Complex system boundaries and where to draw them.

John Mansfield
Adjunct Research Fellow
Systems Engineering and Evaluation Centre
University of South Australia

1. ABSTRACT

In this paper system engineering of complex socio-technical systems is considered as steering the design and implementation of co-evolving sub-systems in the direction of the currently desired system behaviour. The major impediment to a traditional engineering approach is the continual, co-evolutionary change that is fundamental to these systems. Co-evolutionary change dictates that physical laws play a very small part in the behaviour of such systems. In order to gain a better appreciation of the behaviour of an "engineered" system we need to change our mental image of a complex system. This paper describes a novel visualisation of a complex system, discusses the critical concept of the system boundary and a number of fundamental characteristics that affect the system boundaries. The paper contends that complex systems evolve; hence complex system behaviours are grown not engineered and suggests that system engineers discard the pretence that the exact behaviour of a complex system can be specified in advance.

Keywords: Change, co-evolution, complex systems, socio-technical systems, system boundaries.

2. INTRODUCTION

In this paper system engineering of large, complex socio-technical systems is considered as steering the design and implementation of co-evolving sub-systems in the direction of the currently desired system behaviour. The major impediment to a traditional engineering approach is the continual, co-evolutionary change that is fundamental to these systems. Co-evolutionary change dictates that physical laws and their associated mathematics play a very small part in the behaviour of such systems. In order to improve the predictions of the behaviour of an engineered system we need to step back, redefine the attributes of a complex system and then move forward from the redefinition. This paper describes a novel visualisation of a complex system, discusses the critical concept of the system boundary and a number of fundamental characteristics that affect the system behaviour. It recognises that the act of changing a complex system has a high probability of causing unanticipated system behaviour. It suggests we therefore abandon the belief that the exact behaviour of a complex system can be forecast, and hence, we must also abandon the belief that the complete design of a complex system can be specified in advance. Complex systems evolve consequently complex systems are grown, not engineered.

3. CO-EVOLUTIONARY CHANGE

Co-evolution as an extension of evolution is easy to define but the concept, and its ramifications, is difficult to grasp. Kelly devotes a whole chapter of his book *Out of Control* to defining co-evolution, in part he says

> *"The formal definition of co-evolution runs something like this: 'Co-evolution is reciprocal evolutionary change in interacting species,' says John Thompson in Interaction and Coevolution. But what actually happens is more like a tango. The milkweed and monarch* [butterfly] *shoulder to shoulder, lock into a single system, an evolution towards and with each other. … The two become one. Biochemist James Lovelock writes of this embrace, 'The evolution of a species is inseparable from the*

evolution of its environment. The two processes are tightly coupled as a single indivisible process.' "
(Kelly 1994)

This coupling is critical to the concept of co-evolution.

Later in his book Kelly quotes Stuart Brand's introduction to CoEvolution Quarterly, saying, *"Evolution is adapting to meet one's needs. Coevolution, the larger view, is adapting to meet each other's needs".* This anthropomorphising of the process, implying a form of altruism is clearly incorrect. Individual agents adapt, i.e. evolve, to improve their own fitness; they only consider other agents when those agents influence their fitness. Adaptive subsystems are distinguished by their ability to become "fitter" i.e. to become better at some defined task, whereas the product of co-evolution is not necessarily a system with a specific emergent behaviour. Change in a co-evolutionary subsystem is simply a local matter, there is no overall fitness factor and there is no sense of progress of the global system.

There has been considerable research on evolutionary change within software systems but this has mainly been oriented to software quality and maintenance (Zamperoni, Gerritsen et al. 1995; Boehm 1998; Gilb 1998; Arisholm and Sjoberg 2000; Arisholm 2001).

The ubiquity of change in information systems was recognised early on by Fredrick Brookes, in 1975 in his classic "The Mythical Man-Month", (Brooks 1975; Brooks 1995) he says

"The only constancy is change itself. Once one recognizes that a pilot system must be built and discarded, and that a redesign with changed ideas is inevitable, it becomes useful to face the whole phenomenon of change. The first step is to accept the fact of change as a way of life, rather than an untoward and annoying exception. Cosgrove has perceptively pointed out that the programmer delivers satisfaction of a user need rather than any tangible product. And both the actual need and the user's perception of that need will change as programs are built, tested, and used."

Even now, more than 25 years later, because of the complexity of interaction within and between social, organisational and ICT-based systems, change is inevitable and unpredictable. Nevertheless, Brookes' point that change, in computer-based systems, should be accepted as a fact of life is clearly still not acceptable by a large proportion of system owners and builders. Complex, socio-technical systems exist in a dynamic environment; almost everything that affects them is changing. To survive systems must change, i.e. they *must co-evolve with their environment*.

Contrary to discussions about entropy, my contention is that any isolated system will reach internal equilibrium and will not change, but here the key word is "isolated". In fact no real system is unaffected by other systems. Whether the system in question is a galaxy, a bacterium or a computer system it exists in an environment of other systems. These other systems change and those changes impact on the system of interest, and if their impact is sufficiently severe they will change that system. The problem when considering a system is what to include and where to draw the boundary.

4. SO WHAT IS A COMPLEX SYSTEM?

In order to be considered as a complex system a collection of interacting entities must itself be recognised as an entity with a boundary, in the way that an operating system or a transport system is an entity. What is more, it must have a collection of attributes by which it is recognised. This definition is recursive in that a system may be a sub-system in a larger system which in turn may be a sub-system in a yet larger system, and so on.

These attributes can be envisaged as dimensions in a design space. The dimensions can have a range of values, for example: the colour dimension can be red, green and blue; the height dimension can be in millimetres, and the state dimension can be true or false. Each point in the multi-dimensional space represents an instantiation of a system with particular attribute values. Hence, any complex system may be considered an entity occupying a single point in a multi-dimensional space. Where there are multiple sub-systems within a complete system the multi-dimensional space becomes a superset of all the dimensions that occur in one or more sub-systems. For instance, one subsystem may have the dimensions of arms, legs and a head, whilst another may have none of these but has wheels and doors – thus the complete system includes, inter alia, arms, legs, head, wheels and doors. Visualising a system in multiple dimensions is difficult when the number of dimensions exceed three, so a convenient representation in two dimensions is that of a "spider web" with each system dimension a radial line. In the example in Figure 1 the entity that is the system is recognised by five dimensions – number of arms, number of legs, presence of a head, number of wheels and number of doors. The locus of the current dimension values characterises a particular system in a particular state. As the dimensional values change the locus changes and the instantiation of the system moves in multi-dimensional space.

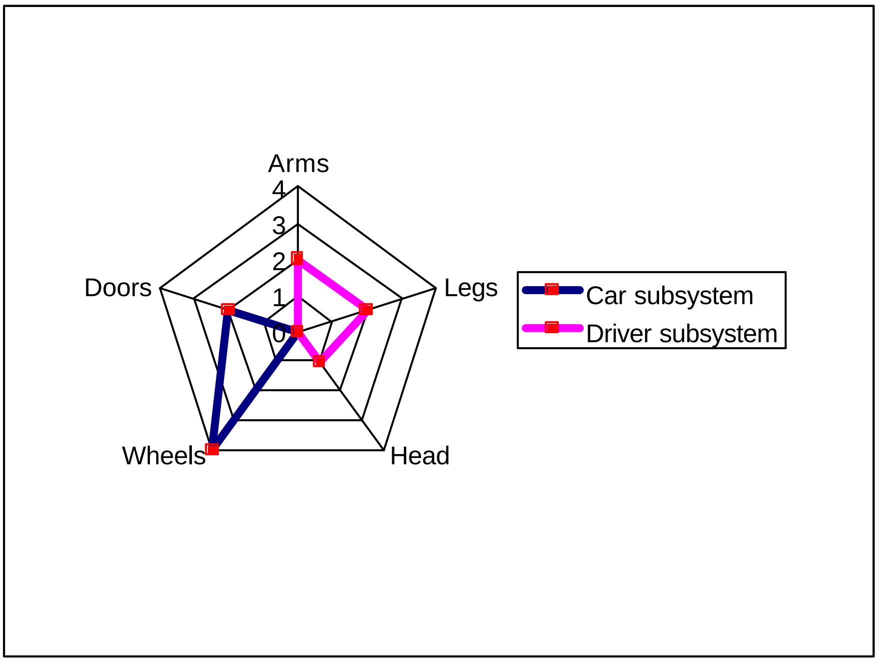

Figure 1 Spider web diagram of a system instantiated with different values for each of the subsystems

These dimensions do not exist in isolation many of them are linked. Being linked means if one value changes in its dimension any linked values move in their dimension. It is clear that in real systems the strength of the link between two specific dimensions may be asymmetric, for example the number of legs of a natural system will influence the speed at which the system can move but the speed at which the system is moving may be completely independent of the number of legs – when sliding down a hill for instance – and hence the strength of the link between dimensions is uni-directional.

Thus dimension values can be acted upon by the dimension values of other sub-systems. These sub-systems can consist of a disjoint set of dimensions to that of the sub-system acted upon. If a dimension exists in two or more linked sets the movement of its value is complex and may be deterministically chaotic, this point was established in an earlier paper (Mansfield 2004). Dimensions of the sub-systems can be also be acted upon by sub-systems that are considered outside the system boundary causing unexpected changes.

5. BOUNDARIES

When studying a system, we first decide on the system's dimensions then the next consideration is where should the boundary be drawn. **The drawing of the boundary is the single most important act in the defining of a complex system.** The boundary of a complex system is arbitrary. Boundaries are drawn to create a manageable, abstract view of the problem from the messiness of the real world. In the real world boundaries are permeable but once a model is constructed the influences from outside the boundary are assumed negligible and hence neglectable. By definition, any influence that is known to have an effect on the complete system must be included within the model's system boundary. The problem lies in deciding how much effect is "an effect". Influences from outside the boundary may fall off with distance like light or gravity over astronomical distances or may be all pervading.

If we are considering the influences on the central system where in Figure 2 do we draw the boundary? At the inner or the outer circle?

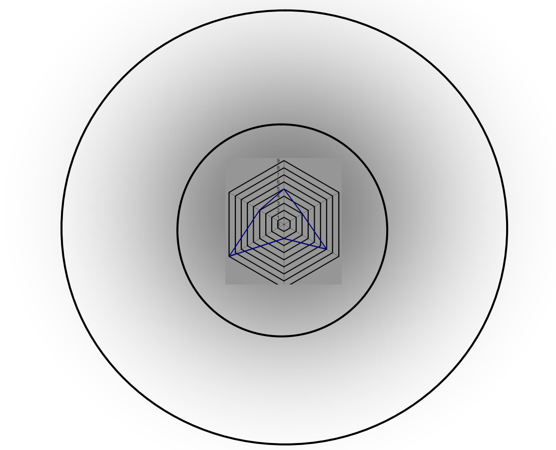

Figure 2 Graduated influence

Influences can be unvarying both inside and outside the system boundary, like the effect of gravity on a system on the surface of the Earth, see Figure 3. Such influences can have a major effect on the system but, because they are all pervasive and apparently immutable, they are usually ignored to simplify the abstract view.

This point of view may have a catastrophic effect when the outside influence changes, as when farmers clear trees to provide a greater fertile acreage and ignore the fact that they are lowering the water table until salinity rises through their soil and makes all of it infertile.

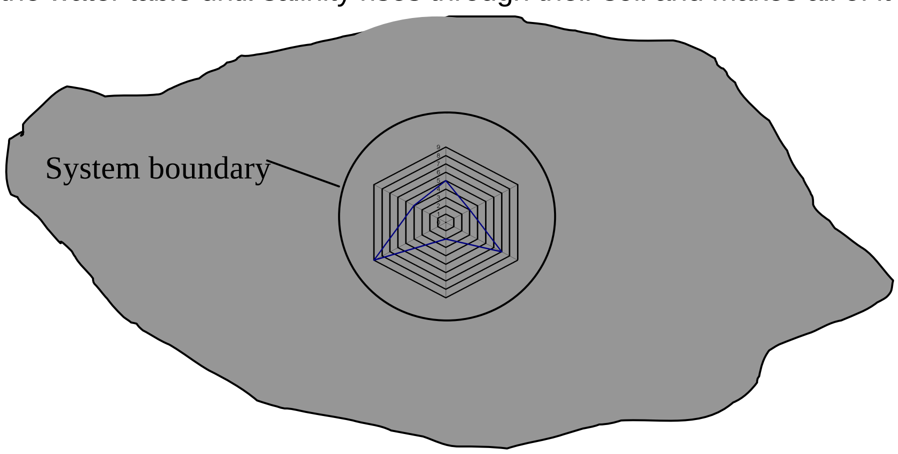

Figure 3 Pervasive influences

It becomes more difficult when there are multiple influences of different strengths. Take, for instance, Figure 4, a light sensitive system with influencing light sources at varying distances. The power of the light falls off as an inverse square of the distance so is this the system boundary,

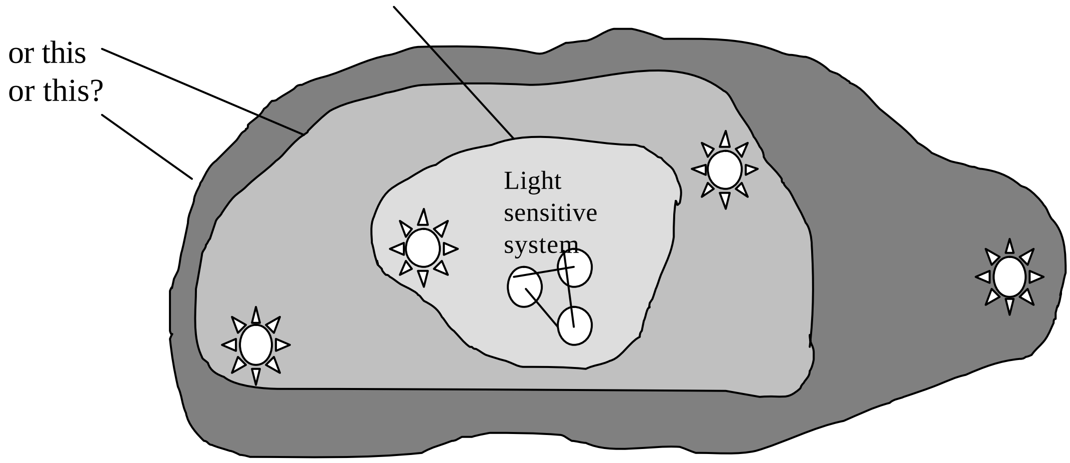

Figure 4 Multiple influences of different strengths

The position of the boundary depends largely on the purpose of the study but drawing that outer boundary is a critical process. In addition to the outer boundary there is an inner boundary, namely, the level of detail represented by the sub-systems. As indicated in the section below the smallest entity in a system depends on the purpose of the model, it could be a country, a tribe, a family or a person, and the attributes of the entity will probably be a subset of all possible attributes. Characteristics to be taken into consideration when defining boundaries include, but are not limited to these layered and hierarchic aspects, system organisation or network structure and temporal features.

6. LAYERED AND HIERARCHIC ASPECTS

When we examine co-evolutionary systems such as ecologies and business organisations, we notice that they divide into layers. Simon described this as a hierarchy saying:

> "*For lack of a better term I shall use " hierarchy " … to refer to all complex systems analyzable into successive sets of subsystems …" (Simon 1969)*

In my view, however, many complex systems are not hierarchical but are networked and they are all layered.

In a building, there is the structure, i.e. foundations plus walls plus roof, the services, space-dividing walls and divisions, furniture and people. In an ecology, there are forests and deserts, trees, shrubs, mammals, birds, annual flowering plants, beetles and butterflies. In each of these layers, we can also notice that the local ecosphere is on a different scale to those above or below it. The forest and the desert respond to long time-scale climatic changes but their response to the death of a flower or butterfly is imperceptible. The death of a tree has a local effect but the effect on the forest is small. In addition to scale, these layers also differ in their rate of change, in general the larger the scale the slower the rate of change. Thus, in a system model, scale, layering, and rate of change become interchangeable concepts (Anderson 1972; Holland 1998). Systems are, probably, best visualised as a single layer with the behaviour of the lower layers apparent only as they affect the visualised layer. Interaction between layers is often small until a critical state is reached then a small change on one layer can cause a change on an upper layer, for example we can push more and more desks into an office but there comes a time when to get more into the room, space-dividing walls must be moved. This may be the precursor to a cascade of changes and can be likened to a phase change where dropping a seed into very cold water may cause it all to change to ice.

In an information system there is a similar layering, the many changes to documents made by users come quickly but changes to applications, changes of applications, changes of operating systems and changes of hardware each happen more slowly and with increasing effect on the total system. Others have found corollaries for layers in economics (Devezas and Corredine 2001) and business (Eisenhardt 1989; Eisenhardt and Galunic 2000).

7. ORGANISATION OR NETWORK STRUCTURE

A network of influences between the subsystems, within and between layers, determines the organisation of a complex system. These networks can take several forms; but they fall into three major classes – random, regular or "small world" networks. Change in the organisation of a complex system can create, destroy, strengthen or weaken these influence links.

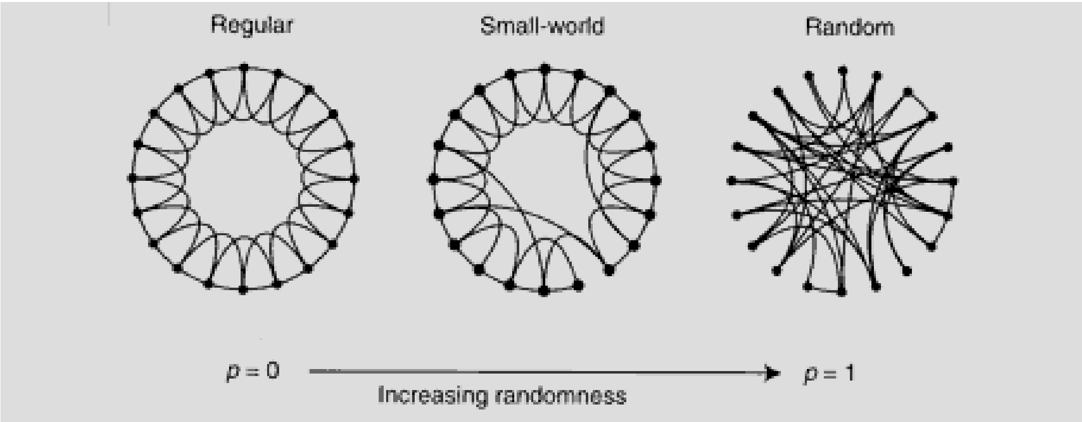

Figure 5 Diagram from Watts and Strogatz (1998)

- Regular - A regular network consists of a symmetric arrangement of a number of components. In Figure 5 the regular network consists of a circle of nodes and 20 edges connecting adjacent and alternate nodes. This ring lattice was used by Watts and Strogatz to demonstrate the characteristics of networks with differing forms of connection between nodes.

- Random – A random network consists of a set of nodes connected at random. More formally, a random network is formed when a lattice is wired such that an edge between two nodes is formed with a probability *p*, duplicate edges and self-connections being excluded. By varying the probability, the transition from regular to random networks can be studied. (Albert and Barabasi 2001) Erdos and Renyi coined the concept of random networks in a paper published in 1959 (Erdos and Renyi 1959) and since that time many papers have been published on random graphs. Unfortunately the random graph as defined by Erdos does not relate well to real world networks such as social networks, the World Wide Web, electrical power systems, etc. (Newman, Watts et al. 2001). This discrepancy led Watts to study the characteristics of small world networks.

- Small world networks. A small world network is part way between a regular network and a random network and includes a high degree of local edges (or local order) between small (when compared to an otherwise sparse network) groups of nodes with a few edges to other similar groups — i.e. clustering is high. (Watts and Strogatz 1998; Watts 1999) (Albert and Barabasi 2001)

In a complex, co-evolutionary system the influence connections, form the network. When a change occurs in a single subsystem, or node, it is because the pressure for change from the influencing subsystems exceeds the resistance to change. The changed subsystem then exerts its own changed influence on other subsystems that may not have been influenced by the original pressure for change. Thus the change is propagated from the first set of subsystems, to the single subsystem and then on to a second set. The length of the path between any two subsystems across this network is defined as the number of subsystems traversed to go from one to the other. Thus, the path length determines how quickly a change in one subsystem is propagated to another across the network.

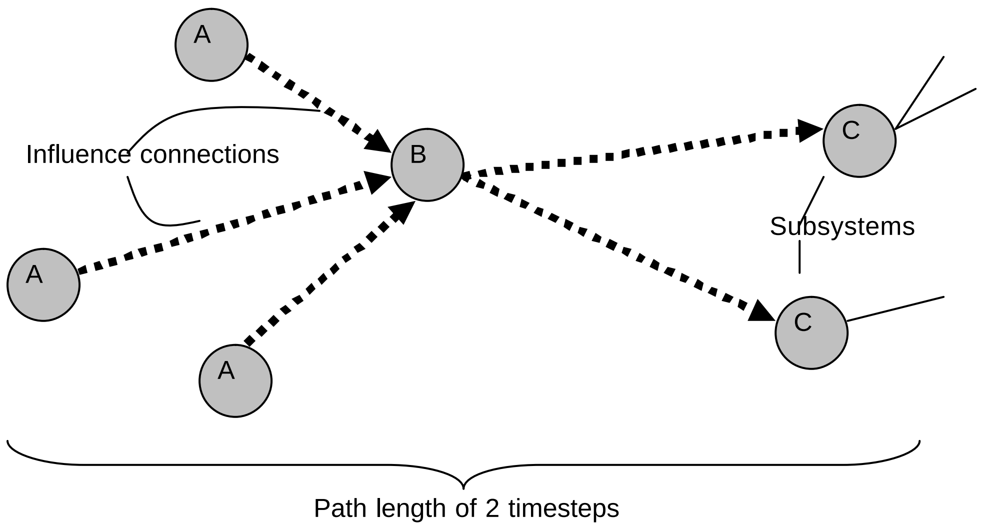

Figure 6 Propagation path lengths

In the case of regular networks, the average path lengths are long, because there is no direct route from one side of the network to the other. So propagation is slow. With random networks, the average path length scales logarithmically with the number of nodes and can be calculated as the average number of neighbours a node has, hence networks with many nodes have increasingly long average paths and propagation is again slow. This characteristic of long average path lengths departs from that shown by naturally occurring complex networks. Small world networks, in contrast, have low average path lengths due to the high degree of clustering. This clustering permits “short cuts” across the network and so propagation of change across the network is fast.

In a previous paper (Mansfield 2005) I showed that systems with a small world structure could change their emergent behaviour more rapidly than systems with random or regular structures.

8. TEMPORAL ASPECTS

In socio-technical systems, the environment of any subsystem is the effect on that subsystem of all the other subsystems. In this type of system there is a lag between the environment changing and the various subsystems perceiving that the environment has changed, **dp,** and between the perception of an environmental change and the reaction to that change, **dr**. In addition, these lags may vary across the subsystems, with one reacting faster than another, $\mathbf{dp_1}$ **?** $\mathbf{dp_2}$ and $\mathbf{dr_1}$ **?** $\mathbf{dr_2}$. Thus if we have a system of many subsystems each with a different perception lag and reaction lag and we make one change to one subsystem then a short time later some of the subsystems will have perceived the change and a fraction of these may even have reacted to the change. As soon as these subsystems react, they change the environment for the rest of the system. Meanwhile, some of the slower subsystems are now reacting to the earlier change and are affecting the environment. Because of the lags, the subsystems react asynchronously and making one small change results in a system that is out of equilibrium. It is changing over time and interaction is taking place between the subsystems, in short a typical complex system (Di Paolo 2001). Once started it cannot be predicted by *analytical* means when, if ever, the system will return to equilibrium.

The rate of change can vary widely but in general, rapid rates of change incur small individual costs. Changes with large costs tend to happen less frequently. Research has shown that for systems as unlike as avalanches in laboratory sand piles (Bak 1997) earthquakes (Bak and Tang 1989) and mortality in wars (Buchanan 2000), graphs of the size of the event against the frequency of the event all show a power law.

9. INSIGHTS FROM BIOLOGY

Stuart Kauffman's interest in biological co-evolution lead him to devise a Boolean network view of this concept, described in *Origins of Order* (Kauffman 1993), which he called NK networks, after the (N)odes and (K)connections in the network. He took his idea further in *At Home in the Universe: The Search for Laws of Self-Organisation and Complexity* and grouped Nodes (individual subsystems) into "*patches*" (larger subsystems) and investigated the interaction of the patches. His algorithm was: divide a system into a "quilt" of non-overlapping patches or groups and try to optimise within each patch. As this optimisation occurs, the couplings between parts in two patches across patch boundaries will mean that finding a 'good' solution in one patch will change the problem to be solved by the parts in the adjacent patches. Since changes in each patch will alter the problems confronted by the neighbouring patches the system is co-evolutionary, the co-evolutionary moves by the neighbouring patches in turn will alter the problem faced by yet other patches, the system is reacting in a similar manner to the co-evolving biological systems of interest to Kauffman.

To imagine this idea in a socio-technical context consider, for example, a multi-group software development organisation - a complex system in which many parts interact. When any software development group optimises itself, by changing a methodology, an interface, or by introducing a new functionality, this causes the environment of all the other groups to change (Tushman and Rosenkopf 1992; Blackler 1994; Hunt and Aldrich 1998). This forces the other groups to re-optimise themselves, and so on. This process of co-evolution drives the behaviour of the system (Axelrod and Cohen 1999). Again, it must be stated there is no '**big picture design**' here; the behaviour of the overall system emerges from the behaviour of individual sub-systems and their interaction. Kelly quotes Alfred Lotka, an early theoretical biologist, who wrote in 1925,

> "*It is not so much the organism or the species that evolves, but the entire system, species plus environment. The two are inseparable.*" (Kelly 1994).

10. DISCUSSION

This paper submits that socio-technical systems are inherently complex, co-evolutionary systems and once such systems reach a particular level of complexity, their behaviour becomes as unpredictable as that of eddies in a stream. Large, information technology systems are built by engineers; systems engineers, software engineers, telecommunication engineers, mechanical engineers, etc. Engineering is goal-directed and based on the laws of physics. There are targets to achieve and requirements to be met. Thus engineering, and systems engineering particularly, is concerned with plans for the future. These plans are based on predicting the behaviour, of the system of interest at specified times in the future. When a decision is made to change a system it may be formalised by using an engineering methodology, but in a truly complex, large socio-technical system, this planned predictability can break down into a morass of unanticipated behaviour.

This unpredictability comes from the propagation of change through co-evolving subsystems. The effect of a change in one subsystem is often deterministic and in general predictable but when influencing links connect many subsystems, the effects become complex.

When changes propagate across a system they move the subsystems in design space and the movement feeds back into the succeeding changes. When a single change is made to one subsystem in a complex system the effects of that change propagate across the system over time as all of the subsystems co-evolve. This situation continues until the system reaches one of three states, namely equilibrium, cycling on an attractor or chaos (Mansfield 2005). If further changes are introduced into the system before it has reached equilibrium, these changes also propagate through the system via co-evolution.

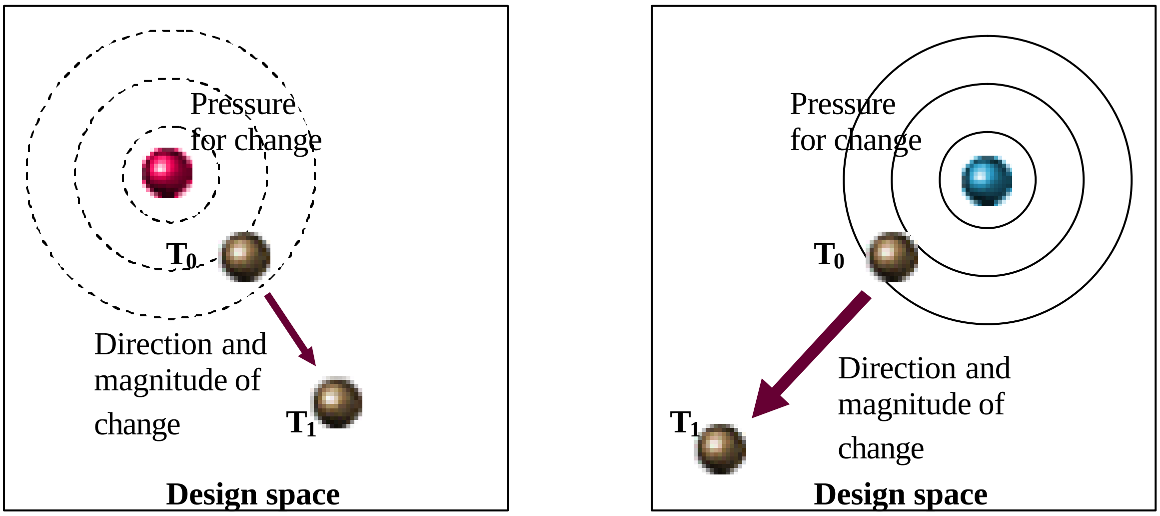

Figure 7 The effect of independent pressures for change

Where a subsystem comes under the influence of other subsystems, which are influenced separately by different sets of propagating changes, the aggregated pressures are clearly different to those pressures that would have occurred had the influences not been in place. The initial subsystem undergoes a different co-evolutionary change to that which it would have, had it been influenced by a different set of pressures. In short, the propagating changes originating from two or more initial changes interfere and alter the outcome.

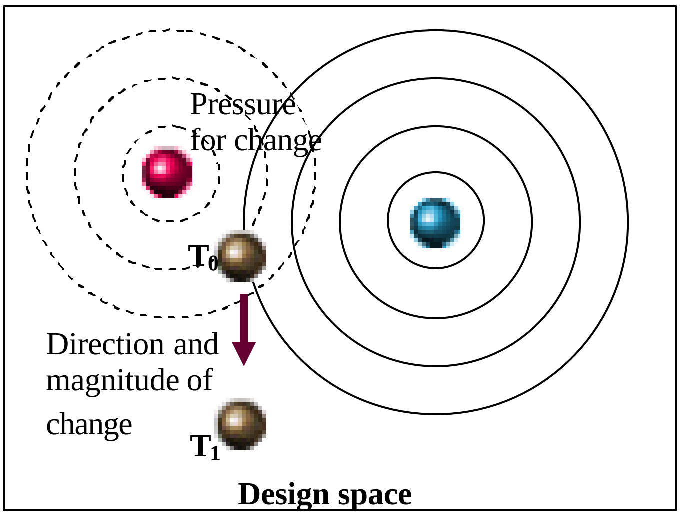

Figure 8 The combined effect of two propagating changes

When two subsystems change they may each enter a state that conflicts with the other's state. These new states apply conflicting pressures for change on each other and on other subsystems inducing them to enter a cyclic or chaotic set of states. These changes to individual subsystems or small groups of subsystems are local in nature but their aggregate effect is evident in the emergent behaviour of the system. The holistic system behaviour is unpredictable because it is dependent on a multitude of local effects.

At the local level the pressure one subsystem has on another is dependent on the degree of coupling between the subsystems. Within a subsystem, there is again conflict, the magnitude of alteration in one dimension being traded off against that of another so that the aggregate effect reduces the pressure for change. The biggest problem in complex system engineering arises when predicting the effects of the introduction of a change into a system, for example, adding a new component. Often the change is deliberate to achieve a predicted effect and the implementer is bewildered when the change achieves a different effect. Probably the most well known example is making a change to one line in a program. In the context of the current module, the change is syntactically and semantically correct but the change causes the system to crash because of a subtle interaction with another module. Is that poor design or poor implementation?

11. CONCLUSION

There would seem to be three ways of determining the effect of all the changes in the time between the current time and a specified time in the future:

(a) Wait until "it happens", i.e. experience all the effects of the changes in the system until the specified time becomes the current time. This has major disadvantages! Particularly when there is a high cost if the system does not behave as anticipated.

(b) Paper and computer-aided analysis. This can take considerable time and can delay the decision to undertake the change beyond that allowed by operational constraints and hence is often ignored in the decision making process.

(c) Computer simulation. This is the fastest and most capable of aggregating the effects when dealing with many thousands of subsystems. Its speed provides a "what if" facility to the decision makers (Mansfield 2005).

In options (b) and (c) possibly the greatest source of error is in the unknowable aspects of initial conditions; closely followed by the difficulty of determining the influence of the environment on a subsystem and the difficulty of determining the trade-offs and thresholds of change within a subsystem.

System engineers need to recognise that the act of changing a complex system has a high probability of causing unanticipated system behaviour and that extensive simulation or other forms of impact analysis will reduce but not eliminate the uncertainty. We must therefore abandon the belief that the exact behaviour of a complex system can be forecast, and hence, we must also abandon the belief that the complete design of a complex system can be specified in advance.

Complex systems evolve; consequently complex systems are grown, not engineered. As noted previously in (Mansfield 2004) when implementing a complex system the systems engineer can only steer its growth in the direction of the currently desired end-point. This paper has indicated that the effects of changes propagating within a complex, socio-technical system are inherently unpredictable and that there is a good probability of the system failing to meet expectations. However, simulations may be a mechanism for improving the success rate and turning the nature of change to advantage.

This field would benefit from more research, concentrating on the interference effects of a change propagating well into the future.

12. ACKNOWLEDGEMENTS

I would like to thank the reviewers for comments and suggestions that improved the paper.

13. REFERENCES

Albert, R. and A.-L. Barabasi (2001) *Statistical Mechanics of Complex networks.*

Anderson, P. W. (1972). "*More is Different*." Science **177**(4047): 393-6.

Arisholm, E. (2001). *Empirical Assessmentof Changeability in Object-Oriented Software, PhD Thesis*. PhD, University of Oslo, Oslo

Arisholm, E. and D. Sjoberg (2000). "*Towards a framework of changeability decay*." The Journal of Systems and Software: 3-14.

Axelrod, R. and M. D. Cohen (1999). *Harnessing Complexity: Organizational Implications of a Scientific Frontier*. New York, Free Press.

Bak, P. (1997). *How Nature Works*. Oxford, Oxford University Press.

Bak, P. and C. Tang (1989). "*Earthquakes as a self-organized critical phenomenon*." J. Geophys. Res. B94: 15635.

Blackler, F. (1994). "*Post(-)Modern Organizations - Understanding How CSCW Affects Organizations.*" Journal of Information Technology **9**(2): 129-36.

Boehm, B. W. (1998). "*A spiral model of software development and enhancement*." IEEE Computer **21**: 61-72.

Brooks, F. P. (1975). *The Mythical Man-Month*. Reading, Mass., Addison-Wesley.

Brooks, F. P. (1995). *The Mythical man-Month Essays on Software Engineering Anniversary Edition p117*. Reading, Massachusetts, Addison-Wesley.

Buchanan, M. (2000). *Ubiquity. The science of history ... or why the world is simpler than we think*. London, Weidenfeld and Nicolson.

Devezas, T. C. and J. T. Corredine (2001). "*The biological determinants of long-wave behaviour in socioeconomic growth and development*." Technological Forecasting and Social Change **68**: 1-57.

Di Paolo, E. A. (2001). "*Rhythmic and non-rhythmic attractors in asynchronous random Boolean networks.*" Biosystems **59**(3): 185-95.

Eisenhardt, K. (1989). "*Making Fast Decisions in High-Velocity Environments*." Academy of Management Journal **32**(3): 543-76.

Eisenhardt, K. and D. C. Galunic (2000). "*Co-evolving: At Last a Way to Make Synergies Work*." Harvard Business Review: 91-101.

Erdos, P. and A. Renyi (1959). "*On random graphs*." Publicationes Mathematicae **6**: 290-297.

Gilb, T. (1998). *Principles of Software Engineering Management*. Reading, MA, Addison-Wesley.

Holland, J. H. (1998). *Emergence: From Chaos to Order*. Oxford, Oxford University Press.

Hunt, C. S. and H. E. Aldrich (1998). "*The second ecology: Creation and evolution of organizational communities.*" Research in Organizational Behavior **20**: 267-301.

Kauffman, S. (1993). *The Origins of Order*. New York, Oxford University Press.

Kauffman, S. (1995). *At Home in the Universe: The Search for Laws of Self-Organisation and Complexity*. Oxford and New York, Oxford University Press.

Kelly, K. (1994). *Out of Control: The New Biology of Machines, Social Systems and the Economic World*. Reading, Mass., Addison-Wesley.

Mansfield, J. (2004). *Interference in propagating change and its effect on system failure*. Systems Engineering and Test Evaluation Conference 2004 (SETE 2004), Adelaide, SESA.

Mansfield, J. (2005). *The nature of change in complex socio-technical systems*. Submitted PhD Thesis, University of Queensland, Brisbane

Newman, M. E. J., D. J. Watts, et al. (2001) *Random graph models of social networks.*

Simon, H. A. (1969). The Architecture of Complexity. *The Sciences of the Artificial,*. Cambridge, Mass., MIT Press**:** 192-229.

Tushman, M. L. and L. Rosenkopf (1992). "*Organizational Determinants of Technological Change - toward a Sociology of Technological Evolution.*" Research in Organizational Behavior **14**: 311-47.

Watts, D. J. (1999). *Small Worlds: The Dynamics of Networks between Order and Randomness*. Princeton, New Jersey, Princeton University Press.

Watts, D. J. and S. H. Strogatz (1998). "*Collective dynamics of 'small-world' networks*." Nature **393**: 440-442.

Zamperoni, A., B. Gerritsen, et al. 1995 *Evolutionary Software Development: an experience report on technical and strategic requirements. Technical Report TR-95-25* Evolutionary Software Development: an experience report on technical and strategic requirements. Technical Report TR-95-25, Leiden University, Leiden, The Netherlands