Report from Dagstuhl Seminar 14512

Collective Adaptive Systems: Qualitative and Quantitative Modelling and Analysis

Edited by
Jane Hillston[1], **Jeremy Pitt**[2], **Martin Wirsing**[3], **and Franco Zambonelli**[4]

1 **University of Edinburgh, GB,** `Jane.Hillston@ed.ac.uk`
2 **Imperial College London, GB,** `j.pitt@imperial.ac.uk`
3 **LMU München, DE,** `wirsing@lmu.de`
4 **University of Modena, IT,** `franco.zambonelli@unimore.it`

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 14512 "Collective Adaptive Systems: Qualitative and Quantitative Modelling and Analysis". Besides presentations on current work in the area, the seminar focused on the following topics: (i) Modelling techniques and languages for collective adaptive systems based on the above formalisms. (ii) Verification of collective adaptive systems. (iii) Humans-in-the-loop in collective adaptive systems.

Seminar December 14–19, 2014 – http://www.dagstuhl.de/14512
1998 ACM Subject Classification C.2.4 Distributed Systems, D.2 Software Engineering, D.2.4 Software/Program Verification, H.1.2 User/Machine Systems
Keywords and phrases Collective Adaptive Systems, Qualitative and Quantitative Modelling and Analysis, Verification, Humans-In-The-Loop
Digital Object Identifier 10.4230/DagRep.4.12.68
Edited in cooperation with Lenz Belzner

1 Executive Summary

Jane Hillston
Jeremy Pitt
Martin Wirsing
Franco Zambonelli

Modern systems are often structured as complex, multi-layered networks of interconnected parts, where different layers interact and influence each other in intricate and sometimes unforeseen ways. It is infeasible for human operators to constantly monitor these interactions and to adjust the system to cope with unexpected circumstances; instead systems have to adapt autonomously to dynamically changing situations while still respecting their design constraints and requirements. Because of the distributed and decentralized nature of modern systems, this usually has to be achieved by collective adaptation of the nodes comprising the system. In open systems exhibiting collective adaptation, unforeseen events and properties can arise, e.g. as side effects of the interaction of the components or the environment. Modelling and engineering collective adaptive systems (CAS) has to take into account such "emergent" properties in addition to satisfying functional and quantitative requirements.

Collective Adaptive Systems: Qualitative and Quantitative Modelling and Analysis, *Dagstuhl Reports*, Vol. 4, Issue 12, pp. 68–113
Editors: Jane Hillston, Jeremy Pitt, Martin Wirsing, and Franco Zambonelli
DAGSTUHL REPORTS Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Finding ways to understand and design CAS, and to predict their behaviour, is a difficult but important endeavour. One goal of this seminar was to investigate techniques for modelling and analysing systems that adapt collectively to dynamically changing environment conditions and requirements. In many cases, these models and analysis techniques should not only capture qualitative properties of the system, such as absence of deadlocks, they should also be able to express quantitative properties such as quality of service.

Research on CAS builds on and integrates previous research efforts from several areas:

- Formal foundations and modelling techniques for concurrent systems deal with problems such as enabling and limiting concurrency, access to shared resources, avoidance of anomalies, communication between processes, and estimation of performance.
- Analysis of concurrent systems typically exploits such notions as bisimilarity of different processes or reasons on stochastic properties of systems consisting of many equivalent processes.
- The area of adaptive systems also investigates systems consisting of interacting entities, but is more concerned with the reaction of whole systems or individual actors in a system to a changing environment.

An important aim of this seminar was to combine research from concurrent systems with results from the adaptive systems community in order to develop formalisms for specifying CAS, to increase the scalability of qualitative and quantitative modelling and analysis techniques to large systems, and to apply them to systems that dynamically change their structure or adapt to novel situations.

The seminar was organised with a mixture of talks and working group sessions which facilitated more in-depth discussions and exploration of topics. In this report we include the abstracts of a selection of the presented talks, and three longer contributions compiled after the meeting which seek to reflect the activities of the working groups. The first group, considering modelling, specification and programming for CAS, start their presentation with brief descriptions of four diverse applications developed on the basis of CAS, ranging from national level power management to personal wearable devices. To complement this identification of application domains, the group also catalogued common and contrasting features that can be found in CAS. This consideration highlights the role of physical space in all the considered domains and the urgent need to develop modelling and analysis techniques which reflect this central role played by space. This was key amongst a number of challenges identified by the group in their conclusions. Spatio-temporal aspects were also identified as a key challenge by the second working group who considered verification of CAS. The report from this group outlines the role of verification within the design and management of CAS ranging from seeking to guarantee global emergent behaviour from local specifications to using online verification to drive adaptation. Two specific challenges were explored in more detail, namely handling the inherent uncertainty in CAS, and specification and verification of spatial properties of systems composed of self-organising patterns. The third working group focused on the issues that arise from the recognition that some of the entities within a CAS may be humans and outside technological control, i.e. the design of socio-technical systems. A number of different scenarios are provided to illustrate the difference between socio-technical CAS and ‘technical’ CAS, and the human factors which must be taken into account. To remediate some of the problems identified, the group propose the idea of a general intervention framework, based around the 3I life-cycle – inspection-innovation-intervention. It was foreseen that intervention would be achieved by shaping mechanisms, and the report goes on to describe some possible shaping mechanisms which were considered. To conclude a number of research challenges are discussed.

2 Table of Contents

Executive Summary

Overview of Talks

Working Group Reports

4 Working Group Reports

4.1 Modelling, Specification, and Programming for Collective Adaptive Systems

Hella Seebach, Lenz Belzner, Marco Gribaudo, Anabelle Klarl, Michele Loreti, Ugo Montanari, Laura Nenzi, Rocco De Nicola, Christophe Scholliers, Petr Tuma, and Martin Wirsing

4.1.1 Introduction

Over the last decades we have witnessed a steep growth in the world population. This increase has a vast impact in the large scale on how cities operate. For example, how to route traffic in the city and where to place parking spots in such a way that the individuals commuting time is minimized. On a smaller scale, big events such as festivals have to be able to predict how the crowd will react in case of a major incident. It is needless to say that each of the individuals at a festival are autonomous entities, yet it is surprising to see that certain patterns can be observed from the group as a whole. Systems consisting out of a large number of individuals exhibiting group behaviour are called collective adaptive systems (CAS). While the collective adaptive systems described above consist solely out of humans the idea is that these systems can consist both out of human entities and/or ICT components.

While our understanding of CAS is getting better over time, the field is not widely understood by the big audience. CAS are omnipresent in current society and it is thus essential to be able to provide the correct set of abstraction in order to model, verify and implement them.

In this paper we show four typical domains of CAS (Sec. 4.1.2). Afterwards we give a description of what collective adaptive system are and how they can be characterized (Sec. 4.1.3). Section 4.1.4 shows how CAS can be modelled and implemented on a computer system. From this overview we conclude that each of the non-trivial collective adaptive systems has a vast need to reason over spatio-temporal properties. Surprisingly, most of the modelling and implementation techniques, do not provide spatio-temporal operations as first class entities. This means that programmers must encode these properties themselves which is time consuming and prone to error. We thus argue that in order to better reason about collective adaptive systems it is essential to focus on these operations. We conclude this paper with perspectives on future research and propose a set of challenges for future researchers to tackle elegantly.

4.1.2 Application Domains

Collective adaptive systems can be found in a lot of different domains. Each application domain naturally leads to different characteristics of CAS which will need multiple new enabling technologies. We just discussed four domains in this workshop which perfectly fit for developing and evaluating techniques for CAS.

Power Management Systems

In current power management systems, big power plants are controlled by electric utilities and other organisations in a flat hierarchy. Utilities and companies manage parts of the overall power system independently from each other. For each of the big power plants, a schedule is created that postulates the output of the power plant at a given time. Schedules are coarse-grained, providing target values in 15 minute intervals. Small power plants and especially DERs (distributed energy resources) under the control of small cooperatives or individuals produce without external control and feed the power produced into the grid. This lack of control by the electric utilities is compensated by powerful controllable power plants. Current plans are to scale the controllable output further by installing more plants, especially flexible gas-powered ones. Geographical distribution is even increasing with the wide-spread installation of DERs such as biogas plants, solar plants, and wind farms. The relative stability in the network is an emergent behaviour. No single entity of the system can provide this stability in the face of load fluctuations, weather changes, and generator outages.

What makes power systems difficult to manage and optimize is the cost of storing energy: since energy consumption varies remarkably with time (day, week and season), the unit cost of production varies also, because less efficient plants are turned on only at peak time. On the other hand, DERs production capacity is also heavily dependent on weather conditions, and thus quite variable in time. Physical storage systems (hydro pumping stations, batteries, e.g. of electric vehicles) are not very practical, thus the best policy is to try to match consumption and production in an integrated or unbundled market. The goal is to make demand active, trying to move it, whenever possible, to slots where energy is less expensive. Therefore, there is a need for a future power grid in which even small power plants, consumers, as well as prosumers (entities that produce and consume power like an electric vehicle) can be controlled or participate in a scheduling scheme or market, since entrance requirements to power markets, such as the lower limit of 100 kW for contracts at the European Energy Exchange (EEX), exclude access for small organisations. Networked measuring equipment must be equipped to allow observing the grid status and make decisions based on current conditions. Power plants and consumers will be networked, too, and provide future production or consumption. Producers, consumers, and prosumers must be combined into groups, e.g., as aggregators [1] or Autonomous Virtual Power Plants (AVPPs) [2],[3],[4] that create schedules to cover a portion of the load (depending on the demand) aiming at: (i) lowering peak energy consumption by exploiting their flexibility; (ii) reducing electricity cost for the whole population of actors; (iii) increasing robustness by locally dealing with load and output fluctuations; and (iv) making a profit. Remaining research challenges comprise the robust autonomous scheduling of large-scale open heterogeneous systems (including spatial and temporal constraints) as well as security, privacy, and safety aspects, amongst others.

Cloud Computing

Contemporary cloud computing platforms rely on server farms that host a number of dedicated workload processing servers together with the necessary networking and storage infrastructure. The infrastructure does not expose the details of virtual server location at the level of individual server racks or individual network ports – these are managed transparently by the platform provider, possibly using mechanisms such as virtual machine migration [5] or software defined networks [6]. In contrast, higher granularity location is exposed – complex application deployment scenarios use it to make sure that both application code and application data is distributed appropriately to accommodate the (often conflicting) requirements on communication efficiency, failure resiliency, cost and other factors.

Although many cloud computing applications adopt somewhat conservative resource management techniques (such as limiting dynamic server allocation to manually selected server farms), many research directions seek to amplify the existing cloud computing benefits by introducing mechanisms such as cloud federations or ad hoc and opportunistic clouds [7, 8].

On many levels, these directions strengthen the collective adaptive system characteristics of the cloud. There are multiple focus areas for research in the domain of cloud computing. *Efficient resource allocation:*Both the cloud platform and the cloud applications seek to maximize utility and minimize cost by sharing resources (servers, storage, network). Efficient resource allocation is a collective task where multiple adaptive entities (platform components and application components) allocate and release resources to meet their specific requirements. The domain offers research challenges in both cooperative and competitive resource allocation algorithms in presence of changing requirements [9, 10, 11, 12, 13, 14, 15]. In turn, these contain challenges in monitoring and predicting the impact of particular resource allocation, needed to perform allocation decisions. *Robustness against failures:* Especially in an open cloud with voluntary participation, node failures and application failures are expected rather than exceptional situations. The domain requires research into efficient failure resilient algorithms, behaviour modelling in presence of (possibly dependent) failures and other challenging problems [16]. *Security against abuse:* As an open environment with heavy resource sharing, cloud computing exposes many opportunities for abuse. These include not only the more traditional security related issues (virtual machine hijacking, data theft and other), but also the possibility of using the available resources beyond fair share or outright free-loading. There is a need for strategies and mechanisms that would prevent such abuse [17, 18, 19, 20]. *Preventing negative emergence:* The cloud environment incorporates entities whose behaviour is largely automated but rarely fully disclosed or even fully understood. Such an environment is easily prone to emergent behaviour with negative consequences, for example oscillations in the adaptive feedback mechanisms. The domain can benefit from research into preventing, detecting or managing cases of negative emergence.

As another practical benefit of the cloud computing domain, the difficulty of the identified challenges varies with the degree of openness and heterogeneity that is considered – a centralized resource allocation in a closed homogeneous cloud faces different issues than a cooperative distributed resource allocation in an open heterogeneous cloud with voluntary participation. Although multiple projects already started tackling some of the listed challenges [21, 22, 23, 24, 25, 26, 27, 28], there are still many topics waiting for advances.

Telecommunication – LTE Resource Allocation

LTE technology, used in 4G mobile phone communications, employs a channel resource allocation scheme based on orthogonal frequency-division multiplexing. In particular it supports for both time-division multiplexing and frequency-division multiplexing, splitting the communication spectrum in a set of resource blocks.

Each participating device is equipped by multiple antennas, and can transmit on more frequencies at the same time. The total bandwidth available to a device depends on the number of blocks it can use to transmit/receive data. The LTE technology envisage the possibility of dynamically allocating the resource blocks depending on the actual demand, to improve the performances that can be achieved by the single user. The service provider usually operates block allocation in a centralized manner: this however limits the bandwidth that could be achieved since interference might arise from the carrier being shared by different providers. An autonomous solution, where each mobile agent can acquire and release block resources, could improve the available bandwidth overcoming these difficulties. This however is not an easy task due to the characteristics of the wireless medium that is affected by limitations such as the hidden terminal problem [29]. The problem can become even more interesting when the cellular infrastructure is complemented with alternative wireless access

<table>
<tr><th></th><th>Homogeneous
Heterogeneous</th><th>Collaborative
Competitive</th><th>Low ind. impact
High ind. impact</th><th colspan="4"></th></tr>
<tr><td>Cloud</td><td><------X</td><td>|--------|</td><td>|-----------|</td><td rowspan="4">Space</td><td rowspan="4">Sync / async</td><td rowspan="4">Continuous / discrete</td><td rowspan="4">Open / closed</td></tr>
<tr><td>Wristband</td><td>X</td><td>|-|</td><td>|-|</td></tr>
<tr><td>4G LTE</td><td><-X</td><td>|--|</td><td>|------------|</td></tr>
<tr><td>Power grids</td><td><---X</td><td>|-------------|</td><td>|---------|</td></tr>
</table>

Figure 1 Categorization of the considered applications.

technologies such as WiFi hotspots [30]. In this way data traffic can be offloaded whenever possible towards such hotspots, at the price of a possible degradation in the quality of service experienced by the users [31, 32]. Preliminary studies of such systems using CAS-based techniques have been proposed in [33, 34]. The key solutions in this direction will also be the basis to the next generation of wireless and cellular communications that exploits more advanced techniques such as *cognitive radio* [35] in which terminals and base stations harvest for unused radio frequencies and spectrum bandwidths to increase their transmission capacity.

Wearable Computational Devices

Due to the advances in hardware technology and the miniaturisation of electronic components it has become feasible to make wearable computational devices. These wearable devices open up opportunities for new and exciting applications. Also in the world of collective adaptive systems this new technology can be exploited. One particular application is the use of wristbands equipped with wireless near field communication. When a large number of people are equipped with such wristbands these wristbands could light up in order to provide additional functionalities to the users. One application of these wristbands could be to make figures at large scale events by lighting up the wristbands at synchronised moments in time. Additionally the same wristbands could be used at mass events to drive people to the exit in case of disaster. Challenging is amongst others the situation of simple, limited nodes (simple communication, limited resources) and the unpredictable position of the nodes.

4.1.3 Common Features and Characteristics of Application Domains

In the workshop, we identified multiple characteristics for CAS. The four mentioned application domains can be categorized against a set of different features that will guide in the selection of the most appropriate modelling technique. Figure 1 summarizes the results.

The first aspect we considered is the type of elements that compose the CAS. The elements can be *homogeneous*: all the cooperating entities can be considered to be identical. This is for example the case of the bracelets in the wristband application, where all the devices are exactly the same. *Heterogeneous* applications are instead composed of agents that are completely different one from the other. A typical example is the power grid scenario, where each producer or consumer is completely different from the others, of course depending on the level of abstraction. Both the cloud and the LTE scenario have some degree of heterogeneity due to the fact that they are composed by different devices, produced by different manufacturers. However all the devices are abstracted by the *role* they are playing in the application: this allows us to consider them homogeneous from a modelling perspective.

The second aspect we discussed is whether the agents are *collaborative* or *competitive*. For example, the wristband application is a clear example of a collaborative system: each agent cannot gain any advantage by not cooperating with the others. The LTE network is instead an example of a competitive application, where each mobile device tries to acquire all the available bandwidth to improve its communication performance. Cloud computing can be either collaborative or competitive depending on the specific application we are considering. A Big Data application might be competitive to gain more resource to parallelize its execution and to reduce its running time. A Platform-as-a-service job can instead be consolidated with other applications to increase the chance of having idle machines that can be switched off, reducing the total energy consumption. The prosumer in the power management systems are first and foremost competitive to optimize their benefit. But in future energy grid scenarios they have to collaborate in organisational structures to be able to participate for example on the energy market.

The third feature is the impact that a local agent can have on the entire community. In the wristband application it can be minimal, since an agent can at most do not propagate a message and do not properly switch the colour of the bracelet. In the power grid example the impact is instead maximum, since other nodes might be relying on the production or consumption of energy of other participants in the network. 4G LTE might have a limited impact, since most of devices are autonomous. However, the presence of a shared environment (the communication spectrum) can have an impact on a large set of devices in case of malicious signals that could be generated to interfere with the regular transmissions. The particular characterization of the cloud-computing scenario depends on the considered application since it can have either a low impact (as the exclusion of a physical node in an Infrastructure-as-a-service scenario), or a high impact (for example when the shut down of a node in a distributed storage application makes a file no longer accessible).

Other important features that characterize an application from a modelling point of view are the presence of space, the fact of being synchronous or asynchronous, discrete or continuous, open or closed. All the examples that we consider in this work rely somehow on a concept of *space*. Space can be either a physical space or a logical space. For example, both the stadium where the concert is held in the wristband application, or the area where base stations are located in the LTE application, are examples of physical spaces. The nodes interconnected by the distribution network in the power grid application, and the servers and routers in the cloud application, are examples of logical space. In both cases the system topology can be described by a graph, where nodes of the system correspond to nodes on the graph. All the considered applications are *asynchronous*, but they build up some level of *synchronism* to allow the system to reach their goals: depending on the level of abstraction that we want to consider, different modelling techniques can be used to specifically target their synchronous or asynchronous features. Most of the applications are *discrete*, in the sense that they change their state according to events that happens in discrete time instants. The power grid application however, requires a *continuous* time approach since problems can arise and must be handled in few milliseconds. This leads to the requirement of radically different modelling techniques. Finally applications can be considered either *open* or *closed*. The wristband is a classical example of a closed application. Also in this case, depending on the level of abstraction and on the features we are interested to consider, different modelling techniques could be employed.

4.1.4 Methods to approach CAS

We identified three different main approaches to model CAS: systems of systems, autonomy, and aggregation. They are not exclusive; rather they should be regarded as a kind of dimensions that a particular system or solution exhibits. Afterwards we discussed several modelling techniques for the different levels of abstraction.

Systems of Systems (Roles)

A core characteristic of CAS to be modelled are behavioural and communicational aspects, both on the individual and the collective level. A key challenge here is the specification of organisational, communicational or behavioural collectives. Also, modelling these structures has to take into account reconfiguration of collectives at runtime due to changing situations (i.e. adaptation). The properties stated above lead to the idea of considering systems of systems [36] as an appropriate way of modelling CAS. The collectives may be organized in hierarchical or overlapping ways, also depending on spatial aspects. In the workshop, aggregation and organization based on roles, communication patterns and spatio-temporal properties have been discussed.

Autonomy (Reasoning)

One way to drive adaptation is to provide learning and planning capabilities to individuals and collectives alike. Reasoning and planning provide ways for autonomic system reorganization according to current needs and system goals. In the context of CAS, this gives rise to questions about individual and collective knowledge gathering and transformation as well reasoning capabilities. In especial, if considering systems of systems the question arise how to compare and evaluate systems e.g. on different or equal levels of hierarchies [37] or in different locations leading to different circumstances.

Aggregation (Quantification)

CAS may consist of extreme numbers of individuals. Also, these numbers may be unknown at design-time and/or runtime. Quantitative analysis approaches identify and abstract away symmetries and structure of the system in order to allow for efficient computation of system properties. While this scalability is highly desirable, it comes at the cost of specializing towards a particular problem or situation – quantitative approaches are strongly coupled to the way a CAS is modelled. Thus, they should drive modelling approaches as well as respect any abstractions made when modelling CAS. Most of the techniques in this field rely on mean field solutions [38, 39]: the system is studied by considering variables that counts the number of elements in the same state, and by studying their evolution using a set of ordinary differential equations. The mean field approximation basically states that a large number of objects that randomly evolve tend to show a deterministic mean behaviour as their count tends to infinity. On this assumption, many higher level modelling techniques based on process algebra [40, 41], or Markovian agents [42] have been developed.

Each of the approaches already provides solutions for problems studied under particular aspects. What remains a mostly open challenge is the combination of modelling and solutions from the different perspectives as well as the integration of spatio-temporal aspects in the mentioned techniques. For example, modelling and collective organization formalisms have to (a) provide methods for integration of reasoning in the modelling process and (b) allow for autonomous, goal- or situation-based reconfiguration. On the other hand, reasoning has to account for structural changes, and has to infer about the collective structure. Also, it seems an interesting challenge how different quantitative approaches could be instrumented autonomously based on current system configuration and accounting for autonomous reconfiguration at runtime.

Modelling CAS at different level of abstractions

Different languages have been proposed or used to support modelling, analysis and deployment of CAS. Some of these are general purpose languages, like Java, that, while providing appropriate API, can be used to program and deploy systems also on complex distributed infrastructures. Others languages are domain specific [43, 44, 45, 46, 47] and are equipped with syntactic constructs specifically thought for modelling relevant aspects of CAS. These domain specific languages are typically more oriented to specification than to deployment and provide formal and automatic tools that can be used to support analysis.

This variety of tools and languages can be used by a CAS designer to tackle the system modelling at different level of abstractions. Indeed, each language can be used to describe and analyse a system from different perspective. However, to take a real advantage from this plethora of tools formal links between the considered formalism are definitively needed. The links, that can be rendered in terms of *model transformation*, will be first of all used to relate the different models described in the different languages. These relations will be then instrumental to use the results of analysis performed in a given language to update/improve the other models.

4.1.5 Conclusion and Open Challenges

After we discussed the different application domains for CAS and considered suitable modelling and specification techniques we identified a set of challenges that must be tackled before CAS will be integrated in today's or future ICT systems. As one main result the working group came to the conclusion that spatio-temporal properties are of great value for the modelling and implementation of CAS but are not yet appropriately integrated in the available methods. One concrete challenge: Investigation of the design of CAS learning spatio-temporal requirements. The idea is to optimise the specification and implementation of the space features in the model in such a way that the satisfiability of a spatio-temporal property is maximised. As we know, the verification of global properties on CAS is often an intractable task from a computational point of view. For this reason, such properties will have to be decomposed in a set of local requirements in the optimisation process.

Further the participants of CAS need methods to reason about local versus global or even conflicting goals of the system. These decisions strongly depend on the organisational structures and the presence or absence of central institutions in the CAS. One concrete challenge: When describing/implementing CAS two aspects are crucial: The specification of the actual actions that the different components have to perform (behavioural specification) and the specification of the goal that the single components or the collectives have to achieve (goal specification). Usually these two kinds of activities are performed by taking advantage of very different tools, the former are performed with classical imperative programming language while the latter rely on declarative specifications. The foreseen challenge is the reconciliation of these two approaches to, e.g., able to take decisions about the next actions after having measured how far the goal is and what is the best choice to get closer to it.

Quantitative approaches for modelling and analysis of CAS help to meet the challenge of state space explosion if considering large-scale CAS. Beside the mean field solutions outlined in the previous section, new physically inspired techniques could be applied. One example could come from fluid dynamics, leading to *fluid approximations to CAS modelling.* If we consider a very large number of agents, densely packed, their evolution can be approximated as the motion of a fluid. To give an idea, let us imagine that agents can evolve through a finite set of modes $i \in \{1, \dots, M\}$. Let us also focus on two-dimensional space where agents can evolve. The state of whole system at time t can be characterized by a set of functions $p_i(x, y, t)$ that describes the density of agents (measured in agents per unit area) in state i at position (x, y). The evolution of $p_i(x, y, t)$ can be described by a set of partial differential equations, similar to the one used by the mass continuity law in fluid dynamics. From the state density $p_i(x, y, t)$ several performance metrics can be derived. These measures can be used to assess several properties, such as for example determining if an emergent behaviour can appear, and which could be the convergence rate as function of the parameters of the model.

References

1 Peeters, E., Belhomme, R., Batlle, C., Bouffard, F., Karkkainen, S., Six, D., Hommelberg, M.: ADDRESS: scenarios and architecture for active demand development in the smart grids of the future. In: Electricity Distribution-Part 1, 2009. CIRED 2009. 20th International Conference and Exhibition on, IET (2009) 1–4

2 Steghöfer, J.P., Anders, G., Siefert, F., Reif, W.: A system of systems approach to the evolutionary transformation of power management systems. In: GI-Jahrestagung. (2013) 1500–1515

3 Bremer, J., Rapp, B., Sonnenschein, M.: Encoding distributed search spaces for virtual power plants. In: Computational Intelligence Applications In Smart Grid (CIASG), 2011 IEEE Symposium on. (April 2011) 1–8

4 Becker, B., Allerding, F., Reiner, U., Kahl, M., Richter, U., Pathmaperuma, D., Schmeck, H., Leibfried, T.: Decentralized energy-management to control smart-home architectures. In Müller-Schloer, C., Karl, W., Yehia, S., eds.: Architecture of Computing Systems – ARCS 2010. Volume 5974 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2010) 150–161

5 Clark, C., Fraser, K., Hand, S., Hansen, J.G., Jul, E., Limpach, C., Pratt, I., Warfield, A.: Live migration of virtual machines. In: Proceedings of NSDI 2005, USENIX

6 Casado, M., Koponen, T., Ramanathan, R., Shenker, S.: Virtualizing the network forwarding plane. In: Proceedings of PRESTO 2010, ACM

7 Grozev, N., Buyya, R.: Inter-cloud architectures and application brokering: Taxonomy and survey. Software: Practice and Experience (2014)

8 Kirby, G., Dearle, A., Macdonald, A., Fernandes, A.: An approach to ad hoc cloud computing

9 Warneke, D., Kao, O.: Exploiting dynamic resource allocation for efficient parallel data processing in the cloud. IEEE Transactions on Parallel and Distributed Systems **22**(6) (2011)

10 Ishakian, V., Sweha, R., Bestavros, A., Appavoo, J.: CloudPack. In: Proceedings of MIDDLEWARE 2012, Springer

11 Wang, J., Hua, R., Zhu, Y., Wan, J., Xie, C., Chen, Y.: RO-BURST: A robust virtualization cost model for workload consolidation over clouds. In: Proceedings of CCGRID 2012, IEEE

12 Beloglazov, A., Abawajy, J., Buyya, R.: Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Journal of Future Generation Computing Systems **28**(5) (2012)

13 Li, J., Shuang, K., Su, S., Huang, Q., Xu, P., Cheng, X., Wang, J.: Reducing operational costs through consolidation with resource prediction in the cloud. In: Proceedings of CCGRID 2012, IEEE

14 Dong, J., Jin, X., Wang, H., Li, Y., Zhang, P., Cheng, S.: Energy-saving virtual machine placement in cloud data centers. In: Proceedings of CCGRID 2013, IEEE

15 Raghavendra, R., Ranganathan, P., Talwar, V., Wang, Z., Zhu, X.: No “power” struggles: Coordinated multi-level power management for the data center. In: Proceedings of ASPLOS 2008, ACM

16 Khalifa, A., Azab, M., Eltoweissy, M.: Resilient hybrid mobile ad-hoc cloud over collaborating heterogeneous nodes. In: Proceedings of COLLABORATECOM 2014, IEEE

17 Velloso, P.B., Laufer, R.P., de O. Cunha, D., Duarte, O.C.M.B., Pujolle, G.: Trust management in mobile ad-hoc networks using a scalable maturity-based model. IEEE Transactions on Network and Service Management **7**(3) (2010)

18 Huang, D., Zhang, X., Kang, M., Luo, J.: MobiCloud: Building secure cloud framework for mobile computing and communication. In: Proceedings of SOSE 2010, IEEE

19 He, Q., Wu, D., Khosla, P.: SORI: A secure and objective reputation-based incentive scheme for ad-hoc networks. In: Proceedings of WCNC 2004, IEEE

20 Buchegger, S., Boudec, J.Y.L.: Self-policing mobile ad-hoc networks by reputation systems. IEEE Communications **43**(7) (2005)

21 Bernard, Y., Klejnowski, L., Hähner, J., Müller-Schloer, C.: Towards trust in desktop grid systems. In: 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, CCGrid 2010, 17-20 May 2010, Melbourne, Victoria, Australia. (2010) 637–642

22 Bulej, L., Bureš, T., Horký, V., Keznikl, J.: Adaptive deployment in ad-hoc systems using emergent component ensembles: Vision paper. In: Proceedings of ICPE 2013, ACM

23 Mayer, P., Velasco, J., Hennicker, R., Puviani, M., Tiezzi, F., Pugliese, R., Keznikl, J., Bureš, T. In: The Autonomic Cloud. Springer (2015)

24 Amoretti, M., Grazioli, A., Senni, V., Zanichelli, F.: Towards a formal approach to mobile cloud computing. In: Proceedings of EUROMICRO PDP 2014

25 Sebastio, S., Amoretti, M., Lluch-Lafuente, A.: A computational field framework for collaborative task execution in volunteer clouds. In: Proceedings of SEAMS 2014, ACM

26 Celestini, A., Lluch-Lafuente, A., Mayer, P., Sebastio, S., Tiezzi, F.: Reputation-based cooperation in the clouds. In: Proceedings of IFIP TM 2014

27 Vassev, E., Hinchey, M., Mayer, P.: Formalizing self-adaptive clouds with knowlang. In: Proceedings of ISoLA 2014

28 Klarl, A., Mayer, P., Hennicker, R.: HELENA@Work: Modeling the science cloud platform. In: Proceedings of ISoLA 2014

29 Tsertou, A., Laurenson, D.I.: Revisiting the hidden terminal problem in a csma/ca wireless network. Mobile Computing, IEEE Transactions on **7**(7) (2008) 817–831

30 Trullols, O., Fiore, M., Casetti, C., Chiasserini, C., Ordinas, J.B.: Planning roadside infrastructure for information dissemination in intelligent transportation systems. Computer Communications **33**(4) (2010) 432–442

31 Han, B., Hui, P., Kumar, V.A., Marathe, M.V., Pei, G., Srinivasan, A.: Cellular traffic offloading through opportunistic communications: A case study. In: Proceedings of the 5th ACM Workshop on Challenged Networks. CHANTS '10, New York, NY, USA, ACM (2010) 31–38

32 Han, B., Hui, P., Kumar, V.S.A., Marathe, M.V., Shao, J., Srinivasan, A.: Mobile data offloading through opportunistic communications and social participation. IEEE Transactions on Mobile Computing **11**(5) (May 2012) 821–834

33 Gribaudo, M., Manini, D., Chiasserini, C.: Studying mobile internet technologies with agent based mean-field models. In: Analytical and Stochastic Modelling Techniques and Applications – 20th International Conference, ASMTA 2013, Ghent, Belgium, July 8–10, 2013. Proceedings. (2013) 112–126

34 Chiasserini, C., Gribaudo, M., Manini, D.: Traffic offloading/onloading in multi-rat cellular networks. In: Proceedings of the IFIP Wireless Days, WD 2013, Valencia, Spain, November 13-15, 2013. (2013) 1–7

35 Mitola, J., Maguire, G.Q., J.: Cognitive radio: making software radios more personal. Personal Communications, IEEE **6**(4) (Aug 1999) 13–18

36 Sage, A.P., Cuppan, C.D.: On the systems engineering and management of systems of systems and federations of systems. Inf. Knowl. Syst. Manag. **2**(4) (December 2001) 325–345

37 Schiendorfer, A., Steghöfer, J.P., Reif, W.: Synthesis and abstraction of constraint models for hierarchical resource allocation problems. In: Proc. of the 6th International Conference on Agents and Artificial Intelligence (ICAART). Volume 2.

38 Benaïm, M., Boudec, J.L.: A class of mean field interaction models for computer and communication systems. Perform. Eval. **65**(11-12) (2008) 823–838

39 Bobbio, A., Gribaudo, M., Telek, M.: Analysis of large scale interacting systems by mean field method. In: Fifth International Conference on the Quantitative Evaluaiton of Systems (QEST 2008), 14-17 September 2008, Saint-Malo, France. (2008) 215–224

40 Bortolussi, L., Hillston, J., Latella, D., Massink, M.: Continuous approximation of collective system behaviour: A tutorial. Perform. Eval. **70**(5) (2013) 317–349

41 Hayden, R.A., Stefanek, A., Bradley, J.T.: Fluid computation of passage-time distributions in large markov models. Theor. Comput. Sci. **413**(1) (2012) 106–141

42 Cerotti, D., Gribaudo, M., Bobbio, A.: Markovian agents models for wireless sensor networks deployed in environmental protection. Rel. Eng. & Sys. Safety **130** (2014) 149–158

43 De Nicola, R., Loreti, M., Pugliese, R., Tiezzi, F.: A formal approach to autonomic systems programming: The SCEL language. TAAS **9**(2) (2014) 7

44 Feng, C., Hillston, J.: PALOMA: A process algebra for located markovian agents. In Norman, G., Sanders, W.H., eds.: Quantitative Evaluation of Systems – 11th International Conference, QEST 2014, Florence, Italy, September 8–10, 2014. Proceedings. Volume 8657 of Lecture Notes in Computer Science., Springer (2014) 265–280

45 Alrahman, Y.A., Nicola, R.D., Loreti, M., Tiezzi, F., Vigo, R.: A calculus for attribute-based communication. In: Proceedings of the ACM Symposium on Applied Computing, SAC 2015, ACM (2015) To appear.

46 Bortolussi, L., Nicola, R.D., Galpin, V., Gilmore, S., Hillston, J., Latella, D., Loreti, M., Massink, M.: Caspa: a collective adaptive stochastic process algebra. In Bertrand, N., Tribastone, M., eds.: Proceedings Twelfth International Workshop on Quantitative Aspects of Programming Languages and Systems, QAPL 2015, April 2015. EPTCS (2014) To appear.

47 Viroli, M., Damiani, F.: A calculus of self-stabilising computational fields. In Eva Kühn, Pugliese, R., eds.: Coordination Models and Languages – 16th IFIP WG 6.1 International Conference, COORDINATION 2014, Held as Part of the 9th International Federated Conferences on Distributed Computing Techniques, DisCoTec 2014, Berlin, Germany, June 3–5, 2014, Proceedings. Volume 8459 of Lecture Notes in Computer Science., Springer (2014) 163–178

4.2 Verification of CAS

Luca Bortolussi, Giacomo Cabri, Giovanna Di Marzo Serugendo, Vashti Galpin, Jane Hillston, Roberta Lanciani, Mieke Massink, Mirco Tribastone, and Danny Weyns

Verification is the process of assessing how well a system meets a specification or requirement. A variety of approaches have appeared in the literature, ranging from model checking to