



Activity Report 2014

Exploratory Action CTRL-A

Control Techniques for Autonomic, Adaptive
and Reconfigurable Computing systems

RESEARCH CENTER
Grenoble - Rhône-Alpes

Table of contents

1. Members	1
2. Overall Objectives	1
2.1. Objective: control support for autonomic computing	1
2.2. Motivation: safe and optimal autonomic management	2
2.2.1. The problem of automating computing systems administration	2
2.2.2. Our model-based control approach	3
2.2.3. Opportunities between computer science and control theory	3
3. Research Program	3
3.1. Modeling and control techniques for autonomic computing	3
3.1.1. Continuous control	3
3.1.2. Discrete control	4
3.2. Design and programming for autonomic computing	4
3.2.1. Reactive programming	4
3.2.2. Component-based approach and domain-specific languages	4
3.3. Infrastructure-level support for autonomic computing	4
3.3.1. Software and adaptive systems	5
3.3.2. Hardware and reconfigurable architectures	5
3.3.3. Applications and autonomic systems	5
4. Application Domains	5
4.1. Distributed systems and High-Performance Computing	5
4.2. Reconfigurable architectures in embedded systems	6
4.3. Smart environments and Internet of Things	6
5. New Software and Platforms	6
6. New Results	7
6.1. Highlights of the Year	7
6.2. Discrete control and reactive language support	7
6.3. Design and programming	8
6.3.1. Component-based approaches	8
6.3.2. Rule-based systems	8
6.4. Infrastructure-level support	9
6.4.1. Autonomic Cloud and Big-Data systems	9
6.4.1.1. Coordination in multiple-loop autonomic Cloud systems	9
6.4.1.2. Control for Big data	9
6.4.2. Reconfiguration control in DPR FPGA	9
6.4.3. Autonomic memory management in HPC	10
6.4.4. Control of smart environments	10
6.4.4.1. Generic supervision architecture	10
6.4.4.2. Rule-based specification	10
7. Bilateral Contracts and Grants with Industry	11
7.1. CIFRE PhD grant Orange	11
7.2. Bilateral Grants with Industry	11
8. Partnerships and Cooperations	11
8.1. Regional Initiatives	11
8.1.1. Projet Exploratoire STAARS	11
8.1.2. Equipe-action HPES	11
8.2. National Initiatives	11
8.2.1. ANR	11
8.2.2. Informal National Partners	12
8.2.3. Informal National Industrial Partners	12

8.3. International Initiatives	12
8.3.1. Inria International Partners	12
8.3.2. Participation In other International Programs	12
9. Dissemination	12
9.1. Promoting Scientific Activities	12
9.1.1. Scientific events organisation	12
9.1.2. Scientific events selection	13
9.1.2.1. responsible of the conference program committee	13
9.1.2.2. member of the conference program committee	13
9.1.2.3. reviewer	13
9.1.3. Journal	13
9.1.4. Invited Conferences	13
9.2. Teaching - Supervision - Juries	13
10. Bibliography	14

Exploratory Action CTRL-A

Keywords: Automatic Control, Autonomic Computing, Component Programming, Programming Languages, Self-adaptive Systems

Ctrl-A is an Inria team, based in Grenoble, exploring a scientific theme at the intersection between, on the one hand, computer science and the topics of the CNRS UMR LIG (Laboratoire d'Informatique de Grenoble), and on the other hand, control theory, specialty of the CNRS UMR Gipsa-lab (Grenoble Images Parole Signal Automatique). Given that a joint team between the two can not be built from a CNRS point of view, Ctrl-A is an Inria team independent of them, and features many of its permanent members from each of the laboratories under the status of external collaborators (in the "Others" category below).

Creation of the Exploratory Action: 2014 January 01.

1. Members

Research Scientist

Eric Rutten [Team leader, Inria, Researcher, HdR]

Post-Doctoral Fellows

Frederico Alvares de Oliveira Junior [Inria]

Julio Angel Cano Romero [Inria, until Mar 2014]

Administrative Assistants

Françoise de Coninck [Inria, until Jul 2014]

Carol Loquet [Inria, from Sep 2014]

Others

Gwenaël Delaval [Univ. Grenoble I, Associate Professor, LIG]

Stéphane Mocanu [INP Grenoble, Associate Professor, Gipsa-lab]

Bogdan Robu [Univ. Grenoble I, Associate Professor, Gipsa-lab]

Hassane Alla [Univ. Grenoble I, Professor, Gipsa-lab]

Nicolas Marchand [CNRS, Senior Researcher, Gipsa-lab]

Mihaly Berekmeri [Univ. Grenoble I, PhD, Gipsa-lab]

Soguy Mak-Kare Gueye [Univ. Grenoble I, PhD]

Adja Sylla [CEA, PhD, from end 2014]

Mengxuan Zhao [Cifre Orange labs, PhD]

Naweiluo Zhou [Univ. Grenoble I, PhD]

2. Overall Objectives

2.1. Objective: control support for autonomic computing

Computing systems are more and more ubiquitous, at scales from tiny embedded systems to large-scale cloud infrastructures. They are more and more adaptive and reconfigurable, for resource management, energy efficiency, or by functionality. Furthermore, these systems are increasingly complex and autonomous: their administration cannot any longer rely on a strong interaction with a human administrator. The correct design and implementation of automated control of the reconfigurations and/or the tuning is recognized as a key issue for the effectiveness of these adaptive systems.

In the last dozen years, the notion of Autonomic Computing has been proposed and supported by industrials like IBM, as a framework for the design of self-adaptive systems. It addresses objectives of self-configuration, w.r.t. deployment issues, self-optimization, w.r.t. resources management, self-healing, w.r.t. robustness and fault-tolerance, and self-protection, w.r.t. security aspects. It relies on a feedback control loop architecture, with: monitors and reconfiguration actions, connected to the API infrastructure of the system under control ; an autonomic management component , transforming flows of monitoring information into adaptation actions, which can be addressed naturally by reactive programming ; a decision mechanism inside the latter manager, which can rely on behavioral models of the managed system.

Our objective is to build methods and tools for the design of safe controllers for autonomic, adaptive, reconfigurable computing systems. To attain this goal, we propose to combine Computer Science and Control Theory, at the levels of systems infrastructures, programming support, and modeling and control techniques. We explore this topic along three main thematic axes: **modeling and control theory** with discrete time, continuous time and hybrid models (distributed control, event-based control, discrete event systems, supervisory control) ; **programming support** with reactive (synchronous languages, controller synthesis, higher-order) and component-based approaches (Fractal framework, language-level support of reconfiguration, Event-Condition-Action (ECA) rules) ; **infrastructure-level support** (operating system, middleware) for monitors/sensors, administration actions/actuators, architectures for controllability, software and hardware reconfiguration mechanisms.

We aim to address applications with reconfiguration control problems from the small scale (facing variability for system on chips (SoCs), reconfigurable architectures, networks on chips (NoCs), etc.) up to the extra large scale (administration, coordination, optimization of data centers and cloud computing, green computing, smart grids, etc.)

We propose to form a team grouping the most active community in France on Control of Computing, with members until now separated by laboratories structure, in order to contribute more efficiently in the local context to the high potential impact on micro- and nanotechnologies in Grenoble, and more widely nationally and internationally in the emerging community on Feedback Computing.

2.2. Motivation: safe and optimal autonomic management

2.2.1. The problem of automating computing systems administration

It lies in the difficulty of manual management in a safe or optimal way, of computing systems which become more and more complex and flexible. There is a deep need for the automation of their management, handled in a closed-loop: the system is monitored by sensors, which enable updating a self-representation of the system, upon which reconfiguration actions are decided, and in turn they are executed, with an effect on the system, that will be measured by sensors. Such dynamically reconfigurable systems, also called adaptive or autonomic computing systems, are characterized by the ability to modify, on-line, their computing structure, in reaction to conditions in their execution environment or platform.

Motivations for dynamic adaptivity are in important questions like : resource management e.g., energy, computation, memory, bandwidth, circuit area, time ; quality of service e.g., levels of precision in computing, of urgency of treatment, graceful degradation ; dependability and fault tolerance, e.g., controlling migrations in response to loss of a processor. Adaptivity concerns systems ranging from hardware to operating systems to services and applications, and in size from tiny embedded systems to large-scale data-centers, from multi-core processors to the Cloud. Their complexity is growing, in scale (software or hardware), but also in interactions between different aspects of reconfiguration.

The design of the adaptation controllers is largely done in an *ad-hoc* fashion, involving lots of different approaches, intuitions, and heuristics. There is an important need for well-founded models and techniques for the safe design of these control loops, which can provide designers with a support to master the complexity of designs, and with guarantees w.r.t. the correctness of the designed controllers.

2.2.2. *Our model-based control approach*

We aim to build general methodologies and tools for the model-based control of reconfigurable computing system, validated in a representative range of reconfigurable systems.

The classical approach in computer science consists of: first programming, and then verifying. We want to explore, in contrast to this, an alternative approach, more effective (easier for the designer) and safer (better guarantees), inspired by control techniques: first model behaviors of the (uncontrolled) system, and its control interfaces, at each component's level ; then specify the adaptation strategy or policy, i.e., control objectives, and possibly check controllability ; finally, derive the controller solution: automatically synthesize the controller (for classes of problems where it is possible).

Our general topic is considering computing systems as object of automatic control, which is a newly emerging scientific theme, often considering continuous models. We will be using our complementary backgrounds in reactive systems and synchronous languages, in Control Theory and in experiences in applying various control techniques to computing systems, as well as a general orientation to apply formal methods to real-world systems. We are reversing the classical view of computer science for control systems, and consider, more originally, *control techniques for computing systems*.

2.2.3. *Opportunities between computer science and control theory*

This new and emerging combination of computing and control can bring novel contributions both ways : adaptive computing can benefit from control techniques, which provide designers with a broad range of results, and begin to be equipped with efficient tools e.g., connected to the synchronous technology, which is essential for concrete impact on real-world systems. Research in Control Theory can benefit from computing systems, embedded or large scale, as new application domain, where theoretical results can be evaluated and transferred, and from where new interesting and relevant problems can come up.

Risks could be in e.g., the fact that such a new and mixed topic of systems and control techniques does not yet correspond to an identified scientific community, making it difficult to find people involving themselves rather than staying in their respective community of origin. But this is in direct relationship to the originality of the subject, and is compensated by the identification of concrete potentials in our ongoing work. This multi-disciplinary work takes much more exploratory time and cooperative discussion than more classical research programs, but it brings all the more original results.

Control for Computing is in its founding phase: the disadvantage is there are no comfortable inherited results and community, the advantages are in the novelty and relevance of founding a new direction.

3. Research Program

3.1. Modeling and control techniques for autonomic computing

3.1.1. *Continuous control*

Continuous control was used to control computer systems only very recently and in few occasions, despite the promising results that were obtained. This is probably due to many reasons, but the most important seems to be the difficulty by both communities to transform a computer system problem into an automatic control problem. The aim of the team is to explore how to formalize typical autonomic computing cases into typical control problems. Many new methodological tools will probably be useful for that, e.g., we can cite the hybrid system approach, predictive control or event-based control approach. Computer systems are not usual for the control system community and they often present non-conventional control aspects like saturation control. New methodological tools are required for an efficient use of continuous-time control in computer science.

3.1.2. Discrete control

Discrete control techniques are explored at long-term, to integrate more control in the BZR language, and address more general control issues, wider than BZR's limitations. Directions are : expressiveness (taking into account in the LTS models value domains of the variables in the program) ; adaptive control (where the controller itself can dynamically switch between different modes) ; distributed control (for classes of problems where communicating controllers can be designed) ; optimal control (w.r.t. weight functions, on states, transitions, and paths, with multicriteria techniques) ; timed and hybrid control bringing a new dimension for modeling and control, giving solutions where discrete models fail.

3.2. Design and programming for autonomic computing

3.2.1. Reactive programming

Autonomic systems are intrinsically reconfigurable. To describe, specify or design these systems, there is a need to take into account this reconfigurability, within the programming languages used. We propose to consider the reconfigurability of systems from the angle of two properties: the notion of time, as we want to describe the state and behavior of the system before, and after its reconfiguration; the notion of dynamicity of the system, i.e., considering that the system's possible behaviors throughout execution are not completely known, neither at design-time nor at initial execution state. To describe and design such reactive systems, we propose to use the synchronous paradigm. It has been successfully used, in industry, for the design of embedded systems. It allows the description of behaviors based on a specific model of time (discrete time scale, synchronous parallel composition), providing properties which are important w.r.t. the safety of the described system: reactivity, determinism, preservation of safety properties by parallel composition (with other parts of the system or with its environment). Models and languages for control, proposed in this framework, provide designers, experts of the application domain, with a user-friendly access to highly technical formal methods of DCS, by encapsulating them in the compilation of concrete programming languages, generating concrete executable code. They are based on discrete models, but also support programming of sampled continuous controllers.

3.2.2. Component-based approach and domain-specific languages

For integration of the previous control kernels into wider frameworks of reconfigurable systems, they have to be integrated in a design flow, and connected on the one side with higher-level specification languages (with help of DSLs), and on the other side with the generated code level target execution machines. This calls for the adoption of a component-based approach with necessary features, available typically in Fractal, for explicitly identifying the control interfaces and mechanisms.

Structuring and instrumentation for controllability will involve encapsulation of computations into components, specification of their local control (activation, reconfiguration, suspension, termination), and exporting appropriate interfaces (including behavior abstraction). Modeling the configurations space requires determining the controlled aspects (e.g., heterogeneous CPUs loads, fault-tolerance and variability, memory, energy/power consumption, communication/bandwidth, QoS level) and their control points, as well as APIs for monitors and actions. Compilation and execution will integrate this in a complete design flow involving : extraction of a reactive model from components; instrumentation of execution platforms to be controllable; combination with other controllers; general "glue" and wrapper code.

Integration of reactive languages and control techniques in component-based systems brings interesting questions of co-existence w.r.t. other approaches like Event-Condition-Action (ECA) rules, or Complex Event Processing (CPE).

3.3. Infrastructure-level support for autonomic computing

The above general kernel of model-based control techniques can be used in a range of different computing infrastructures, representing complementary targets and abstraction levels, exploring the two axes :

- from hardware, to operating system/virtual machine, to middleware, to applications/service level;
- across different criteria for adaptation: resources and energy, quality of service, dependability.

3.3.1. *Software and adaptive systems*

Autonomic administration loops at operating systems or middleware level are already very widespread. An open problem remains in design techniques for controllers with predictability and safety, e.g. w.r.t. the reachable states. We want to contribute to the topic of discrete control techniques for these systems, and tackle e.g. problems of coordination of multiple autonomic loops in data-centers, as in the ANR project CtrlGreen. Another target application is the control of clusters in map-reduce applications. The objective is to use continuous time control in order to tune finely the number of required clusters for an application running on a map-reduce server. This will use results of the ANR project MyCloud that enables to simulate clients on a real map-reduce server. On a longer term, we are interested in control problems in administration loops of event-based virtual machines, or in the deployment of massively parallel computation of the Cloud.

3.3.2. *Hardware and reconfigurable architectures*

Reconfigurable architectures based on Field Programmable Gate Arrays (FPGA) are an active research area, where infrastructures are more and more supportive of reconfiguration, but its correct control remains an important issue. Work has begun in the ANR Famous project on identifying domain-specific control criteria and objectives, monitors and management APIs, and on integrating control techniques in the high-level RecoMARTE environment. On a longer term, we want to work on methods and tools for the programming of **multicore architectures**, exploiting the reconfigurability potentials and issues (because of variability, loss of cores), e.g. in our cooperation with ST Microelectronics, using a Fractal-based programming framework in the P2012 project, and in cooperation with Inria Lille (Adam), or with the CEA and TIMA on integrating control loops in the architecture for a fine control of the energy and of the required nodes for running a given application task.

3.3.3. *Applications and autonomic systems*

In autonomic systems, control systems remain a lively source of inspiration, partly because the notion of control loop implementation is known and practiced naturally. On a wider scale, we started a cooperation with Orange Labs on "intelligent" building automation and control for the Smart Grid, through modeling and control of appliances w.r.t. their power consumption modes, at home, building, and city levels. Other partners on these topics are CEA LETI/DACLE and Schneider Electric.

We could explore more systems and applications e.g., Human-Machine Interfaces, or the orchestration of services. They can help design more general solutions, and result in a more complete methodology.

4. Application Domains

4.1. Distributed systems and High-Performance Computing

Distributed systems have grown to levels of scale and complexity where it is difficult to master their administration and resources management, in dynamic and open environments. One of the growing concerns is that the energy consumption has reached levels where it can not be considered negligible anymore, ecologically or economically. Data centers or high performance computing grids need to be controlled in order to combine minimized power needs with sustained performance and quality of service. As mentioned above, this motivates the automation of their management, and is the major topic of, amongst others, our ANR project Ctrl-Green (see 8.2.1).

Another challenge in distributed systems is in the fast growing amounts of data to process and store. Currently one of the most common ways of dealing with these challenges is the parallel programming paradigm MapReduce which is slowly becoming the de facto tool for Big Data analytics. While its use is already widespread in the industry, ensuring performance constraints while also minimizing costs provides considerable challenges. Current approaches to ensure performance in cloud systems can be separated into three categories: static, reactive, predictive and hybrid approaches. In the industry, static deployments are the standard and usually tuned based on the application peak demand and are generally over-provisioned. Reactive

approaches are usually based on reacting to an input metric such as the current CPU utilisation, request rate, response time by adding and removing servers as necessary. Some public cloud providers offer reactive techniques such as the Amazon Auto Scaler. They provide the basic mechanisms for reactive controllers, but it is up to the user to define the static scaling thresholds which is difficult and not optimal. To deal with this issue, we propose a control theoretical approach, based on techniques that have already proved their usefulness for the control community.

In the domain of parallel systems and High Performance Computing, systems are traditionally less open and more controlled by administrators, but this trend is changing, as they are facing the same challenges in energy consumption, needs for adaptivity in reaction to changing workloads, and security issues in computation outsourcing. Topics of interest for us in this domain concern problem in dynamical management of memory and communications features, which we are exploring in the HPES project of the Labex Persybal-lab (see 8.1).

4.2. Reconfigurable architectures in embedded systems

Dynamically reconfigurable hardware has been identified as a promising solution for the design of energy efficient embedded systems. A common argument in favor of this kind of architecture is the specialization of processing elements, that can be adapted to application functions in order to minimize the delay, the control cost and to improve data locality. Another key benefit is the hardware reuse to minimise the area, and therefore the static power and cost. Further advantages such as hardware updates in long-life products and self-healing capabilities are also often mentioned. In presence of context changes (e.g. environment or application functionality), self-adaptive technique can be applied as a solution to fully benefit from the runtime reconfigurability of a system.

Dynamic Partial Reconfiguration (DPR) of FPGA is another accessible solution to implement and experiment reconfigurable hardware. It has been widely explored and detailed in literature. However, it appears that such solutions are not extensively exploited in practice for two main reasons: i) the design effort is extremely high and strongly depends on the available chip and tool versions; and ii) the simulation process, which is already complex for non-reconfigurable systems, is prohibitively large for reconfigurable architectures. As a result, new adequate methods are required to fully exploit the potential of dynamically reconfigurable and self-adaptive architectures. We are working in this topic, especially on the reconfiguration control aspect, in cooperation with teams specialized in reconfigurable architectures such as the former DaRT team at Inria Lille, and LabSticc in Lorient, as in the recently ended ANR project Famous.

4.3. Smart environments and Internet of Things

Another application domain for autonomic systems design and control is the Internet of Things, and especially the design of smart environments, at the level of homes, buildings, or cities. These domains are often considered at the level of sensors networks, with a strong emphasis on the acquisition of data in massive scales. The infrastructures are sometimes also equipped with actuators, with a wide range of applications, for example concerning lighting or heating, or access and security aspects. We are interested in closing the control loop in such environments, which is less often studied. In particular, rule-based languages are often used to define the automated systems, and we want to contribute to the safe design of such controllers with guarantees on their behaviors. We are working in this topic in cooperation with teams specialized in infrastructures for smart environments at CEA LETI/DACLE and Orange labs.

5. New Software and Platforms

5.1. Heptagon/BZR programming language

Participants: Gwenaël Delaval [Contact person], Eric Rutten.

We want to produce results concretely usable by third parties, either in cooperative projects, or by free diffusion of tools. One perspective is to build tool boxes for the design of continuous control solutions for computing systems: it will be explored in the future. A readily available result concerns discrete control and programming.

HEPTAGON is a dataflow synchronous language, inspired from LUCID SYNCHRONE ¹. Its compiler is meant to be simple and modular, allowing this language to be a good support for the prototyping of compilation methods of synchronous languages.

HEPTAGON has been used to build BZR ², which is an extension of the former with contracts constructs. These contracts allow to express dynamic temporal properties on the inputs and outputs of HEPTAGON node. These properties are then enforced, within the compilation of a BZR program, by discrete controller synthesis, using the SIGALI tool ³. The synthesized controller is itself generated in HEPTAGON, allowing its analysis and compilation towards different target languages (C, Java, VHDL).

Prospects about Heptagon/BZR lie in: support for programming methodologies in BZR: debug, diagnosis, abstraction and composition, ...; extensions towards use of more expressive synthesis tools ; integration of target code into execution platforms (Fractal, reconfigurable FPGA, ...)

6. New Results

6.1. Highlights of the Year

We have been invited to participate to the organization of events, which highlight our active presence in the scientific life in the two domains which we are bridging :

- autonomic computing: Eric Rutten is PC member, as well as workshops chair, of the 12th IEEE International Conference on Autonomic Computing, ICAC 2015 (<http://icac2015.imag.fr/>), and PC co-chair of the 3rd IEEE International Conference on Cloud and Autonomic Computing, CAC 2015 (<http://autonomic-conference.org/>), the two major conferences on the topic.
- control: Eric Rutten is organizer of a special session on discrete control for computing at the 12th IFAC - IEEE International Workshop on Discrete Event Systems, WODES 2014 (<http://wodes2014.lurpa.ens-cachan.fr/>), the main conference specialized in Discrete Event Systems, ; he is on the IFAC Technical Committee 1.3 on Discrete Event and Hybrid Systems, (<http://tc.ifac-control.org/1/3/>) and on the IEEE Control Systems Society Discrete Event Systems Technical Committee (<http://discrete-event-systems.ieeecss.org>).

6.2. Discrete control and reactive language support

Participants: Gwenaël Delaval, Eric Rutten, Stéphane Mocanu.

Concerning language support, we have designed and implemented BZR, a mixed imperative/declarative programming language: declarative contracts are enforced upon imperatively described behaviors (see 5.1). The semantics of the language uses the notion of Discrete Controller Synthesis (DCS) [5]. We target the application domain of adaptive and reconfigurable systems: our language can serve programming closed-loop adaptation controllers, enabling flexible execution of functionalities w.r.t. changing resource and environment conditions. DCS is integrated into a programming language compiler, which facilitates its use by users and programmers, performing executable code generation. The tool is concretely built upon the basis of a reactive programming language compiler, where the nodes describe behaviors that can be modeled in terms of transition systems. Our compiler integrates this with a DCS tool [3]. This work is done in close cooperation with the Inria team Sumo at Inria Rennes (H. Marchand). Ongoing work concerns aspects of compilation and debugging and logico-numeric extension of BZR based on the ReaX tool developed at Inria Rennes in the framework of the ANR Ctrl-Green project (see 8.2.1).

¹<http://www.di.ens.fr/~pouzet/lucid-synchrone>

²<http://bzt.inria.fr>

³<http://www.irisa.fr/vertecs/Logiciels/sigali.html>

We are also currently working on combining maximally permissive discrete control with runtime mechanisms for choosing between valid control values, involving e.g. a classical controller or stochastic aspects ; and on exploring the notion of adaptive discrete control, which is yet an open question in discrete control in contrast to the well-known adaptive continuous control.

Another activity related to discrete control is or work with Leiden University and CWI (N. Khakpour, now at Linnaeus U., and F. Arbab) on enforcing correctness of the behavior of an adaptive software system during dynamic adaptation is an important challenge along the way to realize correct adaptive systems. In this research, we model adaptation as a supervisory control problem and synthesize a controller that guides the behavior of a software system during adaptation. The system during adaptation is modeled using a graph transition system and properties to be enforced are specified using an automaton. To ensure correctness, we then synthesize a controller that imposes constraints on the system during adaptation [14].

6.3. Design and programming

6.3.1. Component-based approaches

Participants: Frederico Alvares, Eric Rutten.

Component-based architectures have shown to be very suited for self-adaptation purposes, not only because of their intrinsic characteristics like reusability and modularity, but also as virtue of their dynamical reconfiguration capabilities. The issue, nevertheless, remains that adaptation behaviors are generally conceived by means of fine-grained reconfiguration actions from the very initial configurations. This way, besides the complexity in managing large-sized architectures, the space of reachable configurations is not known in advance, which prevents ensuring well-mastered adaptive behaviours. We address this problem by designing Ctrl-F, a domain-specific language which objective is to provide high-level support for describing adaptation behaviors and policies in component-based architectures. The proposed language lies on synchronous reactive programming, which means that it benefits of an entire environment and formal tooling allowing for the verification and control of reconfigurations. We show the applicability of Ctrl-F by first integrating it to FraSCAti, a Service Component Architecture middleware platform, and then by applying it to Znn.com, a well known self-adaptive case study.

We work on the topic in cooperation with the Spirals Inria team at Inria Lille (L. Seinturier). It constitutes a follow-up on previous work in the ANR Minalogic project MIND, industrializing the Fractal component-based framework, with a continuation of contacts with ST Microelectronics (V. Bertin). Our integration of BZR and Fractal [4], [2] is at the basis of our current work. On a related topic, we are also starting a cooperation on introducing reactive control in hierarchical autonomic architectures, with A. Diaconescu and E. Najm at TelecomParisTech.

6.3.2. Rule-based systems

Participants: Julio Cano, Adja Sylla, Gwenaël Delaval, Eric Rutten.

Event-Condition-Action (ECA) rules are a widely used language for the high level specification of controllers in adaptive systems, such as Cyber-Physical Systems and smart environments, where devices equipped with sensors and actuators are controlled according to a set of rules. The evaluation and execution of every ECA rule is considered to be independent from the others, but interactions of rule actions can cause the system behaviors to be unpredictable or unsafe. Typical problems are in redundancy of rules, inconsistencies, circularity, or application-dependent safety issues. Hence, there is a need for coordination of ECA rule-based systems in order to ensure safety objectives. We propose a tool-supported method for verifying and controlling the correct interactions of rules, relying on formal models related to reactive systems, and Discrete Controller Synthesis (DCS) to generate correct rule controllers [12].

We work on this topic in cooperation with CEA LETI/DACLE (L. Gurgun) and target the application and experimentation domain of smart environment in the Internet of Things [11].

Another complementary direction on which we are starting a cooperation with CEA LETI/DACLE is the topic of a high-level language for safe rule-based programming in the LINC platform: the PhD of Adja Sylla on this topic will be co-advised with F. Pacull and M. Louvel at CEA.

6.4. Infrastructure-level support

6.4.1. *Autonomic Cloud and Big-Data systems*

This activity continues work started several years ago in the Sardes Inria-team, before it split into Eroads (at LIG) and Ctrl-A (at Inria).

6.4.1.1. *Coordination in multiple-loop autonomic Cloud systems*

Participants: Soguy Gueye, Gwenaël Delaval, Stéphane Mocanu, Bogdan Robu, Eric Rutten.

Complex computing systems are increasingly self-adaptive, with an autonomic computing approach for their administration. Real systems require the co-existence of multiple autonomic management loops, each complex to design. However their uncoordinated co-existence leads to performance degradation and possibly to inconsistency. There is a need for methodological supports facilitating the coordination of multiple autonomic managers. We address this problem in the context of the ANR project Ctrl-Green (see 8.2.1), in cooperation with LIG (N. de Palma) in the framework of the PhD of S. Gueye. We propose a method focusing on the discrete control of the interactions of managers [7] [9]. We follow a component-based approach and explore modular discrete control, allowing to break down the combinatorial complexity inherent to the state-space exploration technique [13]. This improves scalability of the approach and allows constructing a hierarchical control. It also allows re-using complex managers in different contexts without modifying their control specifications. We build a component-based coordination of managers, with introspection, adaptivity and reconfiguration. We validate our method on a multiple-loop multi-tier system.

We are currently working on the distributed execution of modular controllers and on considering more control objectives, beyond purely discrete or logical ones, evaluating the new tool ReaX developed at Inria Rennes (Sumo) (see 6.2) and exploring continuous or stochastic control of servers provisioning.

6.4.1.2. *Control for Big data*

Participants: Bogdan Robu, Mihaly Berekmeri, Nicolas Marchand.

To deal with the issue of ensuring performance constraints while also minimizing costs in systems for Big Data analytics based on the parallel programming paradigm MapReduce, we propose a control theoretical approach, based on techniques that have already proved their usefulness for the control community. We develop an algorithm to create the first linear dynamic model for a Big Data MapReduce system, running a concurrent workload. Furthermore we identify two major performance constraint use cases: relaxed-minimal resource and strict performance constraints. For the first case we developed a feedback control mechanism and, to minimize the number of control actuations, an event-based feedback controller. For the second case we add a feedforward controller that efficiently suppresses the effects of large workload size variations. The work is validated in a simulated Matlab environment build at GIPSA-lab and online on a real 60 node MapReduce cluster (part of GRID 500), running a data intensive Business Intelligence workload. Our experiments demonstrate the success of the control strategies employed in assuring service time constraints [17], [18].

This work is performed in cooperation with LIG (S. Bouchenak) in the framework of the PhD of M. Berekmeri.

6.4.2. *Reconfiguration control in DPR FPGA*

Participant: Eric Rutten.

Dynamically reconfigurable hardware has been identified as a promising solution for the design of energy efficient embedded systems. However, its adoption is limited by the costly design effort including verification and validation, which is even more complex than for non dynamically reconfigurable systems. We work on this topic in the context of a design environment, developed in the framework of the ANR project Famous, in cooperation with LabSticc in Lorient and Inria Lille (DaRT team) [10]. We propose a tool-supported formal method to automatically design a correct-by-construction control of the reconfiguration. By representing system behaviors with automata, we exploit automated algorithms to synthesize controllers that safely enforce reconfiguration strategies formulated as properties to be satisfied by control. We design generic modeling patterns for a class of reconfigurable architectures, taking into account both hardware architecture and applications, as well as relevant control objectives. We validate our approach on two case studies implemented on FPGAs [1].

We are currently valorizing results in more publications, and extending the use of control techniques by evaluating the new tool ReaX developed at Inria Rennes (Sumo) in the framework of the ANR Ctrl-Green project (see 6.2 and 8.2.1).

6.4.3. *Autonomic memory management in HPC*

Participants: Naweiluo Zhou, Gwenaël Delaval, Bogdan Robu, Eric Rutten.

Concurrent programs need to manage the time trade-off between synchronization and computing. A high concurrency level may decrease computing time but at the same time increase synchronization cost among threads. The traditional way to handle synchronization problems is through implementing locks. However locks suffer from the likelihood of deadlocks, vulnerability to failures, faults etc.. Software Transactional Memory (STM) has emerged as a promising technique to address synchronization issues through transactions. In STM, blocks of instructions accessing the shared data are wrapped into transactions. In STM each transaction executes speculatively, and conflicts may be aroused when two transactions are trying to modify the same area simultaneously. A way to reduce conflicts is by adjusting concurrency levels. A suitable concurrency level can maximize program performance. However, there is no universal rule to decide the best concurrency level for a program from an offline view. Hence, it becomes necessary to adopt a dynamical tuning strategy to better manage a STM system, so that a program can achieve a better performance. In the context of the action-team HPES of the Labex Persyval-lab⁴ (see 8.1), we explore the autonomic computing approach and control techniques to address these runtime tuning problems as a feedback control loop to automate the choices of concurrency levels, conflict management policies, and other parameters, with the objective of optimizing program execution time. This work is performed in cooperation with LIG (J.F. Méhaut) in the framework of the PhD of N. Zhou.

6.4.4. *Control of smart environments*

Participants: Julio Angel Cano Romero, Mengxuan Zhao, Eric Rutten, Hassane Alla [Gipsa-lab].

6.4.4.1. *Generic supervision architecture*

New application domains of control, such as in the Internet of Things (IoT) and Smart Environments, require generic control rules enabling the systematization and the automation of the controller synthesis. We are working on an approach for the generation of Discrete Supervisory Controllers for these applications. A general modeling framework is proposed for the application domain of smart home. We formalize the design of the environment manager as a Discrete Controller Synthesis (DCS) problem, w.r.t. multiple constraints and objectives, for example logical issues of mutual exclusion, bounding of power peaks. We validate our models and manager computations with the BZR language and an experimental simulator [15]. This work is performed in cooperation with Orange labs (G. Privat) in the framework of the Cifre PhD of M. Zhao.

6.4.4.2. *Rule-based specification*

In the Internet of things, Event - Condition - Action (ECA) are used as a flexible tool to govern the relations between sensors and actuators. Runtime coordination and formal analysis becomes a necessity to avoid side effects mainly when applications are critical. In cooperation with CEA LETI/DACLE, we have worked on a case study for safe applications development in IoT and smart home environments [11].

⁴<https://persyval-lab.org/en/sites/hpes>

7. Bilateral Contracts and Grants with Industry

7.1. CIFRE PhD grant Orange

This Cifre PhD started in the beginning of 2012, and is going to be defended in april 2015, on the topic of "Discrete Control for Smart Environments through a Generic Finite-State-Models-Based Infrastructure". Hassane Alla and Eric Rutten are advising the PhD student for 10%.

One result of this cooperation is that a patent deposited at the INPI on "Configuration automatique du controle discret d'entites physiques dans un systeme de supervision et de controle", by Gilles Privat et Mengxuan Zhao (Orange labs), Hassane Alla (Gipsa-lab), Eric Rutten (Inria).

7.2. Bilateral Grants with Industry

Our cooperation with CEA LETI/LIST (V. Olive) at Grenoble Minatec is bilateral, involving:

- the Inria Post-Doc grant of Julio Cano, to work with L. Guergen on ECA-based programming in the IoT
- the CEA PhD grant of Adja Sylla, to work with F. Pacull and M. Louvel on high-level programming on top of a rule-based middleware.

8. Partnerships and Cooperations

8.1. Regional Initiatives

The Labex Persyval-lab is a large regional initiative, supported by ANR, where we are contributing through two projects:

8.1.1. *Projet Exploratoire STAARS*

This project, defined for one and a half year, grouped members from Inria, LIG and Gipsa-lab, and concerned the general topic of control for computing, with a special emphasis on relating stochastic models with logical discrete control. It enabled us to organize two international workshops in Grenoble: <https://persyval-lab.org/en/exploratory-project/staars>

8.1.2. *Equipe-action HPES*

This project groups members from Inria, LIG, Gipsa-lab, TIMA and Gipsa-lab, around the topic of High-Performance Computing benefiting from technologies originally developed for Embedded Systems. Ctrl-A is directly involved in the co-advising of the PhD of Naweiluo Zhou, with J.F. Méhaut (LIG), on the topic of autonomic management of software transactional memory mechanisms: <https://persyval-lab.org/en/sites/hpes>

8.2. National Initiatives

8.2.1. ANR

*Ctrl-Green*⁵ is an ANR project on Autonomic energy management for virtualized datacenter. The Coordinator is UJF, and the duration: 2011-2014. Others Partners are: Inria Rennes, IRIT, Eolas.

In Ctrl-A, it is funding the PhD thesis of Soguy Gueye, defended in december 2014; before that it funded the post-doc of Nicolas Berthier in 2012. In both cases, co-advising was done with Noeël de Palma (LIG).

⁵<http://www.en.ctrlgreen.org/>

8.2.2. Informal National Partners

We have contacts with colleagues in France, with whom we are submitting collaboration projects, co-organizing events and workshops, etc. They feature : Avalon Inria team in Lyon (F. Desprez), LIP6 (J. Malenfant), Scales Inria team in Sophia-Antipolis (L. Henrio), LIRRM in Montpellier (A. Gamatié, K. Godary, D. Simon), IRISA/Inria Rennes (J. Buisson, J.L. Pazat, ...), Telecom Paris-Tech (A. Diaconescu, E. Najm), LAAS (Thierry Monteil).

8.2.3. Informal National Industrial Partners

We have ongoing discussions with several industrial actors in our application domains, some of them in the framework of cooperation contracts, other more informal: Eolas/Business decision (G. Dulac), ST Microelectronics (V. Bertin), Schneider Electric (C. El-Kaed, P. Nappey, M. Pitel), Orange labs (J. Pulou, G. Privat).

8.3. International Initiatives

8.3.1. Inria International Partners

8.3.1.1. Informal International Partners

We have ongoing relations with international colleagues in the emerging community on our topic of control for computing e.g., in Sweden at Lund (K.E. Arzen, M. Maggio) and Linnaeus Universities (D. Weyns, N. Khakpour), in the Netherlands at CWI/leiden University (F. Arbab), in China at Heifei University (Xin An), in Italy at University Milano (C. Ghezzi, A. Leva), in the USA at Ann Arbor University (S. Lafortune).

8.3.2. Participation In other International Programs

Eric Rutten is a member of the IFAC Technical Committee 1.3 on Discrete Event and Hybrid Systems, for the 2011-2014 triennium, and for the 2014-2017 triennium <http://tc.ifac-control.org/1/3> ; and of the IEEE Control Systems Society Discrete Event Systems Technical Committee <http://discrete-event-systems.ieeecss.org>.

9. Dissemination

9.1. Promoting Scientific Activities

9.1.1. Scientific events organisation

9.1.1.1. member of the organizing committee

Eric Rutten is

- Workshops chair for the 12th IEEE International Conference on Autonomic Computing, ICAC, July 7-10, 2015 in Grenoble, France <http://icac2015.imag.fr> ;
- proposing the DCDS 2015 Special Session on Dependable Discrete control for adaptive and reconfigurable computing systems <http://www.gdl.cinvestav.mx/dcds2015/submission.html> may 2015 ;
- organizer of the two editions of the Grenoble Workshop on Autonomic Computing and Control, 27 may and 4 november 2014 <https://persyval-lab.org/en/exploratory-project/staars> ;
- organizer, with Thierry Monteil and Khalil Drira (LAAS), of the workshop autonome, 16 - 17 oct. 2014 au LAAS, Toulouse (invited keynote speaker : Xiaoyun Zhu, VMware) <http://projects.laas.fr/autonomique> ;
- organizer of the WODES 2014 Special Session on Discrete control for adaptive and reconfigurable computing systems, 14 may 2014 <http://wodes2014.lurpa.ens-cachan.fr/special-sessions/> ;
- organizer, with J. Buisson (IRISA), of the Atelier Informatique Autonome, satellite of the colloque COMPAS 2014, 22 avril 2014, Neuchatel <http://compas2014.unine.ch/atelier2> ;
- organizer of the Seminar (Grenoble and Rhône-Alpes region, Lyon, Valence, Annecy) on Supervisory control Techniques for Autonomic, Adaptive and Reconfigurable computing Systems <https://team.inria.fr/ctrl-a/members/eric-rutten/labex/staars-seminars-html/>

9.1.2. Scientific events selection

9.1.2.1. responsible of the conference program committee

Eric Rutten is program co-chair of the IEEE International Conference on Cloud and Autonomic Computing, CAC 2015, MIT, USA, September 2015 (<http://autonomic-conference.org>), Part of FAS* - Foundation and Applications of Self* Computing Conferences, Collocated with: the IEEE Self-Adaptive and Self-Organizing System Conference, the IEEE Peer-to-Peer Computing Conference.

9.1.2.2. member of the conference program committee

Eric Rutten is PC member for: 5th IFAC Conference on Analysis and Design of Hybrid Systems (ADHS) October 14-16, 2015, Georgia Tech, Atlanta, Georgia, USA (<http://adhs15.gatech.edu>) ; 12th IEEE International Conference on Autonomic Computing, ICAC 2015, July 7-10, 2015 in Grenoble, France (<http://icac2015.imag.fr>) ; 5th IFAC international workshop on Dependable Control of Discrete Systems , DCDS, Cancun, Mexico, may 27-29, 2015 (<http://www.gdl.cinvestav.mx/dcds2015>) ; 6th Workshop on Adaptive and Reconfigurable Embedded Systems (APRES 2014) April 14, 2014, Berlin, Germany ; Collocated with the CPSWeek (<http://www.mrtc.mdh.se/apres2014>) ; ESLsyn 2014 The 2014 Electronic System Level Synthesis Conference, May 31 - June 1, 2014, San Francisco, CA, USA,co-located with DAC (<http://www.ecsi.org/eslsyn>)

Nicolas Marchand is PC member for Feedback Computing 2015 10th International Workshop on Feedback Computing Co-located with CPSWeek 2015 April 13th, Seattle, WA, USA (<http://cpslab.cs.mcgill.ca/feedbackcomputing15/>)

9.1.2.3. reviewer

Eric Rutten, Nicolas Marchand, Bogdan Robu are reviewers for the IFAC World congress 2014.

9.1.3. Journal

9.1.3.1. reviewer

Eric Rutten is reviewer for ACM Transactions on Autonomous and Adaptive Systems ; Science of Computer Programming ; Journal of Systems and Software.

9.1.4. Invited Conferences

- Eric Rutten was invited to give talks at :
 - groupe de travail COSMAL (<http://gdr-gpl.cnrs.fr/Groupes/COSMAL>), journées du GDR GPL, CNAM Paris du 10 au 13 juin
 - Journée Architectures Logicielles pour la Robotique Autonome, les Systèmes Cyber-Physiques et les Systèmes Auto-Adaptables, 1er décembre 2014, LIP6 Paris (<http://www.lirmm.fr/gtcar/index.php/autres-workshop/workshop-cps-gdr-gpl-robotique>)
 - 11th International Conference on Distributed Computing and Internet Technology (ICDCIT-2015). (<http://www.icdcit.ac.in/>), Bhubaneswar, India, February 5–8, 2015.
- Nicolas Marchand, Bogdan Robu were invited to give talks at the Workshop in Cloud Control, May 7 to May 9, 2014, Lund, Sweden (<http://www.lccc.lth.se/index.php?page=may-2014-2>)

9.2. Teaching - Supervision - Juries

9.2.1. Supervision

- PhD : Soguy Geye ; Coordination Modulaire de Gestionnaires Autonomes par Contrôle Discret ; Université de Grenoble, 3 dec. 2014 ; E. Rutten, N. de Palma (LIG)
- PhD in progress : Adja Ndeye SYLLA ; Generation of coordination rules from an automaton, in the context of the redeployment of distributed contra; applications ; fall 2014 ; E. Rutten, F. Pacull and M. Louvel (CEA)
- PhD in progress : Naweiluo Zhou ; Application-aware policies and control for transactional memory systems ; oct. 2013 ; E. Rutten, G. Delaval, J.F. Mehaut (LIG)

- PhD in progress : Mengxuan Zhao ; Discrete Control in the Internet of things and Smart Environments through a Shared Infrastructure ; nov. 2011 ; E. Rutten, H. Alla (Gipsa-lab), G. Privat (Orange labs)

10. Bibliography

Major publications by the team in recent years

- [1] X. AN, E. RUTTEN, J.-P. DIGUET, N. LE GRIGUER, A. GAMATIÉ. *Discrete Control for the Autonomic Management of Dynamically Partially Reconfigurable FPGA Architectures*, in "Proceedings of the 13th International Conference on Autonomic Computing (ICAC'13), June 26-28, San Jose, California", 2013
- [2] T. BOUHADIBA, Q. SABAH, G. DELAVAL, E. RUTTEN. *Synchronous Control of Reconfiguration in Fractal Component-based Systems – a Case Study*, in "Proceedings of the International Conference on Embedded Software. EMSOFT, Taipei, Taiwan. October 9-14", 2011, pp. 309–318, <http://dx.doi.org/10.1145/2038642.2038690>
- [3] G. DELAVAL, H. MARCHAND, E. RUTTEN. *Contracts for Modular Discrete Controller Synthesis*, in "Proceedings of the ACM SIGPLAN/SIGBED Conference on Languages, Compilers and Tools for Embedded Systems, LCTES, in conjunction with CPSWeek, Stockholm, Sweden, April 12-16", 2010, pp. 57–66, <http://doi.acm.org/10.1145/1755951.1755898>
- [4] G. DELAVAL, E. RUTTEN. *Reactive model-based control of reconfiguration in the Fractal component-based model*, in "Proceedings of the 13th International Symposium on Component Based Software Engineering (CBSE), Prague, Czech Republic, 23-25 June", 2010, pp. 93–112, best paper award, http://dx.doi.org/10.1007/978-3-642-13238-4_6
- [5] G. DELAVAL, E. RUTTEN, H. MARCHAND. *Integrating Discrete Controller Synthesis in a Reactive Programming Language Compiler*, in "Journal of Discrete Event Dynamic System, jDEDS, special issue on Modeling of Reactive Systems", 2013, vol. 23, n^o 4, pp. 385-418, <http://dx.doi.org/10.1007/s10626-013-0163-5>
- [6] E. DUMITRESCU, A. GIRAULT, H. MARCHAND, E. RUTTEN. *Multicriteria optimal discrete controller synthesis for fault-tolerant tasks*, in "Proceedings of the 10th International Workshop on Discrete Event Systems (WODES), Berlin, Germany, August 30 – September 1", 2010, pp. 366–373 [DOI : 10.3182/20100830-3-DE-4013.00059], <http://hal.inria.fr/inria-00510019>
- [7] S. M. K. GUEYE, N. DE PALMA, E. RUTTEN. *Coordination Control of Component-based Autonomic Administration Loops*, in "Proceedings of the 15th International Conference on Coordination Models and Languages, COORDINATION, 3-6 June, Florence, Italy", 2013, <http://dx.doi.org/10.3182/20110828-6-IT-1002.01548>

Publications of the year

Doctoral Dissertations and Habilitation Theses

- [8] S. M. K. GUEYE. *Modular coordination of autonomic managers by discrete control*, Université de Grenoble, December 2014, <https://hal.inria.fr/tel-01091302>

Articles in International Peer-Reviewed Journals

- [9] S. M. K. GUEYE, N. DE PALMA, E. RUTTEN, A. TCHANA, N. BERTHIER. *Coordinating self-sizing and self-repair managers for multi-tier systems*, in "Future Generation Computer Systems", June 2014, vol. 35, pp. 14 - 26 [DOI : 10.1016/J.FUTURE.2013.12.037], <https://hal.inria.fr/hal-00949556>
- [10] G. SÉBASTIEN, F. DE LAMOTTE, N. LE GRIGUER, E. RUTTEN, G. GOGNIAT, J.-P. DIGUET. *Extending UML/MARTE to support Discrete Controller Synthesis, application to Reconfigurable Systems-on-Chip modeling*, in "ACM Transactions on Reconfigurable Technology and Systems", 2014, 17 p. , <https://hal.archives-ouvertes.fr/hal-00972636>

International Conferences with Proceedings

- [11] J. CANO, G. DELAVAL, E. RUTTEN, Y. BENAZZOUZ, L. GURGEN. *ECA rules for IoT environment: a case study in safe design*, in "First Edition of the Workshop on Quality Assurance for Self-adaptive, Self-organising Systems QA4SASO 2014; In conjunction with the Eighth IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO)", London, United Kingdom, September 2014, <https://hal.inria.fr/hal-01091149>
- [12] J. CANO, G. DELAVAL, E. RUTTEN. *Coordination of ECA rules by verification and control*, in "16th International Conference on Coordination Models and Languages", Berlin, Germany, June 2014, 16 p. , <https://hal.archives-ouvertes.fr/hal-01006186>
- [13] G. DELAVAL, S. M. K. GUEYE, E. RUTTEN, N. DE PALMA. *Modular Coordination of Multiple Autonomic Managers*, in "17th International ACM Sigsoft Symposium on Component-Based Software Engineering (CBSE 2014)", Lille, France, June 2014, 291 p. [DOI : 10.1145/2602458.2602465], <https://hal.archives-ouvertes.fr/hal-01006106>
- [14] N. KHAKPOUR, F. ARBAB, E. RUTTEN. *Supervisory Controller Synthesis for Safe Software Adaptation*, in "WODES - 12th IFAC - IEEE International Workshop on Discrete Event Systems", Paris, France, May 2014, <https://hal.inria.fr/hal-01091155>
- [15] M. ZHAO, G. PRIVAT, H. ALLA, E. RUTTEN. *Discrete Control for Smart Environments through a Generic Finite-State-Models-Based Infrastructure*, in "AmI-14 - European Conference on Ambient Intelligence", Eindhoven, Netherlands, Springer, November 2014, vol. LNCS 8850, <https://hal.inria.fr/hal-01091140>

Research Reports

- [16] F. ALVARES DE OLIVEIRA JR., E. RUTTEN, L. SEINTURIER. *High-level Language Support for the Control of Reconfiguration in Component-based Architectures*, Inria Grenoble - Rhône-Alpes ; Inria Lille - Nord Europe ; Laboratoire d'Informatique Fondamentale de Lille, January 2015, n^o RR-8669, <https://hal.inria.fr/hal-01103548>

References in notes

- [17] M. BEREKMERI, D. SERRANO, S. BOUCHENAK, N. MARCHAND, B. ROBU. *A Control Approach for Performance of Big Data Systems*, in "IFAC World Congress", 2014, hal-00940282, <http://hal.archives-ouvertes.fr/hal-00940282>

- [18] M. BEREKMERI, D. SERRANO, S. BOUCHENAK, N. MARCHAND, B. ROBU. *Application du contrôle pour garantir la performance des systèmes Big Data*, in "Conférence francophone d'informatique en parallélisme, architecture et système (ComPAS)", 2014, hal-00982404, <http://hal.archives-ouvertes.fr/hal-00982404>