Do we ever halt when solving complex problems?

Looking beyond the interface horizon

Key Theme

SYSTEM DYNAMICS CONTRIBUTIONS TO THEORY BUILDING IN THE SOCIAL AND NATURAL SCIENCES

Florian Prange

ProjectComplexity

Tresckowstraße 52

20259 Hamburg

florian.prange@projectcomplexity.de

Paper presented at the 24th International Conference of The System Dynamics Society, July 23 – 27, Nijmegen

Do we ever halt when solving complex problems?
Looking beyond the interface horizon

ABSTRACT

Why is it that some problem solving tasks in organisations – though well posed in principle – turn out to be incredibly difficult or impossible to solve when taken on in practice? Why is it that after having followed an otherwise ordered and predictable path we often find ourselves suddenly on an increasingly turbulent stretch of road where we realise – to our horror – that our ability to intervene in the unfolding chaos is rather limited? Yes, complexity theory provides many important insights into the dynamics of complex organisational systems and – over the years – we have become familiar with terms like bifurcation points, strange attractors and phase transitions. However, given a concrete organisational or engineering problem, their use remains largely metaphorical. In fact, the complex dynamics is assumed to be given and no account is offered about its actual emergence. This paper, therefore, aims to serve as a kind of magnifying glass that helps us to study the emergence of complex behaviour in organisations. Also, it gives an account why complexity often out-steps us in many problem solving tasks.

Keywords: Systems, interaction surfaces, combinatorial objects, halting

INTRODUCTION

This paper is motivated by the following simple question: why is it that some systems (like organisations or engineering projects) interact in specific yet unpredictable ways while others seem to function remarkably well? Also, why is it that often our analytical ability seems to be out-paced by the system's dynamics as it evolves in time?
Those are two relevant questions as we constantly interact with a multitude of systems in our every day and professional lives. In fact, the way we deal with them and our decisions to influence their behaviour has often serious consequences which we did not anticipate when we first started acting as part of the system. Now, of course the discussion of organisations and management in the light of complexity and chaos theory has called into doubt the linear relationship between a body of rules and their concrete manifestation in a dynamical system (Levy 1994, Phelan 1995, Murphy 1996, Bardyn et al. 1999, Wheatly 1999).

While this approach has provided a fresh and important perspective to look at the management of systems, it often uses a kind of argument by analogy, namely that '[c]haos is the science of complex, dynamical, non-linear co-creative, far-from-equilibrium systems' and since 'organisations are in their very essence complex, dynamical, non-linear, co-creative, and far-from-equilibrium systems, chaos is the science of organisation' (Fitzgerald 2002: 339). However, judging from my own experience in industry, while many people are generally quite sympathetic to this idea, they attribute the emergence of complex behaviour to the human component, interventions from outside the system, or to noise perturbations due to friction. Still they adhere to the belief that an essentially discrete and finite number of rules exist that, when properly executed and applied to a well-defined problem solving task, in principle, would yield the desired outcome (if they would be just left alone!). This is borne witness by the number of professional bodies that uphold best practice and standard processes in their respective fields (e.g. PMI – the Project Managers Institute, IPMA – the International Project Management Association, INCOSE – the International Council on Systems Engineering and many others). Also, especially, in the engineering community, this widely-held belief is supported by the many challenging tasks that could be solved remarkably well in spite of being non-linear in nature.

This paper, therefore, aims outline how organisational or engineering systems in themselves outdo our ability to capture their dynamics regardless of the body of rules we use to describe them. Also, we contrast them with systems that seem to be immune to those 'holes' in our understanding of the system. To see this, let us have a look at the preliminary definition of a system.

STARTING FROM FIRST PRINCIPLES – what is a system?

Put simply, a system is a set of elements together with a set of relations that hold between those elements. We might further want to add that a system has to be understood as a dynamic and complex whole that interacts as a structured functional unit. It is thus characterised by the information, matter or energy flows between the elements themselves and the outside through the system's boundary. To fully understand a system, therefore, we have to study the interactions and processes between the elements that comprise the whole and the 'conceptual boundary that separates one part of reality [our functional unit] from the rest of the world' (Shaw 1981, p.82). Figure 1 summarises the interplay between the various conceptual building blocks of a system.

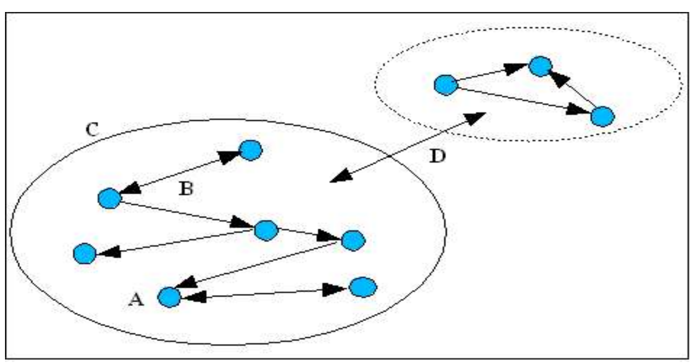

Fig 1: The elements (A) interact through a multitude of relations (B) inside the system's boundary (C). A system usually communicates somehow with the 'outside' (D) thereby making it often difficult to establish its exact operational confines.

Now, given a well posed engineering or organisational problem we are often in a position where we deal with systems whose components are essentially well defined and where the relationship between the components are subject to a familiar body of rules. Using an electrical circuit, as an example, we happen to know the relevant 'ingredients' like resistors, wires and light bulbs etc. as well as the necessary construction principles such as Ohm's law to go about our task. Similarly, using a highly dynamical system this time, a surgeon performs an operation on an inflamed appendix using both his knowledge about the human organs and his surgical tools together with a set of medical procedures.

Thus, to be able to perform a task in a given system requires the knowledge of how the elements interact to form a functional whole that somehow differentiates it from its environment. In other words, how can the elements be put together to form units of higher complexity?

INTERACTION SURFACES – applying the glue

When building systems it is evident that we do not rely on statistical events to create a functional whole. On the contrary, the formation of a system follows what Cervén (1985) has termed 'strong cognitive' processes between a finite and discrete number of elements comprising different properties. What does the somewhat puzzling term 'strong cognitive interaction' mean? To understand how individual building blocks combine and recombine (being a nested and recursive process) to function as a unit of a higher hierarchical level, we have to understand them as being characterised by two things: their properties and their *interaction surface* (ibd.). The interaction surface describes the element's specific ability to form combinatorial bonds. So, interaction between elements can occur, if their mutual interaction surfaces are complementary as visualised in Fig 2. Of course, an element may have several properties or interaction surfaces. Now, the term cognitive is used because the interaction implies a specific *recognition* between elements. This can also be understood as some kind of information exchange between the elements.

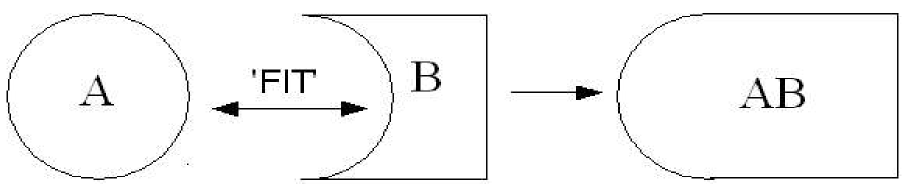

Fig. 2: Two elements with properties 'A' and 'B' are characterised by complementary interaction surfaces that 'recognise' or 'communicate' their fit to form a new unit 'AB'. As such the resulting element may well exhibit emergent properties.

While this recognition of mutual interaction surfaces in living organisms, for example, is carried by biological processes such as symbiosis, often our organisational systems have got that information on the interaction surfaces already built into their very design. For example, it is because we know that the software modules A and B interact nicely that we want them to form a sub-system of our programme; or that it is precisely that I know to interact well with Bob that I want him to be part of my project team. Being in possession of the interaction surfaces of the elements that comprise a given system, therefore, is part of any design process and often it provides us with important conceptual shortcuts.

As the interaction of the system is recursive and as we are interested in its actual evolution, of course, the history of the system becomes important. As time evolves, elements are recombined and new elements of higher hierarchies are formed, which gradually lead to an increased interaction within the system. Now, judging from the way the elements recombine in time, we might identify two types of systems: (i) closed structures are composed of identical or semi-identical elements with fixed coordinates and (ii) open structures that either create new elements of higher order or open the system to the outside (or a combination of the two). Let us now look at what this tells us from an organisational or engineering perspective.

ON CRYSTALS AND ENGINES– rejoicing in predictability

Let us start with those systems that utilise periodic construction rules to combine and recombine families of identical parts where the interaction is characterised by surfaces with fixed or static coordinates. Crystals are an archetypical example to illustrate this type of behaviour as they start from a unit cell and are then packed in a regularly ordered, repeating pattern extending in all three spatial dimensions. Likewise, many of those systems that are the subject of our professional lives as project managers, engineers or administrators are characterised by similar patterns aptly embodied by mechanical artifacts or an efficient and predictable public administration (see Fig. 3). Notice that systems of this kind may well exhibit non-linear behaviour, critical thresholds or turbulence. However, our knowledge of their actual behaviour helps us to devise adequate strategies to cope with this, as demonstrated, e.g., by the effectiveness of a public institution remaining unaffected by the changes or turmoil in the party political system of a country.

Fig. 3: In (a) the unit cell (in this case a cube as the most symmetrical structure of a crystal) recombines with identical structures via any of the 6 surfaces. In case of an engine (b) we add versatility to the system by having families of identical elements with different properties that may also exhibit a wider variety of interaction surfaces.

Now, what does this tell us about our interaction with a system of this kind? We can, in principle, devise a mechanism or set of procedures (technically an algorithm) that identifies and checks the number of elements and corresponding interaction surfaces to then observe the system's evolution in time as interactions occur. Simulations or automatic test routines in the automotive or aviation industry are a good example: Here we use the system elements in conjunction with a description on how they interact (usually expressed in terms of differential equations) to see how the system projects in time. Using the crystal in (3a), for example, such a mechanism would have to identify the interaction surfaces of the unit cell to combine them with additional cubic structures.

As the system starts interacting, at any given time during the system evolution, say n iterations, the number of 'checks' to be performed would be given by

$$max\ number\ of\ interaction\ surfaces = 5n+1 \ .$$

This is quite manageable, as we would expect, both for us humans as problem solvers and for computers as our preferred number crunchers. In terms of algorithmic complexity such a task is manageable in polynomial time as the size of the problem (such as the elements or variables that make up the system) increases proportionally to a multiple of a polynomial. Consequently, in learning theory any procedure to master a new concept or function would have to adhere to such an upper bound in order for us to 'grasp' it in a realistic time frame (Anthony et al. 1992). Also, systems like (3b) behave in a similar manner even if they are more versatile in that they comprise elements with different properties and a larger number of possible transaction surfaces. Still, the number of interaction surfaces is limited by their somewhat static nature or fixed coordinates. I simply cannot add elements randomly and still expect the whole thing to work. Thus, the coordinate structure being fixed, serves as a kind of filter that reduces the number from what is combinatorially feasible to something which can be 'realistically managed' which is essentially what we capture by the polynomial as our upper bound.

So, in dealing with systems of this kind we can keep up with them as they evolve by having an explicit or conceptual mechanism in place that tracks and checks the elements as they recombine in time. Incidentally, the fact that there is a substantial number of systems that behave broadly in this fashion often (mis)leads us to make the assumption that, if adding a known element to a functioning system yields a fault, this must be due to the added element. Knowing that this, alas, is not the case brings us to the next section

OPENING PANDORA's BOX – enumerating its content

As we have seen, there is a large number of important systems whose behaviour satisfies some kind of closure criterion. Parts are combined and our knowledge of the way the elements interact helps us understand or devise the resulting whole as a stable structure. However, elements may combine such that they form new interaction surfaces that open the system to the outside in ways not anticipated. This is due to either the lack of a fixed coordinate structure (as can be found in tumor cells that interact with other cells indiscriminate of their morphology) or because the system boundary itself is susceptible to change. As such, the recursive process of combining and recombining may not settle in a stable state, i.e. being combinatorially closed. In fact, given a system with n elements and, say, families $k_1, k_2, \ldots, k_m$ of identical parts we would end up with an upper bound of interaction surfaces given by

$$max\ number\ of\ interfaces = \frac{N!}{K_1!, K_2!, \ldots, K_k!} \ .$$ [1]

[1] The first element can interact with any of the $n-1$ other elements which in turn may combine with any of the remaining $n-2$ elements thus giving $n \times n-1 \times \ldots \times 1 = n!$. Also, combinations of identical parts are

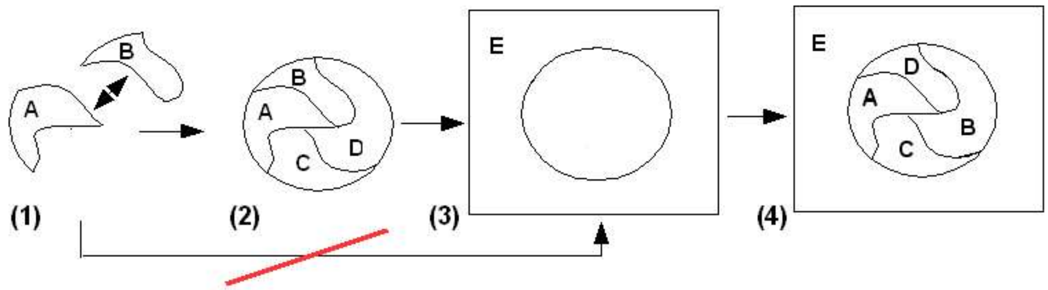

Fig 4: Elements A, B, C, D may combine to create a new interaction surface that opens the system to the outside. While any of the singular elements (1) still remain closed, they may form new links with elements (3) initially outside the system's boundary to form new structures (4).

Clearly this behaviour represents a class of problem solving tasks that, in some way, requires considerably more resources to follow the system's evolution in time. What does this mean for our ability to devise or administer such a system? Unlike the previous case where we ended up with a manageable bound, this behaviour dramatically outpaces anything polynomial given a large enough class of different elements. Certainly, presented with one possible instance of the system, we can check whether it is well-behaved or constitutes a solution to our problem solving task, for example, by inspecting the interaction surfaces. However, as a whole, any inspection mechanism would be out-manoeuvred in the sense that it can not be mastered in a realistic time frame. What does this mean, in principle, for our ability to attain a solution to a given organisational or engineering task?

BEING OUTSIDE OF THE INSIDE – a halting problem for organisations

Let us depict a problem solving task as a sequence of steps to modify a system in such a way that it moves from an initial state to a more desired state. Of course, we do that through interacting with the system's elements. Attaining a solution, therefore, would be equivalent for us to halt in that activity. Note that this way of looking at problem solving is consistent with the engineering paradigm where we construct devices as a sequence of discrete steps using our body of engineering knowledge. Similarly, it applies to an administrative challenge to reduce the healthcare expenditure below a certain threshold through a political process. So, given a description of a finite number of discrete processes, properly executed, and the initial state of the relevant system, can we ever determine whether we halt? The alternative, of course, would be to run forever or to be stopped thanks to an intervention from the *outside* (such as cutting the losses in what turnes out to be an ill-fated project). In other words, is the information available to us *inside* the system sufficient to determine the outcome?

The answer is no. This is a direct consequence of the undecidability results for decision problems in the study of formal systems. Essentially, they ascertain that attributes such as solvability, provability or decidability are attributes that lie outside of the system and the corresponding rule set. In fact, all variations of these results have in common that they speak about the system's ability to speak about itself. This is basically what we do

treated as one so that we have to make sure that the $k_i!$ terms do not contribute to the total number of permutations.

when we ask ourselves whether we halt in whatever we do. Such a statement, in its most abstract sense, is of the form “This problem is not solvable in the system”. The various proofs (for a comprehensive treatment see Bools et al. 1996, e.g.) then establish that if this statement were in fact solvable, it would be solvable in the system, and consequently violate the consistency requirement of our system. Now, if the statement “The problem is not solvable in the system” is in fact unsolvable in the system then, in effect, the statement would be true in the system. The problem with that, however, is that we can not express this concept inside the system itself. Another, more concrete instance of undecidability would be the question whether any given computer programme stops (similar to our remarks on problem solving). Starting with the assumption that such a programme exists, leads us directly to a contradiction and we must conclude that a general programme to solve the halting problem cannot exist. Again, the attribute 'does halt'/'does not halt' is only expressible *outside* the system whereas our rule set lies *inside* it. Would a human be able to determine whether a programme halts? Let's take a simple programme that writes any given natural number as the sum of two prime numbers such has $12=5+7$. Will it ever halt? Well, we don't know as the corresponding formulation in mathematics (Goldbach's conjecture) still defies all attempts to prove it. So, it makes it difficult to see how humans could solve the halting problem as we too would have to resort to concepts that lie *outside* of the system.

The implication of the halting problem, of course, is that we can not answer the question whether we will realistically (and principally) ever attain the solution to a given problem solving task. Put differently, therefore, we have to get used to the fact that often the easiest description for solving problems in complex systems is the actual process of solving it. Thus solving the problem is the only way to determine whether we can solve it. Reformulating this result for standard practices (as propagated by many fields in organisational theory and practice as mentioned above) would read something like this: solvability of a given organisational or engineering task is an attribute *outside* any of the processes that are guided by a specific rule set. Thus, the systems which we deal with as administrators, project leaders or managers can only be understood in terms of their specific context and the local conditions under which they evolve.

REFERENCES

Anthony, M. and Biggs, N. (1992) *Computational Learning Theory: An Introduction.* Cambridge, UK: Cambridge University Press.

Bardyn, J. and Fitzgerald, D. (1999) “Chaos Theory and Project Management: A Perspective on Managing the Complex Project.” In: Lissack, M.R. & Gunz, H.P. (eds.) *Managing Complexity in Organisations*. Westport: Quorum Books: p. 243-258.

Bools, G. S. and Jeffrey R. C. (1996) *Computability and Logic*. Cambridge, UK: Cambridge University Press.

Cervén, E. (1985) “On the stability of cognitive processes” *Experientia* Vol. 41, No. 6, 713-719.

Fitzgerald, L. A. (2002) “Chaos: The Lens that Transcends.” *Journal of Organisational Change Management*. Volume15, No. 4, 339-358.

Levy, D. (1994) “Chaos Theory and Strategy: Theory, Application and Managerial Implications.” *Strategic Management Journal* Vol. 15, 167-178.

Murphy, P. (1996) “Chaos Theory as a Model for Managing Issues and Crises.” *Public Relations Review* Vol. 22, No. 2, 95-113.

Phelan, S.E. (1995) “From Chaos to Complexity in Strategic Planning.” *55th Annual Meeting of the Academy of Management*, August 6-9, 1995. Vancouver, British Columbia, Canada.

Shaw, R. (1981) “Strange Attractors, Chaotic Behaviour, and Information Flow.”*Journal of Physical Sciences* Vol. 36, No. 1, 80-112.

Wheatley, M. J. (1999) *Leadership and the new Science*. San Francisco, USA: Berrett-Koehler Publishers.