Exploiting Loose Horizontal Coupling in Evolutionary Swarm Robotics

Jennifer Owen[1], Susan Stepney[1], Jonathan Timmis[1,2], and Alan Winfield[3]

[1] Department of Computer Science, University of York, York, UK
{jowen,susan,jtimmis}@cs.york.ac.uk
[2] Department of Electronics, University of York, York, UK
[3] Faculty of Environment and Technology, U.W.E., Bristol, UK
Alan.Winfield@uwe.ac.uk

Abstract. We describe a theory from Herbert Simon that links the structure of complex systems to increased speed of evolution, and argue the position that this theory can be beneficial to evolutionary swarm robotic research. We propose a way of applying this theory to evolutionary swarm robotic systems by manually designing the robot to robot communication mechanisms and keeping these constant, whilst evolving the rest of the robots' behaviours. This allows for robots to evolve independently of each other without breaking any inter-dependencies that may exist between robots in the swarm. Finally we address potential criticisms of our suggested approach, and outline a course of future research in this area in order to verify our proposal.

1 Introduction

Here we propose a means of speeding up the evolution of swarm robotic systems by considering the role of communication within complex systems.

The main obstacle to swarm robotics research is known as "the design problem". This is the question of which behaviours we should engineer at the robot level to produce a collective emergent behaviour at the swarm level. One proposed solution [1] is to *evolve* the robot's mapping between its sensor inputs and its motor and actuator outputs. With this approach the designer does not have to worry about what particular behaviours or rulesets should be incorporated into the robot; these things are automatically created during artificial evolution. By extending the evolution of a robot's controller to a whole swarm of evolving robots, we get the field of Evolutionary Swarm Robotics (ESR).

One of the main drawbacks of ESR is that measuring the fitness of a particular swarm phenotype is very slow [2]. Measuring this fitness requires that the swarm be run for long enough to build up a clear picture of how well it is functioning in the world. A standard evolutionary algorithm uses tens or hundreds of candidate solutions (here, swarm phenotypes) in the population at each generation, and the algorithm is run for hundreds or thousands of generations [3]; consequently many thousands of fitness function evaluations are made. If each takes several minutes to evaluate, then a whole evolutionary algorithm will take hours to run, which is likely to be longer than the battery life of the robots. Hence, the speed of the fitness function evaluation and of swarm robot evolution in general is a major hurdle for ESR.

2 Complex Systems and Evolution

Simon [4] argues that complexity in systems will lead to faster evolution than in systems without complexity. He is referring to evolution in the Darwinian sense of the gradual change of a species' genome over multiple generations. However, his argument is worded in general terms and can be applied to the formation of non-biological complex systems. He uses an example of atoms forming molecules, which are then used to form amino-acids, and then proteins. The gradual development of structures with low complexity (atoms) to higher complexity (proteins) is also be referred to as "evolution" by Simon. Consequently Simon's argument can apply to both biological and non-biological systems.

2.1 Complex Systems as Hierarchies

Simon [4, 5] observes that complex systems have "hierarchic structure". He defines a hierarchic structure as: *"A system that is composed of interrelated subsystems, each of the latter being, in turn, hierarchic in structure until we reach some lowest level of elementary subsystem."*

Take as an example the human body: at the highest level is the body itself; within the body are organs that interact to keep the body alive; each organ is made of interacting cells; cells are made of interacting molecules. In a complex hierarchic structure, at each level the interacting components are subsystems that are complex hierarchies too. Therefore each hierarchic complex system is a system of systems.

Simon [5] further observes that a complex system not only has a hierarchic structure, but also its functionality cannot be understood or recreated from examining its individual subsystems. On each hierarchic complexity level there are interactions between the subsystems and it is from the subsystems and their interactions that we get the higher level complex system. The subsystems themselves are the result of interactions between their own subsystems. From the bottom up, at each successive complexity level, we gain some knowledge of behaviour that was not apparent from observing individuals at the level below. Simon [4, 5] calls this "near-decomposability"; we now call it *emergence.*

2.2 Linking Complexity and Evolution

It has been argued that if a system is complex, it will evolve faster than if it is not [4–6]. It is certainly true that complex systems are everywhere in nature—take any cellular organism as an example—so there must be some reason that these organisms have prospered whilst non-complex biological systems have not. Simon [4] examines this relationship between complexity and evolution in depth, and suggests two major reasons why complexity speeds up evolution: stable intermediate subsystems, and loose horizontal coupling.

Stable Intermediate Subsystems. The first observation made in [4], which is also noted in [5,6], is that *"complex systems will evolve from simple systems much more rapidly if there are stable intermediate forms than if there are not"* [5]. Stability in this context means that, despite external perturbation, a system is independently able to maintain some internal state that allows it to continue functioning within its natural environment [7]. In a complex system then, if its subsystems have stability, it means they are able to repair and maintain themselves, and are not as likely to degrade over time. Consequently, stable subsystems are likely to be more prevalent, more available to come together to form something new. Perhaps it is the case that evolution, by developing stable building blocks, speeds the formation of complexity hierarchies, and not the other way around.

Loose Horizontal Coupling. Simon's second observation [4] on how complexity speeds up evolution is to do with functional equivalence and "Loose Horizontal Coupling".

Within a complex system there is "vertical coupling" between levels, in that higher levels are composed of the lower levels. There is also "horizontal coupling": communication and interactions between subsystems on the *same* hierarchic level. If two subsystems on the same level interact with each other in a *fixed* manner, then each is able to evolve independently of the other. These changes may affect the higher level system, and this would direct the evolution. For example, in the human body there is a digestive system and a circulatory system. The digestive system breaks down food and puts it somewhere where it can be absorbed into the bloodstream. If the circulatory system were to route blood more efficiently, then, as long as it still absorbs food from the digestive system, these changes would not affect the functionality of the digestive system. Consequently the higher level system, the body, would be improved.

This relative independence of subsystems due to the fixed nature of their interactions is what Simon calls "Loose Horizontal Coupling" (LHC).

Alphabets. Key to achieving LHC is fixing interaction via a limited "alphabet" of components: *"the flexibility of coupling among subsystems can be further enhanced by limiting the variety of different kinds of components that are incorporated into the larger system"* [4]. Simon uses the example of amino acids. There are 20 types of amino acid, giving an alphabet of size 20 on this complexity level. By repeating and combining parts of this alphabet we can create *"innumerable protein molecules"* [4]. Amino acids are fixed; it is the proteins constructed from them that evolve. An alphabet should be varied enough to be capable of expressing anything, but flexible enough that meaning can be gained from a composite structure of alphabet elements [4]. An alphabet can form the fixed unit of "currency", or information, exchanged between subsystems.

Alphabets are key to LHC because they make it easier for systems to communicate with each other. With an alphabet to dictate what can be communicated, there are fewer types of component to generate in order to pass information between subsystems. There are fewer data types, but the order and structure of the data is what conveys information. In our example of a digestive system with a circulatory system, food is broken down into the message given to the circulatory system. A low complexity level alphabet of amino acids can carry out this communication. A whole range of amino acids are presented by the digestive system; a particular type of protein however, is much more specific and our food is less likely to contain it, so it is less useful for general communication.

In summary, alphabets can regulate the communication between subsystems. They are well suited to this role because alphabet components are more prolific and less specific, compared to things that are composed of the components.

3 Swarm Robots and Speedier Evolution

We have outlined Simon's argument about how complexity in a system can increase the speed of its evolution. A swarm robotic system is complex and dynamic, incorporating feedback and interaction on many hierarchic levels. We conjecture that **the principles of stable hierarchic subsystems with loose horizontal coupling can be applied to evolutionary swarm robotics, resulting in faster evolution of the robot swarm**. The question then arises of how we apply these ideas to swarm robotics.

Addressing the idea of stable subsystems first, the robots in a swarm are themselves stable. Viewing a robot as just a hardware platform for running its controller, the robot will maintain its internal state of being a hardware platform and will continue to do so until either hardware or the battery wears out, or the robot is subjected to destructive external perturbations. The controller within the robot may not be stable, and this could cause the robot to behave erratically. Research has been done into evolutionary algorithms that modularise parts of the genome for reuse, for example [8,9]. Depending on the ability of the chosen evolutionary algorithm to select stable and useful parts of the genome for modularisation and proliferation, stable subsystems within the robot controller could be generated.

Applying loose horizontal coupling (LHC) to robot swarms is more of a challenge because there is so little previous research in this field, particular in the context of evolutionary algorithms and swarm robotics.

Trianni [10] identifies three ways in which swarm robots communicate with each other [10]: *indirect communication* or stigmergy, *direct interaction* where robots physcially interact to communicate and *direct communication* where messages are passed between robots without them needing to physically interact.

We propose that LHC can be implemented by using a fixed means of communication to dictate how the direct communication and interaction between robots should take place. A small fixed alphabet should be used to compose the messages that are conveyed between robots at the relevant subsystem level. This

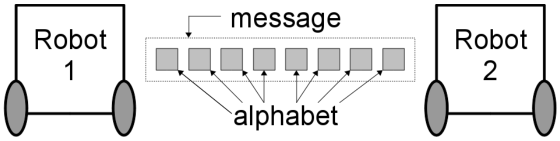

Fig. 1. LHC between two robots. The alphabet and the medium for communication are both designed, and are fixed throughout the evolution of the swarm. Each robot's controller is evolved to map between the robot's sensor inputs (including any received messages) and its outputs. This gives us the robot's behaviour and its interpretation of the message. The message itself emerges from the structure of the combination of several alphabet components and the interpretation that a robot gives to a message.

limits the freedom of the communication, but gives the messages enough flexibility to express meaning in the way the alphabet has been combined. In this manner the functional equivalence of each robot is maintained for as long as that robot can generate and communicate the desired information. If this process is unaffected by the swarm's evolution then LHC will exist between robots, and they can evolve independently of each other without causing other dependant robots to break down. Figure 1 illustrates what we are proposing.

This approach places some restrictions on the freedom of the swarm evolution. LHC is best used when the swarm is evolving in a decentralised way, which is to say, each robot evolves independently with no global time step dictating when to update their genome, using only local information to measure its fitness. LHC in this situation ensures that the robots can evolve independently and still understand each other. If the swarm is evolved centrally, using a global controller to decide robot genomes or fitnesses, (as in [10, 11]) the principles can still be applied but may be of less benefit.

When using LHC, the medium of communication and what alphabet to communicate must be decided *a priori*, so some manual design of the controller is required to support these decisions. This potentially reduces the benefits of evolving the controller as the programmer is still required to design some parts of it. Despite this, it may still be simpler to have to design only the direct communication and interactions, compared to designing the entire controller. Care must therefore be taken when designing the LHC between robots, as the decisions may end up locking the swarm into a behaviour that is less than optimal, and it might never reach maximum fitness. However, over the time frame of a swarm robot experiment we may be able to evolve only over a limited number of generations, so by speeding up the evolution we will hopefully allow the swarm to reach a higher fitness than would have been achievable without LHC. Whilst this may potentially lock us into a lower overall fitness, the benefits should outweigh the costs and we would at least end up with a behaviour that is "good enough".

4 Potential Criticisms

There are some criticisms that might be made when considering our proposal. We address each of them in turn.

The paper from which we are drawing our ideas [4] was written nearly 40 years ago, and so ideas and definitions may be out of date. Simon's hypothesis about how complexity and evolution are linked is based on his understanding of a complex system, described in section 2.1. We have shown that the relevant parts of this definition can be applied to swarm robotics in section 2.2. It therefore follows that Simon's hypothesis can be applied to swarm robotics. Whether his view of complexity is right or not is unimportant. It is the structure and interactions of the complex system that cause the rapid evolution, and we have shown that these are present in swarm robotic systems.

LHC is already implicitly used in evolutionary swarm robotics. Some researchers implicitly use LHC in their experiment. For example, Trianni *et al.* [11] evolve a controller that performs coordinated movement in a swarm of four robots. Before the experiment begins the robots are connected together by the experimenters and they are able to communicate with each other only by exerting a pull on the directly connected robot. The robots are free to evolve the interpretation of this pull, but LHC is implicit in the experimental setup because their means of communication with each other, and the alphabet used to encode the information conveyed, remains fixed throughout the experiment. The alphabet, in this case, consists of pulling forces exerted on the robot in different directions and with different amounts of force.

Similarly, Trianni *et al.* [12] implicitly employ LHC. Each robot in the swarm emits a continuous tone, and the group must evolve their controller to perform swarm aggregation. The continuous tone is used as an "I am here" message communicated between robots, which can be located by other robots using four inbuilt microphones. The robots in this experiment must weight a neural network connecting the microphones and proximity sensors to the motor outputs. In doing so they interpret the signals they receive in order to aggregate together. The direct interaction and direct communication has been explicitly pre-specified by the experimenter.

In experiments where a solitary robot must learn to adapt to a static environment, for example evolving obstacle avoidance behaviour [13], the interaction between the sensed object and the data returned by the robot's proximity sensors is fairly consistent. Hence there is LHC between the robot and its environment because the communication between the two is fixed and does not change over the course of the experiment. In this case the LHC is implicit in the experiment since the experimenter has not specified that the robot's interactions are fixed. LHC is instead just an artefact of how the robot observes its world.

In these examples the LHC is either implicit or unintentional. Although we have not given a very thorough analysis of the field of evolutionary swarm robotics, these examples are sufficient to show that LHC is already implicitly used in some current evolutionary robotics research. We suggest that the implementation of LHC should be *explicitly* considered when conducting future experiments. This is because it affects how the robots evolve, and helps us to understand from what starting point we are evolving the swarm. We can also assess how easy are we making things for the evolutionary algorithm, so that its effectiveness can be assessed and compared to those of other experiments.

Is lack of LHC even a sensible alternative? Are there any circumstances where not using LHC, even implicitly, makes sense?

In the example of [13], where obstacle avoidance behaviour is evolved, the implicit LHC is due to the static environment causing consistent proximity data. If the world were dynamic, the interaction between proximity sensors and objects in the world would not be so consistent, because objects could be moving towards or away from the robot. Consequently, for there to be no implicit LHC present, the environment must be dynamic. But this is the case with swarm robots, because having multiple agents in the environment causes motion and changes to occur in the environment.

In the case of [12] LHC could be removed by stopping the continuous tone, and leaving the robots to evolve the ability to know when to turn them on or off. In [11] the LHC could be removed by separating the robots and leaving them to connect together themselves, although this would completely change the nature of the experiment, which was to measure whether the robots can learn coordinated movement. Essentially in both cases the LHC is removed by leaving the robot-to-robot communications to be completely evolved, and care must be taken to make sure there is no implicit LHC in the experiment.

Not using LHC in the experiment is sensible if the aim is to evolve the robot swarm from nothing, with no pre-established direct interaction and communication between robots. If evolving all this is not practical, or would fundamentally change the nature of the experiment, then removing LHC completely is not a sensible option. It depends on what goals you want to achieve.

5 Conclusion and Future Work

We have presented Simon's idea of how stable subsystems with loose horizontal coupling increase the speed of evolution [4]. The point we make in this paper is that the presence of LHC and stable subsystems may be used speed up evolution in swarm robotic systems by fixing the swarm's methods of robot interaction and communication. Not to use LHC might be desirable in some scenarios, but if the ideas presented in this paper prove correct and LHC is used, whether implicitly or explicitly, we should be aware of the fact and hence maximise its effectiveness.

Our next stage is to investigate Simon's hypothesis using an evolutionary robotic swarm. We will measure the rate of evolution in the case of a swarm using LHC and compare it to the rate of evolution if the swarm were to not use LHC. If we can show that Simon's hypothesis is effective in speeding up the evolution process in robotic systems then our work will help to present evolution as a more viable solution to the "design problem" of swarm robotics. It would also help verify that Simon's hypothesis is correct. This would have consequences in the application of evolutionary algorithms more generally, because the same principles could be applied to the evolution of other classes of solutions.

In this paper we have focused on the robot level of the swarm, and how we could implement LHC. We have not yet considered the possibility of using stable subsystems with LHC *within* the robot controller. If we could develop stable modules within the controller and have these maintain their inputs and outputs as part of their stability, then there is greater potential for speeding up swarm evolution. This is, however, a far more complicated task and further research is required in this area.

Acknowledgments. This work is part of the CoSMoS project, funded by EPSRC grants EP/E053505/1 and EP/E049419/1.

References

1. Trianni, V.: Evolutionary Swarm Robotics: Evolving Self-Organising Behaviours in Groups of Autonomous Robots. Springer (2008)
2. Lund, H.H., Hallam, J.: Evolving sufficient robot controllers. In: 4th IEEE Int. Conf. on Evolutionary Computation, IEEE Press (1997) 495–499
3. Mitchell, M.: An Introduction to Genetic Algorithms. MIT Press (1996)
4. Simon, H.A.: The organization of complex systems. In Pattee, H.H., ed.: Hierarchy Theory. George Braziller (1973) 1–27
5. Simon, H.A.: The architecture of complexity. Proceedings of the American Philosophical Society **106** (1962) 467–482
6. Maturana, H.R., Varela, F.J.: Autopoiesis and Cognition: the Realization of the Living. D. Reidel (1980)
7. Pimm, S.L.: The complexity and stability of ecosystems. Nature **307** (1984) 321–326
8. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison Wesley (1989)
9. Forrest, S., Mitchell, M.: Relative building-block fitness and the building block hypothesis. In: Proc. 2nd Workshop on Foundations of Genetic Algorithms, Morgan Kaufmann (1993) 109–126
10. Trianni, V., Labella, T.H., Dorigo, M.: Evolution of direct communication for a swarm-bot performing hole avoidance. In: ANTS 2004. Volume 3172 of LNCS., Springer (2004) 131–132
11. Trianni, V., Nolfi, S., Dorigo, M.: Hole avoidance: Experiments in coordinated motion on rough terrain. In: Proc. 8th Conference on Intelligent Autonomous Systems (IAS8), IOS Press (2004) 29–36
12. Trianni, V., Groß, R., Labella, T.H., Şahin, E., Dorigo, M.: Evolving aggregation behaviors in a swarm of robots. In: ECAL 2003. Volume 2801 of LNAI., Springer (2003) 865–874
13. Floreano, D., Mondada, F.: Automatic creation of an autonomous agent: genetic evolution of a neural-network driven robot. In: From animals to animats 3, Brighton, UK, MIT Press (1994) 421–430