Critical Critical Systems

Susan Stepney

Department of Computer Science, University of York, Heslington, York, YO10 5DD, UK.
susan@cs.york.ac.uk

Abstract. I discuss the view of communication networks as self-organised critical systems, the mathematical models that may be needed to describe the emergent properties of such networks, and how certain security hygiene schemes may push a network into a super-critical state, potentially leading to large scale security disasters.

1 Introduction

The word “critical” is used in two very different technical senses, both of which are appropriate for considering the security of communication networks, such as the Internet.

The use probably most familiar to delegates at this conference is that of “being indispensable or vital”, of being a “high consequence” system. So we have safety- and security-critical systems, where safety and security are indispensable (or even literally vital) issues of concern to the users of the systems.

The second use is that of “being at a turning point, or a sudden change”. I explore this second sense further here (within the context of the first sense). In particular, I discuss the notion of “super-critical” systems, systems in a state where the potential for the “sudden change” is magnified, and discuss whether certain security defences may be increasing the probability of such states.

2 Critical Systems

2.1 Controlled Critical Systems

Critical systems can exist in two (or more) phases, dependent on some controlling parameter. These phases are separated by a complex boundary state, “the edge of chaos”, where the controlling parameter has a critical value.

The prototypical example is a physical phase change where one phase is an ordered “frozen” solid or liquid, the other is a random “chaotic” liquid or gas (the word *gas* has the same root as the word *chaos*), and the controlling parameter is temperature. In other cases there are analogues of these states. Systems in the boundary state are in some sense “fluid”, and often have particularly interesting behaviour: neither too frozen nor too chaotic. These systems are in a state of constant churning flux and change: they are *far from equilibrium*. The study of physical systems has tended to focus on equilibrium states, however, as these are much easier to model.

Another physical example is the transition from an unmagnetised to a magnetised state through the critical temperature. Road traffic flow exhibits an ordered state of free laminar flow, a chaotic state of gridlock, and a transition state of propagating jams on all scales, controlled by the traffic flow rate parameter [Sole]. Pushing the analogy further, computation may be viewed as the interesting and complex “edge of chaos” between pure frozen memory and pure chaotic process.

2.2 Self Organising Critical Systems

In these physical examples the controlling parameter needs to be externally tuned to hold the system at the critical point. A *self organising* critical system (SOCS), on the other hand, adjusts itself so that its controlling parameter moves to the critical value. Its dynamics has an attractor at the critical point; the critical point is an *emergent property* of the dynamics.

The prototypical example of a SOCS is that of slowly adding simulated sand to a simulated sand pile [Bak]. The controlling parameter is the slope. The frozen state has low slope: nothing happens. The chaotic state has high slope: the pile collapses. The system self-organises to the critical slope: if the slope is low, nothing happens, and adding more sand to the pile increases the slope; if the slope is high, the pile collapses, reducing the slope. At the critical slope, avalanches happen on all scales, from a single grain to the entire pile, with a *power law* distribution in the frequency of their size; so no preferred or typical size is singled out. (It turns out that real sand is too dense to behave in this way; but rice can.)

SOCS are typified by such power laws: they lack any scale in time or space, exhibiting power law temporal fluctuations and fractal spatial organisations [JensenHJ]. This means that events on all scales have the same cause, so no special catastrophic cause is needed to explain the catastrophic events.

Other examples of SOCS may include earthquakes, forest fires (with the controlling parameter being the density of unburned trees), autocatalytic chemical networks [Kauffman93], ecological food webs [Drossel] and industrial supply webs, stock market prices, control systems with in-built self-regulation, and large communication networks.

2.2 Communication Networks

Communication networks and their security are what concern us here. Evidence for a move towards critical behaviour of comms networks includes network traffic jams, computer virus propagation [Kephart], and small changes in administrative policies occasionally having cascading knock-on effects (for example, restricting a protocol that is managing other protocols or resources can introduce a bottleneck; legitimately turning off a data flow can result in a chain of resource exhaustion effects due to upstream application buffer exhaustion; security access controls stopping even a small percentage of system traffic can result in the higher layer protocols causing a cascading halt [Chivers]).

As networks become ever larger and more dynamic, their behaviour becomes ever more internally adaptive [Addis], rather than being hand-tuned by external SysAdmins, and they move from being simple critical systems to full SOCS. Any defences we wish to design need to take this into account.

SOCS have a driving force timescale very much longer than the relaxation force timescale [JensenHJ]. Some kind of pressure slowly builds up on the slower driving force timescale, until it is big enough to overcome a threshold, leading to "cascades" of relaxation on the faster timescale. The difference in timescales means that the entire cascade of avalanche events can be considered to occur between the driving force events.

In the case of attacks on a network, attackers can be considered to be applying the driving force. For example, they may attempt to cause jams by flooding the network with messages.

One particular kind of attack is to attempt to tune a system "to the brink", and then use one small change to push it over the edge. For example, attempting to nearly exhaust each of a range of resources, and then letting a final small resource request (possibly made by an innocent third party) push the entire system over the limit. These kinds of attack can actually be harder to achieve in the context of a SOCS, for two reasons. Firstly, the system is already close to critical, so a small change may simply trigger a cascade, stopping the attacker being able to build up a large pressure. (Of course, the system being close to critical, a small change may well trigger a large cascade. But that is a different issue, and no different for an attacker than for a legitimate user.) Secondly, the attacker may have less access to the detailed behaviour of the system, so may not be able to fine tune an attack. For example, it is difficult to fine tune a resource exhaustion attack in the face of an adaptive stochastic resource allocation policy.

It should be noted that there is not a perfect analogy between intelligent attacks and a classic SOCS. As noted, with a classic SOCS, the driving force timescale is much longer than the relaxation timescale. An intelligent attacker, however, may be able to drive the system on a much shorter timescale, possibly of the same order as the relaxation timescale, and hence build up the pressure much more quickly. New driving force events may occur during the relaxation cascades. For example, an intelligent attacker may be able to design a new virus on a timescale comparable to the defence response time, rather than on the longer evolutionary timescale that nature requires, or simply release multiple diverse viruses simultaneously. This similarity in timescales may result in a qualitatively different behaviour from classic SOCS [Sole] (but not one that is in any way more understandable or predictable).

3. Modelling Critical Systems

3.1 What to model

This move towards SOCS mirrors a move in the kinds of models we need to build to understand and design our systems. Classical models emphasise *static* aspects: entities, states, events, fitness landscapes. SOCS, and nature-inspired models in general, must emphasise *dynamic* aspects: processes, relationships, environment, growth and change, attractors, trajectories [Goodwin].

When we are modelling, designing and predicting a complex network, there are several things we are interested in. There are specific properties we want it to have, such as stability and resilience in the face of errors (parts of a large enough network will always be broken) and growth (new instances and new kinds of nodes, connections, and communications); availability and throughput properties; and the like.

There are more general properties, such as what information the system needs in order to self-organise, and how this information can be made easily available to the system. This raises further concerns, such as whether that availability would compromise privacy requirements. Also this organising information itself becomes a target for attack.

Spatial properties of systems are crucial: a SOCS with spatial extent, where quantities can "diffuse" from one neighbourhood to another, behaves very differently from one that is a homogeneous mass [Sole]; spatially propagating waves of global behaviour can occur. In an artefact such as a communication network, the concept of proximity is not as clear cut as in a natural system: it can mean spatial proximity, but it might also refer to connectivity [Milner02], or even similarity in physical design. It simply needs to be some property that has the potential for supporting some kind of diffusive process.

3.2 Current modelling languages

We have a vast resource of modelling languages and techniques to draw upon.

There are languages for defining computational processes, such as CSP [Hoare] [Roscoe] and CCS [Milner89]. More recently, languages designed to cope with mobility, locality, change, and reconfiguration have appeared, such as the pi-calculus [Milner99] and Ambient Logic [Cardelli].

There are languages and techniques for performance modelling, such as queuing theory, and Markov models [Haggstrom]. We need to remember that SCOSs are far from equilibrium, however.

There are languages for probabilistic reasoning under uncertainty (for example, [JensenFV]). And there are techniques from biology, such as epidemiological models of disease propagation (for example, [Chavez]), and much work on biological and chemical networks (for example [Fontana]).

3.3 Modelling networks, and emergent properties

We need more powerful models of complex networks, both artificial and natural. There is much mathematical theory of networks and graphs, but this tends to be of static, homogeneous, structured, closed networks. SOCS on the other hand needs theories of dynamic, heterogeneous, unstructured, open networks.

- *Dynamic*: it is not in steady state or equilibrium, but is far from equilibrium, governed by attractors and trajectories. Swarm networks may offer insights here [Bonabeau].
- *Heterogeneous*: the nodes, the connections, and the communications can be of many different types.
- *Unstructured*: there is no regularity in the network connectivity: it is not regular, or fully connected, or even random. Some recent advanced in Small World networks offer intriguing new insights [Barabasi] [Watts].
- *Open*: the components are not fixed: nodes and connections may come and go; new *kinds* of nodes and connections may appear.

3.4 Modelling emergent properties

We also need models that let us express emergent properties, and design and build systems that exhibit these properties. Abstract models are needed to gain deeper understanding, and to help derive and state general laws. Such laws would not help predict fine details of behaviour, but would capture general properties that could be used to guide the understanding and design of systems with emergent properties. [Kauffman95] puts it well (in the context of evolution, but the argument holds for other kinds of emergent behaviour):

> *We can never hope to predict the exact branchings of the tree of life, but we can uncover powerful laws that predict and explain their general shape.*

3.5 Simulations

In addition to abstract general models, detailed executable simulations are also necessary.

Simulation is needed in order to gain knowledge about detailed behaviour, and to make detailed predictions. In some cases simulation may be the only way to gain such insight, as the details of emergent properties may not be predictable in general. However, simulations by themselves may not impart much additional understanding of the system – a complicated messy incomprehensible reality has merely been replaced by a complicated messy incomprehensible simulation.

4. Artificial Immune Systems

If we are interested in detecting and preventing security attacks in networks, we can take inspiration from the vertebrate immune system. The vertebrate immune system is much more sophisticated than that of lower animals or plants; it uses antibodies, which allows it to adapt to new previously unseen threats, and remember previously encountered threats. (Nevertheless, it is not infallible.) As well as its defensive function, it also has an important maintenance function.

The classical view of the immune system is that of *passive guardian*. It lies dormant, waiting for attack, and then springs into action, defeating the invader, and then sleeps until the next attack. A more recent view is that of a dynamic SOCS [Cohen], constantly reacting and adapting to its environment.

The view of the vertebrate immune system as SOCS is an illuminating metaphor for network security. Artificial Immune Systems [Dasgupta] [Segel] are currently being developed as general pattern recognition and classifier systems. The application of AIS of interest here is the one that originally spawned the metaphor: intrusion detection and prevention. This takes the analogy of communication network as body, legitimate traffic as the body's ordinary behaviour, and faults and attacks as wounds and infections (see for example, [Somayaji]).

Current AIS are very simplistic models of the incredibly complicated natural immune system. Nevertheless, the metaphor immediately raises some issues. It warns us that homogeneity (lack of diversity in hardware and software) in the network may increase vulnerability to attacks. Certain email viruses and internet worms have already exploited this vulnerability. It also suggests that, even with good defences in place, there may be the possibility of (analogues of) "antibiotic resistant superbugs" caused by an analogue of the evolutionary arms race. Indeed, with the attackers being intelligent agents, this possibility is far greater than for natural immune systems with their intelligent protectors producing vaccines and other medicines. Such agents have all the power of (artificial) evolutionary approaches at their disposal, and in addition, can add in intelligent search strategies.

5. Super Critical Systems

Once we recognise that communication networks may be critical systems, even SOCS, we can use our growing understanding of the dynamics of such systems to see that certain kinds of defences may do more harm than good, in the long term.

The defences are attempts to stop the many cascades that happen at the critical point. However, such defences may simply push the critical system into a *super-critical* state. The system becomes "an accident waiting to happen", and the eventual inevitable cascades are simply bigger [Buchanan].

For example, forest fires are a classical example of SOCS. The models are simple grids of "trees", that randomly "burn", with the fire jumping to neighbouring trees if it can. This is a kind of percolation model [Stauffer], and as such it has a fairly sharply defined density threshold, below which the fires burn out rapidly, above which they consume the entire forest. Natural fires may self-organise real forests to this density threshold. Attempts to control natural fires by putting them out increase the density (particularly of underbrush), which may increase the probability that a future fire will be a large catastrophic cascade. The US is changing its forest fire policy from the traditional Smokey Bear's zero tolerance to one where "naturally ignited wildland fires may burn freely as an ecosystem process". See [Franke] for an in-depth discussion of Yellowstone Park's 1988 fire.

Another example where technology can push a system into a super-critical state is that of traffic management. At the critical point, between smooth free flow and solid jams, the throughput is maximised. A management system artificially moves the critical point by controlling the traffic, enabling higher throughput. But if the management system goes down, the traffic instantly grid-locks, as it finds itself in an unmanaged super-critical state. This is an example of the *fragility of efficiency*, exhibited by many "Just in Time" systems. Additionally, although throughput is maximised at the critical point, it is also the point where the traffic patterns display maximum unpredictability (cascades of jams on all scales) [Sole]. So maximum global efficiency leads to maximum unpredictability, which may not be desirable.

An example of super-critical state problems much closer to the metaphor of immune systems is that of the so-called "hygiene hypothesis". Studies suggest that insufficient exposure of the immune system to challenges in childhood, caused by living in a clean environment, may lead to immune system problems such as allergic asthma [Matricardi]. (To be fair, the chemicals used to make the home environment clean have not yet been ruled out as causes.)

This "use it or lose it" view suggests the immune system needs to be exercised, that SOCS should not be pushed into a super-critical state. If large communication networks are, or are becoming, SOCS, this raises the obvious question: are security hygiene practices making the system super-critical? Are they merely deferring, and magnifying, possible disasters? For example, something as simple as trying to maximise throughput by tuning the system, or allowing it to tune itself, to the critical point could result in accidental denial of service behaviours.

6. Conclusions

Communication networks are, or are becoming, SOCS. We can use our understanding of SOCS to design and predict certain emergent properties of these systems.

That understanding is still growing. We need to develop a theory of complex networks: ones that are heterogeneous, unstructured, dynamic, and open. And we need theories and models of emergent properties.

We must avoid the trap of pushing the systems into super-critical states, possibly in a misguided attempt to prevent problems. Rather, we should discover how to make high consequence systems *sub*-critical. This will mean being willing to accept a constant (but low) level of security "illness". We will need to welcome inefficiency, or "slack" [DeMarco] as a requirement of sub-criticality. We may even learn to welcome naive hackers as an immunisation resource!

Acknowledgments

I would like to thank John Clark, Howard Chivers and Fiona Polack for interesting discussions about, and comments on, this paper.

References

[Addis] M J Addis, P J Allen, Y Cheng, M Hall, M Stairmand, W Hall, D DeRoure Spending less time in Internet traffic jams. *PAAM99 : Proceedings of the Fourth International Conference on the Practical application of Intelligent Agents and Multi-Agents*, 1999.

[Bak] Per Bak. *How Nature Works: the science of self-organized criticality*. Oxford University Press. 1997.

[Barabasi] Albert-Laszlo Barabasi. *Linked: the new science of networks*. Perseus. 2002.

[Bonabeau] Eric W. Bonabeau, Marco Dorigo, Guy Theraulaz. *Swarm Intelligence: from natural to artificial systems*. Oxford University Press. 1999

[Buchanan] Mark Buchanan. *Ubiquity: the science of history*. Weidenfeld & Nicholson. 2000.

[Cardelli] Luca Cardelli and Andrew D. Gordon. Ambient Logic. 2002 (submitted)

[Cohen] Irun R. Cohen. *Tending Adam's Garden: evolving the cognitive immune self*. Academic Press. 2000.

[Chavez] C. Castillo-Chavez, S. Blower, P. vande Driessche, D. Kirschner, eds, *Mathematical Approaches for Emerging and Re-emerging Infectious Diseases Part I: An Introduction to Models, Methods, and Theory*. Springer. 2002.

[Chivers] Howard Chivers. Private communication. 2002.

[Dasgupta] Dipankar Dasgupta, ed. *Artificial Immune Systems and Their Applications*. Springer. 1999

[DeMarco] Tom DeMarco. *Slack*. Broadway. 2001.

[Drossel] B. Drossel, A. J. McKane. Modelling Food Webs. In *Handbook of Graphs and Networks*. Wiley. 2002.

[Fontana] Walter Fontana, Günter Wagner, Leo W. Buss. Beyond digital naturalism. *Artificial Life*, 1:211-227, 1994.

[Franke] Many Ann Franke. *Yellowstone in the Afterglow: lessons from the fires*. 2000.

[Goodwin] Brian Goodwin. *How the Leopard Changed Its Spots: the evolution of complexity*. Phoenix. 1994.

[Haggstrom] Olle Haggstrom. *Finite Markov Chains and Algorithmic Applications*. Cambridge University Press. 2002.

[Hoare] C. A. R. Hoare. *Communicating Sequential Processes*. Prentice-Hall. 1985.

[JensenFV] F. V. Jensen. *Bayesian Networks and Decision Graphs*. Springer. 2001.

[JensenHJ] Henrik Jeldtoft Jensen. *Self-Organized Criticality: emergent complex behaviour in physical and biological systems*. Cambridge University Press. 1998.

[Kauffman93] Stuart A. Kauffman. *The Origins of Order: self-organization and selection in evolution*. Oxford University Press. 1993.

[Kauffman95] Stuart A. Kauffman. *At Home in the Universe: the search for laws of complexity*. Viking. 1995.

[Kephart] Jeffrey O. Kephart and Steve R. White. Directed-Graph Epidemiological Models of Computer Viruses. Proceedings of the *IEEE Computer Society Symposium on Research in Security and Privacy*, Oakland, California, 1991.

[Matricardi] P. M. Matricardi, F. Rosmini, S. Riondino, *et al*. Exposure to foodborne and orofecal microbes versus airborne viruses in relation to atopy and allergic rhinitis: epidemiological study. British Medical Journal. 320:412-417. 2000.

[Milner89] Robin Milner. *Communication and Concurrency*. Prentice-Hall. 1989.

[Milner99] Robin Milner. *Communicating and Mobile Systems: the pi-calculus*. Cambridge University Press. 1999.

[Milner02] Robin Milner. *Bigraphs as a Model for Mobile Interaction*. ICGT: First International Conference on Graph transformation. LNCS vol 2505. Springer. 2002. (electronic edition)

[Roscoe] A. W. Roscoe. *The Theory and Practice of Concurrency*. Prentice-Hall. 1997.

[Segel] Lee A. Segel, Irun R. Cohen, eds. *Design Principles for the Immune System and Other Distributed Autonomous Systems*. Oxford University Press. 2001

[Sole] Ricard Solé, Brian Goodwin. *Signs of Life: how complexity pervades biology*. Basic Books. 2000.

[Somayaji] A. Somayaji, S. Hofmeyr, S. Forrest. Principles of a Computer Immune System. *New Security Paradigms Workshop*. 1998.

[Stauffer] D. Stauffer, A. Aharony. *Introduction to Percolation Theory*. 2nd edition. Taylor and Francis. 1994.

[Watts] Dunan J. Watts. *Small Worlds: the dynamics of networks between order and randomness*. Princeton University Press. 1999.