Self-organization in Software Design

Joel Palmius
Department of Information Technology and Media (ITM), Mid Sweden University
joel.palmius@itk.mh.se, http://www.itm.mh.se/itk/~joepal

Abstract
This paper describes some ideas about possible future research within the area of self-organization. Self-organization can be described as when a system moves from "bad" to "good" organization without explicit pressure from the outside. Self-organizing systems has its closest scientific relatives within AI, AL and CS. The research proposed includes studying strategies for AL, system theories and SOS, vector-orientation of state space, and machine cognition. The applications would include demonstrations of meta-theories, automation, autonomy, intelligence and game programming.

Keywords: Self-organization, design, artificial life, artificial intelligence, automation

1. Introduction

This paper should be viewed as a summary of intentions and ideas about my coming doctoral studies. It is far from a detailed plan on how to move onwards, it's rather a base for discussion to use for the composition of such a plan.

The background for these ideas are partly a paper written to IRIS23 (J. Palmius, 2000b), concerning the development of tools for analysis of state spaces, and mainly the master thesis (J. Palmius, 2000a) I wrote upon self-organization and artificial life.

2. Self-organization

As the complexity of artificial systems around us grow, it gets increasingly hard to keep all the details in the mind at the same time. The human mind cannot grasp infinite complexity (Miller 1956). This might eventually set a limit to what systems can be handled and developed. So far, the usual method of solving the complexity has been by using the analytical approach and trying to split the problem or system into small building blocks (as an example, lines of code), and by combining these blocks into larger blocks (macros).

Implicit in this method is a reduction of the problem in question. To be able to design a system using traditional means, the designer must split the system into parts detailed enough to be possible to describe (as an example in code). However, some systems are not possible to handle this way, as is shown by the work with traditional artificial intelligence (Penrose, 1990). To retain the functionality of the system, the parts will be so many and disperse that the "reduction" resulted in something that was more complex.
To continue with the example of intelligence: Obviously construction of intelligence is possible, since it has been done accidentally by natural evolution (Darwin, 1859).
Since evolution is a mechanical and determined process (Varela and Maturana, 1987), it is possible to emulate it, something that indeed has been shown (Ray, 2000). Further, the principle of evolution is not something that is limited to biological life, but can be employed in practical situation (Husbands, Harvey, Cliff and Miller, 1997).

One alternative to traditional design by the analytical approach is to use a chaotic whole, and let that whole organize according to some internal principle. This principle might as an example be "natural" selection as in the case of biological evolution. The gain in using self-organization would be that detail design should not have to be tinkered with, since the system is meant to move more or less on its own accord towards the desired state. When developing (or rather evolving) systems this way, it is implicit in the method that control by analysis is beside the point.

Self-organizing systems has been an area of research to and from since the mid of the previous century. There has been much debate about its usefulness, and even claims that it theoretically impossible (Foerster, 1960), something that might have a base in the difference between the various definitions of the concept.

Foerster defined the concept of self-organizing system as a system that decreases its level of entropy over time. To obey the second law of thermodynamics ("energy cannot be created nor destroyed"), this entropy would have to be absorbed by the suprasystem. When using this definition of self-organizing system, the question of self-organization largely became academic, since it could be reduced to redistribution of entropy within the suprasystem.

Later, Ashby defined self-organizing system as a system moving from "bad" to "good" organization, claiming that all systems strive for equilibrium (Ashby, 1962). He also added that self-organization was not a special condition, but something that is true for all *dynamic* systems acting under unchanging laws.

After the research discipline of self-organizing systems theory has been established, a FAQ collecting common definitions and questions has been written (Lucas, 2000). In it, self-organizing systems are described as when system structure appears and evolves without explicit pressure from the environment: the change agents are internal to the system.

2.1 The Surrounding Areas

Self-organizing Systems (SOS) is closely knit with a numer of other diciplines, such as Control Systems (CS), Artificial Intelligence (AI) and Artificial Life (AL). Apart from these, general techniques such as Genetic Algorithms (GA), Evolutionary Algoritms (EA), evolutionary design, and Artificial Neural Networks (ANN), are used in both SOS and the surrounding areas. All these areas and techniques could be said to fall within the more general area of Complex Systems (C. Lucas, 2000).

In figure 1 (below), you can see some general relations between SOS and the relevant surrounding areas (no significance is attached to the size of the circles).

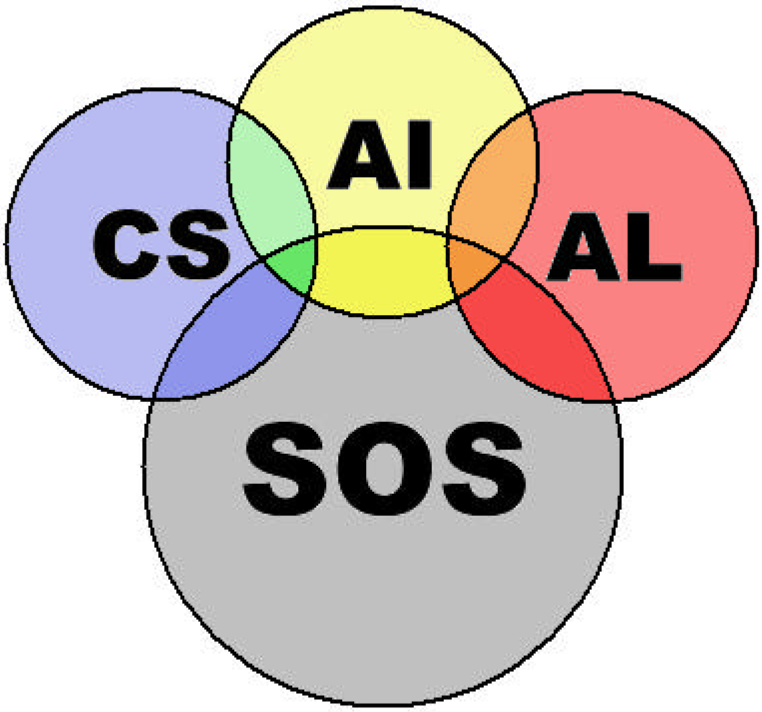

Figure 1: Self-organizing systems (SOS) is closely knit with a number of other areas, such as Control Systems (CS), Artificial Intelligence (AI) and Artificial Life (AL).

Control Systems is the study of regulatory systems in the general form of input, process, output, feedback. Control Systems ar determined usually mechanical systems which maintain their equilibrium by balancing their internal structure according to a prediction of what is to come.

Artificial Intelligence is, in general, the study of how to emulate human intelligence. AI is in itself a number of diciplines and paradigms however. These include learning systems, fuzzy logic and applications such as speech recognition and machine perception.

Artificial Life is the dicipline of trying to emulate natural life in an artificial environment, such as a computer. Artificial Life is more of a collected name for a number of different demonstrations rather than a scientific dicipline.

3. Points of Interest

There are several groups and institutes researching the above topics. However, I feel that more could be said about a number of points within them. It should be noted though, that at this early stage, I have not made a determined effort to find studies describing the research I am proposing. It is possible that these points have been examined before.

3.1 Strategies for AL

Artificial Life (AL) has been the object of several studies, and even more hobby projects. It is an area of some fascination. I have an idea that these different projects build on a couple of different world-views and therefore use different strategies to achieve the intended living system. These strategies do most probably have several points in common as well as points that make them different. I propose a survey to investigate and map these projects and compare them, in order to find similarities and differences, and hopefully build a collected conclusion.

3.2 System Theories and SOS

Traditionally, the study of SOS, AI and AL has been an engineering subject. It has in most cases been approached from the horizon of mathematics, deduction and analysis. Indeed, meta-analysis of the system to be built have not been necessary in order to build the system, in the same way that it is not necessary to know the nature of self-awareness in order to treat spider phobia.

On the other hand, in the study of systems in general (or, in other words, not in the explicit SOS area), many "systems" theories have been proposed. These includes theories and models such as Miller's "Living Systems" (J. Miller, 1978), Ashby's "law of requisite variety" (R. Ashby, 1956) and Maturana/Varela's "Autopoiesis" (F. Varela and H. Maturana, 1978). As far as I know, these has generally not been applied in practise to self-organizing systems, artificial intelligence and artificial life. My theory about the reason for this is that they are too abstract to be of use in practical programming.

I propose that several of these abstract systems theories could be operationalized and brought down to the level of programming in relation to self-organizing systems. The method for this would be first study the relevant systems theories, and then try to map them on SOS applications such as AI and AL.

3.3 Base Principles for Vector Orientation of SOS

One problem with chaotic self-organization is that is difficult to grasp and set going into the desired direction. A couple of tools for mapping the state space and the directions of self-organizing systems is needed. Most of the below points were also presented in a paper presented at IRIS-23 (Palmius, 2000b).

3.3.1 State Space Respresentation

The first tool is a way of representing the state space of a system that is intended to self-organize. In order to plan a path, or at least set the system going in a desired direction with an intended goal, a map of the road is required or at least very usable.
However, this representation could prove to be rather difficult, as a system could be said to have one dimension per internal variable (Lucas, 2000). There is of course always the brute-force approach to the representation: Store information about every possible setting for every available variable in a database, and set indexes according to mutant neighborhood in state space. This might be theoretically possible, but somewhat cumbersome.

Let us picture a chessboard with its 32 pieces as a system with a state space. Each piece is a dimension in the state space, and every move of a piece moves the state space location into a mutant neighbor location. Now, with the brute-force approach, we would be required to store information about 64^{32} (approximately $6.2 * 10^{57}$) locations and their relations (minus the impossible locations such as two pieces in one square and so on). In other words, given that every state is a collection of 32 bytes describing the position of each piece (variable), we would have to store a collection of data several magnitudes larger than anything stored before. This is clearly not economically feasible.

Instead, another way will have to be developed in order to represent the state space of a system. I do not as of yet have a good view of how this representation model will look, but I propose that it might be possible to map key points in the state space, and from there describe their closest mutant neighbors. This would be an rough approximation of the interesting area.

3.3.2 State Space Vectors and Vector Calculus

The second tool, which depends heavily on the first, is a way of representing vectors in the state space of a system. Some points or areas in or of state space could be said to represent attractors for the self-organizing system (Lucas, 2000). With an attractor, there is a force "trying" to make the system move in a direction through state space. This direction and the force that causes the move in the direction, could be viewed as a force vector, and should ideally be representable as such.

When it is possible to represent vectors, it should be possible to treat them in much the same way as force vectors in a physical space. Opposing vectors should result in a standstill or a slight nudge in the direction of the stronger vector. Resultants should be calculable in the case of several attractors dragging into different directions.

With this tool available, it would be possible to set an approximated course through the state space of a system by placing several attractors. This would hopefully vectors with a resultant pointing towards the desired subset of state space.

It is possible the "vector" is a bad metaphor for the method to use to describe the state transitions of a system through its state space. But for the sake of describing what has to be developped, it will have to suffice until the subject has been research further.

3.4 Machine Cognition and Self-awareness

The concept of self-awareness has been a matter of some philosophical debate during the millenia. What is self-awareness in practise? Several answers has been proposed.

The first is, of course, the interpretation of the phenomenon self-awareness as being a "soul", something which is independent of the body and the brain. This explanation could be called the religious explanation, and it effectively disables further research about self-awareness, as it is inherent in the concept of a soul that it is beyond physical reality and therefore not available for observation.

The second, which gained some popularity during the previous century, is that self-awareness is a regulatory function in the brain and that it there-fore should be possible to locate or map physically in the brain. This explanation could be called the machine view of the brain, and has its closest systems-science relatives within cybernetics and control systems. As of today, the attempts to map self-awareness into physical functions has not been vary successful. Certain patterns of brain activation has been possible to study with PET scans in conjunctions with stimuli and activitity (as an example, having the observed person try to remember what he did yesterday), but that is as far as the analysis has come.

The third explanation, which has not gained much acceptance, is that self-awareness is a a higher function, a synergetic effect, of the brain. Inherent in this view is the idea that the brain is a combination of a vast number of uncomplicated systems which together make up a complex whole. Here, the question of self-awareness becomes more a question of quantity rather than quality. Self-awareness cannot be said to be a *part* of the brain. It is rather a *result* of the brain. The difficulty with this view is that it removes the basis for observation. The brain is too complex to study as a whole, but inherent in the view is that analysis of the brain will remove the phenomenon which was the object of the study.

One major issue with artificial intelligence is that of initiative. One common comment upon "intelligent" machine is that "they do only what we tell them to" (or in other words, the intelligence is in the programmer, not in the machine).

I propose that it would be interesting to try gain machine initiative by trying to build some kind of self-awareness into machines. Some requirements for this self-awareness would be:

- Machine cognition; self-awareness is a kind of cognition, or rather a meta-cognition. Meta-cognition is pointless if there isn't anything to be "meta" about.
- Perception, or at least self-perception. My interpretation of self-awareness is that its object often is to place the self in relation to a reality, or to place different parts of oneself (mainly cognitions) in relation to each other. For the machine to be aware, it has to have input data to be aware of.

4. Applications

However interesting, what is the use of research about the above points? I shall here try to give a few examples of the expected outcome and their practical use.

4.1 Demonstrating meta-theories

The first expected outcome of the research is the operationalization and demonstration of several abstract theories and models. In practise this will mean programming examples for the demonstrations, and models for implementation for the operationalization. The application of this is that they will be a help for the building of self-organizing systems.

4.2 Autonomy and Intelligence

The second expected outcome is guidelines and models for implementation of autonomy through self-awareness. The difference between automation and autonomy is that the former implies rigid rules governing action ("if A happens do B, otherwise do C"), while the latter implies an agent with the power to take initiative and use new sollutions. It is my hope that the outcome models will provide a way to build self-aware intelligence with the power to take independent initiatives.

4.3 Automation and Adaptation

As a complement to the previous, the third expected outcome is tools for automation and adaptation. The idea is that systems should be able to handle situations not considered at the time of programming, by having a mechanism for adaptation. This is different from the autonomy principle by the lack of intelligence and initiative. I do rather expect a model built on Ashby's requisite variety, for the maintaining of equilibrium. By implementing self-organizatory and adaptive principles, systems should be able to handle more situations without direct intervention from the programmer.

4.4 Game Programming

The fourth expected outcome is tools usable for artificial life, something that could be used in the computer games industry. Several (often very successful) attempts on implementing NPC AI in games have been made, and I feel that the same could be possible with AL. There are some basic example implementations available[1] already, but I feel that there is more to be done.

1See as an example the WorldForge project at http://www.worldforge.com

5. References

R. Ashby. An Introduction to Cybernetics (1956). From Principia Cybernetica: HTTP://pcp.vub.ac.be/books/IntroCyb.pdf (Link was valid january 2001).

R. Ashby. Principles of the Self-Organizing System (1962). From Klir (edt): Facets of Systems Science (1991). New York, USA, Plenum Press.

C. Darwin. On the Origin of the Species by Means of Natural Selection (1859). From Huxley: The Living Thoughts of Darwin. Cassel, London, Great Britain, 1946.

H. Foerster. On Self-organizing Systems and Their Environments (1960). From H. Foerster (edt): Observing Systems. Intersystems Publications, Seaside, CA, USA 1984.

P. Husbands, I. Harvey, D. Cliff and G. Miller. Artificial Evolution: A New Path for Artificial Intelligence? From Brain and Cognition no. 37 (1997)

C. Lucas. Self-organizing Systems FAQ version 2.4. http://www.calresco.org/sos/sosfaq.htm, 2000. (Link was valid januray 2001)

G. Miller. The Magical Number Seven Plus Minus Two (1956). From HTTP://www.well.com/user/smalin/miller.html. (Link was valid january 2001).

J. Miller. Living Systems (1978). USA: McGraw-hill.

J. Palmius. The Design of Self-organisation (2000). Master thesis presented at ITK, Mid Sweden University. HTTP://www.itk.mh.se/~joepal/longterm/works/tdoso-20000207.pdf

J. Palmius, State Space Vectors (2000). Proceedings of IRIS-23. Uddevalla, Sweden: University of Trollhättan-Uddevalla.

R. Penrose. The Emperor's New Mind. Vintage Books, London, Great Britain, 1990.

T. Ray. Tierra Home Page. HTTP://www.hip.atr.co.jp/~ray/tierra/ (Link was valid january 2001)

F. Varela and H. Maturana. The tree of Knowledge. New Science Library, Boston, MA, USA, 1987