From energy to information and back

Russ Abbott

Question. If complex systems are far from equilibrium, what equilibrium are they far from?

Answer. They are far from energy equilibrium. But being far from an energy equilibrium doesn't necessarily mean that a system is gaining or losing energy. Most complex systems have reasonably stable energy levels.

Question. What then does it mean for a stable system to be far from an energy equilibrium?

Answer. It means that energy flows through it. But most complex systems are not just obstacles in the path of an energy flow for which they must find dissipative mechanisms. Most complex systems maintain their structure by harnessing externally supplied energy and channeling it into well organized internal energy flows. In other words, to understand a complex system requires understanding its internal energy flows.

Complex systems can generally be understood as dynamic systems—systems that transition from state to state. Among other things, a state typically represents a configuration of internally stored energy potentials. In systems that can store energy, Morowitz showed (long ago) that cycles will occur. These cycles provide the power for everything else that happens in these systems. They allow new activities to be built on top of them. The picture I will paint is one in which complex systems harness externally supplied energy and channel it to form reliable power cycles that can then be used to build increasingly complex activities.

This perspective is so obvious that we often tend not even to think about it. Plants use bees for pollination by taking advantage of bee activity. But bee activity is at its heart a complex energy flow built on top of the more primitive biochemical energy flows that power all biological organisms. Our ecological system is organized in terms of such inter-related energy flows. Margulis and Sagan put this more elegantly, "Life did not take over the globe by combat, but by networking." What was networked were energy flows.

We do the same thing when we build infrastructure systems. Consider the physical mail system. When one puts a letter into an envelope, addresses and stamps the envelope, and then puts the envelope where it will be picked up by the mail system, one is making use of an ongoing process. It's arguably more sophisticated than bee flower-hopping, but underneath both are examples of independently powered ongoing processes that can be exploited for parcel delivery. In neither case is the underlying process driven by the use being made of it. Neither the envelope nor the pollen provides the power that moves them.

This way of thinking—producing a result by harnessing and channeling an existing process—may seem unfamiliar. But it's the daily occupation for software developers. Software is symbolic structures that harness and channel the ongoing operation of an underlying computational engine. But just as we often ignore the energy needed to power bees and the mail system we also ignore the energy needed to drive computers. The reason is that the amount of energy required is typically very small compared to the effect produced. The leverage occurs because computers operate in the information realm rather than the physical realm. This is similar to turning on a light. It is not the flipping of the switch that causes the light to go on. Flipping the switch enables energy to flow through the light bulb, which causes it to produce light. The state of the switch is in the information realm. But minimal as the energy required to change its state—especially compared to the effect produced—one still needs energy to flip a switch.

A useful measure of system sophistication is the extent to which its control resides in the information realm (including DNA and neural processes) rather than the physical realm. The more a system's control occurs in the information realm the more leverage that system tends to have over the energy flows it encounters. But it's important to remember that operations within the information realm are still physical—i.e., they require energy—and may therefore have vulnerabilities of their own.

Doursat, R. (2010a) Embryomorphic Engineering: From biological development to self-organized computational architectures. *4th EmergeNET Meeting: Engineering Emergence (EmergeNET4)*, April 19-20, 2010, St William's College, York, UK. ***Keynote address.***

Embryomorphic Engineering: From biological development to self-organized computational architectures

René Doursat

Complex Systems Institute, Paris Ile-de-France
CREA, Ecole Polytechnique & CNRS

Traditional engineered ICT systems are qualitatively different from natural complex systems (CS). The former are made of unique, heterogeneous components assembled in complicated but precise ways, whereas the latter mostly rely on the repetition of agents following identical rules under stochastic dynamics. Thus, while natural CS often generate random patterns (spots, stripes, waves, trails, clusters, hubs, etc.), these patterns generally do not exhibit a true *architecture* like human-made ICT systems possess. Major exceptions, however, blur this dichotomy. (a) ICT-like CS: On the one hand, biology strikingly demonstrates the possibility of combining pure self-organization and elaborate architectures, such as the self-assembly of myriads of cells into the body plans of organisms, the synchronization of neural signals into mental states, or the stigmergic collaboration of swarms of social insects toward giant constructions. (b) CS-like ICT: Conversely, large-scale distributed ICT systems made of a multitude of components at all scales (integrated parts, software agents, network hosts) already exhibit complex emergent effects, albeit still mostly uncontrolled and unwanted. Thus, while some natural CS seemingly exhibit all the attributes of ICT systems, ICT systems are becoming natural objects of study for CS science. Such cross-boundary cases are examples of *self-organized architectures*—i.e., how spontaneous systems need not always be random and engineered systems need not always be directly designed—a hybrid concept insufficiently explored so far. I illustrate this goal with a multi-agent model of *programmable* and *reproducible* morphogenesis in a complex system. This model combines self-assembly (SA) and pattern formation (PF) under the control of a nonrandom gene regulatory network (GRN) stored inside each agent of the system. The differential properties of the agents (division, adhesion, migration) are determined by the regions of gene expression to which they belong, while at the same time these regions further expand and segment into subregions due to the self-assembly of differentiating agents. This model offers a new abstract framework, which I call *Embryomorphic Engineering* (coined after Neuromorphic Engineering), to explore the developmental link from genotype to phenotype that is needed in many emerging computational disciplines, such as 2-D/3-D collective robotics or n-D autonomic network topologies.

- Doursat, R. (2009g) Facilitating evolutionary innovation by developmental modularity and variability. *Generative & Developmental Systems Workshop (GDS 2009)*, at *18th Genetic and Evolutionary Computation Conference (GECCO 2009)*, July 8-12, 2009, Montreal, Canada. http://doursat.free.fr/docs/Doursat_2009_devo_GECCO.pdf

- Doursat, R. & Ulieru, M. (2008b) Emergent engineering for the management of complex situations. *2nd International Conference on Autonomic Computing and Communication Systems (Autonomics 2008)*, September 23-25, 2008, Telecom Italia Labs, Turin, Italy. http://doursat.free.fr/docs/Doursat_Ulieru_2008_nets_Autonomics.pdf

To Boldly Go: an occam-π Adventure

Peter H. WELCH [a], Kurt WALLNAU [b] and Mark KLEIN [b]

[a] *School of Computing, University of Kent, Canterbury, UK*
[b] *Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA., USA*
`p.h.welch@kent.ac.uk`, `kcw@sei.cmu.edu`, `mk@sei.cmu.edu`

Abstract. Future systems will be too complex to design and implement explicitly. Instead, we will have to learn to engineer complex behaviours indirectly: through the discovery and application of local rules of behaviour, applied to simple process components, from which desired behaviours predictably emerge through dynamic interactions between massive numbers of instances.

This talk considers such indirect engineering of emergence using a process-oriented architecture. Different varieties of behaviour may emerge within a single application, with interactions between them provoking ever-richer patterns – almost social systems. We will illustrate with a study based on Reynolds' *boids*: emergent behaviours include *flocking* (of course), directional *migration* (with waves), *fear* and *panic* (of hawks), *orbiting* (points of interest), *feeding frenzy* (when in a large enough flock), *turbulent flow* and *maze solving*.

With this kind of engineering, a new problem shows up: the suppression of the emergence of undesired behaviours. The panic reaction within a flock to the sudden appearance of a hawk is a case in point. With our present rules, the flock loses cohesion and scatters too quickly, making individuals more vulnerable. What are the rules that will make the flock turn almost-as-one and maintain most of its cohesion? There are only the *boids* to which these rules may apply (there being, of course, no design or programming entity corresponding to a flock). More importantly, how do we set about finding such rules in the first place?

Something else that can emerge from emergent engineering are unexpected relationships between quite different domains of study (e.g. bird migration, gas dynamics and maze solving). These relationships may not have been noticed before, as the low-level agents generating the emergent behaviours appear quite unrelated. There is also the potential to discover completely new domains of interest from such studies.

Our architecture and models are written in occam-π, whose processes are sufficiently lightweight to enable a sufficiently large mass to run and be interacted with for real-time experiments on emergent behaviour.

This work is in collaboration with the Software Engineering Institute (at CMU) and is part of the CoSMoS project (at the Universities of Kent and York in the UK).

Keywords. complex systems, emergent behaviour, process orientation, mobile processes, occam-π, boids, flocking, migration, maze solving

Simulation as an experimental design process for emergent systems

Paul Andrews[1], Susan Stepney[1], Alan Winfield[2]

[1]Department of Computer Science, University of York
[2]Bristol Robotics Laboratory

We often wish to engineer electronic systems with emergent properties; swarm robotics being a prime example. Emergent properties, however, are not obvious properties of the underlying system; some would define them as "surprising"[1]. Even if we eschew that definition [2], these properties are not readily deducible: we may know which emergent property is desired for the engineered system, but it is unclear how to engineer this into the system using conventional engineering design processes. So how can we *engineer* emergent systems?

In the CoSMoS project, we have been developing a framework and process [3] for using simulation as a principled scientific instrument, to model, animate, and analyse emergent properties of pre-existing biological and sociological systems. Such simulations incorporate a model of the world being simulated [4]; in this case a model of the real world domain of interest. The simulation can be calibrated and validated against the real world, in order to demonstrate that it is fit for purpose. Simulations always differ from reality: the important point is to demonstrate that they do not differ in ways that invalidate their scientific value.

Inspired by the CoSMoS approach we advocate a process of *co-development* of an emergent engineered system with a simulation of that system. Here, however, the simulation is part of the design process where the model is of an as-yet-unrealised possible world. We need to ensure that the simulation has engineering value: its emergent properties should be sufficiently similar to the proposed engineered system. An additional problem is the non-obviousness of those emergent properties: the purpose of the simulation is to check that the system will have the desired properties.

To address these needs, both the emergent system and its simulation should be *co-designed* using an iterative and experimental approach: the simulation is used to explore possible engineered system prototypes expressing the desired emergent property; engineered system prototypes test, validate, and update, the design expressed by the simulation. This way the engineered system and simulation are developed in parallel with an explicit and traceable relationship, which expresses *why* the simulation behaves in a similar way to the engineered system. By understanding this relationship, the simulation becomes a tool for exploring the newly created world of the engineered system.

This approach is partly inspired by the use of increasingly predictive co-evolved simulations in certain evolutionary robotics developments [5]. Here, however, instead of using a "blind" evolutionary search through simulator and design space, we are proposing a non-evolutionary designed approach based on the rigorous CoSMoS development process. In particular, the resultant simulator of the final system will have been engineered to the same standard as a scientific simulator. This means it can be used as a component of arguments about the engineered properties of the final system. Hence we can *engineer*, rather than simply search for, emergent systems.

[1] EMA Ronald, M Sipper, MS Capcarrère. Testing for emergence in artificial life. *ECAL 1999*, LNCS 1674:13–20. Springer, 1999.
[2] S Stepney, F Polack, H Turner. Engineering Emergence. *ICECCS 2006*, pp 89-97. IEEE, 2006
[3] PS Andrews, F Polack, AT Sampson, S Stepney, J Timmis. The CoSMoS process, version 0.1, University of York technical report, 2010 (submitted)
[4] PS Andrews, S Stepney. Simulation as a scientific instrument (in preparation)
[5] H Lipson, J Bongard, V Zykov, E Malone. Evolutionary Robotics for Legged Machines: From Simulation to Physical Reality. *Proc 9th Intl. Conf on Intelligent Autonomous Systems*. 2006

The UK's strategy for climate and energy is set out in its Low Carbon Transition Plan (DECC, 2009). Its objective is for the UK to become a low carbon country and specifically to "*deliver emission cuts of 80% on 1990 by 2050*". It hopes to achieve this daunting target by the introduction of various schemes and targets which cut carbon emissions incrementally over 3 budget periods (see Carbon Accounting Regulations 2009). These schemes are the responsibility of government departments which must (see Carbon Change Act 2008 which sets limits on the total greenhouse gas emissions allowed from the UK) contribute to five areas of carbon emissions reduction (namely power and heavy industry, transport, homes and communities, workplaces and jobs, and farming, land and waste). However, assigning accountability and measuring activity is simply not enough to prevent the "*widespread human suffering, ecological catastrophes, and political and economic instability*" (DECC, 2009: 22) that is perceived as a rising risk of climate change relating to rising carbon emissions.

It is not surprising therefore that Ofgem (2010) has suggested a range of increasingly transformational options to address doubt over secure and sustainable energy supplies beyond 2015, brought about by the convergence of various factors, including the financial crisis, demanding environmental targets, increasing gas imports, and closure of ageing power stations. Their most radical option is the creation of a central electricity buyer which would essentially regulate the market for sustainable and secure energy generation. Ofgem's proposal gets closer to changing the behaviours of the firms and households which form the backbone of future energy generation.

As with all complex adaptive systems, there is great uncertainty in predicting the emergence of unwanted (and wanted) outcomes, for the very reason that the system has interconnected and nested components with some measure of autonomy. Within a complex system the behaviour of elemental components and their interactions needs to change in order to alter the emergent properties of the system. In the case of the climate the adverse emergent properties are related to the behaviours of people (in their homes and their transport behaviours) which are met by organizations (and their actions) within the built and natural environment. The interaction of these three components drives not only the economy but all the other emergent properties of the system, including undesirable carbon emissions.

So, to achieve transformational change means radically reassessing what there is now (homes, factories and offices, transport infrastructure, etc.) and the changes required to human and organization behaviours (stop, drastically reduce, adapt, create new) which are necessary and sufficient to maintain both the economy and low carbon emissions. A proposal along these lines from McKibbin & Watson (2002) for example proposes permits for firms which in total matches emissions targets. The emergence of a Low Carbon Economy cannot be engineered without assigning 'carbon budgets' to people and organizations. Government departments need to administer the carrots and sticks which will achieve changed behaviours and generate creative destruction which favours the desired new regime. They therefore need new complex, multi-scale, agent based modelling to allow them to do this.

DECC (2009), *The UK Low Carbon Transition Plan,* The Stationery Office, http://www.decc.gov.uk/en/content/cms/publications/lc_trans_plan/lc_trans_plan.aspx.

McKibbin, W. J. and Wilcoxen, P. J. (2002), "The Role of Economics in Climate Change Policy", *The Journal of Economic Perspectives,* 16(2): 107-129.

Ofgem (2010), *Action needed to ensure Britain's energy supplies remain secure,* http://www.ofgem.gov.uk/Media/PressRel/Documents1/Ofgem%20-%20Discovery%20phase%20II%20Draft%20v15.pdf.

Towards New Principles of Unbound Embodied Evolution

A.E. Eiben *M.C. Schut* *S. Kernbach*

Department of Computer Science
VU University Amsterdam
Amsterdam, The Netherlands

Institut für Parallele und
Verteilte Systeme
Universität Stuttgart
Germany

This position paper proposes a road map for extending today's evolutionary computation (EC) in order to address some challenging aspects of engineering emergence in complex systems. We consider EC to be a very fit candidate for contributing to the algorithmic machinery for accomplishing engineered emergence. Evolutionary computation is an powerful paradigm for generating solutions of optimization and design problems inspired by biological evolution. It has been successfully applied in many branches of engineering, business, artificial biological (synthetic biology) and technological (robotics) systems. Evolution-based models of computation in a broader sense are now 'invading' new interdisciplinary research areas in innovative fields within bio-chemistry, on-chip factories, reconfigurable and evolvable hardware, molecular computing, and pervasive adaptive systems. The resulting systems are permanently driven by internal and/or external forces with long-term evolve-ability, unbound developmental processes, and very strong embodiment. Traditional evolutionary computing does not provide knowledge about such novel systems.

Vision – We have a long term vision of having artificially built evolutionary systems for information processing (computing), where: the units (individuals, agents) are physical objects, rather than pieces of code inside a computer; selection and reproduction are asynchronously and autonomously executed by the units themselves, without central control; reproduction creates new objects, rather than replaces existing ones; survivor selection effectively terminates objects so that they really cease to exist; selection is geared towards survival in general as well as user preferences that represent a given computing task; evolution is, in principle, open ended (unbound), but deliberately aiming at continuous improvement of the computing capabilities.

Motivation – The motivation behind this vision lies in the expected benefits of this radically new kind of non-von-Neumannian information processing (computing) systems that could regulate available computing power through adjusting the population size to the requirements of the moment, optimize energy and material consumption by producing units when needed and terminating units when they are not necessary anymore, and finally, undergo a continuous evolution towards better computing capabilities.

Challenges – For successfully pursuing this vision, the following major technical and scientific challenges can be identified: 1) designing the physical units (that can be electro-mechanical, bio-chemical, hybrid of these two), 2) equipping the units with computing capabilities of their own and make the population one big computational entity that can do more than the sum of its parts, 3) designing the (evolutionary) reproduction and inheritance mechanisms for the physical units, 3) managing the population size to prevent explosion as well as implosion (selection), 4) interfacing the user's computational task to the evolutionary system, and 5) striking a good balance between the driving forces towards general survival (improving computing capabilities) and task dependent fitness (improving solution quality).

Engineering Emergent Properties in Artificial Distributed Systems

Ognen Paunovski[1], George Eleftherakis[2], Tony Cowling[3] and Ognen Spiroski[1]

[1] South East European Research Centre (SEERC)
24, Proxenou Koromila str., 54622 Thessaloniki, Greece
({ogpaunovski, ospiroski}@seerc.org)

[2] City College, International Faculty of the University of Sheffield
13, Tsimiski str., 54624 Thessaloniki, Greece
(eleftherakis@city.academic.gr)

[3] University of Sheffield
Regent Court, 211 Portobello Str., Sheffield, S1 4DP. UK.
(a.cowling@dcs.shef.ac.uk)

In the last few decades computer networks have exhibited a significant expansion in both size and diversity. Moreover, in many domains centralized architectures have been replaced by distributed approaches which are able to avoid single point failure and offer significant advantages in utilization of network resources. Nevertheless the increased demand and complexity of applications/services operating within distributed environments has emphasized the need for more efficient, robust and adaptive solutions which will operate without manual configuration and management.

In this direction a possible solution could be achieved by utilizing low level interaction between the components of the distributed system as basis for the emergent processes which will shape the global properties of the system. The ability to guide and use these emergent processes could prove very valuable since emergence is considered to be the basis for a variety of very useful phenomena including self - organization, self - optimization, adaptation as well as other beneficial properties encountered in complex systems. Thus incorporating engineered emergent behaviours in a distributed system could offer significant benefits to the development and performance of the system, making it highly available, scalable and robust.

On the other hand ignoring the appearance of emergent phenomena during the operation of a distributed system could cause many problems by generating unexpected or undesired behaviour which could infringe the system's operation, diminish its functional performance or create behaviour opposite to the one intended at design time. However, at the moment the inability to predict and control emergent phenomena is preventing us from exploring its full potential or avoiding problems in existing complex distributed systems. Furthermore this imposes a considerable strain on the practices employed in the design, development and maintenance of distributed systems which are becoming increasingly complex and unpredictable. Although there have been attempts to deal with emergent phenomena in the context of distributed systems, very few studies have shown any practical results and with very limited success.

In this paper we address several aspects of engineering distributed systems with emergent behaviour. Initially we examine the major elements which comprise a modern distributed system as well as the problems and challenges faced in its development and operation. We also try to establish a practical view on emergence and define the behaviours exhibited by natural systems that could be incorporated into an artificial distributed system in order to satisfy the imposed requirements. Finally we put forward several ideas on how to approach and revise engineering practices when dealing with systems exhibiting emergence.

Validation and simulation: engineering simulations for science

Fiona A. C. Polack [1]
Department of Computer Science, University of York, York, YO10 5DD, UK

Agent-based simulation (ABS) is a potentially-valuable tool in complex systems study, because it can address individual-level variation as well as population-level behaviour. Use of ABS has been limited by a lack of understanding of its strengths and limitations. A widespread view is that ABS of complex systems cannot be validated to a level where they can contribute to scientific research. This paper draws on experience of ABS development and validation in scientific research to identify a way forward.

Validation seeks to show that the right system has been built. A simulation *should be developed for a specific purpose... and its validity determined with respect to that purpose* [4]. Review of simulation engineering [3] shows that validation of conventional simulation does not adapt to ABS for exploring complex systems. Many techniques appeal to comparison between the simulation and other valid models [4, 5]: valid models are rare in the field of complex systems, and are unlikely to be directly comparable to the ABS in hand. Other techniques need deep understanding of the structure and behaviour of the simulated system [4]: in complex systems simulation, the *goal* of simulation is to try to understand deep structure and behaviour. However, conventional simulation engineering does point the way to guidelines for complex systems ABS validation.

A given here is that the goal of ABS validation is to demonstrate *reasonable adequacy*: as in safety-critical systems engineering and certification, it is impossible to demonstrate absolute correctness. Just as it is possible to argue that a system is as safe as it needs to be, ABS validation must be an appropriate justification for trust in the simulation. Validation is an *argument* that the simulation meets its scientific and engineering objectives. Again as in safety critical engineering, the argument is open to scrutiny by whoever needs convincing. For instance, if an ABS supports laboratory research into a biological complex system, then reasonable ABS adequacy is comparable to the adequacy of results of laboratory experimentation (see [2]).

The problem of validation can be broken down into three separable activities:

- *Engineering validation* (and verification) appeals to engineering practice. Its goal is to ensure that the ABS is a sound product.
- *Calibration* is a tuning activity. The goal is to tune the ABS parameters, behaviours, scales of operation, etc, to be appropriate to the scientific context. The fine-tuning also exposes the scientific basis of the ABS: assumptions, abstractions, simplifications, and mappings, as well as performance and outputs.
- *Scientific validation* accepts the strengths and limitations of the simulation, determined through calibration. The goal is to demonstrate the scientific use of the simulation, for instance by simulating real experiments and comparing results.

Validation is thus an argument of adequacy, over engineering and usage evidence. Adequacy is fundamentally tied to the purpose of each simulation, and to the intended criticality and impact of results [1]. *Criticality* relates to the role of the ABS in context. If the simulation is a speculative exploration of possible factors, it has low criticality. However, if the goal of simulation is to identify a missing link in the scientific understanding, then it has high criticality. *Impact* is similar, but not identical: a non-critical ABS might have disproportionate impact in a new or under-researched area, whilst a high-criticality ABS may have low impact because, once it has identified a critical link, this is confirmed by scientific research. If impact or criticality is high, then explicit validation of the engineering and the science must be planned. Modelling and design rationale, test results, calibration evidence etc. can be recorded to support the ABS adequacy argument. However, where criticality and impact is low, the validation evidence can be more implicit, and could rely on basic engineering quality control and normal laboratory lab-book records.

References

[1] P. S. Andrews, F. A. C. Polack, A. T. Sampson, S. Stepney, and J. Timmis. The CoSMoS Process, version 0.1. Technical Report YCS-2010-450, Dept of Computer Science, Univ. of York, 2010. www.cosmos-research.org/publications.

[2] F. A. C. Polack, P. S. Andrews, T. Ghetiu, M. Read, S. Stepney, J. Timmis, and A. T. Sampson. Reflections on the simulation of complex systems for science. In *ICECCS*, page tba. IEEE Press, 2010. to appear.

[3] F. A. C. Polack, P. S. Andrews, and A. T. Sampson. The engineering of concurrent simulations of complex systems. In *CEC*, pages 217–224. IEEE Press, 2009.

[4] R. G. Sargent. Verification and validation of simulation models. In *37th Winter Simulation Conference*, pages 130–143. ACM, 2005.

[5] B. P. Zeigler. A theory-based conceptual terminology for M&S VV&A. Technical Report 99S-SIW-064, Arizona Center for Integrative Modeling and Simulation, 1999. www.acims.arizona.edu/PUBLICATIONS/publications.shtml.

[1] Dr Polack wishes to acknowledge the CoSMoS project and its affiliates: EPSRC grants EP/E053505/1 and EP/E049419/1, www.cosmos-research.org, whose work is the inspiration for this paper.

An Enactive Semiology based Process to Explore Large Heterogeneous Datasets

Fabien Pfaender,
Cognition Research and Enactive Design
University of Technology of Compiègne, France

In reply to our crucial need for making sense of the world outside, one can now rely on a huge quantity of numerical data, ready to be stressed in every possible way by a new kind of explorer only limited by imagination. Tools and skills are no more jungle-ready but made of clever and innovative design and use of infographics and visualizations that are able to reveal the hidden [1]. New fields like EDA and visual analytics encourage the use of an expert and highly visual process to formulate and validate hypothesis on a dataset [2][3]. Once hypothesis are discovered and validated throught visualizations, they become knowledges of the system and can be used for further investigation. The key role of diagrams as part of a process of knowledge discovery increases the need for diagrams to be well conceived and well used so as not to engage on a discovery path leading at best to a dead-end, at worst to a false conclusion. To design relevant and efficient diagrams, one can count on several semiologic works like Bertin[4] or Tufte[5] but these semiologies are still widely based upon the designers experience and are difficult to apply to the extraordinary variety of recent visualizations. Furthermore, they focus on a presenting knows use of diagrams where a process of exploration requires a specific revealing unknows use.

To address these concerns, we propose an experimental enactive semiology that encompasses both perception and cognition to explain graphics reading. It provides a way of analyzing every kind of graphics from a perceptual point of view knowing how it might affect the user. This semiology provides a set of primary graphic structures (list, table, chart, graph and map) and properties of how human perceive and understand them. Then we propose a direct application of these properties and structures to enhance EDA or visual analytics by presenting a new process of data exploration that encompass current visual mining processes by promoting a systematic use of diagrams and visualizations given their perceptual properties. The process is based upon a loop of hypothesis creation and validation that gradually become more and more accurates to form a model of the dataset. The process itself can be managed through a visualization software following the semiology guidelines. However not automatic, It gives a human explorator an insightfull way of mining large dynamic and heterogeneous datasets without getting lost, building a good-fitting model of it, while maximizing his efficiency and minimizing his errors. It also allows him to communicate the discovered knowledges given the specificity and expectations of the target audience. Examples are from various web exploration and experimental work.

[1] MacEachren, A., M.: Visualization in modern cartography: Setting the Agenda. Visualization in Modern Cartography. 1–12 (1994)
[2] Thomas, J., J., Cook, K A.: Illuminating the Path: The Research and Development Agenda for Visual Analytics. National Visualization and Analytics Ctr. (2005)
[3] Andrienko, N., Andrienko, G.: Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach. Birkhauser. (2006)
[4] Bertin, J.: Sémiologie Graphique. les diagrammes, les réseaux, les cartes. First Edition. Mouton, Gauthier-villars Paris-LaHaye (1967)
[5] Tufte, E.: The Visual Display of Quantitative Information Cheschire press. Connecticut (1993)

Biological Metaphors for Critical Infrastructures

Eilidh McAdam[1*], Ruth Falconer[1], John Crawford[2] and Jim Bown[1]
[1] SIMBIOS Centre, University of Abertay Dundee
[2] Faculty of Agriculture, University of Sydney
[*] e.mcadam@abertay.ac.uk

Modern societies are heavily dependent upon a number of critical infrastructures that allow our societies to function in an effective manner. Many real-world examples highlight problems inherent to such infrastructure networks. For example, the problems associated with UK water supply were illustrated with the ‘leakage’ rates experienced by Thames Water where it became clear that the pipeline infrastructure for drinking water was experiencing considerable losses, principally due to a lack of appropriate investment. The dumping of aluminium sulphate into drinking water supplies (1988) also highlighted the extent to which key ‘nodes’ can be attacked. Similarly, the fine imposed on Network Rail (2006) in the aftermath of the Ladbrook Grove rail crash was based upon a lack of effective investment in the maintenance of the rail network and highlighted the manner in which (organisational) defences can be breached across organisational boundaries. Finally, the blockades of refineries in 2001 by UK farmers, highlighted the dependencies and vulnerabilities within the fuel distribution network.

Many life forms have evolved to be dependent on the efficient and effective functionality of network infrastructures. The fungi, an entire kingdom of life, epitomise biological networks where a fungal colony comprises a complex dynamic web of interconnected filaments (or hyphae) that is optimised for growth and survival in heterogeneous environments. It has been shown that global behaviour of the colony arises from local interactions, which are in turn dependent on the global context (Bown et al. 1999). The success of these organisms make them an appealing system to study in the current context; indeed the largest organism on Earth is a fungus, extending over 900 hectares and believed to be over 2500 years old. Colonies are plastic in structure and robust to damage: if parts of the network are disrupted new connections grow; disconnected parts behave in the same way as the connected whole.

We became the first to develop a physiologically-based model for the emergence of colony structure in fungi (Falconer et al. 2005, Falconer et al. 2008). We have shown that the fungal phenotype may have its origins in the defining characteristic of indeterminate organisms, namely their ability to recycle locally immobilized internal resources into a mobilized form capable of being directed to new internal sinks. Moreover that phenotype can be modelled as an emergent phenomenon arising from local processes governing uptake and remobilization of internal resources together with global resource transport processes. Naturally occurring observed complex growth forms are reproduced and the sensitive dependence of phenotype on environmental context may be understood in terms of non-linearities associated with regulation of the recycling apparatus. While this model links colony-scale pattern formation to local and non-local colony processes it does so within a restricted interpretation of the environment: i.e. a 2D or 3D lattice-based representation of soil structure.

This project seeks to generalise the computational model of fungal colony dynamics so that the properties associated with these colonies, e.g. plasticity in form, differential yet interconnected behaviours across the colony extent, self-healing properties, may be exploited in a wide range of systems. To that end, we have converted the lattice-based model into a graph-based formulation. Here, we show that fungal colonies still exhibit the essential emergent phenomena that make the fungal colony such an attractive metaphor. We will present some of the challenges attracted by this conversion, together with the emergent properties that the revised formulation exhibits, for example the biological phenomena of concentric rings in homogeneous environments. We will also highlight the applicability of the graph-based model to a range of network-based problems, such as load-balancing and optimisation of topological investment together with vulnerability to node failure.

References

Bown, J.L., et al. 1999. Evidence for emergent behaviour in the community dynamics of a fungal microcosm. Proceedings of the Royal Society B, 266: 1947-1952.

Falconer, R.E., Bown, J.L., White, N.A. & Crawford, J.W. 2005. Biomass recycling and the origin of phenotype in fungal mycelia. Proceedings of the Royal Society B. 272: 1727-1734.

Falconer, R.E., Bown, J.L., White, N.A. & Crawford, J.W. 2008. Modelling interactions in fungi. Royal Society Interface. 5: 603-615.

The Necessity of Post-construction Management for Complex Systems

Bruce Edmonds
Centre for Policy Modelling, Manchester Metropolitan University

Due to results derivative of those of Gödel and Turing, we can prove that there are apparently simple systems where it is impossible, in general, to "engineer" some well-specified and possible behaviour. That is to say there is no systematic method from going from a well-formed and precise specification of the behaviour to a particular system that corresponds to that specification *even when we know that there is such a system that will do the job* (Edmonds and Bryson 2004). In these kinds of system the outcomes are never determinable in advance without actually running the system and observing the results – that is they display "weak emergence" (Bedau 1997).

These systems can seem very simple. This is illustrated with a very simple finite system of agents with fixed sets of very simple plans, a single action of giving one unit to another agent, and a single decision procedure (looking to see if another agent has zero units to determine which plan to do next). Even if everything about the system is known except how many units the first agent has, then the system is Turing-complete and the above limitations in terms of design can be shown to hold. However this does not stop such systems being evolved by systems of mutation and selection by fitness evaluation.

Given that we can not rule out the possibility that the systems we need to engineer are not of this type, then it is inevitable that dealing with such systems will be somewhat of an experimental nature rather than determinable by "good theory". These systems will not be completely "engineerable" in the sense of good pre-construction planning, but rather will need to be monitored and adjusted (and possibly discarded) post-construction (Edmonds 2005). This goes against the instincts of good engineers, who seek to minimise post-construction adjustment by means of sound design but it may simply be that for some kinds of system this is not the most effective strategy. Biological evolution is an example of this, where there is no natural "genetic engineering" but rather a trial-and-error combined with selection. Similarly it is suggested that science itself is such a process (Popper 1972, Toulmin 1967).

Thus effective means of growing, testing, monitoring, adjusting and discarding complex systems might be as important as efforts to ensure that systems meet specified goals. This process may appear to be more akin to farming than engineering – "system farming". Engineering approaches will still have significant roles to play, for example in the design of parts of the system and in how to grow systems that are more likely to meet the desired goals. However other aspects will take on new importance, for example: identifying key threats to the system, devising effective indicators that would give early indications of system health, evaluating possible intervention strategies, and developing methodologies to continually update clusters of diagnostic simulation models.

Bedau, M. A. (1997) Weak Emergence. In Tomberlin, J. (ed.) *Philosophical Perspectives: Mind, Causation, and World*, vol. 11, Blackwell Publishers, 375-399.

Edmonds, B. and Bryson, J.J. (2004) The Insufficiency of Formal Design Methods – The necessity of an experimental approach for the understanding and control of complex MAS. In *Proc. of the 3rd Int. Joint Conf. on Autonomous Agents and Multiagent Systems* (AAMAS 2004). IEEE Computer Society, 938-945.

Edmonds, B. (2005) Using the experimental method to produce reliable self-organised systems. In Engineering Self-Organising Systems: Methodologies and Applications, *Lecture Notes in Artificial Intelligence*, 3464:84-99.

Popper, K. (1972) *Objective Knowledge: An Evolutionary Approach*, Routledge.

Toulmin, S. (1967) The Evolutionary Development of Natural Science, *American Scientist*, 55:4.

Abstract for Engineering Emergence workshop

The Emergence Engineers' Dilemma: it seems we can evolve emergence, or prove emergent properties, but not both

Alan Winfield, Bristol Robotics Laboratory

The theme of this workshop "engineering emergence" precisely describes the core problem of Swarm Robotics. How do we design the microscopic behaviours of the individual robots of the swarm so that when those robots interact with each other and their environment, the desired macroscopic swarm properties do actually emerge? And given that this is an engineering problem with – we assume – the ultimate aim of real-world applications, how can we be sure that the system will always do the right (emergent) thing and, equally importantly, never do the wrong thing?

At present the swarm engineer has two methods for designing the atomic behaviours of individual robots. The first is better characterised as one of discovery rather than design: a process of trial and error experiments with simulated or real robots, aided by experience, that allows us to find the individual robot behaviours which will lead to our desired emergent properties [1]. The second and arguably more principled approach makes use of a genetic algorithm (GA) to evolve the individual robot behaviours in order to maximise overall swarm fitness [2].

The first approach typically results in a set of discrete behaviours and their relationships that can be described well as a finite state machine (FSM) for individual robots, or a probabilistic finite state machine (PFSM) for the whole swarm. The approach has the advantage that it lends itself to validation with mathematical models [3], and formal verification using logic provers and model checking is starting to show some promise [4].

The second, evolutionary robotics, approach typically requires us to use an Artificial Neural Network (ANN) for the individual robots' controllers, then evolve the connection weights in the ANN. There is of course still a good deal of hand-design in the sense that we need to choose the number of layers, hidden neurons and recurrent connections, etc, in the ANN but - essentially - the ANN has inputs from all of the robot's sensors and outputs to all of its motors. Thus the evolved design cannot be discretised into a finite number of states and - especially if there are hidden layers - can remain inscrutable to analysis.

Thus, it seems possible that we face a dilemma. The unprincipled approach to engineering emergence in robot swarms does allow us to develop good arguments for assuring the safety and dependability of the swarm, yet the principled approach – it appears – does not. This leads us to the question of this talk: is the problem of our being unable to prove the correctness of evolved systems because of some fundamental property of evolved systems, or simply because we haven't yet figured out how to evolve systems that are amenable to mathematical modelling and analysis?

References

[1] M Mataric, Designing emergent behaviors: From local interactions to collective intelligence. In J. Meyers, H.L Roitblat and S.W.Wilson eds, From Animals to Animats II: Proc. 2nd Int. Conf. On simulation of adaptive behavior, pages 432-441, MIT Press, 1992.

[2] V Trianni, Evolutionary Swarm Robotics: evolving self-organising behaviours in groups of autonomous robots, Springer, 2008

[3] AFT Winfield, W Liu, J Nembrini, and A Martinoli. Modelling a wireless connected swarm of mobile robots. Swarm Intelligence, 2(2-4):241–266, 2008.

[4] C Dixon, AFT Winfield and M Fisher, 'Temporal Verification of Emergent Behaviours in Swarm Robotic Systems', submitted to AAAI 2010.

Adam Nellis, University of York
Emergent evolution allows meta-evolution

Evolutionary algorithms are one way of tackling complex engineering problems. Their main weakness is they need customising and parameterising for every particular problem to which they are applied. It would be nice if evolution could be used on a meta-level, to evolve an appropriate evolutionary algorithm for each given problem. The issues so far with meta-evolution are that if done naively, it is difficult to set up and incredibly slow to run.

We propose that meta-evolution can be used in a principled way, allowing the evolutionary algorithm to evolve as it runs, while maintaining fine-grained control of the computational efficiency of the meta-algorithm. The trick is to make evolution an emergent property of a system of organisms that are being evolved, rather than explicitly coding an evolutionary algorithm in a high-level programming language. If evolution is an emergent property of the organisms being evolved, then evolution used to evolve fitter organisms will evolve the “evolutionary algorithm” as well.

Evolution can be made an emergent property of the organisms being evolved: by representing the organisms as artificial life programs performing some task (the problem to be solved). But these artificial life programs are not coded in a high-level programming language. They are implemented as collections of redundant fragments of code written in a simple and evolvable language based on artificial chemistries. The particular language is chosen such that collections of code fragments can represent both solutions to the problem and the ingredients of evolution. The ingredients needed are: a representation of the organism’s genotype, which can be read so it can be expressed into the organism’s phenotype to solve the problem; and a mechanism by which the genotype can be copied imperfectly, so it can reproduce with variation. The problem solution and evolutionary algorithm are both written in the same, evolvable, language. Thus evolution of the evolutionary algorithm can happen, interleaved with regular evolution of the problem solution.

A benefit of using collections of code fragments is that the individual fragments of code have some independence from each other. This means that if a particular fragment is run frequently, it can be singled out. The functionality of this fragment can be re-implemented in a high-level programming language that will execute quickly but will not be evolvable. This allows a tradeoff to be made between the evolvability of the system and the speed at which the program executes. Meta-evolution can be “turned on”, allowing the system to discover novel evolutionary mechanisms, and then later “turned off”so the program can use those mechanisms in a more computationally efficient manner. Moreover, different parts of the system can be turned on and off in this way, to give fine-grained control of the computational efficiency of the program.

These methods are currently being considered for their applicability to problems in control engineering. Different artificial chemistries are being investigated to determine precisely which properties of the chemistry contribute in which ways to the evolvability of the resulting organisms.

Biological markets: a catalyst for the major transitions?

Steve Phelps
Centre for Computational Finance and Economic Agents, University of Essex
sphelps@essex.ac.uk

Many approaches to engineering emergence are biologically inspired. The implicit assumption in such approaches is that natural selection is sufficient to produce emergent complexity. However, one of the puzzles of evolutionary biology is that the complexity we observe in nature cannot always be explained solely by natural selection operating at the level of individual genes. For most of our planet's history, life consisted of simple single-celled organisms and it is only relatively recently that a diverse range of more complex and specialized phenotypes emerged. In early evolutionary history, genes themselves were not the original replicators. Rather, the intricate machinery for replicating strands of DNA itself evolved from more primitive systems of molecules that were able to self-replicate in the absence of enzymes. Complexity in nature emerged from a series of such “major transitions” in the units of selection: genes cooperated to form regulatory networks; similarly cells emerged from networks, multi-cellular organisms from cells and societies from organisms.

Thus we exchange one puzzle for another: if the major transitions require the evolution of cooperation between lower-level selfish replicators, and we wish to explain how the transition to higher levels of selection is systematic, as opposed rather than merely serendipitous, then we need to explain how cooperative strategies can systematically evolve in populations of selfish agents. A great deal of research has uncovered sufficient conditions for cooperative outcomes. However, although there are many stylised scenarios in which cooperation can be shown to stable, there are equally many in which defection prospers.

A new approach to explaining reciprocity in nature appeals to one of the mechanisms that enable reciprocity in human societies, viz. markets. The central insight is that just as trade can give rise to specialisation and mutual benefit in our own species, the same principles apply to interactions in other species. The hypothesis is that this “invisible hand” is a universal phenomena of nature rather than a parochial artefact applying only to humans. If the invisible hand is indeed a universal phenomena of nature, it provides a powerful explanation for the major transitions in nature: for example, biodiversity arises from speciation, which I shall argue arises from economic incentives to specialise. This suggests that ideas from economics and finance may play a role in governing major transitions in *artificial* systems, and hence the *engineering* of emergence.