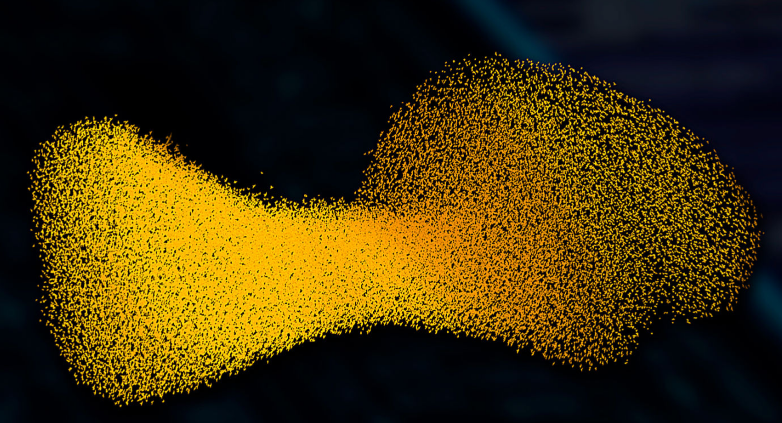
HOW TO PROGRAM A SWARM OF ROBOTS

A robot swarm is a self-organising multi-robot system. Each robot follows a set of pre-optimised simple rules based on its own state and the state of its neighbours.

The system is characterised by swarm intelligence: an emergent form of intelligence that exists in the patterns of interactions between the robots of the swarm, rather than in the brain of any individual robot. In this way, a robot swarm is much like a swarm of ants or bees.

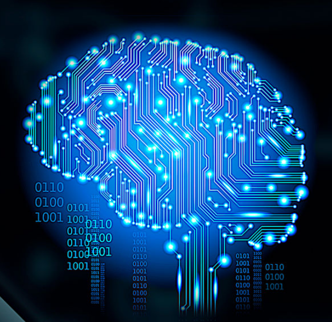
The desirable properties of a robot swarm are the potential for high robustness, flexibility, redundancy and scalability. A swarm can carry out distributed tasks that would be impossible for a single robot. A swarm can also parallelise a task that would take a single robot much longer to complete.

In the future, swarms of robots will be all around us, performing a large share of humanity's labour. Within our lifetimes we can expect to see applications realised in environmental monitoring, crop pollination and pest control, and also *in vivo* applications in medicine.



>>>>> The robot brain >>>>>

The robot "brain" executes a linear sequence of computer instructions. These instructions can be grouped into logical units—building blocks—each of which performs a particular useful function. The function can be something tangible, such as "move forwards", or something more abstract, such as a mathematical formula.



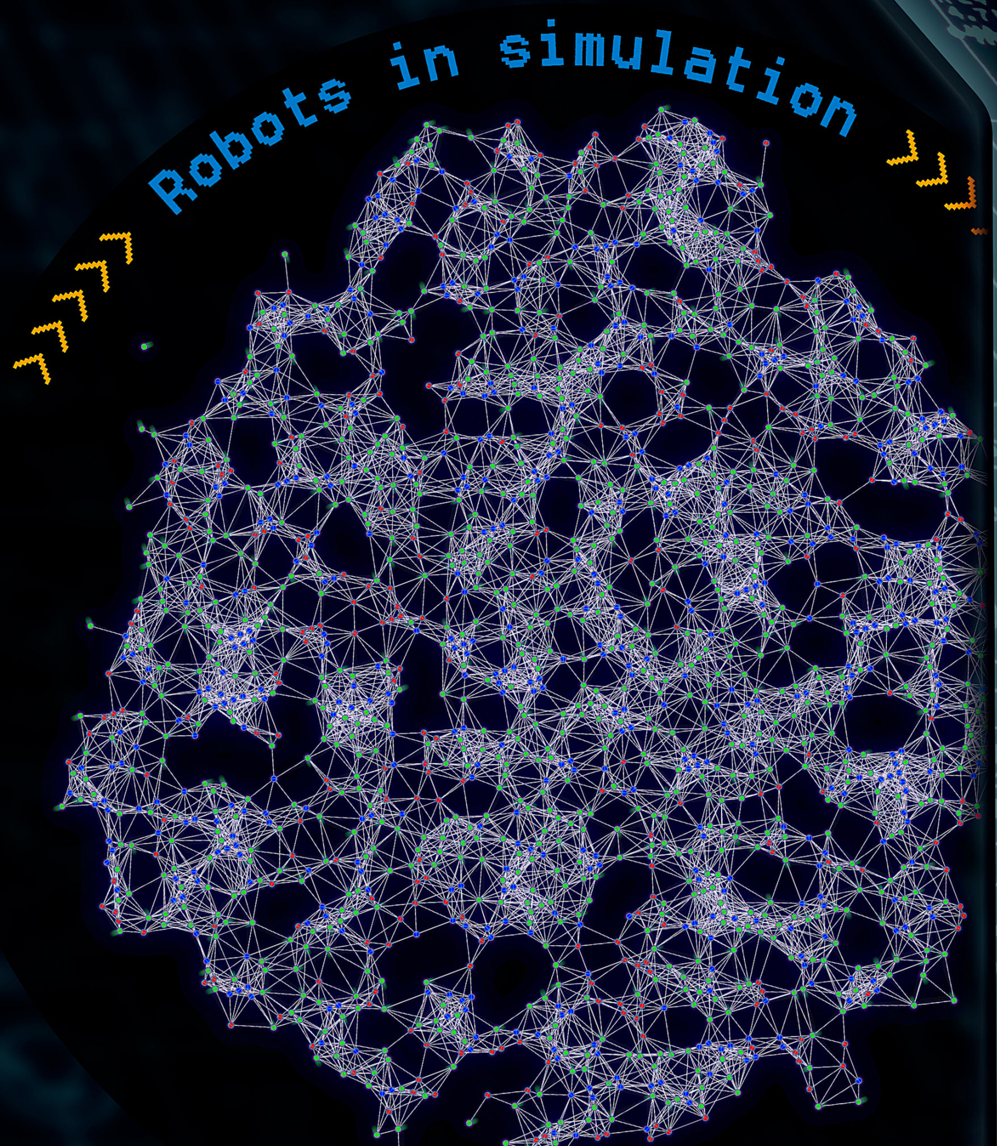
Building blocks can be strung together in series to create a complete swarm algorithm. This is analogous to stringing together a series of words to create a complete sentence.

Computer simulation allows us to assess the fitness of a swarm algorithm.

With modern computers it is possible to perform physically realistic simulations in real time or better, which makes possible rapid prototyping of a swarm algorithm.

The accuracy of the simulation is validated using real robots. For this project we have a test bed of 50 *Kilobot* robots.

By running many randomised variations of a swarm algorithm through the simulation software, a set of training data is built up.



The image shows 2,000 virtual robots running a classic swarm algorithm in which the goal is that robots move around each other while maintaining a specific number of neighbours.

>>>>> Programming the swarm >>>>>

Just like sentences, not all swarm algorithms "make sense". That is, some algorithms result in better swarm behaviour than others. And, as a meaningful sentence must obey the rules of grammar, so we propose that swarm algorithms obey their own form of "robot grammar".

The conceptually difficult problem of discovering a good swarm algorithm is thus reduced to an optimisation problem: find the set of grammar rules that best fit the training data.

These grammar rules can be derived by developing software to automatically scan the training data for patterns. For example, which building block sequences are correlated with fitness? Which building blocks can we group together under the same category, and in what contexts? And so on.

Once we have derived an estimate of the robot grammar, we can apply this grammar to generate new swarm algorithms. These new algorithms are then appended to the training data, and the process is iterated.

Over time we expect this process to converge to close to optimal grammar rules, and to generate swarm algorithms of high mean fitness.

UNIVERSITY OF LIVERPOOL

School of Engineering

Supervisors: Dr. Paolo Paoletti
Dr. Michael Jump

Andrew Alvin
A.R.Alvin@liv.ac.uk