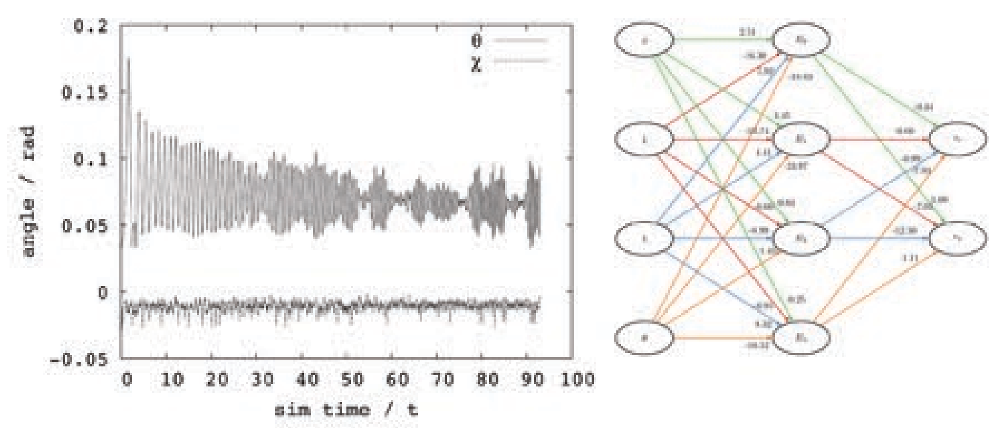
Results

Effective unicycle controllers are found using modest computation on commodity hardware. Evolved controllers employ a range of tactics, some reminiscent of human unicyclists, and seem to be notably more successful than any described in the literature.

It is particularly impressive that **the GA can evolve controllers that are robust in the face of noisy input and actuation**, suggesting that this method (or a close relative) has good chances in the real world!



The ground trajectory of a simulation run with an evolved controller (*left*). Response to manually applied pitch perturbations during a simulation run (*right*).



Pitch and roll angles during a simulation (*left*). Neural network and evolved connection weights for this simulation (*right*).

Limitations and challenges

While this small study has been far more successful than its author expected, the results are so far from simulation only. Transfer to a real-world system is not trivial.

The neural networks described here are "overevolved", in the sense that they work for only one precisely-specified unicycle: a heavier rider or longer crank-arm would render them unstable. **Normalisation (and non-dimensionalisation) of the control inputs could allow the construction of a general controller for unicycles.**

Natural computation

The techniques described here were heavily inspired by work by Torsten Reil and collaborators, who evolved a controller for a simulated bipedal walker. The elegance of their approach, **which co-opts some of nature's learning methods to solve a conventionally insoluble problem**, cannot be denied.

There remains an open problem: whether these techniques, "unprovable" from the perspective of traditional control engineers, are truly effective, or whether they retain hidden pathologies that render them unusable in mission-critical applications such as automotive and aerospace autopilots.

I have no solution to this problem, but one thing does seem clear: as control systems get larger and more complex, **any conventional proved controller is only as reliable as its prover is mathematically competent**. It may be that evolved systems can be "proved" numerically to tolerances acceptable for use in public systems.

My strong instinct is that natural, or bio-mimetic, computational systems will have an increasingly important role to play in many disciplines, control engineering included, where the full weight of mathematical formalism runs aground in the face of increasingly complex problems.

Nick Stenning
nick@whiteink.com

Special thanks to Dr Chris Lester of the High Energy Physics group at the Cavendish Laboratory, University of Cambridge, for sage advice and encouragement.