Facilitating a well-founded approach to autonomic systems

Simon Dobson

Systems Research Group
School of Computer Science and Informatics
UCD Dublin IE
E-mail: simon.dobson@ucd.ie

Abstract

While it is desirable for all computing and communications systems to have well-defined and verifiable behaviour, autonomic systems must additionally guarantee that their adaptive behaviour is correct, both in the sense of responding appropriately to changes in context and in the sense of continuing to meet the high-level requirements of the system. Ensuring such high levels of "process" correctness poses a significant challenge for system designers. Formal methods provide a valuable tool to assist in the design, analysis and verification processes. The goals of the ACF's semantics working group is to identify formal techniques that may be applicable to the development of autonomic systems, and to promote the understanding of these techniques within the research community.

1 Introduction

Autonomic communications pose significant problems for systems designers. The goals of autonomic systems – to exhibit self-management, self-optimisation, self-healing and other "self-*" properties – imply that a system's detailed behaviour changes in order to maintain high-level characteristics. This in turn implies that we can relate the behavioural changes to their impacts in a structured and predictable way.

Whilst it is attractive to think that such properties can be developed simply, it seems unlikely that systems intended for open and dynamic environments will successfully address these challenges unless we can perform substantial simulation, testing and analysis on them to ensure that their behaviour is within acceptable bounds. The well-known limitations of testing [6] strongly support the desire for more abstract, whole-system models of adaptive systems that can be analysed more effectively than code. This in turn suggests a high-level, qualitative, *semantic* understanding of the algorithms and techniques used, independent of the code used to implement them.

In this paper we introduce the rationale and work of the Autonomic Communication Forum's Semantics Working Group. Within the ACF's remit to promote standardisation and interoperability across autonomic systems, the semantics WG aims to ensure that the approaches being proposed offer tractable and predictable properties that can be formally stated and verified, in order to improve the confidence that researchers and practitioners may have in the quality and stability of autonomic solutions going forward.

Section 2 discusses the need for semantic foundations for autonomic systems, highlighting the contribution that applied formal methods can make to practice. Section 3 presents some of the challenges faced in developing both autonomic systems and standards, as seen from the perspective of applied formal methods and semantics, and section 4 describes some of the work proposed for the semantics WG in addressing these issues. Section 5 concludes with some directions for future work.

2 The need for foundations

Why do we need foundations for autonomic systems? Given the demonstrated – if still limited – successes in building self-managed systems, why should we invest research resources in mathematical and other models rather than in algorithm development? There are at least three answers to this question:

Comprehension. Enterprise computing and communications systems – it is growing increasingly meaningless to attempt to differentiate strictly between the two – have a significant economic significance to all stakeholders. The availability of information sys-

tems that can be easily customised to address narrow-window business opportunities, that provide flexible management and robust, predictable service offer significant revenue opportunities for organisations, allowing IT to be a profit centre rather than a cost centre.

However, such opportunities can be capitalised upon only if they can be relied upon. Providers need to know that systems can deliver before committing resources and reputation to them; consumers need confidence that the services they pay for will be delivered. This implies that providers can accurately predict how their systems will behave across a range of situations. Making such predictions with confidence implies an ability to model, simulate and test complex systems. All three approaches are challenging: mathematical modelling remains limited, some systems are too large to simulate effectively, while complete testing is similarly ineffective. However, it is unarguable that having a well-defined, analytic model underpinning a system's design and implementation provides additional tools for building confidence, and can provide insights that make simulation and testing more focused.

Compositionality. No enterprise system is designed, developed or maintained in one piece: *all* interesting systems and their properties arise from composition. Composition can occur in various guises, from the loose coupling of transaction-processing systems to the tighter coupling of high-performance traffic routing, but the problems remain the same: understanding how the (mis-)behaviour of one component will affect another. Whatever development approach is adopted, a well-defined model of interfaces and services provides a basis upon which to perform an analysis of such interactions.

Routing provides a good example. Models such as queueing theory and network calculus [4] allow the interactions between traffic streams to be studied and (in some cases) have their properties proved, and the insights gained can then be codified within IntServ or similar frameworks. The point is that the impact of new services can be studied analytically and used to understand one of the ways in which possibly damaging interactions can occur.

Diversity. Autonomic communications exhibits a fascinating diversity of approaches to the core "self-*" problems (see Dobson *et alia* [3] for a recent survey). This diversity is to be encouraged, since it provides a dynamic ecosystem within which to search for the most effective solutions. However, this scientific argument must be set against commercial requirements for interoperability and the desire to "pick a winner" early rather than being stuck with ineffective technology.

How can these two desires be reconciled? The answer would seem to be to extend compositionality into the realm of the autonomic control system itself, as well as the systems it is controlling. Given the breadth of challenges, it seems unlikely that a single approach will prove universal. Interoperating between control strategies requires that their global behaviours be understood and that their touch points be identifiable. Formal models simplify both tasks.

3 Challenges and approaches

The standard view of autonomic systems is that they "close the loop" of network control. In figure 1, the network *collects* observations and measurements of both the network's environment and its own responses to this environment as it changes over time. These observations are then *analysed* to determine their implications, and used to guide a *decision* process that causes the network to perform some *actions* to change its own behaviour. The consequences of these actions may then be observed, closing the control loop. This is different from traditional network management approaches, in which collected data are fed to human decision-makers.

3.1 Desirable properties

Before we consider the formalisation of this process, we should consider the desirable properties that *any* model should have.

Correctness. All IT systems need to be correct, and this challenge is only highlighted by the depressing regularity with which this is not the case. However, *adaptive* systems have even stronger correctness requirements.

In developing a "standard" desktop, server or embedded system, the context and requirements usually move at human speeds. We can identify what correctness *means* for such a system, and can therefore (at least in principle) decide whether it meets its requirements. If the system evolves, it does so in discrete jumps, each of which can be said to be correct (or not).

We can contrast this *point correctness* against what happens in adaptive systems, whose requirements change continuously. We need to be sure that such a system behaves correctly according to the requirements of the moment, but also that it is correct with respect to the adaptations it makes: it must be *process correct* as well as point correct [1].

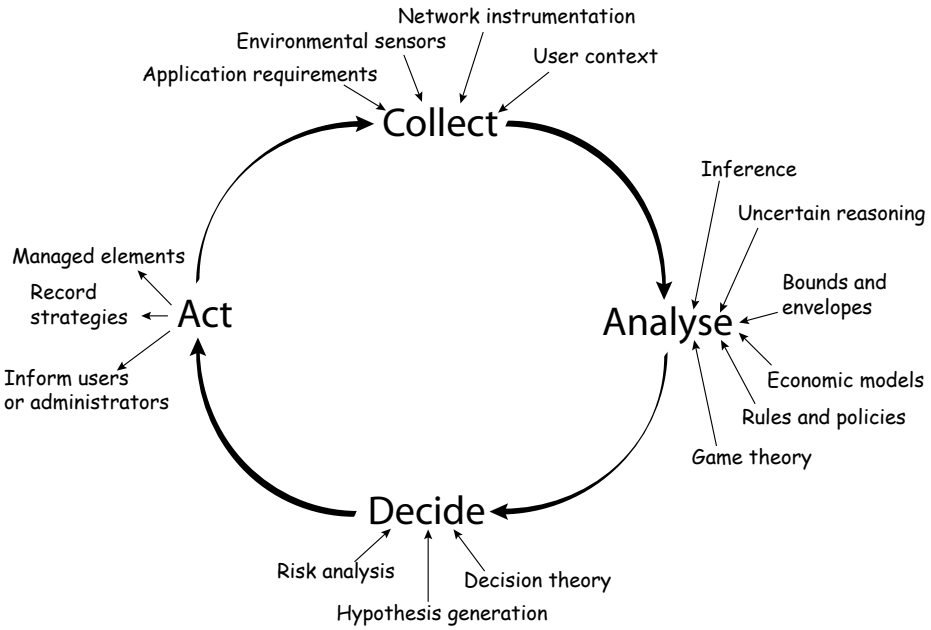

Figure 1. The autonomic control loop (from [3])

Stability. Process correctness manifests itself most clearly in communications. For example, consider a network that must transport a set of traffic streams, each with different priorities, bandwidths, isochrony requirements and so on. As this traffic metadata changes, we might want the network to adapt its transport strategies to maintain the best service possible. What we do *not* want is for the system to suddenly change the ways in which it handles streams, or to diverge to extreme behaviours without notice.

This notion of stability – gradual change that remains within an envelope of correctness – is commonly encountered within control systems, as well as in many physical phenomena. It is extremely desirable as a general rule in adaptive systems, since it provides a measure of confidence that the large-scale behaviour of the system will remain within predictable bounds.

Responsiveness. Set against stability, however, is the desire for responsive change. To continue the network example, adding a new high-priority video stream needed for emergency response (for example) might require that other streams be de-prioritised or even dropped: the priority of the emergency information permits us to break guarantees to other, lower-priority traffic.

This is an obvious statement to make, but a surprisingly difficult one to implement in an open manner. If we want to avoid hard-coding particular behaviours into the network (which damages extensibility), then the network must be able to decide *for itself* which traffic is important and the correct response to take. This requirements a substantial amount of information about traffic, its uses, the techniques that can be applied at a network level and their intended consequences. Clearly this information is inherently cross-layer, in the sense of requiring input from the network, the applications and uses to which the traffic is being put [2].

Predictability. Underlying all these properties is a desire for predictability: being able to say *a priori* how a system will respond to given stimuli.

Why is this important? Is it not enough to provide adaptations that are in some sense "natural" given the conditions? This is an attractive argument both technically and philosophically: the system will exhibit certain low- and mid-level characteristics, and higher-level properties will emerge from their interactions. Unfortunately this is not acceptable within a systems engineering context: both providers and consumers want to be given guarantees as to how their systems will behave, especially if a premium is being charged for specific behaviours ("give this user gold service," for example).

3.2 Example approaches

Part of the excitement of research in autonomic systems is the diversity of techniques that may be applied to addressing the control problems encountered. Simultaneously, of course, this poses the greatest challenge to a standards-defining organisation which must ensure interoperability without stifling innovation. Diversity and openness also reduce (although they do not remove) the applicability of traditional control theory, which does not typically allow compositional solutions.

Considering figure 1 again, we can see that each component of the control loop offers the potential for formal treatments.

Collecting information about the network and its environment requires a substantial investment in sensing. This encompasses both traditional sensors for location and the like, but also "virtual" sensors able to sense the condition of the network itself (for example latency, traffic flow, available bandwidth and so forth). Studies in pervasive computing strongly suggest that sensor readings be collected widely and represented in a standard format. Defining *standard ontologies of sensor information* provides for interchange between subsystems.

The analysis of sensed information must deal with the inherently uncertain nature of most sensor information. It is important to stress that this uncertainty cannot be engineered out of a system, although it can be reduced by careful design. A sensor that provides information about network load, for example, takes a point observation that may be invalidated immediately by future traffic. Managing this uncertainty mandates the use of one of the several techniques available for uncertain reasoning, including Bayesian networks, fuzzy logic, Dempster-Shafer evidence theory, decision theory and so forth. (See the classic work by Pearl [5] for further consideration of this topic.)

A number of other techniques have also been applied to analysis, including game theory, econometrics, category theory, topology, fibre structures and so on. The point is that *the analysis of uncertain data requires structured uncertain reasoning*, and cannot simply be performed *ad hoc*.

The decision aspect of autonomic control is often closely tied to analysis – although it is not completely clear that this should be the case, and such close coupling may mitigate against interoperability. One may regard autonomic decision-making as a process of hypothesis formation: the system hypothesises about the state of the system, its trajectory and the actions that should be taken to correct this, and generates a plan to bring about intended changes. Any such plan will have inherent risks, and it seems sensible to consider these explicitly.

Acting on system is perhaps the most "operational" part of the control system, being concerned with applying control actions through whatever actuators are available. One important (and rather under-studied) implication of autonomic control is the need to explain actions to human supervisors or users.

4 ACF's planned contributions

The ACF's mission is to promote both research and standards in autonomic communications. It should therefore come as no surprise that formal modelling and analysis are seen as key enablers of both research and standardisation.

4.1 The Semantics Working Group

We have established the Semantics Working Group to provide a focus for applying formal techniques to autonomic systems. Such techniques appear in the ACF's programme in two distinct but related ways.

The ACF supports a number of Expert Groups developing standards for particular aspects of autonomic communications, for example in the areas of policy and modelling. Each of these groups will use appropriate models an techniques to ensure that the documents standards they are developing are well-founded and consistent in order to support interoperability.

More broadly, there are (as observed above) a wide variety of techniques being applied to autonomic systems. However, many techniques have very steep learning curves for researchers and practitioners wishing to apply them to communications and systems problems, and this harms their uptake. The often extensive literature of many techniques may not be targeted at – or even comprehensible by – non-specialist readers. We therefore intend to develop a suite of introductory tutorials describing the core principles of techniques that seem to be proving useful, written by domain experts and targeted specifically at providing an introduction and guide for computer scientists and engineers. This will allow us to broaden the appeal and application of formal techniques within autonomic systems.

5 Conclusion

Autonomic systems need strong notions of adaptive correctness, stability and compositionality if they are to be accepted and applied broadly. We believe that this mandates the use of formally well-founded techniques whose properties can be proven and which can

be shown to interoperate. The goal of the ACF's semantics working group is to provide a focus for the formal design analysis of autonomic systems, and to simplify the application of these techniques to real-world problems.

In the immediate future, our goal is to develop an initial set of tutorials for key techniques in the literature, and to make them widely available, perhaps focusing initially on game theory, agent-oriented approaches, uncertain reasoning and ontologies. We welcome collaborators in this process, both in advancing specific techniques and in showing how they may be applied to practical standards-making. A thorough and broadly-based approach for formal modelling and analysis will ensure that the emerging standards in autonomic systems provide a stable and useful platform for future systems and research.

Acknowledgements

This work, and UCD's involvement in the ACF, is partially supported by Science Foundation Ireland under grant numbers 05/RFP/CMS0062 "Towards a Semantics of Pervasive Computing" and 03/CE2/I303-1 "Lero – the Irish Software Engineering Research Centre."

References

[1] J. Coutaz, J. Crowley, S. Dobson, and D. Garlan. Context is key. *Communications of the ACM*, 48(3), March 2005.

[2] S. Dobson. Putting meaning into the network: some semantic issues for the design of autonomic communications systems. In M. Smirnov, editor, *Proceedings of the 1st IFIP Workshop on Autonomic Communications*, volume 3457 of *LNCS*, pages 207–216. Springer Verlag, 2005.

[3] S. Dobson, S. Denazis, A. Fernández, D. Gaïti, E. Gelenbe, F. Massacci, P. Nixon, F. Saffre, N. Schmidt, and F. Zambonelli. A survey of autonomic communications. *ACM Transactions on Autonomous and Adaptive Systems*, 1(2):223–259, December 2006.

[4] J.-Y. Le Boudec and P. Thiran. *Network calculus: a theory of deterministic queuing systems for the internet*, volume 2050 of *Lecture Notes in Computer Science*. Springer-Verlag, 2001.

[5] J. Pearl. *Causality: models, reasoning and inference*. Cambridge University Press, 2000.

[6] G. G. Schulmeyer and J. McManus. *Handbook of software quality assurance*. Prentice Hall, 1998.