

Self-optimized Cognitive Network of Networks

A. MANZALINI^{1,*}, P.H. DEUSSEN², S. NECHIFOR³, M. MAMEI⁴, R. MINERVA^{1,5},
C. MOISO¹, A. SALDEN⁶, T. WAUTERS⁷ AND F. ZAMBONELLI⁴

¹TELECOM ITALIA Future Centre, Via Reiss Romoli, 274, 10148 Turin, Italy

²FOKUS Kaiserin-Augusta-Allee 31, 10589 Berlin, Germany

³Siemens Program and System Engineering SRL, Colina Universitatii 1, Brasov, Romania

⁴Università degli Studi di Modena e Reggio Emilia, Via Amendola 2, 42100 Reggio Emilia, Italy

⁵Institut TELECOM SudParis, 9 Rue Charles Fourier, 91000 Evry, France

⁶Almende B.V., Westerstraat 50, 3016 DJ Rotterdam, The Netherlands

⁷Department of Information Technology (INTEC), Ghent University, IBBT, Gaston Crommenlaan 8, bus 201,
B-9050 Ghent, Belgium

*Corresponding author: antonio.manzalini@telecomitalia.it

Future processing, storage and communication services will be highly pervasive: people, smart objects, machines and the surrounding space (all embedding devices such as with sensors, RFID tags etc.) will define a highly decentralized cyber environment of resources interconnected by dynamic networks of networks. As communications will extend to cover any combination of 'people, machines and things', future networks will be increasingly complex and heterogeneous, yet always endorsed with the challenging task of ensuring end-to-end QoS. This paper proposes the groundwork for an advanced cognitive networking paradigm exploitable in future wired and wireless infrastructures: a decentralized cognitive plane to allow for cross-layer, cross-node and cross-network domain self-management, self-control and self-optimization, while being compatible with legacy management and control systems.

Keywords: cognitive networking; self-management; autonomic; cross-layer; knowledge; network of networks

Received 15 December 2009; revised 12 February 2010

Handling editor: Erol Gelenbe

1. INTRODUCTION

In the future, processing, storage and communication services will be highly pervasive. It is expected that people, smart objects, machines and the surrounding space will all be embedded with devices such as sensors, RFID tags etc., defining a highly decentralized dynamic cyber environment of resources interconnected by pervasive networks of networks (NoNs). This evolution, which has already been pointed out in several papers for both the civilian and military domains [1, 2], is posing challenges and opportunities.

Such future networks will be indeed increasingly complex and heterogeneous, as they will be composed of combinations of a large number of different wireless and wired nodes, devices and a wide variety of applications and protocols. The management of these future networks poses challenges that are not limited to individual point-to-point and client-server applications, but they involve the network as a whole. Let us

think about the Internet: today it is built and operated by a multitude of large and small entities collaborating with each other to process and deliver end-to-end flows originating and terminating in any one of them [1, 3]. However, to globally optimize and manage the network, many further questions emerge, for example: does TCP/IP congestion control impose a Nash equilibrium? Will future networks become an arena of coalitional theoretical games? What is the 'price' of potential anarchy [4, 5]?

The above questions and the need for related analysis suggest a novel viewpoint on network management, heading towards the development of an integrated cognitive networking approach [6, 7], aimed at simplifying management and control, while capable of optimizing resource allocation and use at a more global level. If a node of this cognitive network has a certain level of knowledge of the overall status of the network, operations decisions should be at least as good, if not better

(in terms of global effectiveness and optimization) than those made in ignorance [8, 9]. Of course, for this to be effective, the costs related to exploitation of cognitive enhancements should be gradual and optimized: for example cognition should rely on autonomies, proper filtering and abstraction of network knowledge should be used to reduce the amount of information that has to be exchanged, unnecessary triggering of the cognitive process should be avoided [9].

This paper aims at illustrating a novel approach and a research agenda for introducing cognitive capabilities in future networks. Specifically, it describes the design of the decentralized cognitive plane (DCP) as a new separate plane to provide cross-layer/node/network monitoring, knowledge acquisition and optimization in pervasive future networks. The DCP proposal, while owing credits to previous pioneering ideas and findings in the area of knowledge-based network management [10, 11] and of cognitive network paradigms [7, 9], attempts at leveraging a novel architectural approach, more specifically suited for the outlined future network scenarios.

The remainder of this paper is as follows. Section 2 describes the scenario and use case motivating our research. Section 3 presents the vision and the architectural principles of the DCP, also surveying the potential scientific and technological sources of inspiration towards the actual realization of the DCP idea. Section 4 elaborates on relevant related work in the cognitive area domain, and compares it with the approach proposed by this paper. Section 5 draws some conclusions.

2. SCENARIO AND USE CASE

Future networks will define a highly decentralized cyber environment of dynamically interconnected resources. The following short story illustrates and highlights some relevant aspects of this new scenario.

Imagine that Josip's personal information is available to him anywhere, at any time and in the proper format. His personal network will rely on always-on connectivity, allowing his devices to talk with each other and with the environment.

Local networks can be created on the fly among communicating devices without needing a centralized control. Terminals also can share communication, storage and data resources with the terminals of other people belonging to the same community, or even with unknown people in a trusted and cooperative environment (context data information is disclosed according to contextual privacy policies and settings).

Smart bots running on Josip's terminal will work on refining his profile to provide better personalization and use of services. In particular, in an information retrieval application, those bots will run algorithms to predict what information might be of interest to the user (most of the websites users will visit the following day can be predicted by looking at the surfing habits in the last weeks). Then, the bots might download overnight and cache all information that Josip will likely access

the following day. This would notably speed up information retrieval.

Access to the Internet will not necessarily require a connection from Josip's device to a public infrastructure, but it will just need to contact a nearby terminal and either find directly what he is looking for, be reconnected to a public machines (e.g. kiosk), or to the next in 'line'.

In most cases Josip will find that this hop-by-hop connectivity (through mesh networks formed solely by devices) fits his needs. Kiosks will be widely available, so that Josip can also off-load computation of smart-phone applications. Users will be able to refill their personal storage with the desired contents or receive the latest info on demand (directly from a kiosk).

From the sketched scenario (Fig. 1), it is easy to imagine future NoNs becoming increasingly complex and heterogeneous, with the challenging task of ensuring end-to-end performances. Traditional cross-layer designs perform independent optimizations that may not account for the end-to-end performance goals. Trying to achieve each goal independently is suboptimal, and as the number of cross-layer designs within a node/network grows, leading to adaptation loops.

The thesis put forward by this paper is that, to avoid above the pitfall, advanced cognitive networking capabilities should be exploited in future wired and wireless infrastructures: specifically a new plane, the DCP, should be introduced to allow for cross-layer, cross-node and cross-network domain self-management, self-control and self-optimization of resources (while being compatible with legacy management and control systems).

Let us consider, for example, a video-streaming application. As this application is highly resource-consuming (e.g. CPU, memory and bandwidth), cross-layer optimization and management is advisable to ensure the appropriate end-to-end QoS, and fair resource sharing. In this case, QoS specifications encompass quantitative parameters (e.g. jitter, delay, bandwidth) and qualitative parameters (e.g. CPU scheduling policy, error recovery mechanism), as well as adaptation rules. For example, given certain streaming application requirements, the network resources must ensure certain bit rate, latency, jitter, packet error rate etc.

Let us make a more detailed analysis. First, quality of experience (QoE) of the user should be translated into application QoS specifications (in principle this mapping does not require to know the underlying operating systems and network conditions); these application QoS specifications imply quantitative issues (e.g. video frame rate, image/audio resolution), qualitative issues (e.g. inter/intra-stream synchronization schemes) and adaptation rules (e.g. dropping frames).

Then, in order for the application to be executed in a real OS platform and physical network, the above application-specific QoS parameters need to be translated into more concrete resource requirements, such as bandwidth, delay, jitter, memory allocation, CPU scheduling policies etc. The DCP is the plane enacting these behaviours: it collects/elaborates network

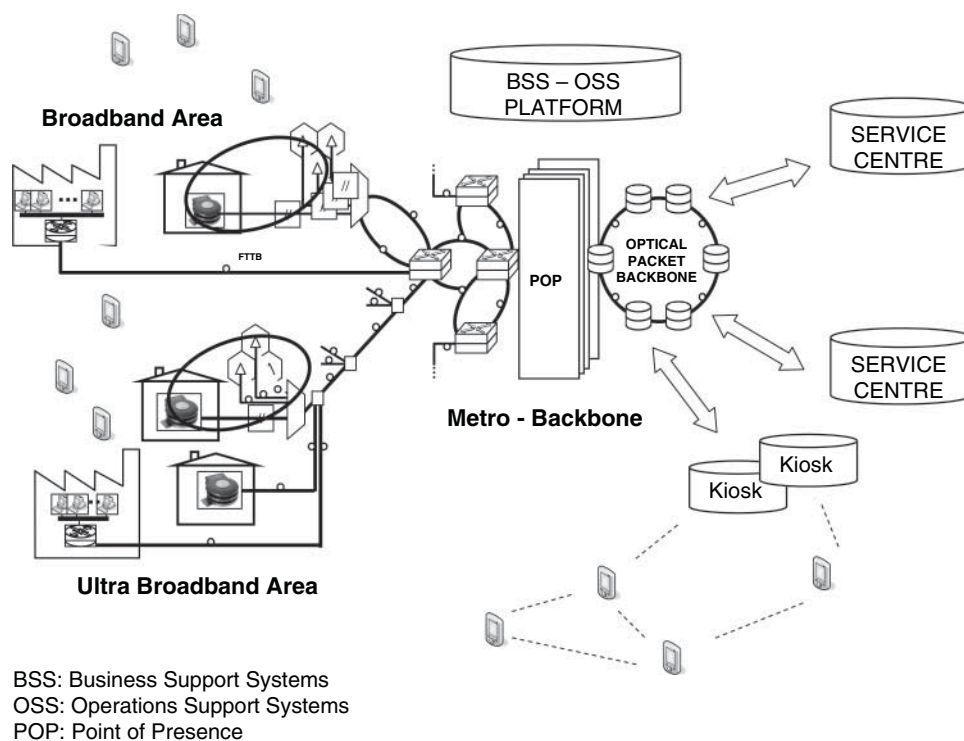


FIGURE 1. Network scenario.

knowledge and, in harmony with management and control tasks, it allows configurations/optimized allocations of resources (to meet the QoE/QoS requirements).

3. VISION AND ARCHITECTURAL CONCEPTS

The DCP is enabling future networks to self-optimize, self-manage and self-control their behaviour, in a reactive-proactive manner, based on a cross-layer/cross-node/cross-network domain of knowledge sharing and processing. The DCP is composed of distributed components in charge of monitoring and effecting the network at different layers.

DCP components run data gathering mining, reasoning and machine learning algorithms/techniques to extract useful information on what is happening in the network and in the distributed services. Eventually, the DCP is in charge of enacting specific autonomic behaviour on the basis of the collected data to assure self-management/control and optimization.

The key concepts at the basis of the DCP are the following:

- (i) a network knowledge representation and diffusion criteria: in particular proper filtering and abstraction of knowledge should be used to limit the amount of information that must be exchanged;
- (ii) tools for the efficient creation and maintenance of the network knowledge by using a combination of recent developments in machine learning and reasoning;

- (iii) optimization algorithms for NoNs, so as to adapt to application and environment changes and meet QoS requirements. Definition of trade-offs might be required: when a problem has multiple objectives, it will not be possible to optimize all metrics indefinitely (multiple objective optimization (MOO)).

A DCP framework, based on the above architectural principles, should be implemented by means of a distributed lightweight middleware, in charge of the following:

- (i) collecting, propagating and aggregating cross-layer data from nodes and software components. This will be useful to get an overall picture of the current status of the network;
- (ii) hosting and running tools, optimization algorithms. This is useful to take concrete actions to manage the network on the basis of the acquired information;
- (iii) interfacing with the management plane (MP), control plane (CP) and hypervisors (or virtual machine managers (VMM)). In particular, the CP is a logical concept that defines the part of the router architecture responsible for building and drawing the network topology map (also known as the routing table) and manifesting it to the forwarding plane (where actual packet forwarding takes place) in the form of the forwarding information base; a hypervisor is a program that allows multiple instances of operating systems to

share a single hardware host. The DCP needs to interact with these low-level services in order to enact network management policies at different levels of the network stack.

The key point of keeping the DCP substrate as minimal as possible is necessary to provide primitives and basic interaction rules (self-organization, clustering, gossiping etc.) that enable the efficient execution and coordination of needed functionalities.

Now, let us survey what are the key issues related to the realization of the DCP idea, and what could be useful sources of scientific and technological inspiration.

3.1. Middleware implementation issues

The DCP should be developed (Fig. 2) to run not only over network nodes (cross-connect, routers, switches, home-gateways etc.), servers (storage, computing) and laptops, but also on users' wireless handheld devices and small computing resources (e.g. mobile phones, PDAs, pervasive sensors etc.).

Accordingly, the DCP should be implemented by means of some minimal middleware substrate, i.e. a software infrastructure deployed on top of the physical resources. Such a middleware substrate should provide for supporting the execution of individual DCP components, and should enforce the concepts of logical and physical distance, locality, local interactions and mobility, coherently to a specific structure of the space. Interaction rules of components should be dynamically plugged in the DCP depending on the application scenario. The advantage of this kind of rule-based approach is that these rules will be applied to all the services in a transparent way, guaranteeing a good separation of concerns in the development.

One important aspect of the DCP implementation will be the absence of a common unifying component model (this goes also in the direction of middleware deconstruction) [12]. DCP design should not necessarily include the definition

of how a component is made internally. Each developer should be able to create components without any constraints and possibly using third-party component frameworks (J2EE, agent-based frameworks etc.). This is the choice adopted by modern Web APIs, only defining XML-based protocols for method invocations and leaving open the possibility of implementing whatever kind of application and services using the API.

3.2. Topological issues and bio-inspired information diffusion

A.L. Barabási pointed out that the dynamics of many social, technological and economic phenomena are driven by human actions, turning the quantitative understanding of human behaviour into a central question of modern science. Studies and simulations on small-world networks (Fig. 3a) represent an interesting effort to model the dynamical behaviour of social, economic and physical networks, and to properly explain why information in social networks tends to diffuse in a very fast and effective way. Accordingly, in order to allow effective and efficient interactions among DCP components enabling fast diffusion of information, it is reasonable to think that DCP components interact over a small-world network.

The way in which components on the DCP live and interact (which may include how they capture and diffuse information, how they move in the environment, how they self-compose and/or self-aggregate with each others) is determined by the set of fundamental 'interaction rules' regulating the DCP model. From this viewpoint, it is interesting to leverage laws of biological systems where evolvability and a close pattern-function relationship provide biological organisms with the plasticity to cope with systemic changes and environmental changes and to learn critical survival strategies under such circumstances by renormalization or multi-scale techniques [13].

Principles like reaction-diffusion could be adopted also in pervasive networks for building cognitive control and

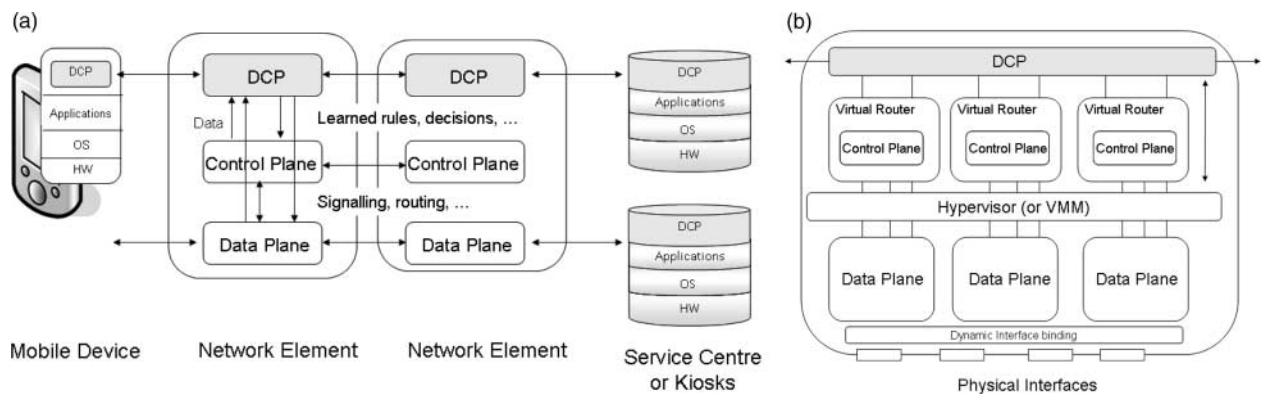


FIGURE 2. (a) DCP in network resources (mobile devices, routers, servers) and (b) DCP in sliced virtual routers.

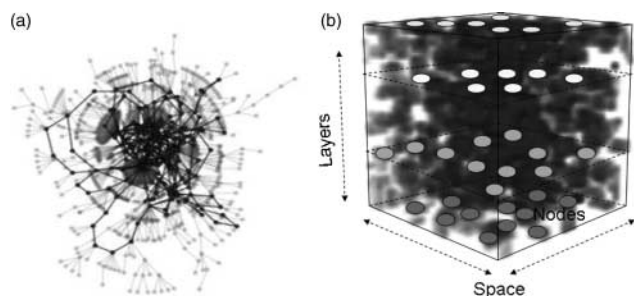


FIGURE 3. (a) Example of small-world topology of DCP components and (b) a 3D simulation (Gray–Scott equations) on information patterning in multi-layered NoNs of nodes/agents (simulations made with Breve tool-kit—<http://www.spiderland.org/>).

optimization [14, 15, 27] capabilities even only based on local interactions (interactions are sometimes a more powerful paradigm than algorithms, since algorithms cannot take into account the time or the interaction events that occur during computation): a network can be seen as ensembles of nodes performing some type of processing (and/or storage) (i.e. reaction) and being linked to other nodes by some communications protocol (i.e. diffusion).

In [12] a middleware is described that relies on spatially distributed tuples for both supporting adaptive and uncoupled interactions between agents, and context-awareness. Nodes/agents 'diffuse' these tuples in the network (to make available some kind of contextual information) and to 'react' with other agents. Tuples are propagated by the middleware, on the basis of application-specific patterns, defining sorts of 'computational fields', and their intended shape is maintained despite network dynamics, such as reconfigurations. Another example is offered by [16], where a Turing-like reaction-diffusion network of smart bots was taught a navigation task under the control of an evolutionary learning algorithm. This perspective opens the possibility that cognitive capabilities could be not only symbolic but also, to a certain extent, structural.

3.3. Network knowledge organization

One of the main problems that have to be addressed in DCP design is about organizing knowledge in an autonomic network. This knowledge is used by the DCP to reason about the current state of the network. This problem has been addressed through the knowledge-based network paradigm [11, 17]. In particular, producers of information describe the available information through ontologies; consumers subscribe to this information through semantic queries. In [18] an interesting three-component architecture is presented which is responsible for collecting the necessary knowledge for each node by querying information present in other remote nodes.

In principle, network knowledge requires a collection of models at different levels. In general the DCP has to filter

its observations as much as possible just to make cognitive processing for a network feasible. The higher these levels, the more abstract such a model needs to be to handle the various cross-scale issues. Examples of models would range from purely statistical descriptions (e.g. co-occurrences and correlations based models) to highly semantic ones (logical representation of networks, descriptions of devices, descriptions of activities etc.).

It is proposed to define a scalable network knowledge representation for NoNs, based on a combination of statistical and semantic models, taking advantage of both bottom-up structures that can emerge from the raw-data stream and top-down models developed by domain experts. The combination will result in self-updating (semi)automatic models representing the state of NoN nodes, devices etc.

The knowledge network prototypes developed (and available in Open Source) in the framework of the CASCADAS [19] project might be a meaningful example of a decentralized solution for managing such knowledge.

3.4. Interfacing with the MP and the CP, and with the VMM

From an architecture viewpoint, the DCP should be able to interact with different network sub-systems in order to enact autonomic management at different levels of the network stack.

Currently, the CP (managing the routing infrastructure) is positioned between the network plane and the MP of the network. The latter, including the operations support systems, configures and supervises the CP and has the ultimate control over all network plane and CP entities (requiring substantial human intervention).

The DCP encompasses capabilities unavailable in the MP and the CP. First, the DCP has a richer and more integrated network knowledge and so it can make the most effective resources configurations and optimizations. Second, the DCP might be implemented using the level of complexity considered appropriate by the Owners. On the other hand, there is the challenge to implement the necessary interfacing and coordination mechanisms (with the MP and the CP) in order to keep the consistency of the actions between various planes.

The DCP can also assist the VMMs or hypervisors (allowing multiple operating systems to share hardware resources) to manage in an optimal way the virtual resources. Network virtualization is a powerful technique as it provides flexibility, promotes diversity and promises security and increased manageability. Nevertheless, there is still a dramatic lack of adequate node performances and programmability, sufficient isolation, cross-layer and cross-network domain interoperability, optimization of physical resources etc.

The introduction of the DCP enhances the current CP and MP, with limited impacts on the expected performance of the execution of specific service sessions. In fact, in general the DCP not involved neither in the run-time control performed

by the resources involved in a service execution, nor in the routing of data across the network. Similarly to the knowledge plane [10], the DCP cooperates with the CP and the MP to activate cross-domain, cross-network, cross-layer 'recognize-explain' and 'recognize-explain-suggest' cycles. In this way, the DCP learns how to improve the network behaviour and may instruct the CP, or even directly the resources, on how to change their configurations in order either to optimize the network performance or to adapt it to the requirements of new services. The configuration task of a network and its resources become a continuous process, adapting them to the evolution of their conditions, priorities and constraints, and of application needs.

In any case, the performances of the DCP must be carefully planned (e.g. by introducing dynamic filters on the information to be collected and processed), to reduce computing, storage and communication resources devoted by the network entities to create and maintain the network knowledge for the process of continuous monitoring and adaptation of configurations of the network and its resources.

3.5. Information security issues

Information security is becoming more and more a vital design aspect of new networking paradigms. A number of well-known solutions are already available today for protecting the Internet: for example firewall tries blocking suspicious traffic and connections; authentication, authorization and an access control mechanism should prevent unauthorized usage of resources and allow access according to privileges. There are also a number of solutions that perform some analysis to detect network intrusions; in general they look for patterns in data observed somewhere in the network. It is widely recognized that the next generation of tools for intrusion detection will require observations from several points in the network that will have to be correlated, in order to get a more robust picture of the situation.

Future NoNs could rely on (active and passive) monitoring tools to develop a kind of multi-layer security system in order to put in place a protection from wide variety of threats and malicious attacks. The DCP provides a basis to implement this data gathering and correlation. DCP should be capable of making global decisions (on its own, i.e. autonomically) by sensing and correlating data of different levels (such as application, user, system, process and packet levels) of the network, and collected from different elements.

On the other hand, since the DCP will have these important tasks, it might become, from the information security point of view, an Achilles's heel. As a consequence, the DCP may need to build, maintain and reason about trust relationships among its components and participants. A trust network can evolve that identifies DCP components to be trustworthy and shuns elements that are not. Introspection would likely require the

development of trust models, and the use of scalable techniques to search a web of trust [10].

Some principles concerning information processing of the biological immune system look like appealing for developing information security against threats and malicious attacks in NoNs. Some of these principles include distributed processing, pathogenic pattern recognition, decentralized control, signalling etc. Actually, some results can be leveraged from the research on artificial immune systems (AISs) [20]. The main idea is simulating a bio-inspired approach, which takes inspiration from the biological immune system to protect distributed computing environments. Interestingly, researches have shown how AISs, which use multi-level information sources as input data, can be used to build effective algorithms for real-time computer intrusion detection.

4. PRIOR ART

Scientific communities (as well as standardization bodies) are increasingly recognizing the need and the potential of introducing cognitive concepts into such future networks. Nevertheless, currently, most efforts focus on cognitive radios (research in this domain is well ahead compared to cognitive networking, which is just taking off).

The vision of E2R project [21] is based on an all-IP network fully integrated with reconfigurable equipment at all network layers. This approach, lacking in scalability, seems to limit the incremental deployment of such networks.

m@ANGEL [22] proposes to implement the cognitive process in the access part of the network, between base stations and mobile users. The degree of cooperation between m@ANGEL elements is usually performed within network elements located in neighbouring cells and it is not propagated to the network core, and this seems to limit the potential of an integrated solution.

In [23] a framework for implementing the cognitive functionality is presented; node architecture implies a logical separation between network nodes and the cognitive engine running in the network. While the cognitive engine performs learning, orientation, planning and decision-making functions, observation and action are left to the reconfigurable node.

In CogNet (<http://adaptive5.ucsd.edu/cognet/>), each protocol layer is extended with software agents performing intra-layer monitoring, control and coordination functions. Modules are interconnected through the cognitive bus to coordinate the cognition modules and are implemented in parallel to the protocol stack. Cognitive functions implemented in intra-layer cognitive elements are distributed between different protocol layers.

A lack of proper coordination or intra-layer cognitive agent monitoring could lead to unpredictable performance results.

Gelenbe *et al.* [6–8, 24] proposed cognitive packet networks, which basically transfer routing and flow control capabilities

from network nodes to the packets. Each cognitive packet contains a cognitive map and a piece of code that is executed every time the packet arrives at the network node. Routing decisions are taken relying on the cognitive map, as well as messages left by other packets or by the network node.

The SPIN [25] architecture is based on three planes interconnected by a layer-2 infrastructure: the forwarding plane is in charge of switching and monitoring and it can provide connectionless and connection-oriented packet forwarding, and tag and label switching; the CP/MP manages forwarding plane devices targeting data, where the forwarding optimization is based on the received measurements; the cognitive plane resides on top of the control/management and forwarding planes, providing intelligence for and administration of the entire system.

Self-NET (<http://www.ict-selfnet.eu/>) aims at defining a paradigm for Internet self-management based on cognitive behaviours, around a novel feedback-control cycle.

The DCP should be exploited not only for L1–L3 network layers, but also for the application-level context information; the DCP design should be exploited as an independent plane, capable of interfacing (and being backward compatible) with the MP, CP and VMM; it should explicitly address not only cross-layer and cross-node but also cross-network domain optimization, by means of advanced algorithms; moreover, the DCP should be deployed on a large variety of nodes (open routers, home gateway, switches) and a user's portable devices.

5. CONCLUSIONS

Future networks will be increasingly complex and heterogeneous, and the task of ensuring end-to-end performances will be even more challenging than today. In this context, the key thesis of this paper is that the above challenges can be faced by introducing an advanced cognitive networking paradigm. A cognitive network is a network composed of elements that, through learning and reasoning, dynamically adapt to varying dynamical conditions in order to optimize end-to-end performances.

A novel plane, the DCP, will be the key architectural enabler: it will allow cross-layer, cross-node and cross-network domain self-management, self-control and self-optimization of resources (while being compatible with legacy management and control systems). Through this cognitive networking, value creation can occur through increased functionality of networks (e.g. always-best-connected-anywhere feature, personal devices networking, network MOO etc.). This value can migrate within the value-chain according to the context in which the devices/nodes/networks are operating in, and the time-varying performance objectives.

The dynamics of value migration within the value chain are complex and difficult to estimate: in any case it is reasonable to admit that the overall value of a cognitive network is potentially much greater (both for the Operator and the Users joining

actively by sharing their resources) than that of the traditional architecture, even taking higher manufacturing and consumer costs into account [26]. In any case, the DCP will enable both a new generation of privately owned and community networks (thus allowing a potential split of local and global connectivity costs) and new business opportunities in synergy with the Internet of Things.

ACKNOWLEDGEMENT

The authors wish to thank Erol Gelenbe, Ricardo Lent (Imperial College of London) and Blaz Fortuna (Jozef Stefan Institut of Ljubljana) for discussions and comments that have contributed to the development of vision and perspectives reported in this paper.

REFERENCES

- [1] Gelenbe, E. (2006) Users and services in intelligent networks. *Proc. IEEE*, **153**, 213–220.
- [2] Ghanea-Hercock, R., Gelenbe, E., Jennings, N.R., Smith, O., Allsopp, D.N., Healing, A., Duman, H., Sparks, S., Karunatilake, N.C. and Vytelingum, P. (2007) Hyperion: next-generation battlespace information services. *Comput. J.*, **50**, 632–645.
- [3] Dobson, S. *et al.* (2006) Autonomic communications. *ACM Trans. Auton. Adapt. Syst.*, **1**, 223–259.
- [4] Papadimitriou, C. (2001) Algorithms, Games, and the Internet. *Proc. 33rd Annual ACM Symp. Theory of Computing*, Hersonissos, Greece, July 6–8, pp. 749–753. ACM, New York.
- [5] Gelenbe, E. (2009) Analysis of single and networked auctions. *ACM Trans. Internet Technol.*, **9**, 1–24.
- [6] Gelenbe, E., Xu, Z. and Seref, E. (1999) Cognitive Packet Networks. *Proc. 11th IEEE Int. Conf. Tools with Artificial Intelligence*, Chicago, IL, November 8–10, pp. 47–54. IEEE Computer Society, Washington.
- [7] Gelenbe, E. (2004) Cognitive Packet Network. U.S. Patent 6,804,201.
- [8] Gelenbe, E., Lent, R. and Nunez, A. (2004) Self-aware networks and QoS. *Proc. IEEE*, **92**, 1478–1489.
- [9] Gelenbe, E. (2009) Steps toward self-aware networks. *Commun. ACM*, **52**, 66–75.
- [10] Clark, D., Partridge, C., Ramming, J. and Wroclawski, J. (2003) A Knowledge Plane for the Internet. *Proc. 2003 Conf. Applications, Technologies, Architectures, and Protocols For Computer Communications (SIGCOMM'03)*. Karlsruhe, Germany, August 25–29, pp. 3–10. ACM, New York.
- [11] Lewis, D., O' Sullivan, D., Feeney, K., Keeney, J. and Power, R. (2006) Ontology-Based Engineering for Self-managing Communications. *Proc. 1st IEEE Int. Workshop on Modeling Autonomic Communications Environments*, Dublin, Ireland, October 25–26. <https://www.cs.tcd.ie/John.Keeney/pubs/MACE06-lewis.pdf> (accessed January 29, 2010).
- [12] Mamei, M. and Zambonelli, F. (2009) Programming pervasive and mobile computing applications: the TOTA approach. *ACM Trans. Softw. Eng. Methodol.*, **18**, 1–56.

- [13] Salden, A.H., ter Haar Romeny, B.M. and Viergever, M.A. (2001) A dynamic scale-space paradigm. *J. Math. Imaging Vis.*, **15**, 127–168.
- [14] Yoshida, A., Aoki, K. and Araki, S. (2005) Cooperative Control Based on Reaction-Diffusion Equation for Surveillance System. *Proc. 9th Int. Conf. Knowledge-Based and Intelligent Information and Engineering Systems (KES)*, Melbourne, Australia, September 14–16, Lecture Notes in Computer Science 3684, pp. 533–539. Springer, Berlin.
- [15] Gelenbe, E., Liu, P. and Lainé, J. (2006) Genetic algorithms for route discovery. *IEEE Trans. Syst. Man Cybern.*, **36**, 1247–1254.
- [16] Kirby, K.G. and Conrad, M. (1986) Intra-neuronal dynamics as a substrate for evolutionary learning. *Physica D*, **22**, 205–215.
- [17] Lewis, D., O’Sullivan, D. and Keeney, J. (2006) Towards the Knowledge-Driven Bench-Marking of Autonomic Communications. *Proc. 2006 Int. Symp. World of Wireless, Mobile and Multimedia Networks*, Niagara-Falls, NY, June 26–29, pp. 500–505. IEEE Computer Society, Washington.
- [18] Latré, S., Verstichel, S., De Vleschauwer, B., De Turck, F. and Demeester, P. (2009) On the Design of an Architecture for Partitioned Knowledge Management in Autonomic Multimedia Access and Aggregation Networks. *Proc. 4th IEEE International Workshop on Modelling Autonomic Communication Environments*, Venice, Italy, October 26–27, Lecture Notes in Computer Science 5844, pp. 105–110. Springer, Berlin.
- [19] Manzalini, A., Zambonelli, F., Baresi, L. and Di Ferdinando, A. (2009) The CASCADAS Framework for Autonomic Communications. In Vasilakos, A., Parashar, M., Karnouskos, S. and Pedrycz, W. (eds) *Autonomic Communication*. Springer, Berlin.
- [20] Dasgupta, D. (2007) Immuno-Inspired Autonomic System for Cyber Defense. Information Security Technical Report 12(4). Elsevier, Amsterdam.
- [21] Bourse, D. and El-Khazen, K. (2005) End-to-end reconfigurability (E2R) research perspectives. *IEICE Trans. Commun. (Special Section on Software Defined Radio Technology and Its Applications)*, **11** 4148–4157.
- [22] Demestichas, P., Stavroulaki, V., Boscovic, D., Lee, A. and Strassner, J. (2006) m@ANGEL: autonomic management platform for seamless cognitive connectivity to the mobile internet. *IEEE Commun. Mag.*, **4**, 118–127.
- [23] Sutton, P., Doyle, L.E. and Nolan, K.E. (2006) A Reconfigurable Platform for Cognitive Networks. *Proc. 1st Int. Conf. Cognitive Radio Oriented Wireless Networks and Communication*, Mykonos, Greece, June 8–10, pp. 1–5. IEEE Computer Society, Washington.
- [24] Gelenbe, E., Sakellari, G. and d’Arienzo, M. (2008) Admission of QoS aware users in a smart network. *ACM Trans. Auton. Adapt. Syst.*, **3**, 1–28.
- [25] Lake, S.M. (2005) Cognitive Networking with Software Programmable Intelligent Networks for Wireless and Wireline Critical Communications. *Proc. IEEE Military Communications Conf.*, Atlantic City, NJ, October 17–20, Vol. 3, pp. 1693–1699. IEEE Computer Society, Washington.
- [26] Nolan, K.E., Mullany, F.J., Ambrose, E. and Doyle, L.E. (2007) Value Creation and Migration in Adaptive Cognitive and Radio Systems. In Arslan, H. (ed.) *Cognitive Radio, Software Defined Radio, and Adaptive Wireless Systems*. Springer, Berlin.
- [27] Gelenbe, E. (2007) A diffusion model for packet travel time in a random multihop medium. *ACM Trans. Sensor Netw.*, **3**, 10.