Autonomic communications

Simon Dobson\textsuperscript{1}, Spyros Denazis\textsuperscript{2}, Antonio Fernández\textsuperscript{3}, Dominique Gaît\textsuperscript{4}, Erol Gelenbe\textsuperscript{5}, Fabio Massacci\textsuperscript{6}, Paddy Nixon\textsuperscript{1}, Fabrice Saffre\textsuperscript{7}, Nikita Schmidt\textsuperscript{1}, Mikhail Smirnow\textsuperscript{8}, and Franco Zambonelli\textsuperscript{9}

\textsuperscript{1} School of Computer Science and Informatics, UCD Dublin IE
simon.dobson@ucd.ie
\textsuperscript{2} Hitachi Research Europe FR
\textsuperscript{3} Universidad Rey Juan Carlos ES
\textsuperscript{4} Université de technologie de Troyes FR
\textsuperscript{5} Imperial College London UK
\textsuperscript{6} Università di Trento IT
\textsuperscript{7} BT Group plc UK
\textsuperscript{8} Fraunhofer FOKUS DE
\textsuperscript{9} Università di Modena e Reggio Emilia IT

Abstract. Autonomic communications seeks to improve networks’ and services’ abilities to cope with unpredicted change, including changes in topology, load, task, the physical and logical characteristics of the networks that can be accessed, and so forth. Broad-ranging autonomic solutions require designers to account for a range of end-to-end issues affecting programming models, network and contextual modeling and reasoning, decentralised algorithms, trust acquisition and maintenance – issues whose solutions may draw on approaches and results from a surprisingly broad range of disciplines. We survey the current state of autonomic communications research and identify significant emerging trends and techniques.

1 Introduction

Modern network infrastructures have achieved a small miracle in presenting a simple and uniform façade to applications. In many ways, the programmer’s view of the network has become simpler over time, with TCP/IP acting as a \textit{de facto} gateway to a wide range of network technologies.

This external simplification has not unfortunately been matched by a corresponding simplification in the construction, management and extension of the network from a provider’s perspective. Adding a new network segment, a new protocol, a new kind of element, or support for a new user- or system-level application have become fraught exercises in managing the complexity of interactions between elements. This in turn both reduces innovation in networks and network-centric services and can directly affect the economic viability of products and services that rely directly on IT and communications agility.
The development of self-managing, self-configuring and self-regulating network and communications infrastructures – collectively referred to as autonomic communications – is an area of considerable research and industrial interest. By analogy to the human autonomic nervous system, which regulates homeostatic functions without conscious intelligent control, autonomic communications seeks to simplify the management of complex communications structures and reduce the need for manual intervention and management. It draws on a number of existing disciplines including protocol design, network management, artificial intelligence, pervasive computing, control theory, game theory, semantics, biology, context-aware systems, sensor networks, trust and security. The distinguishing feature is the fusion of techniques from these fields in pursuit of a goal of simplified systems deployment and management.

It is clear that a topic drawing so diversely from existing disciplines presents a serious learning curve for anyone wanting work with autonomic techniques. Our goal in this paper is to reduce this learning curve by presenting a comprehensive survey of the current state of research in autonomic communications. We address the literature from five separate but interlinked perspectives: the design and analysis of decentralised algorithms; the modelling, handling and use of context; novel and extended programming approaches; issues and approaches for addressing security and trust; and systems evaluation and testing. We highlight some of the emerging trends with a view of informing the evolving research programme, and conclude with some observations on the potential impact of autonomic communications and the fundamental research challenges that remain.

2 Issues in autonomic communications

The increasing density of the global communications network offers industry, network operators, developers and users both dramatic advantages and significant challenges:

For industry The need to maintain diverse and complex networks is often a significant (and increasing) cost of doing business. A more autonomic infrastructure would reduce these costs and facilitate new opportunities, but only if made sufficiently flexible, robust and secure for use across the spectrum of corporate communications.

For operators Increasing interconnectivity potentially allows improved robustness and bandwidth, but also increases the complexity of management and the fragility of protocols in coping with a highly dynamic and largely scale-free environment composed of diverse networks and technologies. Finer-grained mobility and roaming require that the relationships between operators, and between operators and users, be extensively re-thought.

For developers Mobile and pervasive networks allow applications and services to extend into the environment, both providing and benefiting from sensing capabilities and closer integration with the personal and social goals of
users, but at the cost of massively increased programming and configuration complexity.

For users Mobility and ubiquity tilt the balance of communications systems in the users’ direction, placing individually- and socially-focused adaptations at the core of the systems architecture, but with the danger that the increased potential for surveillance and complexity will erode the privacy of individual and further disenfranchise entire social groups.

Existing network paradigms deal poorly with this multi-level tension between complexity and simplicity, diversity and ubiquity.

Traditional networks have been constructed and co-ordinated centrally according to a single plan, and can consequently be architected using a homogeneous population of components with common technical standards and management goals. By contrast, next-generation networks are expected to grow more chaotically, with no centrally-mandated goals or levels of service, with no universally-agreed protocols or other technical standards, and with no a priori knowledge of the topology or component population. This freeing of central control over networks has the potential to release an enormous burst of creativity and new economic activity that is impossible to achieve in a more constrained environment, and consequently has the potential to make networking a vehicle of economic growth and social change.

It is clear, however, that the mathematical, economic and technical bases of networking must be changed radically to address the implied challenges. Specifically, the next-generation network must be radically distributed and decentralised, self-describing, self-organising, self-managing, self-configuring and self-optimising, providing a seamless communications infrastructure composed of multiple technologies and able to leverage local information and decisions without sacrificing global performance, robustness and trustworthiness.

2.1 The emergence of autonamics

The notion of using autonomic techniques – of deploying technology specifically to manage and optimise the functioning of other technology on an on-going basis – has its roots in work on control theory and managed elements. While control theory can provide excellent descriptions of closed systems whose components and desired properties are known and described by certain classes of linear or non-linear mathematical models, it deals poorly with general systems (e.g., discrete and continuous, time-varying, having delayed or uncertain information) even when they can be characterised mathematically. Control theory encounters greater difficulty when the system structure is unknown and is being constantly discovered and modified. Managed elements are essential for controlling a system, but typically require extensive human guidance.

Autonomic design seeks to generalise the control-theoretic view of control by enabling more flexible and adaptive functions in the underlying system. By lever-
aging richer information sources than are typically considered in control systems, autonomic systems should be able both to react to evolving situations and to some extent pre-empt expected future demands.

**Fig. 1. Autonomic control loop**

Autonomic systems form a feedback loop (figure 1). The system collects information from a variety of sources including traditional network sensors and reporting streams but also including higher-level device and user context. These are analysed to construct a model of the evolving situation faced by the network and its services, with this model being used as a basis for adaptation decisions. These decisions are actuated through the network, and will potentially be reported to users or administrators. The impact of the decisions can then be collected to inform the next control cycle.

A high-profile use of autonomic techniques is provided by IBM’s autonomic computing initiative [?]. Autonomic computing is seen as a way of reducing the total cost of ownership of complex IT systems by allowing re-configuration and optimisation to proceed on an on-going basis driven by feedback on the system’s on-going behaviour. It combines a technological vision with a business rationale of increasing the coupling between business goals and IT services.

In the communications arena, the traditional architecture of control and data planes has been expanded in a number of ways. Clark’s influential vision of a “knowledge plane” [?] provides architectural support for integrating low-level (transport and network) knowledge with higher-level applications and user context. One may also view this from a context-aware systems perspective as making the “meaning” of the operations a network is carrying out available to influence how it handles those operations [?]. The use of context from beyond the network makes models “situated”, in that they have a model of their place in a wider scheme. We explore techniques for modelling context in section 4.

**2.2 Challenges to theory and practice**

Networks are traditionally described using a number of theories and technologies, the most influential being classical (Shannon) information theory, communication theory, queuing theory and the Internet protocol (IP) suite. However, such foundational insights must also be accessible to the programmers tasked with generating network services. It is now reasonably well-accepted that traditional programming patterns do not deal well with highly dynamic and adaptive systems, and that new patterns are required in order to create algorithms and services that can both cope with and leverage autonomic platforms. We deal with these issues in sections 3 and 5.
A number of experiments have demonstrated small “autonomic” networks purely at the protocol and/or device level. An integrated autonomic backbone, however, requires strong guarantees on interoperability, scalability, security and stability – guarantees that, in a decentralised technical and managerial environment, can only come from a foundational understanding of the construction and composition of scale-free networks and their emergent properties.

The characteristic features of autonomic communications are the use of highly decentralised algorithms having desirable emergent properties while retaining both a high level of global predictability and a close integration with cognitive and other contextual goals (issues addressed in section 3). The emerging theory and protocols underlying autonomic communications must therefore build on the classical understanding by synthesising results from a range of additional disciplines that are typically pursued independently, including:

**Network information and communications theory** Diversity exploitation for structureless and unsupervised links, information flow and bounds in access channels, with and without state.

**Semantics** The relationship of network architecture to environment, and the impact of task-level knowledge.

**Complexity** Bounds and limitations in open systems with minimal centralisation and information exchange, chaos-based analysis and control of global behaviours.

**Security and privacy** Ensuring privacy in systems collecting extensive contextual information.

**Co-ordination and emergent behaviour** Swarm intelligence and chaotic control.

**Multi-agent systems** Co-operative and competitive models of programming and interaction.

**Decentralised algorithms** Minimal-bottleneck and narrow-channel algorithms.

**Optimisation** Game theory and sub-optimal improvement using only local knowledge.

**Verification and validation** Proofs of correctness and stability in open environments.

**Cognitive, user interface and media theories** The psychological and social interactions facilitated by advanced communications infrastructures.

Some of these topics touch on the most advanced aspects of communication dealing with the problem of signal co-existence without *a priori* agreement. Other areas are more traditionally concerned with programming, reflecting the fact that the availability of robust link-level primitives can be exploited by autonomicity to bring more “meaning” into the network rather than treating it purely as an uninterpreted channel. The use of overlays and semantically-informed protocols and algorithms allow us to treat the network in some sense as a programming language to address specific problems using both programmatic and communications-driven paradigms co-operatively, but also raises the importance
of sound algorithm design and the taking cognisance of security and privacy concerns in the core of applications technology – a topic we address in section 6.

The final issue concerns correctness and validation. Adaptive systems offer additional failure modes over and above those of traditional systems. While a traditional system may be tested or proven to be correct, such a “point” solution is not acceptable in the face of adaptation. Each of a system’s adaptive behaviours must be correct; in addition, the adaptations must occur correctly – a situation referred to as “process” correctness. The testing and evaluation of autonomic systems is still very much in its infancy, and we examine some approaches in section 7.

3 Analysis and design of autonomic algorithms

It has been the rule for many years to assume that communication systems could be monitored and controlled by a central entity. However, with the growth of the size and complexity of communication systems, it has become clear that this assumption will be no longer valid. One of the paradigmatic cases of a communication system that has had to evolve to become autonomic and self-organised is the Internet, where its initial centralized structure having to evolve for fully distributed control and management. This trend is slowly expanding to all kinds of communications systems. This has given way to new models and paradigms and has open the door to migrating new research techniques from other fields to be used in these systems.

In this context, new algorithms that take into account the complexity of the communication systems have to be developed and analysed. These algorithms will have to guarantee the emergent properties of the system with limited contextual information and local control, making the system self-organising in a way that is atypical for most algorithm styles. Similarly, in most cases they will have to deal with very heterogeneous and changing environments, and will be in charge of providing reliability and self-correction capabilities to the systems. All this has to be attained for systems that may be formed of millions of different elements, hence requiring a extraordinary level of scalability.

As an example, classical communication paradigms are adapting to these new models. Classical fault-tolerant communication systems and primitives for group communication (see, for instance [?], [?], [?]) which offered very interesting features such as deterministic guarantees, self-repair, load distribution, and flow control, are being re-thought due to their lack of scalability in a wider and (especially) ad hoc networking context. Hence the appearance of group and multicast communication systems and algorithms with probabilistic instead of deterministic guarantees, like the Spinglass system [?] or the algorithms discussed in [?]. A similar evolution has been observed in publish-and-subscribe systems and algorithms (see [?]), to yield for instance epidemic approaches like Newscast [?].
One very desirable design objective would be that algorithms for autonomic self-organised communication are as independent as possible of the specific communication technology. The context in which these algorithms will have to operate range from classical packet-switching networks, in which routers have to act and react locally to different network traffic behaviours, to sensor networks that have to organise themselves simply in order to operate. In between these extremes we have, for instance, peer-to-peer and overlay networks with high scalability issues, and mobile and ad hoc networks that are very dynamic. Ideally, the same algorithms that are used in these set-ups will be useful in other contexts, and in all of them will provide dependability and efficiency. Unfortunately, new algorithms are currently almost always proposed with a very specific kind of communication system in mind, which prevents its use in other systems. It would be of great interest to define general weak models of communications systems so that any algorithm that works for these models can be used in several real communication systems of different classes.

Most algorithms for large communication systems proposed use classical approaches but with extensions to deal with (and hopefully profit from) the large availability of collaborating agents: hence the proposals to use multi-hop routing in overlay networks to circumvent Internet connectivity failures like [?], which profit of the availability of alternative routes. Similarly, there is a body of algorithms to build and maintain peer-to-peer systems based on Distributed Hash Tables (see, for instance, those presented in [?]) and to manage ad hoc and sensor networks [?]. Other initiatives include the use of spiked neural networks distributed throughout the network routers which learn from on-line measurements using reinforcement learning, so as to achieve adaptive routing and address the users’ QoS needs [?]. Genetic algorithms have also been used and evaluated to discover new network paths that offer better potentially better QoS from previously discovered paths[?].

However other, completely new paradigms of algorithm design are also being considered, some of which are in the early stages of development. One such approach is to base the algorithm’s behaviour on nature, which has given way to a collection of new models, systems and algorithms (for example and section 5 and [?]).

Another line of research is networks with autonomic self-adapting topologies. A thorough study of the optimal topologies for each system configuration would lead to a dynamic adaptation of the topology to the system status [?]. For this, it seems to be a promising line of research to apply the increasing knowledge on small-world and scale free networks. Another line to continue exploring is the application of non-linear (chaotic) dynamics to communication systems, mainly to the lowest levels. Chaotic dynamics provide a way to optimise performance in DS-CDMA-based links used in 3G mobile telephony systems [?], which may be extended to other physical- and transport-layer schemes that exploit simultaneously the code-frequency-time-space diversity that are likely to be adopted for forth-generation telecommunications systems [?].
Decentralisation raises significant consistency challenges, which are perhaps encountered most strongly in the area of distributed databases [??,?]. Most algorithms for transactions and other classical approaches to consistency rely on extensive “barrier” synchronisation which is difficult if not impossible in a decentralised and low-reliability context.

3.1 Trust issues in algorithm design

As well as the more obvious security and trust issues (deferred until section 6), algorithm development in the face of autonomic adaptation poses some unique problems. Most algorithm approaches assume a certain degree of collaboration among the different communicating agents. However in a large communication system like the Internet one may expect to have users with different interests and with the capability of adapting the programs they run in their computers to match their desired behaviour. This will certainly lead to assumptions of adversarial behaviour when designing the algorithms. We have examples of this need in the Berkeley Open Infrastructure for Network Computing (BOINC) [?], used to perform scientific computations with high CPU time requirements in the computers of volunteers, and which had to introduce security mechanisms to prevent being deceived by (possibly malicious) wrong solutions being returned from the (untrusted) clients. In this case the problem is not hard to solve (probabilistically) by assigning the same computation to several users and using a voting strategy to choose the good replies [?].

The problem becomes harder when there is going to be a continuous multi-way collaboration and we want to have all users collaborating in the overall system (for instance a service like the BOINC system without central control). In these models it seems natural to introduce game theory in the design of any algorithm that can be suitable to be tampered with. Then, a well though algorithm in this context will guarantee that even a selfish user will obey the rules of the algorithm because it is in its best interest to do so [?]. An example of game theory in action is the incentives to collaborate included in the BitTorrent system [?]. Interestingly, BitTorrent can still suffer of free riding [?], which basically shows the difficulty of designing selfishness-proof algorithms. Game theory is also been considered in most aspects of communication systems (see, for instance, [?],[?],[?]). Game theory can also be applied to enhance existing protocols. An example is extending the distributed replication group [?] to the case that individual nodes act selfishly, catering to the optimisation of their individual local utilities. Game theory may be used [?],[?],[?] to derive equilibrium object placement strategies that can guarantee improved local utilities for all nodes concurrently as compared to the corresponding local utilities under greedy local object placement, do not suffer from potential mistreatment problems, inherent to centralized strategies that aim at optimising the social utility, and yet do not require the existence of complete information at all nodes.
Economic models are another area of significant interest, essentially treating an algorithm as an economic system. In the area of wireless networks, economic and game-theoretic models may be combined to capture the interaction between network control mechanisms operating at different levels, such as power control, channel selection, and rate control, as well as the interaction between distributed autonomous devices, in order to jointly optimize their overall performance and share resources in an efficient and fair manner [?].

3.2 Limits

Of special interest of the research conducted in this field is the exploration of the limits to the amount of information that can be exchanged in a communication system like those considered here, and the techniques required to approach these limits [?]. This is closely related to the concept of (network) information theory, which studies these issues for single (multipoint) channel communication. The extrapolation of the classical (network) information theory to autonomic networks will lead to a new network information theory discipline.

Finally, we believe that new coding advances at the application level will strongly influence the design and evolution of future algorithms for communication systems. Specific new families of codes are Digital Fountain codes [?] and Network Coding [?]. These are already influencing the next generation of algorithms for peer-to-peer content distribution [?]. Furthermore, it is possible that they will influence the future evolution of the Internet and one of its main protocols: TCP [?, ?].

4 Context awareness and semantics in communication

Autonomic communications implies a stronger degree of self-management and self-optimisation than is found in conventional networks, divorced from human intervention (and in many cases from the possibility of such intervention). To provide self-management and optimisation capabilities it is necessary to investigate the context-aware approach to improve networking properties. Software entity, network components, software agents are used to collect context information related to the presence, the location, the identity and the profile of users and services. A typical context use involves locating services and users, calling-up services according to user behavior, providing information for service composition, facilitating ad hoc communication mechanisms between users, and adaptation of the qualities of service to changes in the environment a result of user and service mobility [?].

Two types of context-aware infrastructure can be proposed:

*Passive context aware infrastructure* Context is raw information that, when correctly interpreted, identifies the characteristics of an entity. An entity can be a
person, place, a device or any object that is relevant to the interaction between a user and the services. Context is a function of time and environment. The environment is in turn a function of the users, services, resources and other entities in the environment. In this phase the focus will be on the context gathering and representation. A data model and dissemination protocol represent, store and manage context information. This includes classifying context sources, providing a unified context structural representation and developing mobile storage strategies with data replication techniques to insure the availability and the proximity of context information.

Active context aware infrastructure An alternative technique is to extend passive collection with smart context information delivery. A context-level agreement protocol can provide automatic context matching with the user’s profile, terminal capabilities and service requirements and offering. The primary aim of such a protocol is the adaptive distribution of context information among multiple mobile and fixed sources and destinations (e.g. devices, service components) using (negotiated) specific dissemination attribute such as power saving and cost. Context dissemination can be achieved in both “pull” and “push” modes.

In order to provide self-adaptive behaviour, an autonomic system must be able to reason about both its context and its behaviour relative to that context. This does not necessarily imply a symbolic structure, but does suggest that the system must be able to reflect on its environment and behaviour in some sense and generate feedback as a result [?]. Thus the autonomic network must accept goals and constraints from, and petition for the attention of, its human governors using terms that are meaningful to their needs and cognitive abilities [?]. However, this must be balanced against the need for the autonomic network to map these semantic terms deterministically to and from the self-managing capabilities of its elements. This requires a high degree of semantic interoperability between expression of adaptive behaviour and of the changing context that drives such adaptation. As context-awareness and semantic based reasoning concerns adaptive, networked systems, it requires research towards models and languages for representing their behaviour expressions and methods for adaptation that operate on such, possibly semantically rich, representations.

The representation of the information in the autonomic network in itself represents a significant challenge. For instance in an optimisation problem, optimisation always occurs relative to some environment or property: one optimises for performance, or for robustness, and so forth, with the improvement in one property often coming at the expense of another. Determining which properties to optimise requires that the optimising agent understands the relative priorities that should be given to the various possible optimisation targets, which in turn implies that the optimising agent is aware of the meaning of the information within the network and its place in the on-going user tasks. In an autonomic system the optimising agent is software, implying that meta-data about the meaning of information, streams and operations is injected into the network infrastructure and used to inform network-level decisions [?].
Adaptive pervasive computing systems attempt to provide information and services that are “distraction-free” for their users. Approaches to adaptability may be loosely classified as “closed-adaptive” when all adaptations are pre-specified and “open-adaptive” when new adaptations may be discovered. Representations of context have varied between sub-symbolic uses of neural networks to more traditional symbolic AI, with an apparent consensus favouring concept graphs modelled (at least externally) using the Resource Description Framework (RDF). This has the advantage of providing a well-understood, triple-based representation together with an open exchange format to facilitate integration into the larger system context – an important consideration given the small part that even highly contextualised services play in an enterprise-scale architecture.

Although pervasive computing is generally regarded as distinct from networking, there is significant convergence. A network is essentially a sensorised system which can observe its own low-level activities and constraints. This may be combined with higher-level contextual information about users, services and applications within a framework of uncertain reasoning.

It is widely recognised that managing the structures of context is a significant challenge. Many systems draw a distinction between context (the low-level information observed or inferred about an environment) and the situation (the high-level scenario in which the system is involved). This frequently involves combining information at different semantic levels. The “trails” model provides one such fusion, based on the insight was the non-hierarchical nature of contextual information, and the need to adopt cross-ontology, cross-layer and cross-tool views in order to obtain meaningful results. An alternative approach is to model complete adaptive spaces whose properties may be analysed a priori for conflicts and other properties.

Contextual variation must be carefully controlled in order to generate intelligible behaviour: context has a direct influence on users’ ability to form well-founded conceptual models of systems, so that adaptive behaviour can be shown to have a direct impact on usability. For a system, network or service to be predictable and usable there must be a clear link between adaptations and their environmental causes both in terms of causation and in the details of the way the adaptation supports working in the new context. The communications industry is well accustomed to defining system semantics in a formal or semi-formal manner, ranging from SDL and Z for signalling systems, TMN’s GDMO and the DMTF’s CIM schema for management models and service-oriented models like ODL (Z.100) and the TeleManagement Forum’s NGOSS technology neutral model for component-oriented communications software. However the diversity of these languages reflects the sectorial divisions within the communications industry that research into SAC must challenge. This challenge is being exacerbated by the introduction of a variety of multi-agent technologies to the communication domains, which introduce further language for capturing semantics such as ACLs and KIF.
At the same time however, the W3C's Semantic Web initiative is addressing semantic interoperability issues through the standardisation of a family of ontology languages for encoding knowledge and services on the web: RDF, RDFS, OWL and OWL-S. There has been little attempt to define generic expression of adaptive behaviour, though development on a Semantic Web Rule Language may provide a suitable starting point [?]. These technologies benefit from wide acceptance and improving toolsets, and have already been suggested as playing an important role in future communication architecture [?]. However, though some initial work has been done in applying ontology-based semantics to communications problems [?], the suitability of these languages for the demanding scalability and real-time requirement of this domain is yet to be proven.

Recently there has been also growing number of propositions to model context for context aware systems in Semantic Web languages like OWL [?]. Although, since the “context-aware” term introduction [?], numerous approaches for context modeling have been proposed in communications society (pervasive computing, peer-to-peer, and so forth), the most popular is seeing context as some function or mode of the parameters of the environment, such as time, place, etc. Successful creation of autonomic communications will require fuller interpretation of context as concept dynamically changing rather than static [?].

5 Novel co-ordination and communications programming models

Situated and autonomic communication calls for novel paradigms of communication and coordination, and consequently calls for novel modeling approaches and for novel supporting infrastructures and programming languages. Traditional paradigms based on approaches such as message-passing, client-server, or distributed shared memory – on which most practice of distributed programming has relied so far – appear inadequate when dealing with the new systems challenges being faced. These include:

**Network dynamics** The mobility, addition and removal of nodes imply a more dynamic treatment of service location, service provisioning and network topology.

**Situation** Situating a network and its services in a context of use increases the need for a well-founded understanding of resource requirements and adaptations.

**Self-configuration** The lack of administration implies that services must manage their own initial and evolving configuration.

**Adaptation and inspection** A network must provide both individual and collective behaviour, which implies maintaining a running view of its own performance against its current and anticipated goals.
In fact, traditional programming models typically rely on static assumptions and *a priori* knowledge about the system (e.g., spatial-temporal coupling [?] and referential awareness [?]) that hardly apply to modern ICT scenarios and will definitely be unrealistic for future scenarios. Also, these models are not situated *per se*, in that they do not intrinsically account for context-aware interactions [?]. It is also rather hard (both conceptually and in practice) to map mechanisms such as self-organisation and self-adaptation onto models that do not accommodate such notions in their basic premises.

Perhaps more importantly, the above models rely on a traditional layered perspective of communication systems. This typically prevents the communication medium from adapting to network and application dynamics. On the one hand, the layered architecture makes the higher layers blind with regard to underlying changing conditions (such as a bandwidth reduction caused by a network glitch); on the other, lower layers are unaware of the kinds of services in which they are involved, and so they cannot customise their activities accordingly (for example by the transport layer switching autonomously between TCP and UDP depending on the application it is supporting). The vision of autonomic communication affects both the lower network layers – calling for programmable components capable of adapting their behavior in concerted way to provide autonomic features – as well as the higher (from transport to application) layers, in that software components have to interact in a very dynamic world. This implies re-thinking the traditional layered network models: strong cross layer interactions between application components and network components are required to (i) have applications access and understand low-level information about the situation of the network and *vice versa*, (ii) achieve “cross-layer” tuning of their respective behaviors that enables services to adapt to the current network characteristics, (iii) conversely to have the network understand the current needs of services and adapt to them, and thus (iv) to achieve overall an orchestrated activity of the network as a whole.

These requirements provide a compelling case for novel programming and communications approaches, in that they must enable a vision and a programming style in which the distinction between “network” and “service” components is blurred, and in which network components will become an integral part and will interact with those software components that execute on them in a semantic world.

### 5.1 Emerging themes

A variety of research groups are already proposing a number of innovative communication and coordination approaches, and the associated programming languages and infrastructures, more suitable to the needs of future situated and autonomic communication scenarios.
**Event-based models** Distributed event-based approaches (such as Milan [?], or Siena[?]). These models and middleware support and event-based style of programming by providing distributed event-dispatching services in the context of mobile ad hoc networks, in which mobile nodes engage a distributed algorithm to self-organise event-dispatching routes and to maintain such routes despite network dynamics. There is considerable scope for adaptation within such systems, as the event service abstracts most of the network issues. Top-down influences are weak, however, and typically the event service will support only a small number of interaction patterns efficiently.

**Tuple-space models** Distributed tuple spaces (for example Lime and Egospace [?]) exploit transiently-shared tuple spaces as the basis for programming interactions in dynamic network scenarios. Each mobile device, as well as each network node, owns a private tuple space. Upon connection with other devices or with network nodes, the privately-owned tuple spaces can merge in a federated tuple space, to be used as a common data space to exchange information.

Tuple-space models may be encapsulated within higher-level environments, for example within Sun’s Jini framework. There remain issues in providing efficient and fault-tolerant implementations [?], especially in highly dynamic environments, as the semantics of tuple-space interactions require extensive use of synchronisation.

**Field- and morphogen-based models** Field-based approaches (for example Co-Fields, TOTA and MMASS [?]) can be regarded as a general framework to program and engineer co-ordinated behaviors in dynamic and distributed computing systems. The key idea in field-based coordination is to have agents’ actions driven by computational “force fields”, generated by the agents themselves and/or by some infrastructure, and propagated across the environment. Field-based approaches enable the programming of adaptive and effective coordination schemes ranging from motion coordination to routing in dynamic networks. Middleware infrastructures like TOTA [?] allow services to define and propagate field data structures across a dynamic network and to maintain such data structures automatically in the presence of local failures. If one of the nodes supporting the field fails, for example, the field will re-configure to reflect this local change in the global configuration of the field, while local instabilities are “damped” at the global level.

Morphogen-gradient approaches (for example [?]) draw their inspiration from the original works on the amorphous computing project [?]. They propose driving the activities of autonomic components in dynamic networks by means of data structures similar in concept to fields, and define specific gradient-oriented programming languages accordingly. The primary application scenario being addressed is that of pattern formation among mobile robots and computational
particles, but the model also finds useful applications in sensor networks and pervasive computing.

**Biologically-inspired models** There is an increasingly widespread agreement within the community that biologically-inspired (or bio-mimetic) solutions are likely to play a key role in autonomic computing and communication (see e.g. [?]). Indeed, biological complex systems tend to be fully decentralised and rely primarily on local information transfer, which translates into desirable properties such as scalability, adaptability (to changing conditions) and robustness (to partial failure and/or hostile disruption of normal activity).

A historical example of using biologically-inspired models to devise original and efficient solutions to relevant routing and scheduling problems in networks is provided by the so-called swarm intelligence or “ant colony” paradigm [?]. However, many studies in the field are explicitly targeting “toy” problems like, e.g., the Travelling Salesman Problem [?] or the Graph Colouring Problem [?]. Such approaches often incorporate some form of reinforcement learning, as in the distributed neural network approach that has been implemented in a large packet network test-bed [?]; in view of the encouraging experimental results that have been already obtained with this bio-inspired approach, further investigation of the interaction between neural network control and packet networks seems well justified. Other very encouraging results on concrete applications (e.g. server farm management [?]) have been obtained using Monte Carlo simulation techniques but have never been tested experimentally.

So beyond identifying the many similarities between the problems faced by autonomic distributed systems and those already solved by their biological counterparts (see e.g. [?]), there remains a critical need for in-depth, quantitative investigation of the performance of specific bio-mimetic algorithms in a practical deployment scenario. This is made especially challenging by the fact that evaluating the complex system properties of bio-inspired solutions requires a paradigm shift from deterministic to statistical predictions (see for example [?]), which entails accepting some level of uncertainty as far as the behaviour and fate of individual system components is concerned. In practice, using bio-mimetic techniques in artificial systems requires strict evaluation of the cost of trial-and-error approaches to problem-solving, as well as that of “losing” some units in the process of system self-organization, both fundamental features of most biological models, from morphogenesis to collective phenomena. Temporarily increased delays and waste of resources (bandwidth, storage, CPU) must be adequately compensated by improved long-term efficiency and/or responsiveness to unpredictable fluctuations for biologically inspired solutions to outperform conventional, centralised alternatives.

In the context of network-based services, Saffre and Blok [?] describe a bio-mimetic peer-to-peer provision framework (SelfService) based on on-demand service instantiation and featuring emergent load-balancing. However, the high rate of creation and termination of local access points, as well as the demands
on bandwidth incurred by broadcasting requests in the early stages of the system’s evolution toward steady state, dictate that SelfService is only a viable option if fluctuations in demand are genuinely unpredictable and access point creation/termination is relatively lightweight. So in this particular case, the biomimetic approach could be comparatively powerful in a highly dynamic, low security environment, but probably not in a more static resource sharing scenario, or if confidentiality considerations require a participating peer to be taken offline and “scrubbed” of any sensitive data each time that it stops hosting a particular service.

Stigmergic approaches (for example Anthill and SwarmLinda) rely on stigmergic coordination derived from interactions in insect colonies to drive the activities of autonomic application components in dynamic networks. As an example of this class of approaches (based on artificial ants), Anthill supports the design and development of adaptive peer-to-peer applications by relying on distributed mobile components (“ants”) that can travel and can indirectly interact and co-operate with each other by leaving and retrieving bunches of information (to act as synthetic pheromones) in the visited hosts. The key objective of anthill is to build robust and adaptive “semi-structured” networks of peer-to-peer services by exploiting the capabilities of ants to re-organize their activity patterns accordingly to the changes in the network structure. As another example, SwarmLinda is an ant-inspired system to program distributed application components that can adaptively coordinate with each other. Application components on the Internet can access a global distributed tuple space that is realized by set of independent local tuple spaces to retrieve and deposit information. Swarms of ant-agents that represent tuples or templates roam across the network of spaces performing a kind of foraging activity that create routes to guide application components in accessing the proper tuple space location.

Probabilistic and metabolic approaches Epidemic and probabilistic approaches, while not directly related to programming, aim at overcoming the burden related in maintaining global data structures such as routing tables and pheromone paths over dynamic networks, as it may be required in stigmergic and field-based approaches, and thus may have impact on nearly all models presented above. They propose to rely on epidemiology theories to provide a “probabilistic guarantees” that data structures and routes will be eventually maintained.

Metabolic approaches such as Fraglets apply a chemical execution model to the implementation of communication protocols. One guiding question is the creation of robust execution circuits which can distribute over a dynamic network and which continue their service despite parts of the implementation being knocked out. Like packets that can be lost (which can be recovered by the appropriate protocols) it is possible to envisage an environment where parts of a protocol’s execution can be lost. The remaining implementation elements should
continue to operate and be able to recover by themselves for restoring full services again.

**Structurally-based approaches** A general concern for decentralised programming is the ability of designers to predict – and more importantly, guarantee – that a service will exhibit and maintain certain desirable properties over its lifetime regardless of any adaptations it may make. One approach is to encode explicitly the context to which a system will adapt, and to derive its adaptive behaviour in a way that respects both this structure and the overall goals of the service. Early research on using category theory to describe adaptive behaviour [?] suggests that such approaches may combine adaptivity with stronger guarantees on the adaptive “envelope” of systems, although this remains to be demonstrated in larger cases and may not provide sufficient dynamism for many applications.

Spatial and environment-based approaches, of which approaches based on overlay networks are a specific case [?], propose to exploit spatial and environmental abstractions as a primary mean to drive components interactions. In these approaches, spatial concepts are realised by means of self-organising and self-adapting overlay data structures that provide components with context information suitable for driving and co-ordinating their activities. Overlay data structures are distributed data structures that generalize the idea of overlay networks [?]. Overlay networks provide distributed routing management, providing components with a suitable application-specific or network-specific view of the network (for example providing the perception of a specific, application-specific overlay topology of the network). To some extent, spatial approaches can be considered as a specific instance of the general concept of semantics-oriented autonomic communications, in which the adaptive space is limited to physical (metric) spaces.

### 6 Trust and security

Autonomic communications offers an open environment for rapid and dynamic resource integration where federations of heterogeneous systems are formed with no central authority and no unified security infrastructure – a similar situation as that which pertains in pervasive and ubiquitous computing. The architecture of autonomic systems in general considers them as consisting of autonomic elements, each performing a fixed function and interacting with other elements, possibly in a very dynamic environment. An autonomic element is commonly viewed as comprising two units: a functional unit that performs the element’s basic function, and a management unit that controls the functional unit’s configuration, inputs and outputs. In this architecture, security tasks are performed by the management unit governed by the element’s policies, of which security policies are a subset [?].
Setting aside the standard properties of integrity, confidentiality and authentication residing on message and network levels and assuming the availability of standards and protocols for that (e.g. WS-Security, SSL etc.), new security challenges are placed ahead by autonomic communication.

6.1 Identity management

Essentially all security properties and services – integrity, confidentiality, authentication, trust, reputation – hinge on identity. There is hardly any point in encrypting communication if we are not sure who we are talking to. While in a static scenario digital identity management does not present much of a problem, it emerges as an important issue when autonomic nodes dynamically join different alliances.

A very widespread technique of identity management is the Single Sign-On (SSO) mechanism. The main idea behind SSO is to eliminate the need of storing and remembering multiple users’ IDs and passwords for each online service by pushing the burden onto a trusted identity provider (IP) for a primary authentication that is then forwarded to the partners offering the desired services.

OASIS Security Assertion Markup Language (SAML) is an open XML-based security standard that provides a way of exchanging user authentication information. SAML on its own is the most widely used standard for bilateral identity management. It offers one-off SSO relationships in which two partners establish an SSO with each other.

Microsoft .Net Passport is a proof-of-concept user identity management infrastructure. It takes a centralized approach and associates a unique ID with every user (mapping this ID to the user’s personal profile) that is used for signing in and accessing all online services also part of the .Net infrastructure. All users’ personal data is centrally stored on Microsoft servers which play the role of IDs.

The Liberty Alliance project and WS-Federation take a decentralized approach for cross-domain identity management. It enables a multilateral federation of partners sharing the same domain (circle) of trust. Each federation supports multiple identity providers and within a federation (circle of trust) a user may traverse all involved partners’ services with a single authentication. Liberty’s specifications are based on SAML standard and extend it with a number of protocols and features enabling multilateral identity management while WS-Federation relies on WS-Trust, WS-Policy, WS-SecureConversation etc. for describing trust relationships and policies of entities in a federation.

In a single federation each service provider is responsible for the management and enforcement of its own security policies with a high degree of autonomy. Hence, for many services no partner may guess a priori what will be sent by clients and clients may not know a priori what credentials are demanded for completing a service, which may require the orchestration of many different autonomic nodes.
The work on interactive access control [?,?] proposes that servers should be able to get back to clients asking for missing credentials, whereas the latter may decide to supply or decline the requested credentials and so on until a final decision is taken. One may also analyse the causal dependencies between software agents using techniques such as versioning vectors [?].

Though there are a number of industrial proposals for identity solutions, such as those described above, they do not cater well for dynamic autonomic scenarios. Therefore a research challenge is to adapt and apply the existing technology to the case of autonomic communication.

6.2 Trust management and negotiation

Trust is an important aspect for making decision on security in information systems, particularly influencing the specification of security policies. Trust management is an approach to manage distributed access control by combining policies, digital credentials and logical deduction.

Since in an autonomic network there are multilateral communications among self-managing and self-preserving partners, there is a pressing need of suitable models/schemes for establishing and maintaining trust relationships between those partners. The highly dynamic nature of autonomic systems requires novel dynamic trust management models [?,?,?,,?] for establishing trust relationships and managing access rights.

Capability-based systems approach distributed authorization by basing their access decisions on user’s capabilities (access rights) expressed as digital credentials. So, management of credentials emerges as the key issue for a distributed authorization framework, and credential-based access control [?,?,?,?,,?] becomes a suitable model for a trust management system.

A number of frameworks have been developed for designing trust management systems. They mainly focus on different aspects of trust by adopting different notions of trust relationships, and so implementing different mechanisms for propagating trust and deducing new security statements. The key focus of these proposals is usually the policy and credential language as in KeyNote, PolicyMaker and REFEREE [?]. A number of later proposals have refined the languages used for policies, single credentials or hierarchies thereof and for their evaluation [?,?].

Extending the concept of trust management from the level of transactions and message exchange to the context of semantics of communications opens up approaches to establish trust by agreeing on an upper layer ontology. The challenge is to find a level of abstraction that encompasses the range of concepts used by different mechanisms to establish trust, but which captures enough semantics to usefully support interoperation between those different approaches at a later date.
The last few years have seen the emergence of a new concept in trust management methodology called trust negotiation [?]. It enables iterative disclosures of credentials between a requester and a provider in order to establish the necessary level of trust allowing exchange of data. This makes it particularly suitable for autonomic communication systems.

Further, digital reputation methods (for preliminary results and related work see [?]) will be used for the continual self-monitoring of autonomic entities (AEs) and the services they provide. This system will rely on the dissemination, throughout the network, of trust information gathered through transactions between AEs. In this way AEs can build knowledge about the behaviour of other AEs (with whom they may never have interacted before) and the available services. This information can then be used to decide whether to interact with another AE and whether to continue supporting an existing service or to start supporting a new service.

**Does reputation management imply self-monitoring, or exchange of observed trust with others?**

This self-monitoring is done at two levels. The first level monitors long-term actions such as the offering of new services and the discontinuing of unwanted services. The services will need to be continually evaluated for the utility they provide and the resources they consume. The second, short-term level is more reactive with a shorter response time that will incorporate mechanisms to realise services, improve them (for example by appropriating more resources for services that are more popular), identify misbehaving AEs, modify resource allocation etc.

**Intrusion tolerance?**

### 6.3 Self-protection and self-healing

Recent work in security management has revealed the necessity of designing a new generation of self-adaptive security solutions. In this context, these solutions can be based on multi-agent systems and intelligent agent technology. An example of this approach is the work of Gelenbe et al. which proposes and evaluates a scheme for denial of service detection and defence based on a self-healing autonomic approach using the Cognitive Packet Network paradigm [?].

**References?**

Biological models of resilience follow the analogy with nervous and immune systems in biological organisms. A nervous system provides sensing (problem detection) and self-protection through reflexes (autonomic responses). An immune system is responsible for anomaly detection (“self” versus “non-self”) and self-healing. Effective artificial immune systems have been developed which model very closely elements and processes of natural immune systems, such as lymphocytes, their generation, maturation, circulation, binding to pathogens and activation, as well as both primary and secondary immune responses [?].
Research on traditional intrusion detection systems in the context of autonomic communications is looking at knowledge representation of service providers, consumers, services and threats. Object-oriented and ontology-based models have been proposed [?].

6.4 Self-organized public key management

In general, the usage of public key cryptography requires the presence of a centralized certification authority. However, such an authority (and the infrastructure it requires) is incompatible with decentralized nature of autonomic communications. To solve this problem, some authors propose an approach similar to PGP in the sense that users issue certificates for each other based on their personal acquaintances. However, unlike the PGP, certificates are stored and distributed by the users themselves in a completely self-organized fashion. When two users want to verify the public keys of each other, they merge their local certificate repositories and try to find appropriate certificate chains within the merged repository that make the verification possible. The success of this approach depends very much on the algorithm for the construction of the local certificate sets and on the characteristics of the certificate graph, i.e. a graph whose vertices represent public-keys of the users and the edges represent public-key certificates issued by the users. The analysis of the two typical algorithms shows that even a simple construction algorithm can achieve high performance. Moreover, the certificate graph exhibits small-world features, so that good scalability of the approach is obviously expected.

7 Evaluation and testing

Autonomic networks open a new area of investigation for experimental network evaluation because, contrary to research on conventional IP networks, it is no longer sufficient to interconnect systems via existing wired, optical and wireless modalities, and to measure the effect of carrying perhaps novel traffic flows in the presence of incremental changes in the protocols. In autonomic networks, both the user’s perception of context awareness, and the lower level network perception of autonomic access and management of resources need to be addressed using novel approaches, and the methods used to evaluate them need to be sufficiently pragmatic and empirical to be convincing.

Much of European academic research in computer and communication networks has been based in the past on the use of theoretical tools such as queueing models, software models and studies of protocols, and simulation studies. Networking research in industry and at telecommunications operators has used similar approaches but, in addition, it has widely benefited from experimental facilities and the possibility to test experimental networks and measure networks which are already deployed. These differences have been justified in the past by the high
costs of hardware related to experimental networks and to the need for software and system development when one builds or modifies experimental systems. However, this balance is now modified by the advent of open system platforms and the possibility we have of using low cost off the shelf hardware to design and interconnect network routers. Thus, the threshold of resources needed to conduct meaningful research on network evaluation using test-beds is now definitely lower, probably by an order of magnitude, than a decade ago. Several recent projects in Europe consider low cost wireless test-beds [5,6], or specific fiber optics networks with local connectivity, that could support experimentation on adaptive autonomic networks. Other work examines multimedia traffic [7], while wireless power management and QoS are investigated in some existing projects from the autonomic perspective [8,9]. Precise traffic measurement on long haul networks is considered in [10]. An ambitious project is currently planning a large regional wired/optical and wireless test-bed that would support a variety of social activities [11]. Finally, usability issues for autonomic network in Europe could be developed as an extension of work concerning virtual environments[12]. On the other hand the sheer size of US research in this area, with the possibility of interconnecting hundreds of nodes, and the sophistication of some of the Japanese test-beds with respect to context awareness and usability of autonomic networking environments of some experimental US and Canadian projects [13,14] should guide us to a higher level of ambition.

The Ubiquitous Home [15] project in Japan has created a wireless home where context aware services are offered dynamically and ubiquitously. The “self-aware network” project at Imperial [16] investigates how lower level networking functions can be implemented using context-aware measurement based techniques combined with adaptive neural network based reinforcement learning. Between these two extremes at the high (application) and low (network) levels, there remains very substantial work to be done in all areas of the performance evaluation of autonomic networks, from services, to protocols, and from robust networking to QoS.

8 Conclusion

... 

Does autonomic systems constitute a discipline? Probably not: there are as yet few techniques deriving directly from the study of autonomic regulation of computing or communications. There is, however, a considerable body of knowledge arising from the ways in which techniques from different disciplines can interact to provide emergent properties and other autonomic features. This in turn may generate insights that would not arise from the individual disciplines in isolation.
8.1 Emerging trends and expectations

Decentralisation is the *sine qua non* of autonomic systems: any centralised resource or control point will act as a brake on a system's ability to adapt, especially in terms of robustness of performance. While this statement is hardly contentious, it is important to understand that decentralisation is poorly understood at the algorithmic, analysis and programming levels. New techniques are needed to understand the exact robustness, performance and complexity characteristics of decentralised algorithms. Moreover the existing repertoire of programming techniques must be significantly enhanced to escape from the tendency—sometimes very well-concealed—to inadvertently introduce a centralised element into even the most distributed computation. Even superficially innocuous techniques such as iteration can conceal problems which almost demand a centralised resource of some kind.

Indeed, one might reasonably argue that decentralisation is a goal worth pursuing even at the cost of performance. Although many people advocate (for example) peer-to-peer solutions on scalability grounds, such solutions are often intrinsically more able to withstand adaptation, treating service relocation and other changes as failures to be recovered from.

The traditional layered models of networks, while valuable conceptually and pedagogically, were never realistic implementation strategies, and this is even more true for autonomic systems. Layering breaks-up the holistic nature of context and reasoning, whereas valuable strategies can derive from the ability to correlate (for example) traffic-level properties against the applications that generate that traffic. A more integrated model of network monitoring and control need not generate awkward dependencies and loops—although that is of course a significant danger that must be avoided.

8.2 Research priorities

What then are the remaining research priorities for autonomic communications?

Classical (Shannon) information theory has provided an excellent basis for modelling and reasoning about slowly-changing networks. Autonomic systems do not however respect the Shannon view of communications on uninterpreted channels: in an autonomic system the message affects the medium so as to improve the latter's ability to function. There is little understanding of the impact this will have on information-theoretic properties, and in particular on the limits of the channel in terms of information flow and robustness. Understanding these issues will require significant foundational research.

Many autonomic systems have only weak guarantees on the properties they present. They may not, for example, be able to guarantee delivery of messages
under all circumstances, or be able to bound end-to-end properties such as latency or security. In part this is derived from the inadequate modelling mechanisms mentioned above; in part it derives from the increased dynamism in the protocols and management approaches being utilised. However, users’ satisfaction with a system derives largely from such end-to-end properties. De-layering and other techniques must be improved to allow better end-to-end guarantees.

Most systems can be both modelled and evaluated, with the latter being considerably more popular than the former at enterprise scales. Autonomic networks and services are unusual in being resistant to both approaches. Without the ability to evaluate systems in a controlled manner, developers will find it hard to fine-tune adaptive strategies and service offerings. In tandem with improved modelling, it is vital that improved ways of measuring and evaluating autonomic systems in controlled environments be found.

For any adaptive system there is an issue of stability: will adaptations converge to a steady-state, or will the system’s behaviour oscillate uncontrollably? While stability is a well-understood property of traditional control theory deriving from the basic differential equations, it is significantly less well-understood in the general case of autonomic systems. Without stability guarantees (and methods of obtaining them) adaptation carries too high a risk to be used on a wide scale over critical infrastructure.

Finally, there is the issue of complexity. In principle an adaptive system may be significantly less variable to a user’s eyes than a traditional, non-adaptive system, as the adaptations will be used to mask otherwise-significant observable differences. It is vital that this paradox be carried over into the development and management domains so that those who develop for, deploy and maintain complex networks can focus on the value-add that their activities bring without being consumed by the complexity of the mechanisms that underlie them. It is only in this way that autonomic communications systems can become leveraged partners in delivering the next generation of pervasive and reliable services.

Acknowledgements

The authors acknowledge the members of the ACEnet and ACCA European project consortia, many of whose ideas have significantly informed the understanding of the issues presented here.

About the authors

Simon Dobson’s research centres around pervasive computing and advanced software technology, addressing both theory and practice and being supported by an extensive record of published work (including papers in CACM, JPDC, EHCI and ECOOP) and a direct involvement in grants worth around €7M. He has been closely involved with developing the Irish and European-level research
initiatives in autonomic and pervasive systems. He is a chartered engineer and member of the BCS, IEEE and ACM.

Spyros Denazis . . .

Antonio Fernández Anta received his PhD in Computer Science from University of SW Louisiana in 1994 and his MS in Computer Science from Universidad Politécnica de Madrid in 1991. He is currently Associate Professor at Universidad Rey Juan Carlos. He has been visiting researcher in several institutions including MIT, Bell Labs, University of Connecticut, and University of Cyprus. He has received several academic and research awards, including a Fulbright grant. He is Senior Member of the IEEE and member of the ACM and EATCS. His research effort is concentrated in distributed and parallel computing, communications and algorithms.

Dominique Gaïti is a researcher and full professor at the University of Technology in Troyes (UTT) in France and conducts researches in the CNRS lab ISTIT (scientific institute of information technology in Troyes). She received her PhD from the University of Paris 6 in 1991 and the “habilitation à diriger des recherches” from the University of Paris 9 in 1995. Her main interests are on network control.

Erol Gelenbe is the Dennis Gabor Chair and Head of Intelligent Systems and Networks at Imperial, where he conducts work in smart computer networks, stochastic modelling and data fusion. Prior to this, he held an endowed chair at Duke University and a University Chair in Florida. A Member of Academia Europaea, and Fellow of IEEE, ACM and IEE, he has received the IFIP Silver Core. His scientific awards include the Grand Prix France Telecom of the French Academy, the Parlar Foundation Award of Turkey, the Doctorate Honoris Causa in “Ars Computandi” from the University of Rome, the PhD Honoris Causa from Bogazici University (Istanbul) and the Doctor of Science Honoris Causa from the University of Liège (Belgium). His honours include the Chevalier and Officier de l’Ordre du Mérite of France, and the Palmes Académiques.

Fabio Massacci received an MSc and a PhD in Computer Engineering from the University of Roma “La Sapienza”. In 2001 he joined Trento as associate professor and was habilitated as full professor in 2003. In 2001 he was awarded the “Intelligenza Artificiale” prize by the Italian Association of Artificial Intelligence (AI*IA) for young researchers in AI under 35. His main research interests are in the area of automated reasoning and formal verification, network and computer security, and security requirements engineering.

Paddy Nixon holds a BSc, MA and PhD in Computer Science, is a chartered engineer and a member of the BCS, ACM and IEEE. He leads the Systems Research Group at UCD Dublin supported by a €2.5M Science Foundation Ireland
endowment. Previously he led the Global and Pervasive Computing Group at Strathclyde University and was research Director of the Kelvin Institute.

*Fabrice Saffre* is a Senior Research Scientist within BT’s Pervasive ICT Research Centre, a Member of the Institute of Physics and a Member of the British Computer Society. He holds a BA and MA in Philosophy of Science and a postgraduate degree in Computer Science from the Université Libre de Bruxelles, a post-graduate degree in Neuroscience from the Université de Nancy I and a Science PhD in Theoretical Biology from the Université Libre de Bruxelles. His primary areas of expertise are modelling and analysing complex systems, especially using numerical techniques (Monte Carlo simulations and cellular automata), and the study of collective phenomena.

*Nikita Schmidt . . .

*Michael Smirnov* is a Direktorium member of Fraunhofer FOKUS. His research interests include advances in Internet services, distributed and autonomic communication, self-organisation, scalable group communication, policy based networking, etc. He has authored 9 books and over 70 reviewed paper. He is a founding member of the Autonomic Communications Forum and has been closely involved in research funding programmes for autonomic communications.

*Franco Zambonelli* has been a professor in Computer Science at the University of Modena and Reggio Emilia since 2001. He obtained the Laurea degree in Electronic Engineering in 1992, and the PhD in Computer Science in 1997, both from the University of Bologna. His current research interests include distributed and pervasive computing, autonomic computing and communication, software engineering for large-scale agent systems. In these areas, he has published over 130 papers in international fora, co-edited 7 books, and received several best paper awards. He is a member of IEEE, ACM, AIIA, and TABOO.