# A Survey of Autonomic Communications

SIMON DOBSON
UCD Dublin, IE
SPYROS DENAZIS
University of Patras, GR and
Hitachi Research Europe, FR
ANTONIO FERNÁNDEZ
Universidad Rey Juan Carlos, ES
DOMINIQUE GAÏTI
Université de technologie de Troyes, FR
EROL GELENBE
Imperial College London, UK
FABIO MASSACCI
Università di Trento, IT
PADDY NIXON
UCD Dublin, IE
FABRICE SAFFRE
BT Group plc, UK
NIKITA SCHMIDT
UCD Dublin, IE
and
FRANCO ZAMBONELLI
Università di Modena e Reggio
Emilia, IT

Autonomic communications seek to improve the ability of network and services 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.

A. Fernández is partially supported by the Spanish MEC under grant number TIN2005-09198-C02-01, and by the Comunidad de Madrid under grant number S-0505/TIC/0285. E. Gelenbe, F. Massacci, F. Saffre, and F. Zambonelli wish to acknowledge the CASCADAS (IST-027807) Project funded by the Future and Emerging Technologies Programme of the European Commission. E. Gelenbe's work was supported by grants from EPSRC (UK) GR/S52360/01 and EU FP6 SAPAD MIRG-CT-2004-506602 on "Self-Aware Networks and Quality of Service". F. Saffre acknowledges the EPSRC grant EP/D003105/1. N. Schmidt is funded by Science Foundation Ireland under grant 04/RPI/1544 on "Secure and Predictable Pervasive Computing". S. Dobson, S. Denazis and P. Nixon are partially supported by the ACCA co-ordination action on autonomic communications (IST-6475) funded by the Future and Emerging Technologies Programme of the European Commission. S. Dobson and P. Nixon are also partially supported by Science Foundation Ireland under grant number 03/CE2/1303-1, "LERO: the Irish Software Engineering Research Centre".
Author's address: S. Dobson, Systems Research Group, School of Computer Science and Informatics, UCD Dublin, Belfield, Dublin 4, Ireland; email: simon.dobson@ucd.ie.

Categories and Subject Descriptors: C.2.1 [**Computer-Communications Networks**]: Network Architecture and Design

General Terms: Algorithms, Design, Management

Additional Key Words and Phrases: Autonomic communication

## 1. INTRODUCTION

Modern network infrastructures have achieved a small miracle in presenting a simple and uniform fa cade to applications. In many ways, the programmer's view of the network has become simpler over time with TCP/IP acting as a 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 article is to reduce this learning curve by presenting a comprehensive survey of the current state of research in autonomic communications. We motivate both autonomic communications and our approach to the literature in Section 2, and then address the five interlinked perspectives of the design and analysis of decentralized algorithms (Section 3); the modeling, handling and use of context (Section 4); novel and extended programming approaches (Section 5); issues and approaches for addressing security and trust (Section 6); and systems evaluation and testing (Section 7). In Section 8, we highlight some of the emerging trends with a view to informing the evolving research program 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. An infrastructure to reduce these costs and facilitate new opportunities is urgently needed, and it must at the same time be 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 as well as between operators and users be extensively rethought.

*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 multilevel tension between complexity and simplicity, diversity and ubiquity.

Traditional networks have been constructed and coordinated 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, no universally-agreed upon protocols or other technical standards, and 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 decentralized, self-describing, self-organizing, self-managing, self-configuring, and self-optimizing, 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 Autonomic Systems

The notion of using autonomic techniques—of deploying technology specifically to manage and optimize the functioning of other technology on an ongoing 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 nonlinear 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 characterized mathematically. Control theory encounters even 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 generalize the control-theoretic view of control by enabling more flexible and adaptive functions in the underlying system. By leveraging 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, preempt expected future demands.

An autonomic system offers an open environment for rapid and dynamic resource integration where federations of heterogeneous systems are formed with no central authority or unified infrastructure, a similar situation to that which pertains to pervasive and ubiquitous computing. The architecture of autonomic system, 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 being comprised of one or more *managed elements* (also referred to as *functional units*) that perform the element's operational function and an *autonomic manager* (*management unit*) that controls the managed elements' configuration, inputs, and outputs.

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 analyzed to construct a model of the evolving situation faced by the network and its services with this model 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 [Kephart and Chess 2003]. Autonomic computing is seen as a way of reducing the total cost of ownership of complex IT systems by allowing reconfiguration and optimization to proceed on an ongoing basis driven by feedback on the system's ongoing behavior. It combines a technological vision with a business rationale for increasing the coupling between business goals and IT services.

*Autonomic communication*, by contrast, generally refers to all these research thrusts involved in a deep foundational rethinking of communication, networking, and distributed computing paradigms to face the increasing complexities

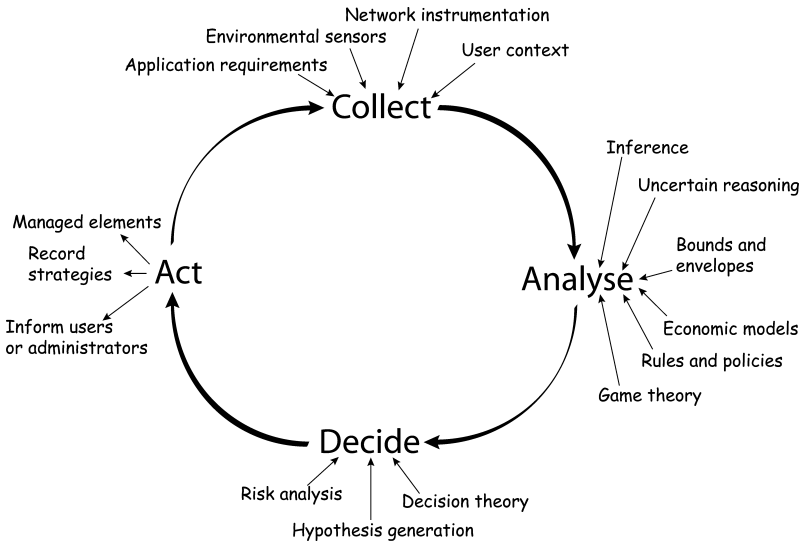


Fig. 1. Autonomic control loop.

and dynamics of modern network scenarios. The ultimate vision of autonomic communication research is that of a networked world in which networks and associated devices and services will be able to work in a totally unsupervised manner, able to self-configure, self-monitor, self-adapt, and self-heal—the so-called *self-* properties*. On the one hand, this will deliver networks capable of adapting their behaviors dynamically to meet the changing specific needs of individual users; on the other, it will dramatically decrease the complexity and associated costs currently involved in the effective and reliable deployment of networks and communication services.

Despite their evident similarities, there are significant differences between autonomic computing and communication. While autonomic communication is more oriented towards distributed systems and services and to the management of network resources at both the infrastructure and the user levels [Quitadamo and Zambonelli 2007], autonomic computing is more directly oriented towards application software and management of computing resources. Nevertheless, both research areas recognize that traditional software systems are facing a decreasing incremental benefit from technological advances (powerful CPUs, large memories, and so forth) because the complexities of development and management are overwhelming the technical gains. Accordingly, the twin visions of autonomic communications and computing are aligned in identifying the need for decentralized algorithms and control, context-awareness, novel programming paradigms, end-to-end privacy management, and comprehensive evaluation in order to deliver the desired self-* properties.

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 [Clark et al. 2003] 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 [Dobson 2005]. The use of context from beyond the network makes models situated in that they have a model of their place in a wider scheme.

### 2.2 Challenges to Theory and Practice

Networks are traditionally described using a number of theories and technologies, the most influential of which are classical (Shannon) information theory, communication theory, queuing theory and the IP suite. However, such foundational insights must also be accessible to the programmers tasked with generating network services.

What are the core challenges for next-generation networking? Network researchers have to a large extent developed approaches for tackling purely technical challenges such as bandwidth and authentication. We consider that many of the core challenges lie at the boundary between networking and what might traditionally be regarded as systems or applications, including, the following.

*Interaction with strangers*. Authentication allows one to identify users but not their motives and can build secure paths between services without deciding whether the services selected are the most reliable and least prone to information leakage. Next-generation wireless, mobile, and ad hoc networks will need to manage the trust and privacy of users and services end-to-end, without any a priori knowledge of the parties involved.

*Information reflection and collection*. In order to decide between competing demands, the network needs to embody information about the data it is transporting and the uses being made of the data. It must also reflect on its own behavior so as to, for example, recover from—or preferably predict and avoid—routing failures and other events.

*Lack of centralized goals and control*. Decentralization increases the robustness of individual services at a microscale and, as was seen with the emergence of the World Wide Web, encourages new applications and services at a macroscale.

*Meaningful adaptation*. Designing an adaptive system implies being able to gain confidence that the system will adapt correctly to given stimuli, maintain key behaviors and avoid deleterious ones. Designers also need to understand adaptation at a system level so that adaptations that optimize individual services do not cause undesirable interactions with those of other services.

*Cooperative behavior in the face of competition*. Open systems architectures allow agents to join networks dynamically and both offer and consume services. The growth of peer-to-peer services and the withering of centralized control and billing both make it vital that cooperation structures are intrinsically proof against free-riding and other selfish behaviors.

*Heterogeneous services and semantics*. Agents cannot guarantee the existence of particular services or their precise behaviors.

| | Interaction with strangers | Lack of centralised goals and control | Information reflection and collection | "Meaningful" adaptation | Engineering co-operative behaviour | Heterogeneous services and semantics |
|---|---|---|---|---|---|---|
| Analysis and design of autonomic algorithms | | *Self-monitoring*<br>*Self-stabilisation* | *Self-optimisation* | *Computational steering* | *Self-stabilisation*<br>*Game theory*<br>*Econometrics* | |
| Context-awareness and semantics in communication | *Metadata acquisition*<br>*Task and user modelling* | | *Self-description* | *Self-analysis* | | *Interface description* |
| Novel co-ordination and communications programming models | *Uncoupled interactions* | *Novel abstractions*<br>*Negotiation* | *Localised interactions* | *Stigmergy* | *Field-based programming* | *Interface matching and brokering* |
| Trust and security | *Trust formation* | *Policy matching* | *Information visibility and leakage* | | *Algorithm and information disclosure* | *End-to-end trust maintenance* |
| Evaluation and testing | | *Scalabilty testing*<br>*Stability testing* | | *Stability under perturbation* | *Process- versus point-correctness* | |

Fig. 2. Cross-cutting issues

It would be attractive if each of these challenges could be dealt with independently, but unfortunately this does not seem to be the case. Routing involves measuring the network, interacting with untrusted nodes, and applying adaptive strategies; service provision must deal with noncooperative behavior, heterogeneity, and a lack of centralized direction. And the same seems to be true for most other communications activities. To borrow a phrase from computer science, the concerns of autonomic communications are *cross-cutting* and must be woven into a coherent solution.

To focus on the cross-cutting concerns is to risk fragmenting the discussion across accepted technical domains and so, for the remainder of this article, we focus on five core domains within the research literature: algorithm design, context collection and modeling, novel programming paradigms, trust, and testing and evaluation. This allows the reader to focus on their area of technical interest while still being exposed to the core cross-cutting concerns of autonomic communications.

Figure 2 presents a road map to the rest of the article, matching the challenges against the cross-cutting issues and showing the technical ideas emerging from each issue when addressing each challenge.

Why these five domains? Although any choice is somewhat arbitrary, we believe that these best capture the current research themes within the evolving body of knowledge.

The characteristic features of autonomic communications are the use of highly decentralized algorithms that have 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 synthesizing results from a range of additional disciplines that are typically pursued independently.

Some of these topics touch on the most advanced aspects of communication dealing with the problem of signal coexistence 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. We address the issues of collecting and modeling context in Section 4. 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 cooperatively. We discuss programming paradigms in Section 5. However, such approaches also raise the importance of being cognizance 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 behaviors 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 is no longer valid. One of the paradigmatic cases of a communication system that has had to evolve to become autonomic and self-organized is the Internet whose initial centralized structure has had to evolve for fully distributed control and management. This trend is slowly expanding to all kinds of communications systems. It has given way to new models and paradigms and has open the door to migrating new research techniques from other fields for use in these systems.

In this context, new algorithms that take into account the complexity of the communication systems have to be developed and analyzed. These algorithms will have to guarantee the emergent properties of the system with limited contextual information and local control, making the system self-organizing in a way that is atypical for most algorithm styles. Similarly, in most cases, they will have to deal with very heterogenous 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 an 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, e.g., Chockler et al. [2001], Défago et al. [2004], and Birman [2005]) which offered very interesting features such as deterministic guarantees, self-repair, load distribution, and flow control are being rethought 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 [Spinglass Project 2005] or the algorithms discussed in Eugster et al. [2003]. A similar evolution has been observed in publish-and-subscribe systems and algorithms (see Eugster et al. [2003]) to yield, for instance, epidemic approaches like Newscast [Jelasity et al. 2003].

One very desirable design objective would be that algorithms for autonomic self-organized communication he 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 behaviors, to sensor networks that have to organize themselves 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.

### 3.1 Extending Classical Design Techniques

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 multihop routing in overlay networks to circumvent Internet connectivity failures such as in Andersen et al. [2002] that profit from 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, e.g., those presented in Balakrishnan et al. [2003]) and to manage ad hoc and sensor networks [Chrobak et al. 2004; Chlebus et al. 2005; von Rickenbach et al. 2005; Tschudin et al. 2005; Àlvarez et al. 2004; Tschudin et al. 2005; Chelius et al. 2005]. Other initiatives include the use of spiked neural networks distributed throughout the network routers which learn from online measurements using reinforcement learning so as to achieve adaptive routing and address the users' QoS needs [Gelenbe 2004a, 2004b, 2004c; Gelenbe et al. 2001, 2004; Gelenbe and Lent 2004; Gelenbe and Nunez 2003]. Genetic algorithms have also been used and evaluated to discover new network paths that offer potentially 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 behavior on nature, which has given way to a collection of new models, systems, and algorithms (e.g., see Section 5 and Babaoglu et al. [2005]).

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 [Cholvi et al. 2005; Rodero et al. 2006]. This seems to be a promising line of research for applying the increasing knowledge on small-world and scale-free networks. Another area to continue exploring is the application of nonlinear (chaotic) dynamics to communication systems, mainly to the lowest levels. Chaotic dynamics provide a way to optimize performance in DS-CDMA-based links used in 3G mobile telephony systems [Setti et al. 2002; Kennedy et al. 2000], 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 [Setti et al. 2004; Rovatti et al. 1998, 2000, 2001, 2004a, 2004b; Mazzini et al. 1997, 1999, 2000, 2001].

Decentralization raises significant consistency challenges which are perhaps encountered most strongly in the area of distributed databases [Milan-Franco et al. 2004; Jiménez-Peris et al. 2002; Baldoni et al. 2004]. Most algorithms for transactions and other classical approaches to consistency rely on extensive barrier synchronization which is difficult if not impossible in a decentralized and low-reliability context.

### 3.2 Collaboration Issues in Algorithmic 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 on their computers to match their desired behavior. This will certainly lead to assumptions of adversarial behavior when designing the algorithms. We have examples of this need in the Berkeley Open Infrastructure for Network Computing (BOINC) [BOINC Project 2006], used to perform scientific computations with high CPU time requirements in the computers of volunteers, and which had to introduce security mechanisms to prevent deception caused by (possibly malicious) wrong solutions 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 [Fernández et al. 2005, 2006].

The problem becomes harder when there is to be a continuous multiway collaboration and we want to have all users collaborating in the overall system (e.g., a service like the BOINC system without central control). In these models, it seems natural to introduce game theory into the design of any algorithm that it is suitable to tamper with. A well thought our 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 [Lücking et al. 2004]. An example of game theory in action is the incentives to collaborate included in the BitTorrent system [Cohen 2003]. Interestingly, BitTorrent can still suffer from free riding [Qiu and Srikant 2004], which attests to the difficulty of designing selfishness-proof

algorithms. Game theory is also being considered in most aspects of communication systems (see, e.g., Lücking et al. [2004], Laoutaris et al. [2004a], and Siris and Courcoubetis [2004])

Game theory can also be applied to enhance existing protocols. An example is extending the distributed replication group [Leff et al. 1993] to the case that individual nodes act selfishly, catering to the optimization of their individual local utilities. Game theory may be used [Laoutaris et al. 2004, 2004b, 2005] 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. Such approaches do not suffer from potential mistreatment problems inherent in centralized strategies that aim at optimizing 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 can 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 equitable manner [Siris and Courcoubetis 2004; Siris 2002; Siris et al. 2002].

### 3.3 Limits

Of special interest to 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 [Mestre et al. 2003; Madueño and Vidal 2005]. 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 [Luby 2002, 2003] and Network Coding [Ahlswede et al. 2000]. These are already influencing the next generation of algorithms for peer-to-peer content distribution [Byers et al. 2004; Gkantsidis and Rodriguez 2005]. Furthermore, it is possible that they will influence the future evolution of the Internet and one of its main protocols, TCP [Byers et al. 2002; López et al. 2005].

### 3.4 Stability and Reliability

Autonomic systems strive to provide self-managing and self-optimizing services. The most basic service any network can provide is communication between nodes with traffic management and routing providing the most visible target for decentralized algorithms. A basic question for such systems is the extent to which it is possible to maintain a given quality of service in the face of

changing conditions. This essentially characterizes the stability of the system under perturbation, and often, specifically, its reliability.

An algorithm designer can adopt two basic positions in the face of a given quality requirement. On the one hand, an algorithm may adapt its behavior to achieve the best performance available under the prevailing circumstances. This may involve trading off conflicting requirements at both network and application/user levels, for instance, reducing sound fidelity (at application level) to maintain video frame rate (a user preference) in the face of packet loss (a network concern). On the other hand, an algorithm may fail in a controlled fashion if the required quality of service cannot be maintained. Both of these approaches might be termed reliable: the former focuses on best-effort behavior no matter what the circumstances, while the latter will not display an undesirable behavior.

Somewhat orthogonally to these two approaches, a designer need to understand how a given stimulus will affect an algorithm's behavior. Adaptation is not an excuse for incorrectness: one wants an algorithm's behavioral "envelope" to be well-defined, regardless of the possible conditions that occur in the network [Dobson and Nixon 2004]. Allowing free adaptation within an envelope provides a more precise notion of reliable behavior, while placing clear limits on the degree to which self-management is allowed to weaken (or compromise) service guarantees.

### 3.5 Summary

Traditional algorithms tend to require steering or placement in order to behave optimally in a given configuration. The challenge of autonomic self-management and self-optimization is to provide these optimizations automatically and with dynamic self-reconfiguration as the platform changes. It is also not possible to assume that all agents will behave cooperatively. Recognizing these constraints explicitly in algorithm design leads to a stress on algorithms, which are inherently self-stabilizing, in drawing on insights from other self-stabilizing systems in games, economics, and biology, and also drawing on cross-layer requirements and context information necessary to address concerns across the complete system.

## 4. CONTEXT AWARENESS AND SEMANTICS IN COMMUNICATION

Autonomic communications implies a stronger degree of self-management and self-optimization than is found in conventional networks which are divorced from human intervention (and, in many cases, from the possibility of such intervention). To provide self-management and optimization capabilities, it is necessary to investigate the context-aware approach to improve networking properties. Software entity, network components, and 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 as a result of user and service mobility [Coutaz et al. 2005].

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 attributes such as power saving and cost. Context dissemination can be achieved in both "pull" and "push" modes.

In order to provide self-adaptive behavior, an autonomic system must be able to reason about both its context and its behavior 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 behavior in some sense and generate feedback as a result [de Castro et al. 2004]. 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 [Pujolle et al. 2004]. 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 the expression of adaptive behavior and the changing context that drives such adaptation. As context-awareness and semantic-based reasoning concerns adaptive, networked systems, it requires research about models and languages for representing their behavior 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 optimization problem, optimization always occurs relative to some environment or property: one optimizes 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 optimize requires that the optimizing agent understands the relative priorities that should be given to the various possible optimization targets which, in

turn, implies that the optimizing agent is aware of the meaning of the information within the network and its place in the ongoing user tasks. In an autonomic system, the optimizing agent is software, implying that metadata about the meaning of information, streams, and operations is injected into the network infrastructure and used to inform network-level decisions [Merghem et al. 2003].

Adaptive pervasive computing systems attempt to provide information and services that are distraction-free for users. Approaches to adaptability may be loosely classified as *closed-adaptive* when all adaptations are prespecified and *open-adaptive* when new adaptations can be discovered [Oreizy et al. 1999]. Representations of context have varied between subsymbolic uses of neural networks to more traditional symbolic AI [Henrickcon et al. 2002] with an apparent consensus favoring concept graphs modeled (at least externally) using the Resource Description Framework (RDF) [Lassila and Swick 1999]. 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-contextualized 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 sensorized 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 recognized that managing the structures of context is a significant challenge [Coutaz et al. 2005]. 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) [Gonzalez and Ahlers 1999]. This frequently involves combining information at different semantic levels. The trails model [Driver and Clarke 2004] provides one such fusion, based on the insight of the nonhierarchical 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 analyzed a priori for conflicts and other properties [Jensen and Milner 2003; Dobson and Nixon 2004].

As observed in Section 3, context-driven adaptation must be carefully controlled in order to generate intelligible behavior: context has a direct influence on users' ability to form well-founded conceptual models of systems, so adaptive behavior 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 [Dobson and Nixon 2004]. The communications industry is accustomed to defining system semantics in a formal or semiformal 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 standardization 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 behavior, though development on a Semantic Web rule Language may provide a suitable starting point [de Bruijn et al. 2005].

These technologies benefit from wide acceptance and improving toolsets and have already been suggested as playing an important role in future communication architecture [Clark et al. 2003]. However, though some initial work has been done is applying ontology-based semantics to communications problems [de Vergara et al. 2004; Lewis et al. 2005], the suitability of these languages for the demanding scalability and real-time requirements of this domain is yet to be proven.

Recently there has been a growing number of propositions to model context for context-aware systems in semantic Web languages like OWL [Strang et al. 2003]. Although, since the introduction of the context-aware term [Weiser 1991], numerous approaches for context modeling have been proposed in communications society (pervasive computing, peer-to-peer, and so forth), the most popular of which is viewing 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 a dynamically-changing concept rather than static one [Sterritt et al. 2005].

### 4.1 Summary

Context models provide explicit representations of concerns from a number of different semantic levels. While not all algorithms require explicit context modeling in order to exhibit self-managing behavior, a context model can uniformly inform autonomic decision-making across the spectrum, allowing whole-system self-optimization. It also provides a useful approach for open-adaptive behavior and collaboration across tools through standard formats and protocols.

## 5. NOVEL COORDINATION AND COMMUNICATIONS PROGRAMMING MODELS

The core challenges of autonomic communications calls for novel paradigms of communication and coordination and, consequently, calls for novel modeling approaches and novel supporting infrastructures and programming languages.

### 5.1 Inadequacy of Traditional Models

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 core challenges being faced.

First, traditional programming models typically rely on static assumptions and a priori knowledge about the system, that is, spatial-temporal coupling [Cabri et al. 2000] and referential awareness [Tanenbaum 2004]): components are assumed to live in the same network at the same period and are assumed to know each other. Such assumptions can hardly apply to modern and future network scenarios where interaction with strangers is the norm and where the dynamics of the network (due to the presence of mobile and ephemeral nodes) and its lack of centralized control make the adoption of any a priori strategy useless. Rather, suitable programming and coordination models must assume a dynamic treatment of components' identification and location and must pay attention to overall collective behaviors and coordination rather than individual behaviors and pairwise interactions. While Web services provide a partial approach to this problem, they still rely to some extent on shared assumptions about partners' semantic capabilities in a manner that is in some senses very similar to that of traditional distributed middleware [Baker and Dobson 2005].

Second, traditional models such as message-passing or distributed shared-memories do not account in any way for context-awareness and meaningful interactions [Mamei and Zambonelli 2004]. Put simply, components are assumed to live in a void, where the only things that exist are the other components (or some shared portions of memory). Such models do not enforce per se any form of context-aware computing or context-aware interactions nor do they account for the presence of infrastructures to support context-awareness (as from previous section). Clearly, this makes it very hard to enforce reflective behaviors via analysis of current context. Also, it prevents (both conceptually and in practice) the enforcement of distributed applications, and it services all those properties of self-configuration, self-adaptation, self-healing that by definition (having to rely on phenomena of adaptive self-organization) are need the modeling of components situated in some environments and capable of reacting to its properties.

Third, and again strictly related to the adaptation issue, the previously discussed traditional 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 customize their activities accordingly (e.g., 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 a concerted way to provide autonomic features—as well as the higher layers (from transport to application), in that software components have to interact in a very dynamic world. This implies rethinking the traditional layered network models: strong cross-layer interactions between application components and network components are required to: (1) have applications access and underztand low-level information about the situation of the network and vice versa; (2) achieve cross-layer tuning of their respective behaviors that enables services to adapt to the current network characteristics;

(3) conversely, to have the network understand the current needs of services and adapt to them, and thus achieve overall orchestrated activity of the network as a whole.

The previous considerations provide a compelling case for novel programming and communications approaches in that they must enable a vision and a programming style in which components must be able to interact in dynamic scenarios, where the distinction between network and service components is blurred, and where network components will become an integral part and interact with those software components that execute on them in a semantic world.

### 5.2 Uncoupling Coordination

Although not explicitly conceived to face the challenges of autonomic communications, some well-assessed coordination paradigms exist that, by enforcing uncoupled and dynamic coordination, can to some extent suit modern network scenarios a bit more than message-passing or distributed shared memories do.

—*Event-based models*. In event-based publish/subscribe models, a distributed application is modeled by a set of components interacting with each other by generating events and by reacting to events of interest. This clearly supports flexible and uncoupled interactions which are suitable for a dynamic network scenario.

Modern distributed event-based approaches (such as Milan [G. Picco 2003; Eugster et al. 2003] or Siena [Carzaniga et al. 2001]) also support an 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-organize event-dispatching routes and to maintain such routes despite network dynamics.

Clearly, there is considerable scope for adaptation within such systems, as the event service can abstract most of the network issues. However, top-down influences are weak, and typically the event service will support only a small number of interaction patterns efficiently, while several phenomena of self-organization can hardly be mapped in terms of publish-subscribe patterns because of the lack of any high-level concept of environment.

—*Tuple-space models*. Tuple-space-based coordination models exploit localized data structures (tuple spaces) in order to let agents deposit information, access information via a pattern-matching mechanism, and thus achieve both some forms of context-awareness and the possibility of interacting with unknown agents in a fully uncoupled way. These are indeed desirable characteristics for an autonomic communications programming and coordination model.

Although early proposals consider centralized tuple space or a limited set of well-localized distributed tuples space, more recent proposals adapt the model to better fit dynamic network scenarios. Systems such as Lime and Egospace [Curino et al. pear; Picco et al. 2001; Roman et al. 2002]) 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 can be encapsulated within higher-level environments, for example, within Sun's Jini framework. There remain issues in providing efficient and fault-tolerant implementations [Rowstron 1999], especially in highly-dynamic environments as the semantics of tuple-space interactions require extensive use of synchronization. Also, tuple space models can hardly be used to effectively program and enforce phenomena of self-organization and self-adaptation. In fact, although the tuple space can act as a sort of shared environment, its lack of structuring make mapping self-organization phenomena on it very hard.

### 5.3 Emerging Models

Beside event-based and tuple-based models, a variety of research groups have started proposing a number of innovative communication and coordination models, and the associated programming languages and infrastructures which appear much more suitable to the needs of future situated and autonomic communication scenarios. In a word, all these innovative models build on the lessons of event-based and tuple-space models and enrich them by giving meaning and/or some structure to interactions.

—*Field- and morphogen-based models*. Field-based approaches (for example, Co-Fields, TOTA and MMASS [Bandini et al. 2002; Mamei and Zambonelli 2004; Mamei et al. 2004]) can be regarded as a general framework to program and engineer coordinated behaviors in dynamic and distributed computing systems. The key idea in field-based coordination is to have components' actions driven by computational force fields generated by the components themselves and/or by some infrastructure and propagated across the environment according to specific propagation structures. To some extent, fields can be considered as sorts of distributed data structures that can play both the role of events and of shared data with the added value of giving a meaning to their distributed structuring over the network.

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 [Mamei and Zambonelli 2004] 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 reconfigure to reflect this local change in the global configuration of the field, while local instabilities are damped at the global level.

Morphogen-gradient approaches (e.g., Stoy and Nagpal [2004a, 2004b]) draw their inspiration from the original works on the amorphous computing project [Abelson et al. 2000]. 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 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 biomimetic) solutions are likely to play a key role in autonomic computing and communication (see, e.g., Babaoglu et al. [2005]. Indeed, biologically complex systems tend to exploit fully decentralized and uncoupled coordination models, relying primarily on environment-mediated local information transfer. This translates into desirable properties such as scalability, adaptability to changing conditions and dynamic scenarios, 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 [Bonabeau et al. 1999; Merloti 2004]. However, many studies in the field are explicitly targeting toy problems like, for instance, the Travelling Salesman Problem [Dorigo and Gambardella 1997] or the Graph Coloring Problem [Costa and Hertz 1997]. 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 testbed [Gelenbe et al. 2001; Gelenbe 2004a]; 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 [Nakrani and Tovey 2004]) 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., Shackleton et al. [2004]), there remains a critical need for in-depth, quantitative investigation of the performance of specific biomimetic 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, e.g., Bullock and Cliff [2004]), which entails accepting some level of uncertainty as far as the behavior and fate of individual system components is concerned. In practice, using biomimetic 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, centralized alternatives.

In the context of network-based services, Saffre and Blok [2005] describe a biomimetic 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 (e.g., Anthill and Swarmlinda [Babaoglu et al. 2002; Menezes and Tolksdorf 2003; Omicini et al. 2004; Trianni et al. 2004]) 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 cooperate 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 semistructured networks of peer-to-peer services by exploiting the capabilities of ants to reorganize 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 a 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 [Castro et al. 2003; Costa and Picco pear; Eugster et al. 2004; Eugster et al. 2004], 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 may be required in stigmergic and field-based approaches, and thus may impact on nearly all models presented previously. They propose relying on epidemiology theories to provide a probabilistic guarantees that data structures and routes will be eventually maintained.

Metabolic approaches such as Fraglets [Tschudin and Yamamoto 2004] 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 very general issue in distributed programming of large systems is the possibility for designers to predict and guarantee that a service will exhibit and maintain certain desirable properties over its lifetime regardless of any adaptations it might make. One approach is to encode explicitly the context to which a system will adapt and to derive its adaptive behavior in a way that respects both this structure and the overall goals of the service. Early research on using category theory to describe adaptive behavior [Dobson and Nixon 2004] 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 [Bandini et al. 2002; Castro et al. 2003; Rao et al. 2003; Weyns et al. 2005; Zambonelli and Mamei 2004], propose to exploit spatial and environmental abstractions as a primary means to drive components interactions. In these approaches, spatial concepts are realized by means of self-organizing and self-adapting overlay data structures that provide components with context information suitable for driving and coordinating their activities. Overlay data structures are distributed data structures that generalize the idea of overlay networks [Ratsanamy et al. 2001, 2002; Rowstron and Druschel 2001]. Overlay networks provide distributed routing management, providing components with a suitable application-specific or network-specific view of the network (e.g., 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 semantic-oriented autonomic communications in which the adaptive space is limited to physical (metric) spaces.

### 5.4 Summary

Autonomic communications presents very different challenges to traditional desktop, server, or embedded programming, and it is perhaps unsurprising that novel programming models are evolving to meet them. Such models can provide a useful substrate on which to construct autonomic control systems and algorithms, and, in many cases, may embody the structure of a given class of algorithms to simplify their construction. In and of itself, however, a programming model will not give rise to self-organizing or self-optimizing behavior without suitable algorithm design and context modeling: it is important that the model complements the algorithms and vice versa, and it is unclear which models will prove most suited to given applications.

## 6. TRUST AND SECURITY

As mentioned earlier (Section 2), the components of an autonomic system are typically regarded as consisting of a functional and a management component with the latter taking responsibility for monitoring and influencing the behavior of the former. In this architecture, security tasks are performed by the management unit governed by the element's and the system's policies, of which security policies are a subset [Chess et al. 2003].

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) [OASIS Security Services TC 2004] 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 that are 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 [IBM, Microsoft, BEA, RSA Security and VeriSign 2003] 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 can 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* [Koshutanski and Massacci 2004a, 2004b] 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 analyze the causal dependencies between software agents using techniques such as versioning vectors [Almeida et al. 2002].

Though there are a number of industrial proposals for identity solutions, such as those described, they do not cater well to 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 managing 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 for 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 [English et al. 2003a, 2003b; Cahill et al. 2003; Terzis et al. 2004] for establishing trust relationships and managing access rights.

Capability-based systems approach distributed authorization by basing their access decisions on the 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 [ITU-T 2001; Ellison et al. 1999; Thompson et al. 1999; Chadwick et al. 2003; Park and Sandhu 1999; Karabulut 2003] 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 [Blaze et al. 1999]. A number of later proposals have refined the languages used for policies, single credentials, or hierarchies thereof and for their evaluation [Li et al. 2002; Yao 2003; Becker and Sewell 2004].

Extending the concept of trust management from the level of transactions and message exchange to the context of the 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 [Seamons and Winsborough 2002; Yu et al. 2003; Bonatti and Samarati 2002; Winsborough and Jacobs 2003; Koshutanski and Massacci 2004c]. It enables iterative disclosures of credentials between a requester and a provider in order to establish the necessary level of trust to allow the exchange of data. This makes it particularly suitable for autonomic communication systems.

Further, digital reputation methods (for preliminary results and related work see Garg et al. [2004] and Michiardi and Molva [2002]) 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 behavior 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.

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 realize services, improve them (e.g., by appropriating more resources for services that are more popular), identify misbehaving AEs, modify resource allocation, etc.

### 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. [2001] 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.

Biological models of resilience provide an 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 very closely model 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 [Hofmeyr and Forrest 2000; Esponda et al. 2004].

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 [Undercoffer et al. 2003; McGibney et al. 2005].

### 6.4 Self-Organized Public Key Management

In general, the use of public key cryptography requires the presence of a centralized certification authority. However, such an authority (and the infrastructure it requires) is incompatible with the decentralized nature of autonomic communications. To solve this problem, some authors propose an approach similar to the *web of trust* of Pretty Good Privacy (PGP) [Abdul-Rahman 1996] 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, that is, 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.

### 6.5 Summary

Effective trust management is vital to the acceptability of highly pervasive applications and networking. Traditional trust and security models are highly centralized, while autonomic systems require significantly more decentralized approaches if they are to offer sufficient self-management. In particular, mechanisms are needed to establish rich collaborations between agents—not generally people—in a way that does not assume bona fides and allows autonomic identification of unacceptable patterns of behavior.

## 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 of testing experimental networks and measuring 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 has now been 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 testbeds is now definitely lower (probably by an order of magnitude) than a decade ago. Several recent projects in Europe consider low-cost wireless testbeds [Vidales et al. 2005; Tsarampopoulos et al. 2005] or specific fiber optics networks with local connectivity that could support experimentation on adaptive autonomic networks. Other work examines multimedia traffic [Magedanz et al. 2005], while wireless power management and QoS are investigated in some existing projects from the autonomic perspective [Gelenbe and Lent 2004; Gelenbe et al. 2004]. Precise traffic measurement on long haul networks is considered in Morató et al. [2005]. An ambitious project is currently planning a large regional wired/optical and wireless testbed that would support a variety of social activities [Carreras et al. 2005]. Finally, usability issues for autonomic network in Europe could be developed as an extension of work concerning virtual environments [O'Neill et al. 2005, 2006]. 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 testbeds with respect to context awareness and usability of autonomic networking environments of some experimental US and Canadian projects [Takai et al. 2005; Ionescu et al. 2005] should guide us to a higher level of ambition.

The Ubiquitous Home [Yamazaki 2005] project in Japan has created a wireless home where context-aware services are offered dynamically and ubiquitously. The self-aware network project at Imperial [Gelenbe et al. 2004] investigates how lower-level networking functions can be implemented using context-aware measurement-based techniques combined with an 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.

### 7.1 Summary

In many ways, existing network testbeds provide a good basis for comparing autonomic systems with other, more traditional approaches. The use of open standards and formats simplifies simulation and allows realistic scale simulation to answer quite complex questions about an autonomic system's reactions

to stimuli, although this realism may be limited by the fidelity of modeling the user interactions.

## 8. CONCLUSION

The notion of autonomic computing arose in specific reaction to the increasing cost of ownership of enterprise-grade systems and is leading to significant innovations in systems administration and application configuration. However, emerging applications in domains such as pervasive computing, ad hoc networking, and wireless sensors must place equal emphasis on their communications elements whose exposure to change is at least as great as that of individual computing elements. Autonomic communications encompasses a range of techniques whose application impacts computing and communications equally.

In this article, we have surveyed the state-of-the-art in autonomic communications from five complementary perspectives. Following, we draw some specific trends. As a general conclusion, however, these (and other) techniques fundamentally change the ways in which those designing, implementing, deploying, administering, and using highly-distributed adaptive systems will interact with those systems in the future. The emphasis is clearly shifting away from systems which are developed against a set of requirements agreed a priori, and towards platforms that can adapt to the changing demands placed upon them with greatly reduced human interaction and steering. Although more complex in the development phases, such systems offer enormous labor, complexity, and cost savings over the medium-and long-terms.

Does autonomic systems engineering 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 Research Priorities

We conclude by extracting the trends and challenges emerging from our foregoing survey of the autonomic communications landscape.

*Decentralization, complexity and analysis.* Traditional algorithm design has focused on issues such as space- and time-complexities, but such analyzes often require assumptions that are hard to guarantee in open environments. Of particular significance is the breakdown of the assumptions of cooperative behavior, which is being replaced by less trusting models. It seems likely that game theory and economics-derived models of interaction will increasingly become core components of algorithm analysis.

Classical (Shannon) information theory has provided an excellent basis for modeling 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 as yet 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 and will inevitably involved a richer model of how information is described and manipulated end-to-end throughout a network.

Decentralization is the sine qua non of autonomic systems: any centralized 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 decentralization is poorly understood at the algorithmic, analytic, and programming levels. New techniques are urgently needed to understand the exact robustness, performance, and complexity characteristics of decentralized 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 centralized element into even the most distributed computation. Even superficially innocuous techniques such as iteration can conceal problems which almost demand a centralized resource of some kind.

Indeed, one might reasonably argue that decentralization 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.

*Cross-layer impacts*. 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.

The use of context information is no longer confined to pervasive computing, and is instead becoming a key part of systems design. Techniques developed for pervasive applications can be applied to other, less interactive systems: the core innovations are in the use of sensor fusion and uncertain reasoning, where *sensor* is taken generally to mean any data-collecting element that can provide information about the context of a system. (*Contextor* is another, perhaps less loaded word.)

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 modeling mechanisms mentioned previously; in part, it derives from the increased dynamism in the protocols and management approaches being utilized. However, users' satisfaction with a system derives largely from such end-to-end properties. Delayering and other techniques must be improved to allow better end-to-end guarantees.

A consequence of this is that the traditional reductionist strategies for designing, constructing, and managing systems break down as systems strive to become more autonomic. The self-* properties are inherently holistic and require whole-system treatments to be developed. This poses a significant challenge for both theory and practice: autonomic systems need theories that can handle information at multiple levels of abstraction that are traditionally described using completely different formalisms, while any resulting theoretical analysis must be suitable for being maintained dynamically and robustly within an operating network.

*Paradigms*. Traditional programming languages evolved in a milieu radically different from that targeted by most autonomic systems, and it seems unlikely that their abstractions and constructs will be adequate for dealing with the emerging challenges. The range of new programming styles will require innovation in the core of programming language design, not simply new libraries.

Biologically-inspired models are likely to play an increasingly important role as the very concept of autonomic computing is itself inspired by biological models such as the autonomic nervous system. Swarm intelligence has been applied to routing and scheduling in autonomic communications, and artificial immune systems provide self-healing and self-protection. Biological models also tend to exhibit scalable convergence properties which can simplify analysis, but the interactions between subsystems can remain surprisingly subtle and need more precise characterization.

Trust and security have never evoked the public response that the research community feels they merit. This is certain to change as the permeation of daily life by advanced IT support accelerates. The emerging standards provide a platform for expressing the requirements and policies of applications—but only if these policies can be captured and balanced on a systemwide basis. We contend that the handling of trust issues must necessarily lead public concerns since a high-profile failure could have significant consequences for future adoption.

Despite significant existing research, we still lack clear and convincing trust and privacy models for highly-interactive open systems. While the core technologies may provide a proper basis, there is a clear challenge remaining in engaging in a dialogue with stakeholders to acquire, maintain, and evolve trust and privacy constraints in a simple, unobtrusive, and modular fashion.

In principle, an adaptive system may be significantly *less* variable to a user's eyes than a traditional, nonadaptive 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 deploy, and maintain complex networks can focus on the value added 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.

ACKNOWLEDGMENTS

We would like to acknowledge the support of the EU Future and Emerging Technologies initiative and the assistance of the members of the ACEnet and ACCA European project consortia, many of whose ideas have significantly informed the understanding of the issues presented here.

REFERENCES

ABDUL-RAHMAN, A. 1996. The PGP trust model. http://www.cs.ucl.ac.uk/staff/F.AbdulRahman/docs/pgptrust.html.

ABELSON, H., ALLEN, D., COORE, D., HANSON, C., HOMSY, G., KNIGHT, T., NAGPAL, R., RAUCH, E., SUSSMAN, G., AND WEISS, R. 2000. Amorphous computing. *Comm. ACM 43*, 5 (May), 74–82.

AHLSWEDE, R., CAI, N., LI, S.-Y. R., AND YEUNG, R. W. 2000. Network information flow. *IEEE Trans. Inform. Theory 46*, 4, 1204–1216.

ALMEIDA, P. S., BAQUERO, C., AND FONTE, V. 2002. Version stamps—decentralized version vectors. In *Proceedings of the 22nd International Conference on Distributed Computing Systems (ICDCS'02)*. IEEE Computer Society, 544.

ÀLVAREZ, C., DÍAZ, J., PETIT, J., ROLIM, J., AND SERNA, M. 2004. Efficient and reliable high level communicatioon in deployed sensor networks. In *ACM International Workshop on Mobility Management and Wireless Systems (Mobiwac'04)*. ACM Press, 106–110.

ANDERSEN, D. G., BALAKRISHNAN, H., KAASHOEK, M. F., AND MORRIS, R. 2002. Resilient overlay networks. *Comput. Comm. Rev. 32*, 1, 66.

BABAOGLU, Ö., JELASITY, M., AND MONTRESOR, A. 2005a. Grassroots approach to self-management in large-scale distributed systems. In *Unconventional Programming Paradigms*. Lecture Notes in Computer Science, vol. 3566. Springer-Verlag, 286–296.

BABAOGLU, Ö., JELASITY, M., MONTRESOR, A., FETZER, C., LEONARDI, S., VAN MOORSEL, A. P. A., AND VAN STEEN, M., EDS. 2005b. Self-star properties in complex information systems, conceptual and practical foundations. Lecture Notes in Computer Science, vol. 3460. Springer.

BABAOGLU, Ö., MELING, H., AND MONTRESOR, A. 2002. A framework for the development of agent-based peer-to-peer systems. In *22nd International Conference on Distributed Computing Systems*. Vienna, Austria. IEEE Computer Socity Press, 15–22.

BAKER, S. AND DOBSON, S. 2005. Comparing service-oriented and distributed object architectures. In *Proceedings of the International Symposium on Distributed Objects and Applications*, R. Meersman and Z. T. et al, Eds. Lecture Notes in Computer Science, vol. 3760. Springer Verlag, 631–645.

BALAKRISHNAN, H., KAASHOEK, M. F., KARGER, D. R., MORRIS, R., AND STOICA, I. 2003. Looking up data in P2P systems. *Comm. ACM 46*, 2, 43–48.

BALDONI, R., QUERZONI, L., PATIÑO-MARTÍNEZ, M., AND JÍMENEZ-PERIS, R. 2004. Low-load dynamic hierarchical quorums for P2P networks. Tech. rep. (Dec).

BANDINI, S., MANZONI, S., AND SIMONE, C. 2002. Heterogeneous agents situated in heterogeneous spaces. *Appl. Artificial Intellig. 16*, 9–10 (Oct.–Dec.), 831–852.

BECKER, M. Y. AND SEWELL, P. 2004. Cassandra: Flexible trust management, applied to electronic health records. In *Proceedings of the 17th IEEE Computer Security Foundations Workshop (CSFW)*. 139–154.

BIRMAN, K. 2005. *Reliable Distributed Systems Technologies, Web Services, and Applications*. Springer.

BLAZE, M., FEIGENBAUM, J., IOANNIDIS, J., AND KEROMYTIS, A. D. 1999. The role of trust management in distributed systems security. In *Secure Internet Programming: Security Issues for Mobile and Distributed Objects*. Springer-Verlag, 185–210.

BOINC PROJECT. 2006. Berkeley open infrastructure for network computing. http://boinc.berkeley.edu/.

BONABEAU, E., DORIGO, M., AND THERAULAZ, G. 1999. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press.

BONATTI, P. AND SAMARATI, P. 2002. A unified framework for regulating access and information release on the web. *J. Comput. Security 10*, 3, 241–272.

BULLOCK, S. AND CLIFF, D. 2004. Complexity and emergent behaviour in ICT systems. Tech. rep. HPL-2004-187.

BYERS, J., HORN, G., LUBY, M., MITZENMACHER, M., AND SHAVER, W. 2002. FLID/DL: Congestion control for layered multicast. *IEEE J. Select. Areas Comm. 20*, 8 (Oct.), 1558–1570.

BYERS, J. W., CONSIDINE, J., MITZENMACHER, M., AND ROST, S. 2004. Informed content delivery across adaptive overlay networks. *IEEE/ACM Trans. Netw. 12*, 5, 767–780.

CABRI, G., LEONARDI, L., AND ZAMBONELLI, F. 2000. Mobile-agent coordination models for internet applications. *IEEE Comput. 33*, 2.

CAHILL, V., SHAND, B., GRAY, E., DIMMOCK, N., TWIGG, A., BACON, J., ENGLISH, C., WAGEALLA, W., TERZIS, S., NIXON, P., BRYCE, C., DI MARZO SERUGENDO, G., SEIGNEUR, J.-M., CARBONE, M., KRUKOW, K., JENSEN, C., CHEN, Y., AND NIELSEN, M. 2003. Using trust for secure collaboration in uncertain environments. *IEEE Pervasive Comput. 2*, 3 (July), 52–61.

CARRERAS, I., GRASSO, R., AND SZABO, C. A. 2005. Design considerations on the CREATE-NET testbed. In *Proceedings of TRIDENTCOM 2005*. 44–53.

CARZANIGA, A., ROSENBLUM, D., AND WOLF, A. 2001. Design and evaluation of a wide-area event notification service. *ACM Trans. Comput. Syst. 19*, 3 (Aug.), 332–383.

CASTRO, M., JONES, M., KERMARREC, A., ROWSTRON, A., THEIMER, M., WANG, H., AND WOLMAN, A. 2003. An evaluation of scalable application-level multicast built using peer-to-peer overlays. In *IEEE Infocom*. San Francisco, CA. IEEE Computer Society Press.

CHADWICK, D., OTENKO, A., AND BALL, E. 2003. Role-based access control with X.509 attribute certificates. *IEEE Internet Comput. 7*, 2 (March/April), 62–69.

CHELIUS, G., JELGER, C., FLEURY, E., AND NOEL, T. 2005. IPv6 addressing scheme and self-configuration for multi-hops wireless ad hoc network. In *Proceedings of the International Conference on Information Networking (ICOIN'05)*. Jeju, Korea.

CHESS, D. M., PALMER, C. C., AND WHITE, S. R. 2003. Security in an autonomic computing environment. *IBM Syst. J. 42*, 1, 107–118.

CHLEBUS, B. S., GASIENIEC, L., KOWALSKI, D. R., AND RADZIK, T. 2005. On the wake-up problem in radio networks. In *ICALP*, L. Caires, G. F. Italiano, L. Monteiro, C. Palamidessi, and M. Yung, Eds. Lecture Notes in Computer Science, vol. 3580. Springer, 347–359.

CHOCKLER, G., KEIDAR, I., AND VITENBERG, R. 2001. Group communication specifications: A comprehensive study. *ACM Comput. Surv. 33*, 4, 427–469.

CHOLVI, V., LADERAS, V., LÓPEZ, L., AND FERNÁNDEZ, A. 2005. Self-adapting network topologies in congested scenarios. *Physical Rev. 71*, 3.

CHROBAK, M., GASIENIEC, L., AND RYTTER, W. 2004. A randomized algorithm for gossiping in radio networks. *Networks 43*, 2, 119–124.

CLARK, D., PARTRIDGE, C., RAMMING, J., AND WROCLAWSKI, J. 2003. A knowledge plane for the internet. In *Proceedings of ACM SIGCOMM*.

COHEN, B. 2003. Incentives build robustness in bittorrent. `http://www.bittorrent.com/bittorrentecon.pdf`.

COSTA, D. AND HERTZ, A. 1997. Ants can colour graphs. *J. Operation. Resear. Soc. 48*, 295–305.

COSTA, P. AND PICCO, G. 2005 (to appear). Semi-probabilistic content-based publish-subscribe. In *IEEE International Conference on Distributed Computing Systems (ICDCS)*, Columbus, OH. IEEE Computer Society Press.

COUTAZ, J., CROWLEY, J., DOBSON, S., AND GARLAN, D. 2005. Context is key. *Comm. ACM 48*, 3 (March), 49–53.

CURINO, C., GIANI, M., GIORGETTA, M., GIUSTI, A., MURPHY, A., AND PICCO, G. 2005. Tinylime: Bridging mobile and sensor networks through middleware. In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Kauai Island, Hawaii. IEEE Computer Society Press.

DE BRUIJN, J., LAUSEN, H., POLLERES, A., AND FENSEL, D. 2005. The WSML rule languages for the semantic web. In *Proceedings of the W3C Workshop on Rule Languages for Interoperability*. Washington, DC.

DE CASTRO, M. F., MERGHEM, L., GAÏTI, D., AND MHAMED, A. 2004. The basis for an adaptive IP QoS management. *IEICE Trans. Comm. E87-B*, 3 (March), 564–572.

DE VERGARA, J., VILLAGRÉ, V., AND BERROCAL, J. 2004. Applying the Web Ontology Language to management information definitions. *IEEE Comm. Mag. 42*, 7 (July), 68–74.

DÉFAGO, X., SCHIPER, A., AND URBÁN, P. 2004. Total order broadcast and multicast algorithms: Taxonomy and survey. *ACM Comput. Sur. 36*, 4, 372–421.

DOBSON, S. 2005. Putting meaning into the network: Some semantic issues for the design of autonomic communications systems. In *Proceedings of the 1st IFIP Workshop on Autonomic Communications* (Berlin, Germany), M. Smirnov, Ed. Lecture Notes in Computer Science, vol. 3457. Springer Verlag.

DOBSON, S. AND NIXON, P. 2004. More principled design of pervasive computing systems. In *Human Computer Interaction and Interactive Systems*, R. Bastide and J. Roth, Eds. Lecture Notes in Computer Science, vol. 3425. Springer Verlag.

DORIGO, M. AND GAMBARDELLA, L. M. 1997. Ant colonies for the travelling salesman problem. *Biosystems 43*, 73–81.

DRIVER, C. AND CLARKE, S. 2004. Context-aware trails. *IEEE Comput. 37*, 8 (Aug.), 97–99.

ELLISON, C., FRANTZ, B., LAMPSON, B., RIVEST, R., THOMAS, B., AND YLONEN, T. 1999. SPKI certificate theory. IETF RFC 2693.

ENGLISH, C., TERZIS, S., WAGEALLA, W., LOWE, H., NIXON, P., AND MCGETTRICK, A. 2003a. Trust dynamics for collaborative global computing. In *Proceedings of 12th IEEE International Workshop on Enabling Technologies (WETICE'03)*, Linz, Austria. IEEE Computer Society. 283–288.

ENGLISH, C., WAGEALLA, W., NIXON, P., TERZIS, S., LOWE, H., AND MCGETTRICK, A. 2003b. Trusting collaboration in global computing systems. In *Proceedings of 1st International Conference on Trust Management (iTrust'03)*, Heraklion, Crete, Greece. Lecture Notes in Computer Science, vol. 2692. Springer, 136–149.

ESPONDA, F., FORREST, S., AND HELMAN, P. 2004. A formal framework for positive and negative detection. *IEEE Trans. Syst., Man, Cybernet. 34*, 1 (Feb.).

EUGSTER, P., GUERRAOUI, R., KERMARREC, A., AND MASSOULIE, L. 2004. Epidemic information dissemination in distributed systems. *IEEE Comput. 37*, 5, 60–67.

EUGSTER, P., GUERRAOUI, R., AND KOUZNETSOV, P. 2004. D-reliable broadcast: A probabilistic measure of broadcast reliability. In *IEEE International Conference on Distributed Computing Systems (ICDCS)*. Tokio, Japan.

EUGSTER, P. T., FELBER, P., GUERRAOUI, R., AND KERMARREC, A.-M. 2003. The many faces of publish/subscribe. *ACM Comput. Surv. 35*, 2 (June), 114–131.

EUGSTER, P. T., GUERRAOUI, R., HANDURUKANDE, S. B., KOUZNETSOV, P., AND KERMARREC, A.-M. 2003. Lightweight probabilistic broadcast. *ACM Trans. Comput. Syst. 21*, 4, 341–374.

FERNÁNDEZ, A., GEORGIOU, C., LÓPEZ, L., AND SANTOS, A. 2005. Reliably executing tasks in the presence of malicious processors. In *DISC*, P. Fraigniaud, Ed. Lecture Notes in Computer Science, vol. 3724. Springer, 490–492.

FERNÁNDEZ, A., GEORGIOU, C., LÓPEZ, L., AND SANTOS, A. 2006. Reliably executing tasks in the presence of untrusted entities. In *Proceedings of the 25th IEEE Symposium on Reliable Distributed Systems (SRDS'06)*. Leeds, UK.

G. PICCO, G. CUGOLA, A. M. 2003. Efficient content-based event dispatching in presence of topological reconfigurations. In *Proceedings of the IEEE International Conference on Distributed Computing Systems*, Providence, RI. IEEE Computer Society Press.

GARG, A., BATTITI, R., AND COSTANZI, G. 2004. Dynamic self-management of autonomic systems: The reputation, quality and credibility (RQC) scheme. In *Proceedings of the 1st IFIP TC6 WG6.6 International Workshop on Autonomic Communication (WAC)*. Springer-Verlag, Berlin, Germany.

GELENBE, E. 2004a. Cognitive packet network, U.S. Patent 6,804,201.

GELENBE, E. 2004b. Cognitive routing in packet networks. In *ICONIP*, N. R. Pal, N. Kasabov, R. K. Mudi, S. Pal, and S. K. Parui, Eds. Lecture Notes in Computer Science, vol. 3316. Springer, 625–632.

GELENBE, E. 2004c. Sensible decisions based on QoS. *Comput. Manage. Science 1*, 1, 1–14.

GELENBE, E., GELLMAN, M., LENT, R., LIU, P., AND SU, P. 2004. Autonomous smart routing for network QoS. In *Proceedings of the First International Conference on Autonomic Computing*. IEEE Computer Society, 232–239.

GELENBE, E. AND LENT, R. 2004. Power aware ad hoc cognitive packet networks. *Ad Hoc Netw. J. 2*, 3, 205–216.

GELENBE, E., LENT, R., AND NUNEZ, A. 2004. Self-aware networks and quality of service. *Proceedings of the IEEE 92*, 9, 1479–1490.

GELENBE, E., LENT, R., AND XU, Z. 2001. Design and performance of cognitive packet networks. *Perform. Evaluat. 46*, 155–176.

GELENBE, E. AND NUNEZ, A. 2003. Adaptive web service for QoS improvement. In *Proceedings of IADIS International WWW/Internet Conference*. Algarve, Portugal.

GKANTSIDIS, C. AND RODRIGUEZ, P. 2005. Network coding for large scale content distribution. In *IEEE/INFOCOM 2005*. Miami, FL.

GONZALEZ, A. AND AHLERS, R. 1999. Context-based representation of intelligent behaviour in training simulations. *Trans. Society Comput. Simulat. Int. 15*, 4 (March).

HENRICKCON, K., INDULSKA, J., AND RAKOTONIRAINY, A. 2002. Modeling context information in pervasive computing systems. In *Proceedings of the 1st International Conference on Pervasive Computing Systems*, F. Mattern and M. Naghshineh, Eds. Lecture Notes in Computer Science, vol. 2414. Springer Verlag.

HOFMEYR, S. A. AND FORREST, S. 2000. Architecture for an artificial immune system. *Evolution. Computat. 8*, 4, 443–473.

IBM, MICROSOFT, BEA, RSA SECURITY AND VERISIGN. 2003. *Web Services Federation Language (WS-Federation)*. IBM, Microsoft, BEA, RSA Security and VeriSign. `http://www-106.ibm.com/developerworks/webservices/library/ws-fed`.

IONESCU, B., IONESCU, M., VERES, S., IONESCU, D., CUERVO, F., AND LUIKEN-MILLER, M. 2005. A testbed and research network for next generation services over next generation networks. In *Proceedings of TRIDENTCOM 2005*. 21–31.

ITU-T. 2001. The directory: Public-key and attribute certificate frameworks. ITU-T Recommendation X.509:2000(E) | ISO/IEC 9594-8:2001(E).

JELASITY, M., KOWALCZYK, W., AND VAN STEEN, M. November 2003. Newscast computing. Tech. rep., Vrije Universiteit IR-CS-006.

JENSEN, O. H. AND MILNER, R. 2003. Bigraphs and mobile processes. Tech. rep. UCAM-CL-TR-570, University of Cambridge Computer Laboratory.

JIMÉNEZ-PERIS, R., PATIÑO-MARTÍNEZ, M., AND ALONSO, G. 2002. An algorithm for non-intrusive, parallel recovery of replicated data and its correctness. In *Proceedings of 21st IEEE Int. Conference on Reliable Distributed Systems (SRDS'02)*. Osaka, Japan, 150–159.

KARABULUT, Y. 2003. Implementation of an agent-oriented trust management infrastructure based on a hybrid PKI model. In *iTrust*, P. Nixon and S. Terzis, eds. Lecture Notes in Computer Science, vol. 2692. Springer, 318–331.

KENNEDY, M. P., ROVATTI, R., AND SETTI, G. 2000. *Chaotic Electronics in Telecommunications*. CRC Press.

KEPHART, J. AND CHESS, D. 2003. The vision of autonomic computing. *IEEE Comput. 36*, 1, 41–50.

KOSHUTANSKI, H. AND MASSACCI, F. 2004a. E pluribus unum: Deduction, abduction and induction, the reasoning services for access control in autonomic communication. In *Proceedings of the 1st IFIP TC6 WG6.6 International Workshop on Autonomic Communication (WAC)*. Springer-Verlag, Berlin, Germany.

KOSHUTANSKI, H. AND MASSACCI, F. 2004b. Interactive access control for Web Services. In *Proceedings of the 19th IFIP Information Security Conference (SEC'04)*, Toulouse, France. Kluwer Press, 151–166.

KOSHUTANSKI, H. AND MASSACCI, F. 2004c. An interactive trust management and negotiation scheme. In *Proceedings of the 2nd International Workshop on Formal Aspects in Security and Trust (FAST)*, Toulouse, France. Kluwer Press, 139–152.

LAOUTARIS, N., PANAGAKIS, A., AND STAVRAKAKIS, I. 2004a. Content distribution through autonomic content and storage management. In *WAC 2004*, M. Smirnov, Ed. Lecture Notes in Computer Science, vol. 3457. Springer, Berlin, Germany, 69–78.

LAOUTARIS, N., PANAGAKIS, A., AND STAVRAKAKIS, I. 2004b. Content distribution through autonomic content and storage management. In *WAC 2004*.

LAOUTARIS, N., TELELIS, O., ZISSIMOPOULOS, V., AND STAVRAKAKIS, I. 2004. Local utility aware content replication. In *IFIP Networking 2005*.

LAOUTARIS, N., TELELIS, O., ZISSIMOPOULOS, V., AND STAVRAKAKIS, I. 2005. Distributed selfish replication. *IEEE Trans. Paral. Distrib. Syst*. [under submission].

LASSILA, O. AND SWICK, R. 1999. Resource Description Framework model and syntax specification. Tech. rep., World Wide Web Consortium.

LEFF, A., WOLF, J., AND YU, P. 1993. Replication algorithms in a remote caching architecture. *IEEE Trans. Paral. Distrib. Syst. 4*, 11 (Nov.), 1185–1204.

LEWIS, D., FEENEY, K., CAREY, K., TIROPANIS, T., AND COURTENAGE, S. 2005. Semantic-based policy engineering for autonomic systems. In *Proceedings of 1st IFIP WG6.6 International Workshop on Autonomic Communication*, M. Smirnow, Ed. Springer Verlag.

LI, N., MITCHELL, J. C., AND WINSBOROUGH, W. H. 2002. Design of a role-based trust-management framework. In *Proceedings of IEEE Symposium on Security and Privacy, 2002. S&P*. IEEE Press.

LÓPEZ, L., FERNÁNDEZ, A., AND CHOLVI, V. 2005. A game theoretic analysis of protocols based on fountain codes. In *The 10th IEEE Symposium on Computers and Communications (ISCC'05)*. La Manga del Mar Menor, Spain.

LUBY, M. 2002. LT codes. In *FOCS*. IEEE Computer Society, 271.

LUBY, M. 2003. Fast, reliable data transport. In *USENIX Symposium on Internet Technologies and Systems*.

LÜCKING, T., MAVRONICOLAS, M., MONIEN, B., AND RODE, M. 2004. A new model for selfish routing. In *Proceedings of STACS 2004*. 547–558.

MADUEÑO, M. AND VIDAL, J. 2005. Joint physical-MAC layer design of the broadcast channel protocol in adhoc networks. *IEEE J. Select. Areas in Comm.* (Special Issue on Ad-Hoc Networking).

MAGEDANZ, T., WITASZEK, D., AND KNUETTEL, K. 2005. The IMS playground at FOKUS—an open test-bed for next generation network multimedia services. In *Proceedings of TRIDENTCOM 2005*.

MAMEI, M. AND ZAMBONELLI, F. 2004. Programming pervasive and mobile computing applications with the TOTA middleware. In *IEEE International Conference On Pervasive Computing (Percom)* Orlando, FL. IEEE Computer Socity Press.

MAMEI, M., ZAMBONELLI, F., AND LEONARDI, L. 2004. Co-fields: A physically inspired approach to distributed motion coordination. *IEEE Pervasive Comput. 3*, 2, 52–61.

MAZZINI, G., ROVATTI, R., AND SETTI, G. 2000. A tensor approach to higher order expectations of chaotic trajectories—part II: Application to chaos-based DS-CDMA in multipath environements. *IEEE Trans. Circuits and Syst.—Part I 47*, 1584–1596.

MAZZINI, G., ROVATTI, R., AND SETTI, G. 2001. Chaos-based asynchronous DS-CDMA systems and enhanced rake receivers: Measuring the improvements. *IEEE Trans. Circuits Syst.—Part I 48*, 12, 1445–1453.

MAZZINI, G., ROVATTI, R., AND SETTI, G. 1999. Interference minimization by auto-correlation shaping in asynchronous DS-CDMA systems: Chaos-based spreading is nearly optimal. *IEEE Electronics Lett. 35*, 13 (June), 1054–1055.

MAZZINI, G., SETTI, G., AND ROVATTI, R. 1997. Chaotic complex spreading sequences for asynchronous DS-CDMA—Part I: System modeling and results. *IEEE Trans. Circuits Syst.—Part I 44*, 10, 937–947.

MCGIBNEY, J., SCHMIDT, N., AND PATEL, A. 2005. A service-centric model for intrusion detection in next-generation networks. *Comput. Stand. Interf. 27*, 5 (June), 513–520.

MENEZES, R. AND TOLKSDORF, R. 2003. A new approach to scalable linda-systems based on swarms. *ACM Symposium on Applied Computer*, Orlando, FL. ACM Press, 375–379.

MERGHEM, L., GAÏTI, D., AND PUJOLLE, G. 2003. On using multi-agent systems in end to end adaptive monitoring. In *Proceedings of E2EMON (End to End MONitoring) Workshop, in Conjunction with MMNS2003* (Belfast, UK). Lecture Notes in Computer Science, vol. 2839. Springer Verlag, 422–435.

MERLOTI, P. E. 2004. Optimization algorithms inspired by biological ants and swarm behavior.

MESTRE, X., FONOLLOSA, J. R., AND PAGÈS, A. 2003. Capacity of MIMO channels: Asymptotic evaluation under correlated fading. *IEEE J. Select. Areas Comm.* (Special Issue on MIMO Systems and Applications) *21*, 5 (June).

MICHIARDI, P. AND MOLVA, R. 2002. Core: A COllaborative REputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Proceedings of Communication and Multimedia Security Conference*.

MILAN-FRANCO, J. M., JIMÉNEZ-PERIS, R., PATIÑO-MARTÍNEZ, M., AND KEMME, B. 2004. Adaptive distributed middleware for database replication. In *Proceedings of 5th ACM/IFIP/USENIX Middleware Conference*. Toronto, Canada, 175–194.

MORATÓ, D., MAGANA, E., IZAL, M., ARACIL, J., NARANJO, F. J., ASTIZ, P., ALONSO, U., CSABAI, I., HÁGA, P., SIMON, G., STEGER, J., AND VATTAY, G. 2005. The European traffic observatory measurement infrastructure (ETOMIC): A testbed for universal active and passive measurements. In *Proceedings of TRIDENTCOM 2005*.

NAKRANI, S. AND TOVEY, C. 2004. On honey bees and dynamic server allocation in internet hosting centers. *Adaptive Behavior 12*, 3–4, 223–240.

OASIS SECURITY SERVICES TC. 2004. Security Assertion Markup Language (SAML). `http://www.oasis-open.org/committees/security`.

OMICINI, A., RICCI, A., VIROLI, M., CASTELFRANCHI, C., AND TUMMOLINI, L. 2004. Coordination artifacts: Environment-based coordination for intelligent agents. *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. New York (NY).

O'NEILL, E., KLEPAL, M., LEWIS, D., O'DONNELL, T., O'SULLIVAN, D., AND PESCH, D. 2005. A testbed for evaluating human interaction with ubiquitous computing environments. In *Proceedings of TRIDENTCOM 2005*.

O'NEILL, E., LEWIS, D., MCGLINN, K., AND DOBSON, S. 2006. Rapid user-centred evaluation for context-aware systems. In *Proceedings of the 13th International Workshop on Design, Specification and Verification of Interactive Systems (DSVIS'06)*, G. Doherty and A. Blandford, Eds. Lecture Notes in Computer Science, Springer Verlag.

OREIZY, P., GORLICK, M., TAYLOR, R., HEIMBIGNER, D., JOHNSON, G., MEDVIDOVIC, N., QUILLIEI, A., ROSENBLUM, D., AND WOLF, A. 1999. An architecture-based approach to self-adaptive software. *IEEE Intel. Syst. 14*, 3 (May/June), 54–62.

PARK, J. S. AND SANDHU, R. 1999. RBAC on the Web by smart certificates. In *Proceedings of the 4th ACM Workshop on Role-Based Access Control*. ACM Press, 1–9.

PICCO, G., MURPHY, A., AND ROMAN, G. 2001. Lime: A middleware for logical and physical mobility. In *Proceedings of the IEEE International Conference on Distributed Computing Systems*, Providence, RI. IEEE Computer Society Press, 524–536.

PUJOLLE, G., CHAOUCHI, H., AND GAÏTI, D. 2004. Beyond TCP/IP: A context-aware architecture. In *Proceedings of NetCon 2004*, Palma de Mallorca. Kluwer Academic Publishers, 337–347.

QIU, D. AND SRIKANT, R. 2004. Modeling and performance analysis of bittorrent-like peer-to-peer networks. In *SIGCOMM*, R. Yavatkar, E. W. Zegura, and J. Rexford, Eds. ACM, 367–378.

QUITADAMO, R. AND ZAMBONELLI, F. 2007. Autonomic communication services: a new challenge for software agents. *J. Autonom. Agents Multiagent Syst*.

RAO, A., PAPADIMITRIOU, C., RATNASAMY, S., SHENKER, S., AND STOICA, I. 2003. Geographic routing without location information. In *ACM Mobicom*. San Diego, CA. ACM Press.

RATSANAMY, S. AND AL. 2002. Ght: A geographic hash table for data-centric storage. In *1st ACM International Workshop on Wireless Sensor Networks and Applications*. ACM Press, Atlanta, Georgia, USA.

RATSANAMY, S., FRANCIS, P., HANDLEY, M., AND KARP, R. 2001. A scalable content-addressable network. In *ACM SIGCOMM Conference*, San Diego, CA. ACM Press.

RODERO, L., LÓPEZ, L., FERNÁNDEZ, A., AND CHOLVI, V. 2006. Dante: A self-adapting peer-to-peer system. In *Proceedings of the 5th International Workshop on Agents and Peer-to-Peer Computing (AP2PC06)*, Hakodate, Japan. Lecture Notes in Computer Science. Springer.

ROMAN, G., JULIEN, C., AND HUANG, Q. 2002. Network abstractions for context-aware mobile computing. *24th International Conference on Software Engineering*, Orlando, FL. ACM Press.

ROVATTI, R., MAZZINI, G., AND SETTI, G. 2000. A tensor approach to higher order expectations of chaotic trajectories—Part I: General theory and specialization to piecewise affine Markov systems. *IEEE Trans. Circuits Syst.—Part I 47*, 1571–1583.

ROVATTI, R., MAZZINI, G., AND SETTI, G. 2001. Enhanced rake receivers for chaos-based DS-CDMA. *IEEE Trans. Circuits Syst.—Part I 48*, 818–829.

ROVATTI, R., MAZZINI, G., AND SETTI, G. 2004a. On the ultimate limits of chaos-based asynchronous DS-CDMA—Part I: Basic definitions and results. *IEEE Trans. Circuits Syst.—Part I 52*, 7, 1336–1347.

ROVATTI, R., MAZZINI, G., AND SETTI, G. 2004b. On the ultimate limits of chaos-based asynchronous DS-CDMA—Part II: Analytical results and asymptotics. *IEEE Trans. Circuits Syst.—Part I 52*, 7, 1348–1364.

ROVATTI, R., SETTI, G., AND MAZZINI, G. 1998. Chaotic complex spreading sequences for asynchronous DS-CDMA—Part II: Some theoretical performance bounds. *IEEE Trans. Circuits Syst.—Part I 45*, 4, 496–506.

ROWSTRON, A. 1999. Mobile coordination: Providing fault tolerance in tuple space based coordination language. *Coordination Languages and Models (Coordination'99)*, P. Ciancarini and P. Wolf, Eds. Lecture Notes in Computer Science, vol. 1594. Springer-Verlag, 196–210.

ROWSTRON, A. AND DRUSCHEL, P. 2001. Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems. *18th IFIP/ACM Conference on Distributed Systems Platforms*. Heidelberg, Germany. ACM Press.

SAFFRE, F. AND BLOK, H. R. 2005. SelfService—A theoretical protocol for autonomic distribution of services in P2P communities. In *Proceedings of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05)*. 528–534.

SEAMONS, K. AND WINSBOROUGH, W. 2002. Automated trust negotiation. Tech. rep., US Patent and Trademark Office. IBM Corporation, patent application filed March 7, 2000.

SETTI, G., MAZZINI, G., ROVATTI, R., AND CALLEGARI, S. 2002. Statistical modeling and design of discrete time chaotic processes: Basic finite-dimensional tools and applications. In *Proceedings of the IEEE*. Vol. 90. 662–690.

SETTI, G., ROVATTI, R., AND MAZZINI, G. 2004. Performance of chaos-based asynchronous DS-CDMA with different pulse shapes. *IEEE Comm. Lett. 8*, 7 (July), 416–418.

SHACKLETON, M., SAFFRE, F., TATESON, R., BONSMA, E., AND ROADKNIGHT, C. 2004. Autonomic computing for pervasive ICT—A whole system perspective. *BT Tech. J. 22*, 3, 191–199.

SIRIS, V. A. 2002. Resource control for elastic traffic in CDMA networks. In *Proceedings of 8th ACM Conference on Mobile Computing and Networking (MOBICOM'02)*. Atlanta, GA.

SIRIS, V. A., BRISCOE, B., AND SONGHURST, D. 2002. Economic models for resource control in wireless networks. In *Proceedings of IEEE Int. Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC'02)*.

SIRIS, V. A. AND COURCOUBETIS, C. 2004. Resource control for loss-sensitive traffic in CDMA networks. In *Proceedings of 23rd Conference of the IEEE Communications Society (INFOCOM)*. Hong Kong.

SPINGLASS PROJECT. 2005. Spinglass: Adaptive probabilistic tools for advanced networks. `http://www.cs.cornell.edu/Info/Projects/Spinglass/`.

STERRITT, R., MULVENNA, M., AND LAWRYNOWICZ, A. 2005. A role for contextualised knowledge in autonomic communications. In *Proceedings of 1st IFIP TC6 WG6.6 International Workshop on Autonomic Communications*, M. Smirnow, Ed. Springer Verlag.

STOY, K. AND NAGPAL, R. 2004a. Self-reconfiguration using directed growth. In *7th International Symposium on Distributed Autonomous Robotic Systems (DARS)*. Toulouse, France. ACM Press, 149–160.

STOY, K. AND NAGPAL, R. 2004b. Self-repair through scale independent self-reconfiguration. In *Proceedings of IEEE/RSJ International Conference on Robots and Systems (IROS)*, Sendai, Japan. IEEE Press.

STRANG, T., LINNHOFF-POPIEN, C., AND FRANK, K. 2003. CoOL: A context ontology language to enable contextual interoperability. In *Proceedings of 4th IFIP International Conferece on Distributed Applications and Interoperable Systems (DAIS'03)*.

TAKAI, M., BAGRODIA, R., GERLA, M., DANESHRAD, B., FITZ, M. P., SRIVASTAVA, M. B., BELDING-ROYER, E. M., KRISHNAMURTHY, S. V., MOLLE, M., MOHAPATRA, P., RAO, R. R., MITRA, U., SHEN, C.-C., AND EVANS, J. B. 2005. Scalable testbed for next generation wireless networking technologies. In *Proceedings of TRIDENTCOM 2005*. 162–171.

TANENBAUM, A. 2004. *Distributed Systems*, 2nd ed. Addison Wesley.

TERZIS, S., WAGEALLA, W., ENGLISH, C., AND NIXON, P. 2004. Trust lifecycle management in a global computing environment. In *Global Computing, IST/FET International Workshop GC 2004*, Rovereto, Italy. 9–12 (March), C. Priami and P. Quaglia, Eds. Lecture Notes in Computer Science, vol. 3267. 291–313.

THOMPSON, M., JOHNSTON, W., MUDUMBAI, S., HOO, G., JACKSON, K., AND ESSIARI, A. 1999. Certificate-based access control for widely distributed resources. In *Proceedings of 8th USENIX Security Symposium (Security'99)* (23–26). 215–228.

TRIANNI, V., LABELLA, T., AND DORIGO, M. 2004. Evolution of direct communication for a swarm-bot performing hole avoidance. In *ANTS Workshop*. Brussels, Belgium.

TSARAMPOPOULOS, N., KALAVROS, I., AND LALIS, S. 2005. A low-cost and simple to deploy peer-to-peer wireless network based on open source linux routers. In *Proceedings of TRIDENTCOM 2005*.

TSCHUDIN, C., GUNNINGBERG, P., LUNDGREN, H., AND NORDSTROM, E. 2005. Lessons from experimental MANET research. *Ad Hoc Netw. J. 3*, 2 (March).

TSCHUDIN, C. AND YAMAMOTO, L. 2004. A metabolic approach to protocol resilience. In *Proceedings of the 1st IFIP Workshop on Autonomic Communication (WAC'04)*. Berlin, Germany.

TSCHUDIN, C. F., GUNNINGBERG, P., LUNDGREN, H., AND NORDSTRÖM, E. 2005. Lessons from experimental MANET research. *Ad Hoc Netw. 3*, 2 (March), 221–233.

UNDERCOFFER, J., JOSHI, A., AND PINKSTON, J. 2003. Modeling computer attacks: An ontology for intrusion detection. In *Proceedings of the 6th International Symposium on Recent Advances in Intrusion Detection (RAID'03)*. Pittsburgh, PA. Lecture Notes in Computer Science, vol. 2820, 113–135.

VIDALES, P., MAPP, G., STAJANO, F., CROWCROFT, J., AND BERNARDOS, C. 2005. A practical approach for overlay systems: Deployment of overlay networks. In *Proceedings of TRIDENTCOM 2005*.

VON RICKENBACH, P., SCHMID, S., WATTENHOFER, R., AND ZOLLINGER, A. 2005. A robust interference model for wireless ad-hoc networks. In *IPDPS*. IEEE Computer Society.

WEISER, M. 1991. The computer for the 21st century. *Scientific American 265*, 3, 94–104.

WEYNS, D., PARUNAK, H., MICHEL, F., HOLVOET, T., AND FERBER, J. 2005. *Environments for Multiagent Systems, State-of-the-art and Research Challenges*. Lecture Notes in Artificial Intelligence, vol. 3374. Springer Verlag, Berlin, Germany.

WINSBOROUGH, W. AND JACOBS, J. 2003. Automated trust negotiation in attribute-based access control. In *Proceedings of DARPA Information Survivability Conference and Exposition*. Vol. 2. IEEE Press, 252–257.

YAMAZAKI, T. 2005. Ubiquitous home: Real-life testbed for home context-aware services. In *Proceedings of TRIDENTCOM 2005*. 54–69.

YAO, W. 2003. Fidelis: A policy-driven trust management framework. In *1st International Conference on Trust Management (iTrust)*. Lecture Notes in Computer Science, vol. 2692. Springer, 301–317.

YU, T., WINSLETT, M., AND SEAMONS, K. E. 2003. Supporting structured credentials and sensitive policies through interoperable strategies for automated trust negotiation. *ACM Trans. Inform. Syst. Secur. (TISSEC) 6*, 1, 1–42.

ZAMBONELLI, F. AND MAMEI, M. 2004. Spatial computing: An emerging paradigm for autonomic computing and communication. In *International Workshop on Autonomic Communication*. Berlin, Germany.

Received January 2006; revised July 2006; accepted August 2006