

Inria

RESEARCH CENTER

FIELD

**Networks, Systems and Services,
Distributed Computing**

Activity Report 2019

Section Application Domains

Edition: 2020-03-21

DISTRIBUTED SYSTEMS AND MIDDLEWARE

1. COAST Project-Team 5
2. CTRL-A Project-Team 6
3. DELYS Project-Team (section vide) 7
4. MIMOVE Project-Team 8
5. Myriads Project-Team 11
6. SPIRALS Project-Team 12
7. STACK Project-Team 13
8. WHISPER Project-Team 15
9. WIDE Project-Team (section vide) 16

DISTRIBUTED AND HIGH PERFORMANCE COMPUTING

10. ALPINES Project-Team 17
11. AVALON Project-Team 19
12. DATAMOVE Project-Team 21
13. HIEPACS Project-Team 29
14. KERDATA Project-Team 32
15. POLARIS Project-Team 33
16. ROMA Project-Team 35
17. STORM Project-Team 36
18. TADAAM Project-Team 37

DISTRIBUTED PROGRAMMING AND SOFTWARE ENGINEERING

19. DIVERSE Project-Team (section vide) 38
20. EASE Project-Team 39
21. FOCUS Project-Team 41
22. INDES Project-Team 42
23. RMOD Project-Team 43

NETWORKS AND TELECOMMUNICATIONS

24. AGORA Project-Team 44
25. COATI Project-Team 45
26. DANTE Project-Team 46
27. DIANA Project-Team (section vide) 48
28. DIONYSOS Project-Team 49
29. DYOGENE Project-Team 50
30. EVA Project-Team 51
31. FUN Project-Team (section vide) 54
32. GANG Project-Team 55
33. GANG Project-Team 56
34. MARACAS Team 57
35. NEO Project-Team 59
36. RESIST Team 60
37. SOCRATE Project-Team (section vide) 62

38. TRIBE Project-Team (section vide) 63

COAST Project-Team

4. Application Domains

4.1. Crisis Management

Crisis management research investigates all the dimensions regarding the management of unexpected catastrophic events like floods, earthquake or terrorist attacks. All the phases of a crisis, from preparedness to recovery require collaboration between people from many organizations. This provides opportunities to study inter-organizational collaboration at a large scale and to propose and evaluate mechanisms that ensure secure and safe collaboration. The work of Béatrice Linot provides us with a deep understanding of the factors that encourage collaboration and help to maintain trustworthy collaboration between stakeholders. This work is continued by Clélie Amiot who studies the effects of human chat-bot collaboration in this kind of setting.

4.2. Collaborative Editing

Collaborative editing is a common application of optimistic replication in distributed settings. The goal of collaborative editors, irrespective of the kind of document, is to allow a group of users to update a document concurrently while ensuring that they eventually get all the same copy at the end. Our algorithm allows to implement collaborative editor in a peer to peer way. It avoids the need for a central server ensuring a higher level of privacy among collaborators. In this context, it requires to consider the problem of authentication and authorization of participants[4] and of trust between them[7].

CTRL-A Project-Team

4. Application Domains

4.1. Self-adaptive and reconfigurable computing systems in HPC and the IoT

We are attacking the problem of designing well-regulated and efficient self-adaptive computing systems by the development of novel strategies for systems management.

The kind of systems we typically target involve relatively coarse grained computation tasks (e.g. image processing or HPC tasks, components or services), assembled in workflows, application dependency graphs, or composites. At that level, there can be parallel and conditional branches, as well as choices that can be made between alternative branches, corresponding to different ways to perform that part of the application. Such tasks can be achieved following a choice of implementations or versions, such as in service oriented approaches. Each implementation has its own characteristics and requirements, e.g., w.r.t. resources consumed and QoS offered. The systems execution infrastructures present heterogeneity, with different computing processors, a variety of peripheral devices (e.g., I/O, video port, accelerators), and different means of communications. This hardware or middleware level also presents adaptation potential e.g., in varying quantities of resources or sleep and stand-by modes.

The kinds of control problems encountered in these adaptive systems concern the navigation in the configurations space defined by choice points at the levels of applications, tasks, and architecture. Upon events or conditions triggering reconfiguration and adaptation, the controller has to choose a next configuration where, on the one hand, all consistency constraints are satisfied w.r.t. dependencies and resources requirements. On the other hand, it has to apply a policy or strategy deciding between correct configurations which one to chose e.g. by optimizing one or more criteria, or by insuring reachability of some later configuration (goal or fallback). This targeted class of computing systems we consider is mid-sized, in the sense that the combinatorial complexity is large enough for manual solving to be impossible, while remaining within the range where supervisory control techniques are tractable. The pace of control is more sporadic, and slower than the instruction-level computation performance within the coarse-grained tasks.

The objectives of CTRL-A will be achieved and evaluated in both of our main application domains, thereby exhibiting their similarities from the point of view of reconfiguration control. A first application domain is High Performance Computing. In this area, we currently focus especially on the management of Dynamic Partial Reconfiguration in FPGA based hardware, at the level of middleware. Here the particular application we consider is, as in our ANR project HPeC starting end of 2015, video image flow processing for smart cameras implemented on DPR FPGAs themselves embedded in drones.

A second general application domain to confront our models is the Internet of Things (IoT), more specifically self-adaptive middleware platforms for Smart Environments, or Industry 4.0 related topics, like SCADA. We focus on providing coordination components and controllers of software components and services, or rule-based middleware platforms. The adaptation problems concern both the functional aspects of the applications in a smart building, and the middleware support deployment and reconfiguration issues. We are considering perspectives concerning self-protection and security.

DELYS Project-Team (section vide)

MIMOVE Project-Team

4. Application Domains

4.1. Mobile urban systems for smarter cities

With the massive scale adoption of mobile devices and further expected significant growth in relation with the Internet of Things, mobile computing is impacting most – if not all – the ICT application domains. One such domain is the one of "smart cities". The smart city vision anticipates that the whole urban space, including buildings, power lines, gas lines, roadways, transport networks, and cell phones, can all be wired together and monitored. Detailed information about the functioning of the city then becomes available to both city dwellers and businesses, thus enabling better understanding and consequently management of the city's infrastructure and resources. This raises the prospect that cities will become more sustainable environments, ultimately enhancing the citizens' well being. There is the further promise of enabling radically new ways of living in, regulating, operating and managing cities, through the increasing active involvement of citizens by ways of crowd-sourcing/sensing and social networking.

Still, the vision of what smart cities should be about has been and keeps evolving at a fast pace in close concert with the latest technology trends. It is notably worth highlighting how mobile and social network use has reignited citizen engagement, thereby opening new perspectives for smart cities beyond data analytics that have been initially one of the core foci for smart cities technologies. Similarly, open data programs foster the engagement of citizens in the city operation and overall contribute to make our cities more sustainable. The unprecedented democratization of urban data fueled by open data channels, social networks and crowd sourcing enables not only the monitoring of the activities of the city but also the assessment of their nuisances based on their impact on the citizens, thereby prompting social and political actions. However, the comprehensive integration of urban data sources for the sake of sustainability remains largely unexplored. This is an application domain that we focus on, further leveraging our research on emergent mobile distributed systems, large-scale mobile sensing & actuation, and mobile social crowd-sensing.

In particular, we concentrate on the following specialized applications:

- **Democratization of urban data for healthy cities.** We integrate the various urban data sources, especially by way of crowd-Xing, to better understand city nuisances. This goes from raw pollution sensing (e.g., sensing noise) to the sensing of its impact on citizens (e.g., how people react to urban noise and how this affects their health).
- **Social applications.** Mobile applications are being considered by sociologists as a major vehicle to actively involve citizens and thereby prompt them to become activists. We study such a vehicle from the ICT perspective and in particular elicit relevant middleware solutions to ease the development of such "civic apps".

4.2. Home network diagnosis

With the availability of cheap broadband connectivity, Internet access from the home has become a ubiquity. Modern households host a multitude of networked devices, ranging from personal devices such as laptops and smartphones to printers and media centers. These devices connect among themselves and to the Internet via a local-area network—a home network—that has become an important part of the "Internet experience". In fact, ample anecdotal evidence suggests that the home network can cause a wide array of connectivity impediments, but their nature, prevalence, and significance remain largely unstudied.

Our long-term goal is to assist users with concrete indicators of the quality of their Internet access, causes of potential problems and—ideally—ways to fix them. We intend to develop a set of easy-to-use home network monitoring and diagnosis tools. The development of home network monitoring and diagnosis tools brings a number of challenges. First, home networks are heterogeneous. The set of devices, configurations, and applications in home networks vary significantly from one home to another. We must develop sophisticated techniques that can learn and adapt to any home network as well as to the level of expertise of the user. Second, Internet application and services are also heterogeneous with very diverse network requirements. We must develop methods that can infer application quality solely from the observation of (often encrypted) application network traffic. There are numerous ways in which applications can fail or experience poor performance in home networks. Often there are a number of explanations for a given symptom. We must devise techniques that can identify the most likely cause(s) for a given problem from a set of possible causes. Finally, even if we can identify the cause of the problem, we must then be able to identify a solution. It is important that the output of the diagnosis tools we build is “actionable”. Users should understand the output and know what to do.

In our partnership with Princeton University (associate team HOMENET) we have deployed monitoring infrastructure within users’ homes. We are developing a mostly passive measurement system to monitor the performance of user applications, which we call Network Microscope. We are developing Network Microscope to run in a box acting as home gateway. We have deployed these boxes in 50 homes in the US and 10 in France. The US deployment was ran and financed by the Wall Street Journal. They were interested in understanding the relationship between Internet access speed and video quality. We have been discussing with Internet regulators (in particular, FCC, ACERP, and BEREK) as well as residential access ISP in how Network Microscope can help overcome the shortcomings of existing Internet quality monitoring systems.

4.3. Mobile Internet quality of experience

Mobile Internet usage has boomed with the advent of ever smarter handheld devices and the spread of fast wireless access. People rely on mobile Internet for everyday tasks such as banking, shopping, or entertainment. The importance of mobile Internet in our lives raises people’s expectations. Ensuring good Internet user experience (or Quality of Experience—QoE) is challenging, due to the heavily distributed nature of Internet services. For mobile applications, this goal is even more challenging as access connectivity is less predictable due to user mobility, and the form factor of mobile devices limits the presentation of content. For these reasons, the ability to monitor QoE metrics of mobile applications is essential to determine when the perceived application quality degrades and what causes this degradation in the chain of delivery. Our goal is to improve QoE of mobile applications.

To achieve this goal, we are working on three main scientific objectives. First, we are working on novel methods to monitor mobile QoE. Within the IPL BetterNet we are developing the HostView for Android tool that runs directly on mobile devices to monitor network and system performance together with the user perception of performance. Second, we plan to develop models to predict QoE of mobile applications. We will leverage the datasets collected with HostView for Android to build data-driven models. Finally, our goal is to develop methods to optimize QoE for mobile users. We are currently developing optimization methods for interactive video applications. We envision users walking or driving by road-side WiFi access points (APs) with full 3G/LTE coverage and patchy WiFi coverage (i.e., community Wifi or Wifi APs on Lampposts) or devices with multiple 3G/LTE links. To achieve this goal, we plan to leverage multi-path and cross-layer optimizations.

4.4. Internet Scanning

Internet-wide scanning has enabled researchers to answer a wealth of new security and measurement questions ranging from “How are authoritarian regimes spying on journalists?” to “Are security notifications effective at prompting operators to patch?” Most of these studies have used tools like ZMap, which operates naively, scanning every IPv4 address once. This simplicity enables researchers to easily answer a question once, but the methodology scales poorly when continually scanning to detect changes, as networks change at dramatically

different rates. Service configurations change more frequently on cloud providers like Amazon and Azure than on residential networks. Internet providers in developing regions often have extremely short DHCP windows. Some networks are unstable with host presence varying wildly between different hours and others have distinct periodic patterns, e.g., hosts are only available during regional business hours. A handful of large autonomous systems have not had hosts present in decades. Our work in collaboration with Stanford University is developing more intelligent Internet-wide scanning methods to then implement a system that can scan continuously. Such a system will allow for up-to-date analysis of Internet trends and threats with real-time alerts of important events.

Myriads Project-Team

4. Application Domains

4.1. Main application domains

The Myriads team investigates the design and implementation of system services. Thus its research activities address a broad range of application domains. We validate our research results with selected use cases in the following application domains:

- Smart city services,
- Smart grids,
- Energy and sustainable development,
- Home IoT applications,
- Bio-informatics applications,
- Data science applications,
- Computational science applications,
- Numerical simulations.

SPIRALS Project-Team

4. Application Domains

4.1. Introduction

Although our research is general enough to be applied to many application domains, we currently focus on applications and distributed services for the retail industry and for the digital home. These two application domains are supported by a strong expertise in mobile computing and in cloud computing that are the two main target environments on which our research prototypes are built, for which we are recognized, and for which we have already established strong collaborations with the industrial ecosystem.

4.2. Distributed software services for the retail industry

This application domain is developed in relation with the **PICOM** (*Pôle de compétitivité Industries du Commerce*) cluster. We have established strong collaborations with local companies in the context of former funded projects, such as Cappucino and **Macchiato**, which focused on the development of a new generation of mobile computing platforms for e-commerce. We are also involved in the Datalyse and OCCIware funded projects that define cloud computing environments with applications for the retail industry. Finally, our activities in terms of crowd-sensing and data gathering on mobile devices with the APISENSE[®] platform share also applications for the retail industry.

4.3. Distributed software services for the digital home

We are developing new middleware solutions for the digital home, in particular through our long standing collaboration with Orange Labs. We are especially interested in developing energy management and saving solutions with the POWERAPI software library for distributed environments such the ones that equip digital homes. We are also working to bridge the gap between distributed services hosted on home gateways and distributed services hosted on the cloud to be able to smoothly transition between both environments. This work is especially conducted with the SALOON platform.

STACK Project-Team

4. Application Domains

4.1. Overview

Supporting industrial actors and open-source communities in building an advanced software management stack is a key element to favor the advent of new kinds of information systems as well as web applications. Augmented reality, telemedicine and e-health services, smart-city, smart-factory, smart-transportation and remote security applications are under investigations. Although, STACK does not intend to address directly the development of such applications, understanding their requirements is critical to identify how the next generation of ICT infrastructures should evolve and what are the appropriate software abstractions for operators, developers and end-users. STACK team members have been exchanging since 2015 with a number of industrial groups (notably Orange Labs and Airbus), a few medical institutes (public and private ones) and several telecommunication operators in order to identify both opportunities and challenges in each of these domains, described hereafter.

4.2. Industrial Internet

The Industrial Internet domain gathers applications related to the convergence between the physical and the virtual world. This convergence has been made possible by the development of small, lightweight and cheap sensors as well as complex industrial physical machines that can be connected to the Internet. It is expected to improve most processes of daily life and decision processes in all societal domains, affecting all corresponding actors, be they individuals and user groups, large companies, SMEs or public institutions. The corresponding applications cover: the improvement of business processes of companies and the management of institutions (*e.g.*, accounting, marketing, cloud manufacturing, etc.); the development of large “smart” applications handling large amounts of geo-distributed data and a large set of resources (video analytics, augmented reality, etc.); the advent of future medical prevention and treatment techniques thanks to the intensive use of ICT systems, etc. We expect our contributions will favor the rise of efficient, correct and sustainable massively geo-distributed infrastructures that are mandatory to design and develop such applications.

4.3. Internet of Skills.

The Internet of Skills is an extension of the Industrial Internet to human activities. It can be seen as the ability to deliver physical experiences remotely (*i.e.*, via the Tactile Internet). Its main supporters advocate that it will revolutionize the way we teach, learn, and interact with pervasive resources. As most applications of the Internet of Skills are related to real time experiences, latency may be even more critical than for the Industrial Internet and raise the locality of computations and resources as a priority. In addition to identifying how Utility Computing infrastructures can cope with this requirement, it is important to determine how the quality of service of such applications should be defined and how latency and bandwidth constraints can be guaranteed at the infrastructure level.

4.4. e-Health

The e-Health domain constitutes an important societal application domain of the two previous areas. The STACK teams is investigating distribution, security and privacy issues in the fields of systems and personalized (aka. precision) medicine. The overall goal in these fields is the development of medication and treatment methods that are tailored towards small groups or even individual patients.

We are working, as part of the ongoing PrivGen CominLabs collaborative project on new means for the sharing of genetic data and applications in the Cloud. More generally, we are applying and developing corresponding techniques for the medical domains of genomics, immunobiology and transplantology in the international network SHLARC and the regional networks SysMics and Oncoshare: there, we investigate how to secure and preserve privacy if potentially sensitive personal data is moved and processed by distributed biomedical analyses.

We are also involved in the SyMeTRIC regional initiative where preliminary studies have been conducted in order to build a common System Medicine computing infrastructure to accelerate the discovery and validation of bio-markers in the fields of oncology, transplantation, and chronic cardiovascular diseases. The challenges were related to the need of being able to perform analyses on data that cannot be moved between distinct locations.

The STACK team will continue to contribute to the e-Health domain by harnessing advanced architectures, applications and infrastructures for the Fog/Edge.

4.5. Network Virtualization and Mobile Edge Services.

Telecom operators have been among the first to advocate the deployment of massively geo-distributed infrastructures, in particular through working groups such as Mobile Edge Computing at the European Telecommunication Standards Institute⁰. The initial reason is that geo-distributed infrastructures will enable Telecom operators to virtualize a large part of their resources and thus reduce capital and operational costs. As an example, we are investigating through the I/O Lab, the joint lab between Orange and Inria, how can a Centralized Radio Access Networks (*a.k.a.* C-RAN or Cloud-RAN) be supported for 5G networks. We highlight that our expertise is not on the network side but rather on where and how we can deploy, allocate and reconfigure software components, which are mandatory to operate a C-RAN infrastructure, in order to guarantee the quality of service expected by the end-users. Finally, working with actors from the network community is a valuable advantage for a distributed system research group such as STACK. Indeed, achievements made within one of the two communities serve the other.

⁰<http://www.etsi.org/news-events/news/1078-2016-04-etsi-mobile-edge-computing-publishes-foundation-specifications>.

WHISPER Project-Team

4. Application Domains

4.1. Linux

Linux is an open-source operating system that is used in settings ranging from embedded systems to supercomputers. The most recent release of the Linux kernel, v4.14, comprises over 16 million lines of code, and supports 30 different families of CPU architectures, around 50 file systems, and thousands of device drivers. Linux is also in a rapid stage of development, with new versions being released roughly every 2.5 months. Recent versions have each incorporated around 13,500 commits, from around 1500 developers. These developers have a wide range of expertise, with some providing hundreds of patches per release, while others have contributed only one. Overall, the Linux kernel is critical software, but software in which the quality of the developed source code is highly variable. These features, combined with the fact that the Linux community is open to contributions and to the use of tools, make the Linux kernel an attractive target for software researchers. Tools that result from research can be directly integrated into the development of real software, where it can have a high, visible impact.

Starting from the work of Engler et al. [41], numerous research tools have been applied to the Linux kernel, typically for finding bugs [40], [57], [67], [76] or for computing software metrics [46], [78]. In our work, we have studied generic C bugs in Linux code [10], bugs in function protocol usage [51], [52], issues related to the processing of bug reports [71] and crash dumps [45], and the problem of backporting [66], [77], illustrating the variety of issues that can be explored on this code base. Unique among research groups working in this area, we have furthermore developed numerous contacts in the Linux developer community. These contacts provide insights into the problems actually faced by developers and serve as a means of validating the practical relevance of our work.

4.2. Device Drivers

Device drivers are essential to modern computing, to provide applications with access, via the operating system, to physical devices such as keyboards, disks, networks, and cameras. Development of new computing paradigms, such as the internet of things, is hampered because device driver development is challenging and error-prone, requiring a high level of expertise in both the targeted OS and the specific device. Furthermore, implementing just one driver is often not sufficient; today's computing landscape is characterized by a number of OSes, *e.g.*, Linux, Windows, MacOS, BSD and many real time OSes, and each is found in a wide range of variants and versions. All of these factors make the development, porting, backporting, and maintenance of device drivers a critical problem for device manufacturers, industry that requires specific devices, and even for ordinary users.

The last fifteen years have seen a number of approaches directed towards easing device driver development. Réveillère, who was supervised by G. Muller, proposes Devil [8], a domain-specific language for describing the low-level interface of a device. Chipounov *et al.* propose RevNic, [34] a template-based approach for porting device drivers from one OS to another. Ryzhyk *et al.* propose Termite, [68], [69] an approach for synthesizing device driver code from a specification of an OS and a device. Currently, these approaches have been successfully applied to only a small number of toy drivers. Indeed, Kadav and Swift [47] observe that these approaches make assumptions that are not satisfied by many drivers; for example, the assumption that a driver involves little computation other than the direct interaction between the OS and the device. At the same time, a number of tools have been developed for finding bugs in driver code. These tools include SDV [25], Coverity [41], CP-Miner, [56] PR-Miner [57], and Coccinelle [9]. These approaches, however, focus on analyzing existing code, and do not provide guidelines on structuring drivers.

In summary, there is still a need for a methodology that first helps the developer understand the software architecture of drivers for commonly used operating systems, and then provides tools for the maintenance of existing drivers.

WIDE Project-Team (section vide)

ALPINES Project-Team

4. Application Domains

4.1. Compositional multiphase Darcy flow in heterogeneous porous media

We study the simulation of compositional multiphase flow in porous media with different types of applications, and we focus in particular on reservoir/bassin modeling, and geological CO₂ underground storage. All these simulations are linearized using Newton approach, and at each time step and each Newton step, a linear system needs to be solved, which is the most expensive part of the simulation. This application leads to some of the difficult problems to be solved by iterative methods. This is because the linear systems arising in multiphase porous media flow simulations cumulate many difficulties. These systems are non-symmetric, involve several unknowns of different nature per grid cell, display strong or very strong heterogeneities and anisotropies, and change during the simulation. Many researchers focus on these simulations, and many innovative techniques for solving linear systems have been introduced while studying these simulations, as for example the nested factorization [Appleyard and Cheshire, 1983, SPE Symposium on Reservoir Simulation].

4.2. Inverse problems

We focus on methods related to the blend of time reversal techniques and absorbing boundary conditions (ABC) used in a non standard way. Since the seminal paper by [M. Fink et al., Imaging through inhomogeneous media using time reversal mirrors. Ultrasonic Imaging, 13(2):199, 1991.], time reversal is a subject of very active research. The principle is to back-propagate signals to the sources that emitted them. The initial experiment was to refocus, very precisely, a recorded signal after passing through a barrier consisting of randomly distributed metal rods. In [de Rosny and Fink. Overcoming the diffraction limit in wave physics using a time-reversal mirror and a novel acoustic sink. Phys. Rev. Lett., 89 (12), 2002], the source that created the signal is time reversed in order to have a perfect time reversal experiment. In [39], we improve this result from a numerical point of view by showing that it can be done numerically without knowing the source. This is done at the expense of not being able to recover the signal in the vicinity of the source. In [40], time dependent wave splitting is performed using ABC and time reversal techniques. We now work on extending these methods to non uniform media.

All our numerical simulations are performed in FreeFem++ which is very flexible. As a byproduct, it enables us to have an end user point of view with respect to FreeFem++ which is very useful for improving it.

4.3. Numerical methods for wave propagation in multi-scale media

We are interested in the development of fast numerical methods for the simulation of electromagnetic waves in multi-scale situations where the geometry of the medium of propagation may be described through characteristic lengths that are, in some places, much smaller than the average wavelength. In this context, we propose to develop numerical algorithms that rely on simplified models obtained by means of asymptotic analysis applied to the problem under consideration.

Here we focus on situations involving boundary layers and *localized* singular perturbation problems where wave propagation takes place in media whose geometry or material characteristics are submitted to a small scale perturbation localized around a point, or a surface, or a line, but not distributed over a volumic sub-region of the propagation medium. Although a huge literature is already available for the study of localized singular perturbations and boundary layer phenomena, very few works have proposed efficient numerical methods that rely on asymptotic modeling. This is due to their functional framework that naturally involves singular functions, which are difficult to handle numerically. The aim of this part of our research is to develop and analyze numerical methods for singular perturbation methods that are prone to high order numerical approximation, and robust with respect to the small parameter characterizing the singular perturbation.

4.4. Data analysis in astrophysics

We focus on computationally intensive numerical algorithms arising in the data analysis of current and forthcoming Cosmic Microwave Background (CMB) experiments in astrophysics. This application is studied in collaboration with researchers from University Paris Diderot, and the objective is to make available the algorithms to the astrophysics community, so that they can be used in large experiments.

In CMB data analysis, astrophysicists produce and analyze multi-frequency 2D images of the universe when it was 5% of its current age. The new generation of the CMB experiments observes the sky with thousands of detectors over many years, producing overwhelmingly large and complex data sets, which nearly double every year therefore following Moore's Law. Planck (<http://planck.esa.int/>) is a keystone satellite mission which has been developed under auspices of the European Space Agency (ESA). Planck has been surveying the sky since 2010, produces terabytes of data and requires 100 Petaflops per image analysis of the universe. It is predicted that future experiments will collect on the order of petabyte of data around 2025. This shows that data analysis in this area, as many other applications, will keep pushing the limit of available supercomputing power for the years to come.

4.5. Molecular simulations

Molecular simulation is one of the most dynamic areas of scientific computing. Its field of application is very broad, ranging from theoretical chemistry and drug design to materials science and nanotechnology. It provides many challenging problems to mathematicians and computer scientists.

In the context of the ERC Synergy Grant EMC2 we address several important limitations of state of the art molecular simulation. In particular, the simulation of very large molecular systems, or smaller systems in which electrons interact strongly with each other, remains out of reach today. In an interdisciplinary collaboration between chemists, mathematicians and computer scientists, we focus on developing a new generation of reliable molecular simulation algorithms and software.

AVALON Project-Team

4. Application Domains

4.1. Overview

The AVALON team targets applications with large computing and/or data storage needs, which are still difficult to program, maintain, and deploy. Those applications can be parallel and/or distributed applications, such as large scale simulation applications or code coupling applications. Applications can also be workflow-based as commonly found in distributed systems such as grids or clouds.

The team aims at not being restricted to a particular application field, thus avoiding any spotlight. The team targets different HPC and distributed application fields, which brings use cases with different issues. This will be eased by our various collaborations: the team participates to the INRIA-Illinois Joint Laboratory for Petascale Computing, the Physics, Radiobiology, Medical Imaging, and Simulation French laboratory of excellence, the E-Biothon project, the INRIA large scale initiative Computer and Computational Sciences at Exascale (C2S@Exa), and to BioSyL, a federative research structure about Systems Biology of the University of Lyon. Moreover, the team members have a long tradition of cooperation with application developers such as CERFACS and EDF R&D. Last but not least, the team has a privileged connection with CC-IN2P3 that opens up collaborations, in particular in the astrophysics field.

In the following, some examples of representative applications that we are targeting are presented. In addition to highlighting some application needs, they also constitute some of the use cases that will be used to validate our theoretical results.

4.2. Climatology

The world's climate is currently changing due to the increase of the greenhouse gases in the atmosphere. Climate fluctuations are forecasted for the years to come. For a proper study of the incoming changes, numerical simulations are needed, using general circulation models of a climate system. Simulations can be of different types: HPC applications (*e.g.*, the NEMO framework [28] for ocean modelization), code-coupling applications (*e.g.*, the OASIS coupler [33] for global climate modeling), or workflows (long term global climate modeling).

As for most applications the team is targeting, the challenge is to thoroughly analyze climate-forecasting applications to model their needs in terms of programming model, execution model, energy consumption, data access pattern, and computing needs. Once a proper model of an application has been set up, appropriate scheduling heuristics can be designed, tested, and compared. The team has a long tradition of working with CERFACS on this topic, for example in the LEGO (2006-09) and SPADES (2009-12) French ANR projects.

4.3. Astrophysics

Astrophysics is a major field to produce large volumes of data. For instance, the Vera C. Rubin Observatory (<https://www.vro.org/>) will produce 20 TB of data every night, with the goals of discovering thousands of exoplanets and of uncovering the nature of dark matter and dark energy in the universe. The Square Kilometer Array (<http://www.skatelescope.org/>) produces 9 Tbits/s of raw data. One of the scientific projects related to this instrument called Evolutionary Map of the Universe is working on more than 100 TB of images. The Euclid Imaging Consortium (<https://www.euclid-ec.org/>) will generate 1 PB data per year.

AVALON collaborates with the *Institut de Physique des deux Infinis de Lyon* (IP2I) laboratory on large scale numerical simulations in astronomy and astrophysics. Contributions of the AVALON members have been related to algorithmic skeletons to demonstrate large scale connectivity, the development of procedures for the generation of realistic mock catalogs, and the development of a web interface to launch large cosmological simulations on GRID'5000.

This collaboration, that continues around the topics addressed by the CLUES project (<http://www.clues-project.org>), has been extended thanks to the tight links with the CC-IN2P3. Major astrophysics projects execute part of their computing, and store part of their data on the resources provided by the CC-IN2P3. Among them, we can mention SNFactory, Euclid, or VRO. These applications constitute typical use cases for the research developed in the AVALON team: they are generally structured as workflows and a huge amount of data (from TB to PB) is involved.

4.4. Bioinformatics

Large-scale data management is certainly one of the most important applications of distributed systems in the future. Bioinformatics is a field producing such kinds of applications. For example, DNA sequencing applications make use of MapReduce skeletons.

The AVALON team is a member of BioSyL (<http://www.biosyl.org>), a Federative Research Structure attached to University of Lyon. It gathers about 50 local research teams working on systems biology. Moreover, the team cooperated with the French Institute of Biology and Chemistry of Proteins (IBCP <http://www.ibcp.fr>) in particular through the ANR MapReduce project where the team focuses on a bio-chemistry application dealing with protein structure analysis. AVALON has also started working with the Inria Beagle team (<https://team.inria.fr/beagle/>) on artificial evolution and computational biology as the challenges are around high performance computation and data management.

DATAMOVE Project-Team

4. Application Domains

4.1. Data Aware Batch Scheduling

Large scale high performance computing platforms are becoming increasingly complex. Determining efficient allocation and scheduling strategies that can adapt to technological evolutions is a strategic and difficult challenge. We are interested in scheduling jobs in hierarchical and heterogeneous large scale platforms. On such platforms, application developers typically submit their jobs in centralized waiting queues. The job management system aims at determining a suitable allocation for the jobs, which all compete against each other for the available computing resources. Performances are measured using different classical metrics like maximum completion time or slowdown. Current systems make use of very simple (but fast) algorithms that however rely on simplistic platform and execution models, and thus, have limited performances.

For all target scheduling problems we aim to provide both theoretical analysis and complementary analysis through simulations. Achieving meaningful results will require strong improvements on existing models (on power for example) and the design of new approximation algorithms with various objectives such as stretch, reliability, throughput or energy consumption, while keeping in focus the need for a low-degree polynomial complexity.

4.1.1. Algorithms

The most common batch scheduling policy is to consider the jobs according to the First Come First Served order (FCFS) with backfilling (BF). BF is the most widely used policy due to its easy and robust implementation and known benefits such as high system utilization. It is well-known that this strategy does not optimize any sophisticated function, but it is simple to implement and it guarantees that there is no starvation (i.e. every job will be scheduled at some moment).

More advanced algorithms are seldom used on production platforms due to both the gap between theoretical models and practical systems and speed constraints. When looking at theoretical scheduling problems, the generally accepted goal is to provide polynomial algorithms (in the number of submitted jobs and the number of involved computing units). However, with millions of processing cores where every process and data transfer have to be individually scheduled, polynomial algorithms are prohibitive as soon as the polynomial degree is too large. The model of *parallel tasks* simplifies this problem by bundling many threads and communications into single boxes, either rigid, rectangular or malleable. Especially malleable tasks capture the dynamicity of the execution. Yet these models are ill-adapted to heterogeneous platforms, as the running time depends on more than simply the number of allotted resources, and some of the common underlying assumptions on the speed-up functions (such as monotony or concavity) are most often only partially verified.

In practice, the job execution times depend on their allocation (due to communication interferences and heterogeneity in both computation and communication), while theoretical models of parallel jobs usually consider jobs as black boxes with a fixed (maximum) execution time. Though interesting and powerful, the classical models (namely, synchronous PRAM model, delay, LogP) and their variants (such as hierarchical delay), are not well-suited to large scale parallelism on platforms where the cost of moving data is significant, non uniform and may change over time. Recent studies are still refining such models in order to take into account communication contentions more accurately while remaining tractable enough to provide a useful tool for algorithm design.

Today, all algorithms in use in production systems are oblivious to communications. One of our main goals is to **design a new generation of scheduling algorithms fitting more closely job schedules according to platform topologies.**

4.1.2. Locality Aware Allocations

Recently, we developed modifications of the standard back-filling algorithm taking into account platform topologies. The proposed algorithms take into account locality and contiguity in order to hide communication patterns within parallel tasks. The main result here is to establish good lower bounds and small approximation ratios for policies respecting the locality constraints. The algorithms work in an online fashion, improving the global behavior of the system while still keeping a low running time. These improvements rely mainly on our past experience in designing approximation algorithms. Instead of relying on complex networking models and communication patterns for estimating execution times, the communications are disconnected from the execution time. Then, the scheduling problem leads to a trade-off: optimizing locality of communications on one side and a performance objective (like the makespan or stretch) on the other side.

In the perspective of taking care of locality, other ongoing works include the study of schedulers for platforms whose interconnection network is a static structured topology (like the 3D-torus of the BlueWaters platform we work on in collaboration with the Argonne National Laboratory). One main characteristic of this 3D-torus platform is to provide I/O nodes at specific locations in the topology. Applications generate and access specific data and are thus bounded to specific I/O nodes. Resource allocations are constrained in a strong and unusual way. This problem is close for actual hierarchical platforms. The scheduler needs to compute a schedule such that I/O nodes requirements are filled for each application while at the same time avoiding communication interferences. Moreover, extra constraints can arise for applications requiring accelerators that are gathered on the nodes at the edge of the network topology.

While current results are encouraging, they are however limited in performance by the low amount of information available to the scheduler. We look forward to extend ongoing work by progressively increasing application and network knowledge (by technical mechanisms like profiling or monitoring or by more sophisticated methods like learning). It is also important to anticipate on application resource usage in terms of compute units, memory as well as network and I/Os to efficiently schedule a mix of applications with different profiles. For instance, a simple solution is to partition the jobs as "communication intensive" or "low communications". Such a tag could be achieved by the users themselves or obtained by learning techniques. We could then schedule low communications jobs using leftover spaces while taking care of high communication jobs. More sophisticated options are possible, for instance those that use more detailed communication patterns and networking models. Such options would leverage the work proposed in Section 4.2 for gathering application traces.

4.1.3. Data-Centric Processing

Exascale computing is shifting away from the traditional compute-centric models to a more data-centric one. This is driven by the evolving nature of large scale distributed computing, no longer dominated by pure computations but also by the need to handle and analyze large volumes of data. These data can be large databases of results, data streamed from a running application or another scientific instrument (collider for instance). These new workloads call for specific resource allocation strategies.

Data movements and storage are expected to be a major energy and performance bottleneck on next generation platforms. Storage architectures are also evolving, the standard centralized parallel file system being complemented with local persistent storage (Burst Buffers, NVRAM). Thus, one data producer can stage data on some nodes' local storage, requiring to schedule close by the associated analytics tasks to limit data movements. This kind of configuration, often referred as *in-situ analytics*, is expected to become common as it enables to switch from the traditional I/O intensive workflow (batch-processing followed by *post mortem* analysis and visualization) to a more storage conscious approach where data are processed as closely as possible to where and when they are produced (in-situ processing is addressed in details in section 4.3). By reducing data movements and scheduling the extra processing on resources not fully exploited yet, in-situ processing is expected to have also a significant positive energetic impact. Analytics codes can be executed in the same nodes than the application, often on dedicated cores commonly called helper cores, or on dedicated nodes called staging nodes. The results are either forwarded to the users for visualization or saved to disk through I/O nodes. In-situ analytics can also take benefit of node local disks or burst buffers to reduce data movements.

Future job scheduling strategies should take into account in-situ processes in addition to the job allocation to optimize both energy consumption and execution time. On the one hand, this problem can be reduced to an allocation problem of extra asynchronous tasks to idle computing units. But on the other hand, embedding analytics in applications brings extra difficulties by making the application more heterogeneous and imposing more constraints (data affinity) on the required resources. Thus, the main point here is to develop efficient algorithms for dealing with heterogeneity without increasing the global computational cost.

4.1.4. Learning

Another important issue is to adapt the job management system to deal with the bad effects of uncertainties, which may be catastrophic in large scale heterogeneous HPC platforms (jobs delayed arbitrarily far or jobs killed). A natural question is then: *is it possible to have a good estimation of the job and platform parameters in order to be able to obtain a better scheduling ?* Many important parameters (like the number or type of required resources or the estimated running time of the jobs) are asked to the users when they submit their jobs. However, some of these values are not accurate and in many cases, they are not even provided by the end-users. In DataMove, we propose to study new methods for a better prediction of the characteristics of the jobs and their execution in order to improve the optimization process. In particular, the methods well-studied in the field of big data (in supervised Machine Learning, like classical regression methods, Support Vector Methods, random forests, learning to rank techniques or deep learning) could and must be used to improve job scheduling in large scale HPC platforms. This topic received a great attention recently in the field of parallel and distributed processing. A preliminary study has been done recently by our team with the target of predicting the job running times (called wall times). We succeeded to improve significantly in average the reference EASY Back Filling algorithm by estimating the wall time of the jobs, however, this method leads to big delay for the stretch of few jobs. Even if we succeed in determining more precisely hidden parameters, like the wall time of the jobs, this is not enough to determine an optimized solution. The shift is not only to learn on dedicated parameters but also on the scheduling policy. The data collected from the accounting and profiling of jobs can be used to better understand the needs of the jobs and through learning to propose adaptations for future submissions. The goal is to propose extensions to further improve the job scheduling and improve the performance and energy efficiency of the application. For instance preference learning may enable to compute on-line new priorities to back-fill the ready jobs.

4.1.5. Multi-objective Optimization

Several optimization questions that arise in allocation and scheduling problems lead to the study of several objectives at the same time. The goal is then not a single optimal solution, but a more complicated mathematical object that captures the notion of trade-off. In broader terms, the goal of multi-objective optimization is not to externally arbitrate on disputes between entities with different goals, but rather to explore the possible solutions to highlight the whole range of interesting compromises. A classical tool for studying such multi-objective optimization problems is to use *Pareto curves*. However, the full description of the Pareto curve can be very hard because of both the number of solutions and the hardness of computing each point. Addressing this problem will opens new methodologies for the analysis of algorithms.

To further illustrate this point here are three possible case studies with emphasis on conflicting interests measured with different objectives. While these cases are good representatives of our HPC context, there are other pertinent trade-offs we may investigate depending on the technology evolution in the coming years. This enumeration is certainly not limitative.

Energy versus Performance. The classical scheduling algorithms designed for the purpose of performance can no longer be used because performance and energy are contradictory objectives to some extent. The scheduling problem with energy becomes a multi-objective problem in nature since the energy consumption should be considered as equally important as performance at exascale. A global constraint on energy could be a first idea for determining trade-offs but the knowledge of the Pareto set (or an approximation of it) is also very useful.

Administrators versus application developers. Both are naturally interested in different objectives: In current algorithms, the performance is mainly computed from the point of view of administrators, but the users should be in the loop since they can give useful information and help to the construction of better schedules. Hence, we face again a multi-objective problem where, as in the above case, the approximation of the Pareto set provides the trade-off between the administrator view and user demands. Moreover, the objectives are usually of the same nature. For example, *max stretch* and *average stretch* are two objectives based on the slowdown factor that can interest administrators and users, respectively. In this case the study of the norm of stretch can be also used to describe the trade-off (recall that the L_1 -norm corresponds to the average objective while the L_∞ -norm to the max objective). Ideally, we would like to design an algorithm that gives good approximate solutions at the same time for all norms. The L_2 or L_3 -norm are useful since they describe the performance of the whole schedule from the administrator point of view as well as they provide a fairness indication to the users. The hard point here is to derive theoretical analysis for such complicated tools.

Resource Augmentation. The classical resource augmentation models, i.e. speed and machine augmentation, are not sufficient to get good results when the execution of jobs cannot be frequently interrupted. However, based on a resource augmentation model recently introduced, where the algorithm may reject a small number of jobs, some members of our team have given the first interesting results in the non-preemptive direction. In general, resource augmentation can explain the intuitive good behavior of some greedy algorithms while, more interestingly, it can give ideas for new algorithms. For example, in the rejection context we could dedicate a small number of nodes for the usually problematic rejected jobs. Some initial experiments show that this can lead to a schedule for the remaining jobs that is very close to the optimal one.

4.2. Empirical Studies of Large Scale Platforms

Experiments or realistic simulations are required to take into account the impact of allocations and assess the real behavior of scheduling algorithms. While theoretical models still have their interest to lay the groundwork for algorithmic designs, the models are necessarily reflecting a purified view of the reality. As transferring our algorithm in a more practical setting is an important part of our creed, we need to ensure that the theoretical results found using simplified models can really be transposed to real situations. On the way to exascale computing, large scale systems become harder to study, to develop or to calibrate because of the costs in both time and energy of such processes. It is often impossible to convince managers to use a production cluster for several hours simply to test modifications in the RJMS. Moreover, as the existing RJMS production systems need to be highly reliable, each evolution requires several real scale test iterations. The consequence is that scheduling algorithms used in production systems are mostly outdated and not customized correctly. To circumvent this pitfall, we need to develop tools and methodologies for alternative empirical studies, from analysis of workload traces, to job models, simulation and emulation with reproducibility concerns.

4.2.1. Workload Traces with Resource Consumption

Workload traces are the base element to capture the behavior of complete systems composed of submitted jobs, running applications, and operating tools. These traces must be obtained on production platforms to provide relevant and representative data. To get a better understanding of the use of such systems, we need to look at both, how the jobs interact with the job management system, and how they use the allocated resources. We propose a general workload trace format that adds jobs resource consumption to the commonly used SWF⁰ workload trace format. This requires to instrument the platforms, in particular to trace resource consumptions like CPU, data movements at memory, network and I/O levels, with an acceptable performance impact. In a previous work we studied and proposed a dedicated job monitoring tool whose impact on the system has been measured as lightweight (0.35% speed-down) with a 1 minute sampling rate. Other tools also explore job monitoring, like TACC Stats. A unique feature from our tool is its ability to monitor distinctly jobs sharing common nodes.

⁰Standard Workload Format: <http://www.cs.huji.ac.il/labs/parallel/workload/swf.html>

Collected workload traces with jobs resource consumption will be publicly released and serve to provide data for works presented in Section 4.1 . The trace analysis is expected to give valuable insights to define models encompassing complex behaviours like network topology sensitivity, network congestion and resource interferences.

We expect to join efforts with partners for collecting quality traces (ATOS/Bull, Ciment meso center, Joint Laboratory on Extreme Scale Computing) and will collaborate with the Inria team POLARIS for their analysis.

4.2.2. Simulation

Simulations of large scale systems are faster by multiple orders of magnitude than real experiments. Unfortunately, replacing experiments with simulations is not as easy as it may sound, as it brings a host of new problems to address in order to ensure that the simulations are closely approximating the execution of typical workloads on real production clusters. Most of these problems are actually not directly related to scheduling algorithms assessment, in the sense that the workload and platform models should be defined independently from the algorithm evaluations, in order to ensure a fair assessment of the algorithms' strengths and weaknesses. These research topics (namely platform modeling, job models and simulator calibration) are addressed in the other subsections.

We developed an open source platform simulator within DataMove (in conjunction with the OAR development team) to provide a widely distributable test bed for reproducible scheduling algorithm evaluation. Our simulator, named Batsim, allows to simulate the behavior of a computational platform executing a workload scheduled by any given scheduling algorithm. To obtain sound simulation results and to broaden the scope of the experiments that can be done thanks to Batsim, we did not chose to create a (necessarily limited) simulator from scratch, but instead to build on top of the SimGrid simulation framework.

To be open to as many batch schedulers as possible, Batsim decouples the platform simulation and the scheduling decisions in two clearly-separated software components communicating through a complete and documented protocol. The Batsim component is in charge of simulating the computational resources behaviour whereas the scheduler component is in charge of taking scheduling decisions. The scheduler component may be both a resource and a job management system. For jobs, scheduling decisions can be to execute a job, to delay its execution or simply to reject it. For resources, other decisions can be taken, for example to change the power state of a machine i.e. to change its speed (in order to lower its energy consumption) or to switch it on or off. This separation of concerns also enables interfacing with potentially any commercial RJMS, as long as the communication protocol with Batsim is implemented. A proof of concept is already available with the OAR RJMS.

Using this test bed opens new research perspectives. It allows to test a large range of platforms and workloads to better understand the real behavior of our algorithms in a production setting. In turn, this opens the possibility to tailor algorithms for a particular platform or application, and to precisely identify the possible shortcomings of the theoretical models used.

4.2.3. Job and Platform Models

The central purpose of the Batsim simulator is to simulate job behaviors on a given target platform under a given resource allocation policy. Depending on the workload, a significant number of jobs are parallel applications with communications and file system accesses. It is not conceivable to simulate individually all these operations for each job on large platforms with their associated workload due to implied simulation complexity. The challenge is to define a coarse grain job model accurate enough to reproduce parallel application behavior according to the target platform characteristics. We will explore models similar to the BSP (Bulk Synchronous Program) approach that decomposes an application in local computation supersteps ended by global communications and a global synchronization. The model parameters will be established by means of trace analysis as discussed previously, but also by instrumenting some parallel applications to capture communication patterns. This instrumentation will have a significant impact on the concerned application performance, restricting its use to a few applications only. There are a lot of recurrent applications executed on HPC platform, this fact will help to reduce the required number of instrumentations and captures. To assign

each job a model, we are considering to adapt the concept of application signatures as proposed in. Platform models and their calibration are also required. Large parts of these models, like those related to network, are provided by Simgrid. Other parts as the filesystem and energy models are comparatively recent and will need to be enhanced or reworked to reflect the HPC platform evolutions. These models are then generally calibrated by running suitable benchmarks.

4.2.4. Emulation and Reproducibility

The use of coarse models in simulation implies to set aside some details. This simplification may hide system behaviors that could impact significantly and negatively the metrics we try to enhance. This issue is particularly relevant when large scale platforms are considered due to the impossibility to run tests at nominal scale on these real platforms. A common approach to circumvent this issue is the use of emulation techniques to reproduce, under certain conditions, the behavior of large platforms on smaller ones. Emulation represents a natural complement to simulation by allowing to execute directly large parts of the actual evaluated software and system, but at the price of larger compute times and a need for more resources. The emulation approach was chosen in to compare two job management systems from workload traces of the CURIE supercomputer (80000 cores). The challenge is to design methods and tools to emulate with sufficient accuracy the platform and the workload (data movement, I/O transfers, communication, applications interference). We will also intend to leverage emulation tools like Distem from the MADYNES team. It is also important to note that the Batsim simulator also uses emulation techniques to support the core scheduling module from actual RJMS. But the integration level is not the same when considering emulation for larger parts of the system (RJMS, compute node, network and filesystem).

Replaying traces implies to prepare and manage complex software stacks including the OS, the resource management system, the distributed filesystem and the applications as well as the tools required to conduct experiments. Preparing these stacks generate specific issues, one of the major one being the support for reproducibility. We propose to further develop the concept of reconstructability to improve experiment reproducibility by capturing the build process of the complete software stack. This approach ensures reproducibility over time better than other ways by keeping all data (original packages, build recipe and Kameleon engine) needed to build the software stack.

In this context, the Grid'5000 (see Sec. 6.4) experimentation infrastructure that gives users the control on the complete software stack is a crucial tool for our research goals. We will pursue our strong implication in this infrastructure.

4.3. Integration of High Performance Computing and Data Analytics

Data produced by large simulations are traditionally handled by an I/O layer that moves them from the compute cores to the file system. Analysis of these data are performed after reading them back from files, using some domain specific codes or some scientific visualisation libraries like VTK. But writing and then reading back these data generates a lot of data movements and puts under pressure the file system. To reduce these data movements, **the in situ analytics paradigm proposes to process the data as closely as possible to where and when the data are produced.** Some early solutions emerged either as extensions of visualisation tools or of I/O libraries like ADIOS. But significant progresses are still required to provide efficient and flexible high performance scientific data analysis tools. Integrating data analytics in the HPC context will have an impact on resource allocation strategies, analysis algorithms, data storage and access, as well as computer architectures and software infrastructures. But this paradigm shift imposed by the machine performance also sets the basis for a deep change on the way users work with numerical simulations. The traditional workflow needs to be reinvented to make HPC more user-centric, more interactive and turn HPC into a commodity tool for scientific discovery and engineering developments. In this context DataMove aims at investigating programming environments for in situ analytics with a specific focus on task scheduling in particular, to ensure an efficient sharing of resources with the simulation.

4.3.1. Programming Model and Software Architecture

In situ creates a tighter loop between the scientist and her/his simulation. As such, an in situ framework needs to be flexible to let the user define and deploy its own set of analysis. A manageable flexibility requires to favor simplicity and understandability, while still enabling an efficient use of parallel resources. Visualization libraries like VTK or Visit, as well as domain specific environments like VMD have initially been developed for traditional post-mortem data analysis. They have been extended to support in situ processing with some simple resource allocation strategies but the level of performance, flexibility and ease of use that is expected requires to rethink new environments. There is a need to develop a middleware and programming environment taking into account in its foundations this specific context of high performance scientific analytics.

Similar needs for new data processing architectures occurred for the emerging area of Big Data Analytics, mainly targeted to web data on cloud-based infrastructures. Google Map/Reduce and its successors like Spark or Stratosphere/Flink have been designed to match the specific context of efficient analytics for large volumes of data produced on the web, on social networks, or generated by business applications. These systems have mainly been developed for cloud infrastructures based on commodity architectures. They do not leverage the specifics of HPC infrastructures. Some preliminary adaptations have been proposed for handling scientific data in a HPC context. However, these approaches do not support in situ processing.

Following the initial development of FlowVR, our middleware for in situ processing, we will pursue our effort to develop a programming environment and software architecture for high performance scientific data analytics. Like FlowVR, the map/reduce tools, as well as the machine learning frameworks like TensorFlow, adopted a dataflow graph for expressing analytics pipe-lines. We are convinced that this dataflow approach is both easy to understand and yet expresses enough concurrency to enable efficient executions. The graph description can be compiled towards lower level representations, a mechanism that is intensively used by Stratosphere/Flink for instance. Existing in situ frameworks, including FlowVR, inherit from the HPC way of programming with a thinner software stack and a programming model close to the machine. Though this approach enables to program high performance applications, this is usually too low level to enable the scientist to write its analysis pipe-line in a short amount of time. The data model, i.e. the data semantics level accessible at the framework level for error check and optimizations, is also a fundamental aspect of such environments. The key/value store has been adopted by all map/reduce tools. Except in some situations, it cannot be adopted as such for scientific data. Results from numerical simulations are often more structured than web data, associated with acceleration data structures to be processed efficiently. We will investigate data models for scientific data building on existing approaches like Adios or DataSpaces.

4.3.2. Resource Sharing

To alleviate the I/O bottleneck, the in situ paradigm proposes to start processing data as soon as made available by the simulation, while still residing in the memory of the compute node. In situ processings include data compression, indexing, computation of various types of descriptors (1D, 2D, images, etc.). Per se, reducing data output to limit I/O related performance drops or keep the output data size manageable is not new. Scientists have relied on solutions as simple as decreasing the frequency of result savings. In situ processing proposes to move one step further, by providing a full fledged processing framework enabling scientists to more easily and thoroughly manage the available I/O budget.

The most direct way to perform in situ analytics is to inline computations directly in the simulation code. In this case, in situ processing is executed in sequence with the simulation that is suspended meanwhile. Though this approach is direct to implement and does not require complex framework environments, it does not enable to overlap analytics related computations and data movements with the simulation execution, preventing to efficiently use the available resources. Instead of relying on this simple time sharing approach, several works propose to rely on space sharing where one or several cores per node, called *helper cores*, are dedicated to analytics. The simulation responsibility is simply to handle a copy of the relevant data to the node-local in situ processes, both codes being executed concurrently. This approach often lead to significantly better performance than in-simulation analytics.

For a better isolation of the simulation and in situ processes, one solution consists in offloading in situ tasks from the simulation nodes towards extra dedicated nodes, usually called *staging nodes*. These computations are said to be performed *in-transit*. But this approach may not always be beneficial compared to processing on simulation nodes due to the costs of moving the data from the simulation nodes to the staging nodes.

FlowVR enables to mix these different resources allocation strategies for the different stages of an analytics pipeline. Based on a component model, the scientist designs analytics workflows by first developing processing components that are next assembled in a dataflow graph through a Python script. At runtime the graph is instantiated according to the execution context, FlowVR taking care of deploying the application on the target architecture, and of coordinating the analytics workflows with the simulation execution.

But today the choice of the resource allocation strategy is mostly ad-hoc and defined by the programmer. We will investigate solutions that enable a cooperative use of the resource between the analytics and the simulation with minimal hints from the programmer. In situ processings inherit from the parallelization scale and data distribution adopted by the simulation, and must execute with minimal perturbations on the simulation execution (whose actual resource usage is difficult to know a priori). We need to develop adapted scheduling strategies that operate at compile and run time. Because analysis are often data intensive, such solutions must take into consideration data movements, a point that classical scheduling strategies designed first for compute intensive applications often overlook. We expect to develop new scheduling strategies relying on the methodologies developed in Sec. 4.1.5 . Simulations as well as analysis are iterative processes exposing a strong spatial and temporal coherency that we can take benefit of to anticipate their behavior and then take more relevant resources allocation strategies, possibly based on advanced learning algorithms or as developed in Section 4.1 .

In situ analytics represent a specific workload that needs to be scheduled very closely to the simulation, but not necessarily active during the full extent of the simulation execution and that may also require to access data from previous runs (stored in the file system or on specific burst-buffers). Several users may also need to run concurrent analytics pipe-lines on shared data. This departs significantly from the traditional batch scheduling model, motivating the need for a more elastic approach to resource provisioning. These issues will be conjointly addressed with research on batch scheduling policies (Sec. 4.1).

4.3.3. Co-Design with Data Scientists

Given the importance of users in this context, it is of primary importance that in situ tools be co-designed with advanced users, even if such multidisciplinary collaborations are challenging and require constant long term investments to learn and understand the specific practices and expectations of the other domain.

We will tightly collaborate with scientists of some application domains, like molecular dynamics or fluid simulation, to design, develop, deploy and assess in situ analytics scenarios, as already done with Marc Baaden, a computational biologist from LBT.

HIEPACS Project-Team

4. Application Domains

4.1. Material physics

Participants: Olivier Coulaud, Pierre Esterie.

Due to the increase of available computer power, new applications in nano science and physics appear such as study of properties of new materials (photovoltaic materials, bio- and environmental sensors, ...), failure in materials, nano-indentation. Chemists, physicists now commonly perform simulations in these fields. These computations simulate systems up to billion of atoms in materials, for large time scales up to several nanoseconds. The larger the simulation, the smaller the computational cost of the potential driving the phenomena, resulting in low precision results. So, if we need to increase the precision, there are two ways to decrease the computational cost. In the first approach, we improve algorithms and their parallelization and in the second way, we will consider a multiscale approach.

A domain of interest is the material aging for the nuclear industry. The materials are exposed to complex conditions due to the combination of thermo-mechanical loading, the effects of irradiation and the harsh operating environment. This operating regime makes experimentation extremely difficult and we must rely on multi-physics and multi-scale modeling for our understanding of how these materials behave in service. This fundamental understanding helps not only to ensure the longevity of existing nuclear reactors, but also to guide the development of new materials for 4th generation reactor programs and dedicated fusion reactors. For the study of crystalline materials, an important tool is dislocation dynamics (DD) modeling. This multiscale simulation method predicts the plastic response of a material from the underlying physics of dislocation motion. DD serves as a crucial link between the scale of molecular dynamics and macroscopic methods based on finite elements; it can be used to accurately describe the interactions of a small handful of dislocations, or equally well to investigate the global behavior of a massive collection of interacting defects.

To explore i.e. to simulate these new areas, we need to develop and/or to improve significantly models, schemes and solvers used in the classical codes. In the project, we want to accelerate algorithms arising in those fields. We will focus on the following topics (in particular in the currently under definition **OPTIDIS** project in collaboration with CEA Saclay, CEA Ile-de-france and SIMaP Laboratory in Grenoble) in connection with research described at Sections 3.4 and 3.5 .

- The interaction between dislocations is long ranged ($O(1/r)$) and anisotropic, leading to severe computational challenges for large-scale simulations. In dislocation codes, the computation of interaction forces between dislocations is still the most CPU time consuming and has to be improved to obtain faster and more accurate simulations.
- In such simulations, the number of dislocations grows while the phenomenon occurs and these dislocations are not uniformly distributed in the domain. This means that strategies to dynamically construct a good load balancing are crucial to achieve high performance.
- From a physical and a simulation point of view, it will be interesting to couple a molecular dynamics model (atomistic model) with a dislocation one (mesoscale model). In such three-dimensional coupling, the main difficulties are firstly to find and characterize a dislocation in the atomistic region, secondly to understand how we can transmit with consistency the information between the two micro and meso scales.

4.2. Co-design of algorithms in scientific applications

Participants: Emmanuel Agullo, Aurélien Falco, Olivier Beaumont, Lionel Eyraud-Dubois, Mathieu Faverge, Marek Felsoci, Luc Giraud, Gilles Marait, Van Gia Thinh Nguyen, Pierre Ramet, Guillaume Sylvand, Alena Shilova.

4.2.1. High performance simulation for ITER tokamak

Scientific simulation for ITER tokamak modeling provides a natural bridge between theory and experimentation and is also an essential tool for understanding and predicting plasma behavior. Recent progresses in numerical simulation of fine-scale turbulence and in large-scale dynamics of magnetically confined plasma have been enabled by access to petascale supercomputers. These progresses would have been unreachable without new computational methods and adapted reduced models. In particular, the plasma science community has developed codes for which computer runtime scales quite well with the number of processors up to thousands cores. The research activities of HIEPACS concerning the international ITER challenge have started in the Inria Project Lab C2S@EXA in collaboration with CEA-IRFM and were related to two complementary studies: a first one concerning the turbulence of plasma particles inside a tokamak (in the context of GYSELA code) and a second one concerning the MHD instability edge localized modes (in the context of JOREK code). The activity concerning GYSELA was completed at the end of 2018.

Other numerical simulation tools designed for the ITER challenge aim at making a significant progress in understanding active control methods of plasma edge MHD instability Edge Localized Modes (ELMs) which represent a particular danger with respect to heat and particle loads for Plasma Facing Components (PFC) in the tokamak. The goal is to improve the understanding of the related physics and to propose possible new strategies to improve effectiveness of ELM control techniques. The simulation tool used (JOREK code) is related to non linear MHD modeling and is based on a fully implicit time evolution scheme that leads to 3D large very badly conditioned sparse linear systems to be solved at every time step. In this context, the use of PaStiX library to solve efficiently these large sparse problems by a direct method is a challenging issue.

This activity continues within the context of the EoCoE2 project, in which the PaStiX solver is identified to allow the processing of very larger linear systems for the nuclear fusion code TOKAM3X from CEA-IRFM. Contrary to the JOREK code, the problem to be treated corresponds to the complete 3D volume of the plasma torus. The objective is to be competitive, for complex geometries, compared to an Algebraic MultiGrid approach designed by one partner of EoCoE2.

4.2.2. Numerical and parallel scalable hybrid solvers in large scale calculations

Parallel and numerically scalable hybrid solvers based on a fully algebraic coarse space correction have been theoretically studied and various advanced parallel implementations have been designed. Their parallel scalability has been initially investigated on large scale problems within the EoCoE project thanks to a close collaboration with the BSC and the integration of MaPhyS within the Alya software. This activity will further develop in the EoCoE2 project. The performance has also been assessed on PRACE Tier-0 machine within a PRACE Project Access through a collaboration with CERFACS and Laboratoire de Physique des Plasmas at Ecole Polytechnique for the calculation of plasma propulsion. A comparative parallel scalability study with the Algebraic MultiGrid from Petsc has been conducted in that framework.

4.2.3. Aeroacoustics Simulation

This domain is in the context of a long term collaboration with Airbus Research Centers. Wave propagation phenomena intervene in many different aspects of systems design at Airbus. They drive the level of acoustic vibrations that mechanical components have to sustain, a level that one may want to diminish for comfort reason (in the case of aircraft passengers, for instance) or for safety reason (to avoid damage in the case of a payload in a rocket fairing at take-off). Numerical simulations of these phenomena play a central part in the upstream design phase of any such project. Airbus Central R & T has developed over the last decades an in-depth knowledge in the field of Boundary Element Method (BEM) for the simulation of wave propagation in homogeneous media and in frequency domain. To tackle heterogeneous media (such as the jet engine flows, in the case of acoustic simulation), these BEM approaches are coupled with volumic finite elements (FEM). We end up with the need to solve large (several millions unknowns) linear systems of equations composed of a dense part (coming from the BEM domain) and a sparse part (coming from the FEM domain). Various parallel solution techniques are available today, mixing tools created by the academic world (such as the Mumps and Pastix sparse solvers) as well as parallel software tools developed in-house at Airbus (dense solver SPIDO, multipole solver, \mathcal{H} -matrix solver with an open sequential version available online). In the current state of

knowledge and technologies, these methods do not permit to tackle the simulation of aeroacoustics problems at the highest acoustic frequencies (between 5 and 20 kHz, upper limits of human audition) while considering the whole complexity of geometries and phenomena involved (higher acoustic frequency implies smaller mesh sizes that lead to larger unknowns number, a number that grows like f^2 for BEM and f^3 for FEM, where f is the studied frequency). The purpose of the study in this domain is to develop advanced solvers able to tackle this kind of mixed dense/sparse linear systems efficiently on parallel architectures.

4.2.4. Optimization for Deep Convolutional Neural Networks

The training phase of Deep Convolutional Neural Networks represents nowadays a significant share of the computations performed on HPC supercomputers. It introduces several new problems concerning resource allocation and scheduling issues, because of the specific pattern of task graphs induced by the stochastic gradient descent and because memory consumption is particularly critical when performing training. As of today, the most classical parallelization methods consists in partitioning mini-batches, images, filters,... but all these methods induce high synchronization and communication costs, and only very partially resolve memory issues. Within the framework of the Inria IPL on HPC Big Data and Learning convergence, we are working on re-materialization techniques and on the use of model parallelism, in particular to be able to build on the research that has been carried out in a more traditional HPC framework on the exploitation of resource heterogeneity and dynamic runtime scheduling.

KERDATA Project-Team

4. Application Domains

4.1. Application Domains

The KerData team investigates the design and implementation of architectures for data storage and processing across clouds, HPC and edge-based systems, which address the needs of a large spectrum of applications. The use cases we target to validate our research results come from the following domains.

- Climate and meteorology
- Earth science
- Energy and sustainable development
- Smart cities
- Data science

POLARIS Project-Team

4. Application Domains

4.1. Large Computing Infrastructures

Supercomputers typically comprise thousands to millions of multi-core CPUs with GPU accelerators interconnected by complex interconnection networks that are typically structured as an intricate hierarchy of network switches. Capacity planning and management of such systems not only raises challenges in term of computing efficiency but also in term of energy consumption. Most legacy (SPMD) applications struggle to benefit from such infrastructure since the slightest failure or load imbalance immediately causes the whole program to stop or at best to waste resources. To scale and handle the stochastic nature of resources, these applications have to rely on dynamic runtimes that schedule computations and communications in an opportunistic way. Such evolution raises challenges not only in terms of programming but also in terms of observation (complexity and dynamicity prevents experiment reproducibility, intrusiveness hinders large scale data collection, ...) and analysis (dynamic and flexible application structures make classical visualization and simulation techniques totally ineffective and require to build on *ad hoc* information on the application structure).

4.2. Next-Generation Wireless Networks

Considerable interest has arisen from the seminal prediction that the use of multiple-input, multiple-output (MIMO) technologies can lead to substantial gains in information throughput in wireless communications, especially when used at a massive level. In particular, by employing multiple inexpensive service antennas, it is possible to exploit spatial multiplexing in the transmission and reception of radio signals, the only physical limit being the number of antennas that can be deployed on a portable device. As a result, the wireless medium can accommodate greater volumes of data traffic without requiring the reallocation (and subsequent re-regulation) of additional frequency bands. In this context, throughput maximization in the presence of interference by neighboring transmitters leads to games with convex action sets (covariance matrices with trace constraints) and individually concave utility functions (each user's Shannon throughput); developing efficient and distributed optimization protocols for such systems is one of the core objectives of Theme 5.

Another major challenge that occurs here is due to the fact that the efficient physical layer optimization of wireless networks relies on perfect (or close to perfect) channel state information (CSI), on both the uplink and the downlink. Due to the vastly increased computational overhead of this feedback – especially in decentralized, small-cell environments – the ongoing transition to fifth generation (5G) wireless networks is expected to go hand-in-hand with distributed learning and optimization methods that can operate reliably in feedback-starved environments. Accordingly, one of POLARIS' application-driven goals will be to leverage the algorithmic output of Theme 5 into a highly adaptive resource allocation framework for next-generation wireless systems that can effectively "learn in the dark", without requiring crippling amounts of feedback.

4.3. Energy and Transportation

Smart urban transport systems and smart grids are two examples of collective adaptive systems. They consist of a large number of heterogeneous entities with decentralised control and varying degrees of complex autonomous behaviour. We develop an analysis tools to help to reason about such systems. Our work relies on tools from fluid and mean-field approximation to build decentralized algorithms that solve complex optimization problems. We focus on two problems: decentralized control of electric grids and capacity planning in vehicle-sharing systems to improve load balancing.

4.4. Social Computing Systems

Social computing systems are online digital systems that use personal data of their users at their core to deliver personalized services directly to the users. They are omnipresent and include for instance recommendation systems, social networks, online medias, daily apps, etc. Despite their interest and utility for users, these systems pose critical challenges of privacy, security, transparency, and respect of certain ethical constraints such as fairness. Solving these challenges involves a mix of measurement and/or audit to understand and assess issues, and modeling and optimization to propose and calibrate solutions.

ROMA Project-Team

4. Application Domains

4.1. Applications of sparse direct solvers

Sparse direct (e.g., multifrontal solvers that we develop) solvers have a wide range of applications as they are used at the heart of many numerical methods in computational science: whether a model uses finite elements or finite differences, or requires the optimization of a complex linear or nonlinear function, one often ends up solving a system of linear equations involving sparse matrices. There are therefore a number of application fields, among which some of the ones cited by the users of the sparse direct solver MUMPS are: structural mechanics, seismic modeling, biomechanics, medical image processing, tomography, geophysics, electromagnetism, fluid dynamics, econometric models, oil reservoir simulation, magneto-hydro-dynamics, chemistry, acoustics, glaciology, astrophysics, circuit simulation, and work on hybrid direct-iterative methods.

STORM Project-Team

4. Application Domains

4.1. Application domains benefiting from HPC

The application domains of this research are the following:

- Bioinformatics (see ADT Gordon 9.2.2)
- Environment, in particular CO_2 capture (see Exa2PRO, 9.3.1)
- Health and heart disease analysis (see EXACARD, 9.2.1)
- Software infrastructures for Telecommunications (see AFF3CT, 7.9 , 9.2.2)
- Aeronautics (collaboration with Airbus, J.-M. Couteyen)

4.2. Application in High performance computing/Big Data

Most of the research of the team has application in the domain of software infrastructure for HPC and BigData (ANR SOLHAR 9.2.1 , Inria ADT SwLoc and Gordon 9.2.2 , IPL HAC-SPECIS and BigData 9.2.3 , PIA project ELCI 9.2 , H2020 projects INTERTWinE and Exa2Pro 9.3.1 and PRACE project PRACE5IP 9.3.1).

TADAAM Project-Team

4. Application Domains

4.1. Mesh-based applications

TADAAM targets scientific simulation applications on large-scale systems, as these applications present huge challenges in terms of performance, locality, scalability, parallelism and data management. Many of these HPC applications use meshes as the basic model for their computation. For instance, PDE-based simulations using finite differences, finite volumes, or finite elements methods operate on meshes that describe the geometry and the physical properties of the simulated objects. This is the case for at least two thirds of the applications selected in the 9th PRACE. call ⁰, which concern quantum mechanics, fluid mechanics, climate, material physic, electromagnetism, etc.

Mesh-based applications not only represent the majority of HPC applications running on existing supercomputing systems, yet also feature properties that should be taken into account to achieve scalability and performance on future large-scale systems. These properties are the following:

Size Datasets are large: some meshes comprise hundreds of millions of elements, or even billions.

Dynamicity In many simulations, meshes are refined or coarsened at each time step, so as to account for the evolution of the physical simulation (moving parts, shockwaves, structural changes in the model resulting from collisions between mesh parts, etc.).

Structure Many meshes are unstructured, and require advanced data structures so as to manage irregularity in data storage.

Topology Due to their rooting in the physical world, meshes exhibit interesting topological properties (low dimensionality embedding, small maximum degree, large diameter, etc.). It is very important to take advantage of these properties when laying out mesh data on systems where communication locality matters.

All these features make mesh-based applications a very interesting and challenging use-case for the research we want to carry out in this project. Moreover, we believe that our proposed approach and solutions will contribute to enhance these applications and allow them to achieve the best possible usage of the available resources of future high-end systems.

⁰<http://www.prace-ri.eu/prace-9th-regular-call/>

DIVERSE Project-Team (section vide)

EASE Project-Team

4. Application Domains

4.1. Pervasive applications in Smart Building

A Smart Building is a living space equipped with information-and-communication-technology (ICT) devices conceived to collaborate in order to anticipate and respond to the needs of the occupants, working to promote their comfort, convenience, security and entertainment while preserving their natural interaction with the environment.

The idea of using the Pervasive Computing paradigm in the Smart Building domain is not new. However, the state-of-the-art solutions only partially adhere to its principles. Often the adopted approach consists in a heavy deployment of sensor nodes, which continuously send a lot of data to a central elaboration unit, in charge of the difficult task of extrapolating meaningful information using complex techniques. This is a *logical approach*. EASE proposed instead the adoption of a *physical approach*, in which the information is spread in the environment, carried by the entities themselves, and the elaboration is directly executed by these entities "inside" the physical space. This allows performing meaningful exchanges of data that will thereafter need a less complicated processing compared to the current solutions. The result is a smart environment that can, in an easier and better way, integrate the context in its functioning and thus seamlessly deliver more useful and effective user services. Our contribution aims at implementing the physical approach in a smarter environment, showing a solution for improving both comfort and energy savings.

4.2. Automation in Smart City

The domain of Smart Cities is still young but it is already a huge market which attracts number of companies and researchers. It is also multi-fold as the words "smart city" gather multiple meanings. Among them one of the main responsibilities of a city, is to organize the transportation of goods and people. In intelligent transportation systems (ITS), ICT technologies have been involved to improve planification and more generally efficiency of journeys within the city. We are interested in the next step where efficiency would be improved locally relying on local interactions between vehicles, infrastructure and people (smartphones).

For the future autonomous vehicle are now in the spotlight, since a lot of works has been done in recent years in automotive industry as well as in academic research centers. Such unmanned vehicles could strongly impact the organisation of the transportation in our cities. However, due to the lack of a definition of what is an "autonomous" vehicle, it is still difficult to see how these vehicles will interact with their environment (eg. road, smart city, houses, grid, etc.). From augmented perception to fully cooperative automated vehicle, the autonomy covers various realities in terms of interaction the vehicle relies on. The extended perception relies on communication between the vehicle and surrounding roadside equipments. That helps the driving system to build and maintain an accurate view of the environment. But at this first stage the vehicle only uses its own perception to make its decisions. At a second stage, it will take advantages of local interaction with other vehicles through car-to-car communications to elaborate a better view of its environment. Such "cooperative autonomy" does not try to reproduce the human behavior anymore, it strongly relies on communication between vehicles and/or with the infrastructure to make decision and to acquire information on the environment. Part of the decision could be centralized (almost everything for an automatic metro) or coordinated by a roadside component. The decision making could even be fully distributed but this put high constraints on the communications. Automated vehicles are just an example of smart city automated processes that will have to share information within the surrounding to make their decisions.

4.3. Pervasive applications in uncontrolled environments

Some limitations of existing RFID technology become challenging: unlike standard RFID application scenarios, pervasive computing often involves uncontrolled environment for RFID, where tags and reader have to operate in much more difficult situations that those usually encountered or expected for classical RFID systems.

RFID technology is to avoid missing tags when reading multiple objects, as reading reliability is affected by various effects such as shadowing or wave power absorption by some materials. The usual applications of RFID operate in a controlled environment in order to reduce the risk of missing tags while scanning objects.

In pervasive computing applications, a controlled reading environment is extremely difficult to achieve, as one of the principles is to enhance existing processes "in situ", unlike the controlled conditions that can be found in industrial processes. Consider for example a logistic application, where RFID tags could be used on items inside a package in order to check for its integrity along the shipping process. Tags would likely be placed randomly on items inside the package, and reading conditions would be variable depending on where the package is checked.

RFID operation in uncontrolled environments is challenging because RFID performance is affected by multiple parameters, in particular:

- Objects materials (on which tags are attached to),
- Materials in the surrounding environment,
- RFID frequency spectrum,
- Antenna nature and placement with respect to the tags.

In a controlled environment, the difficulty to read tags can be limited by using the appropriate parameters to maximize the RFID performance for the application. But in many cases, it is needed to read a large number of objects of various nature, arranged randomly in a given area or container. **Most pervasive computing applications fall in this context.**

FOCUS Project-Team

4. Application Domains

4.1. Ubiquitous Systems

The main application domain for Focus are ubiquitous systems, broadly systems whose distinctive features are: mobility, high dynamicity, heterogeneity, variable availability (the availability of services offered by the constituent parts of a system may fluctuate, and similarly the guarantees offered by single components may not be the same all the time), open-endedness, complexity (the systems are made by a large number of components, with sophisticated architectural structures). In Focus we are particularly interested in the following aspects.

- *Linguistic primitives* for programming dialogues among components.
- *Contracts* expressing the functionalities offered by components.
- *Adaptability and evolvability* of the behaviour of components.
- *Verification* of properties of component systems.
- Bounds on component *resource consumption* (e.g., time and space consumed).

4.2. Service Oriented Computing and Cloud Computing

Today the component-based methodology often refers to Service Oriented Computing. This is a specialized form of component-based approach. According to W3C, a service-oriented architecture is “a set of components which can be invoked, and whose interface descriptions can be published and discovered”. In the early days of Service Oriented Computing, the term services was strictly related to that of Web Services. Nowadays, it has a much broader meaning as exemplified by the XaaS (everything as a service) paradigm: based on modern virtualization technologies, Cloud computing offers the possibility to build sophisticated service systems on virtualized infrastructures accessible from everywhere and from any kind of computing device. Such infrastructures are usually examples of sophisticated service oriented architectures that, differently from traditional service systems, should also be capable to elastically adapt on demand to the user requests.

INDES Project-Team

4. Application Domains

4.1. Web

The Web is the natural application domain of the team. We are designing and implementing multitier languages for helping the development of Web applications. We are creating static and dynamic analyses for Web security. We are conducting empirical studies about privacy preservation on the Web.

4.2. Internet of Things

More recently, we have started focusing on *Internet of Things* (IoT) applications. They share many similarities with Web applications so most of the methodologies and expertises we have developed for the Web apply to IoT but the restricted hardware resources made available by many IoT devices demand new developments and new research explorations.

RMOD Project-Team

4. Application Domains

4.1. Programming Languages and Tools

Many of the results of RMoD are improving programming languages or development tools for such languages. As such the application domain of these results is as varied as the use of programming languages in general. Pharo, the language that RMoD develops, is used for a very broad range of applications. From pure research experiments to real world industrial use (the **Pharo Consortium** has more than 25 company members).

Examples are web applications, server backends for mobile applications or even graphical tools and embedded applications

4.2. Software Reengineering

Moose is a language-independent environment for reverse and re-engineering complex software systems. Moose provides a set of services including a common meta-model, metrics evaluation and visualization. As such Moose is used for analyzing software systems to support understanding and continuous development as well as software quality analysis.

AGORA Project-Team

4. Application Domains

4.1. Smart Cities

One major characteristic of modern societies is that they are prevalently urban. Consequently, the contributions of the Agora team are in particular applied to provide solutions tailored to the emergence of the Internet of Things (IoT) and to Smart Cities applications. A major motivation of the team is the forthcoming explosion of the number of connected devices. In particular, low cost - small data devices are supposed to be densely deployed in our environment, fostering the interest for a convergence of the traditional wireless networking paradigms.

Smart City is a constantly reshaped concept, embracing the future of dense metropolitan areas, with references to efficient and sustainable infrastructure, improving citizens' quality of life and protecting the environment. A consensus on the Smart City philosophy is however that it will be primarily achieved by leveraging a clever integration of Information and Communication Technologies (ICT) in the urban tissue. Indeed, ICTs are enabling an evolution from the current duality between the real world and its digitized counterpart to a continuum in which digital contents and applications are seamlessly interacting with classical infrastructures and services. Smart Cities are often described by the digital services that should be provided which are inherently dependent on dense measurements of the city environment and activities, the collection of these data, their processing into information, and their redistribution. The networking infrastructure plays therefore a critical role in enabling advanced services, in particular the wireless infrastructure supporting density and mobility.

From a wireless networking viewpoint, the digitization of cities can be seen as a paradigm shift extending the Internet of Things (IoT) to a citizen-centric model in order to leverage the massive data collected by pervasive sensors, connected mobiles or fixed devices, and social applications.

COATI Project-Team

4. Application Domains

4.1. Telecommunication Networks

COATI is mostly interested in telecommunications networks but also in the network structure appearing in social, molecular and transportation networks.

We focus on the design and management of heterogeneous physical and logical networks. The project has kept working on the design of backbone networks (optical networks, radio networks, IP networks). However, the fields of Software Defined Networks and Network Function Virtualization are growing in importance in our studies. In all these networks, we study routing algorithms and the evolution of the routing in case of any kind of topological modifications (maintenance operations, failures, capacity variations, etc.).

4.2. Other Domains

Our combinatorial tools may be well applied to solve many other problems in various areas (transport, biology, resource allocation, chemistry, smart-grids, speleology, etc.) and we collaborate with experts of some of these domains.

For instance, we collaborate with project-team ABS (Algorithms Biology Structure) from Sophia Antipolis on problems from Structural Biology (co-supervision of a PhD student). In the area of transportation networks, we collaborate with SMEs Benomad and Instant-System on dynamic car-pooling combined with multi-modal transportation systems in the context of ANR project Multimod started in January 2018. Last, we have started a collaboration with GREDEG (Groupe de Recherche en Droit, Economie et Gestion, Univ. Nice Sophia Antipolis) on the analysis of collaboration networks.

DANTE Project-Team

4. Application Domains

4.1. Life Science & Health

In parallel to the advances in modern medicine, health sciences and public health policy, epidemic models aided by computer simulations and information technologies offer an increasingly important tool for the understanding of transmission dynamics and of epidemic patterns. The increased computational power and use of Information and Communication Technologies make feasible sophisticated modelling approaches augmented by detailed in vivo data sets, and allow to study a variety of possible scenarios and control strategies, helping and supporting the decision process at the scientific, medical and public health level. The research conducted in the DANTE project finds direct applications in the domain of LSH since modelling approaches crucially depend on our ability to describe the interactions of individuals in the population.

Within PhD work of G. Frusque, we collaborate with Dr. Julien Jung from Hôpital de Neurologie de Bron (HCL) and with Nadine Ravel, DR CNRS (CRNL, INSERM).

4.2. Network Science / Complex networks

In the last ten years the science of complex networks has been assigned an increasingly relevant role in defining a conceptual framework for the analysis of complex systems. Network science is concerned with graphs that map entities and their interactions to nodes and links. For a long time, this mathematical abstraction has contributed to the understanding of real-world systems in physics, computer science, biology, chemistry, social sciences, and economics. Recently, however, enormous amounts of detailed data, electronically collected and meticulously catalogued, have finally become available for scientific analysis and study. This has led to the discovery that most networks describing real world systems show the presence of complex properties and heterogeneities, which cannot be neglected in their topological and dynamical description. This has called forth a major effort in developing the methodology to characterise the topology and temporal behaviour of complex networks, to describe the observed structural and temporal heterogeneities, to detect and measure emerging community structure, to see how the functionality of networks determines their evolving structure, and to determine what kinds of correlations play a role in their dynamics. All these efforts have brought us to a point where the science of complex networks has become advanced enough to help us to disclose the deeper roles of complexity and gain understanding about the behaviour of very complicated systems.

In this endeavour the DANTE project targets the study of dynamically evolving networks, concentrating on questions about the evolving structure and dynamical processes taking place on them. During the last year we developed several projects along these lines concerning three major datasets:

- **Mobile telephony data:** In projects with academic partners and Grandata we performed projects based on two large independent datasets collecting the telephone call and SMS event records for million of anonymised individuals. The datasets record the time and duration of mobile phone interactions and some coarse grained location and demographic data for some users. In addition one of the dataset is coupled with anonymised bank credit information allowing us to study directly the socioeconomic structure of a society and how it determines the communication dynamics and structure of individuals.
- **Skype data:** Together with Skype Labs/STACC and other academic groups we were leading projects in the subject of social spreading phenomena. These projects were based on observations taken from a temporally detailed description of the evolving social network of (anonymised) Skype users registered between 2003 and 2011. This data contains dates of registration and link creation together with gradual information about their location and service usage dynamics.

- Twitter data: In collaboration with ICAR-ENS Lyon we collected a large dataset about the microblogs and communications of millions of Twitter users in the French Twitter space. This data allows us to follow the spreading of fads/opinions/hashtags/ideas and more importantly linguistic features in online communities. The aim of this collaboration is to set the ground for a quantitative framework studying the evolution of linguistic features and dialects in an social-communication space mediated by online social interactions.

4.3. Social Sciences / Epistemology

Political impacts of the internet and of networks begin to be well known (Cambridge Analytica, Russian trolls, etc.). Hence the public at large begins to be aware of the abuses of the leaders of the internet (privacy by firms and advertising, surveillance by states, fake news by activists, etc.). In the same time, on-line exchanges now give scientific estimations of the political life [60] and political sciences begin to consider the internet as a relevant subject of study. As the internet is a *technology*, philosophy is the best approach to understand what socially happens (or can be made) with the internet. We develop it by two ways:

- political philosophy. Some Dante members are working with Triangle laboratory (social sciences, philosophy and politics; Ens de Lyon and CNRS).
- epistemology, because computer sciences discoveries are related with the evolution of science(s): we discovered that, in the case of the internet, political philosophy can do few if not strongly related to epistemology [61]. Epistemological approach is developed in collaboration with Jean Dhombres (who holds a seminar at Enssib: <http://barthes.enssib.fr/cours/Dhombres2018-2019.html> and with Hcéres (new relations between social and exact sciences, cf. point 9.2.8).

This approach should help computer scientists to understand how their research may depend on foreign initiatives and to create new links between social sciences and Inria.

DIANA Project-Team (section vide)

DIONYSOS Project-Team

4. Application Domains

4.1. Networking

Our global research effort concerns networking problems, both from the analysis point of view, and around network design issues. Specifically, this means the IP technology in general, with focus on specific types of networks seen at different levels: wireless systems, optical infrastructures, peer-to-peer architectures, Software Defined Networks, Content Delivery Networks, Content-Centric Networks, clouds.

A specific aspect of network applications and/or services based on video or voice content, is our PSQA technology, able to measure the Perceptual Quality automatically and in real time. PSQA provides a MOS value as close as it makes sense to the value obtained from subjective testing sessions. The technology has been tested in many environments, including one way communications as, for instance, in video streaming, and bi-directional communications as in IP telephony, UDP- or TCP-based systems, etc. It has already served in many collaborative projects as the measuring tool used.

4.2. Stochastic modeling

Many of the techniques developed at Dionysos are related to the analysis of complex systems in general, not only in telecommunications. For instance, our Monte Carlo methods for analyzing rare events have been used by different industrial partners, some of them in networking but recently also by companies building transportation systems. We develop methods in different areas: numerical analysis of stochastic models, bound computations in the same area, Discrete Event Simulation, or, as just mentioned, rare event analysis.

DYOGENE Project-Team

4. Application Domains

4.1. Physical communication networks

Internet, wireless, mobile, cellular networks, transportation networks, distributed systems (cloud, call centers).
In collaboration with Nokia Bell Labs and Orange Labs.

4.2. Abstract networks

Social interactions, human communities, economic networks.

4.3. Power grids

Energy networks. In collaboration with EDF.

EVA Project-Team

4. Application Domains

4.1. Industrial Process Automation

Wireless networks have become ubiquitous and are an integral part of our daily lives. These networks are present in many application domains; the most important are detailed in this section.

Networks in industrial process automation typically perform **monitoring and control** tasks. Wired industrial communication networks, such as HART⁰, have been around for decades and, being wired, are highly reliable. Network administrators tempted to “go wireless” expect the same reliability. Reliable process automation networks – especially when used for control – often impose stringent latency requirements. Deterministic wireless networks can be used in critical systems such as control loops, however, the unreliable nature of the wireless medium, coupled with their large scale and “ad-hoc” nature raise some of the most important challenges for low-power wireless research over the next 5-10 years.

Through the involvement of team members in standardization activities, protocols and techniques are proposed for the standardization process with a view to becoming the *de-facto* standard for wireless industrial process automation. Besides producing top level research publications and standardization activities, EVA intends this activity to foster further collaborations with industrial partners.

4.2. Environmental Monitoring

Today, outdoor WSNs are used to monitor vast rural or semi-rural areas and may be used to detect fires. Another example is detecting fires in outdoor fuel depots, where the delivery of alarm messages to a monitoring station in an upper-bounded time is of prime importance. Other applications consist in monitoring the snow melting process in mountains, tracking the quality of water in cities, registering the height of water in pipes to foresee flooding, etc. These applications lead to a vast number of technical issues: deployment strategies to ensure suitable coverage and good network connectivity, energy efficiency, reliability and latency, etc.

We work on such applications in an associate team “REALMS” comprising members from EVA, the university of Berkeley and the university of Michigan.

4.3. The Internet of Things

The general agreement is that the Internet of Things (IoT) is composed of small, often battery-powered objects which measure and interact with the physical world, and encompasses smart home applications, wearables, smart city and smart plant applications.

It is absolutely essential to (1) clearly understand the limits and capabilities of the IoT, and (2) develop technologies which enable user expectation to be met.

The EVA team is dedicated to understanding and contributing to the IoT. In particular, the team maintains a good understanding of the different technologies at play (Bluetooth, IEEE 802.15.4, WiFi, cellular), and their trade-offs. Through scientific publications and other contributions, EVA helps establish which technology best fits which application.

⁰Highway Addressable Remote Transducer

4.4. Military, Energy and Aerospace

Through the HIPERCOM project, EVA has developed cutting-edge expertise in using wireless networks for military, energy and aerospace applications. Wireless networks are a key enabling technology in the application domains, as they allow physical processes to be instrumented (e.g. the structural health of an airplane) at a granularity not achievable by its wired counterpart. Using wireless technology in these domains does however raise many technical challenges, including end-to-end latency, energy-efficiency, reliability and Quality of Service (QoS). Mobility is often an additional constraint in energy and military applications. Achieving scalability is of paramount importance for tactical military networks, and, albeit to a lesser degree, for power plants. EVA will work in this domain.

Smart cities share the constraint of mobility (both pedestrian and vehicular) with tactical military networks. Vehicular Ad-hoc NETWORKS (VANETs) will play an important role in the development of smarter cities.

The coexistence of different networks operating in the same radio spectrum can cause interference that should be avoided. Cognitive radio provides secondary users with the frequency channels that are temporarily unused (or unassigned) by primary users. Such opportunistic behavior can also be applied to urban wireless sensor networks. Smart cities raise the problem of transmitting, gathering, processing and storing big data. Another issue is to provide the right information at the place where it is most needed.

4.5. Emergency Applications

In an “emergency” application, heterogeneous nodes of a wireless network cooperate to recover from a disruptive event in a timely fashion, thereby possibly saving human lives. These wireless networks can be rapidly deployed and are useful to assess damage and take initial decisions. Their primary goal is to maintain connectivity with the humans or mobile robots (possibly in a hostile environment) in charge of network deployment. The deployment should ensure the coverage of particular points or areas of interest. The wireless network has to cope with pedestrian mobility and robot/vehicle mobility. The environment, initially unknown, is progressively discovered and may contain numerous obstacles that should be avoided. The nodes of the wireless network are usually battery-powered. Since they are placed by a robot or a human, their weight is very limited. The protocols supported by these nodes should be energy-efficient to maximize network lifetime. In such a challenging environment, sensor nodes should be replaced before their batteries are depleted. It is therefore important to be able to accurately determine the battery lifetime of these nodes, enabling predictive maintenance.

4.6. Types of Wireless Networks

The EVA team will distinguish between opportunistic communication (which takes advantage of a favorable state) and collaborative communication (several entities collaborate to reach a common objective). Furthermore, determinism can be required to schedule medium access and node activity, and to predict energy consumption.

In the EVA project, we will propose **self-adaptive wireless networks** whose evolution is based on:

- optimization to minimize a single or multiple objective functions under some constraints (e.g. interference, or energy consumption in the routing process).
- machine learning to be able to predict a future state based on past states (e.g. link quality in a wireless sensor network) and to identify tendencies.

The types of wireless networks encountered in the application domains can be classified in the following categories.

4.6.1. Wireless Sensor and Mesh Networks

Standardization activities at the IETF have defined an “upper stack” allowing low-power mesh networks to be seamlessly integrated in the Internet (6LoWPAN), form multi-hop topologies (RPL), and interact with other devices like regular web servers (CoAP).

Major research challenges in sensor networks are mostly related to (predictable) power conservation and efficient multi-hop routing. Applications such as monitoring of mobile targets, and the generalization of smart phone devices and wearables, have introduced the need for WSN communication protocols to cope with node mobility and intermittent connectivity.

Extending WSN technology to new application spaces (e.g. security, sports, hostile environments) could also assist communication by seamless exchanges of information between individuals, between individuals and machines, or between machines, leading to the Internet of Things.

4.6.2. Deterministic Low-Power Networks

Wired sensor networks have been used for decades to automate production processes in industrial applications, through standards such as HART. Because of the unreliable nature of the wireless medium, a wireless version of such industrial networks was long considered infeasible.

In 2012, the publication of the IEEE 802.15.4e standard triggered a revolutionary trend in low-power mesh networking: merging the performance of industrial networks, with the ease-of-integration of IP-enabled networks. This integration process is spearheaded by the IETF 6TiSCH working group, created in 2013. A 6TiSCH network implements the IEEE 802.15.4e TSCH protocol, as well as IETF standards such as 6LoWPAN, RPL and CoAP. A 6TiSCH network is synchronized, and a communication schedule orchestrates all communication in the network. Deployments of pre-6TiSCH networks have shown that they can achieve over 99.999% end-to-end reliability, and a decade of battery lifetime.

The communication schedule of a 6TiSCH network can be built and maintained using a centralized, distributed, or hybrid scheduling approach. While the mechanisms for managing that schedule are being standardized by the IETF, which scheduling approach to use, and the associated limits in terms of reliability, throughput and power consumption remain entirely open research questions. Contributing to answering these questions is an important research direction for the EVA team.

4.6.3. MANETs and VANETs

In contrast to routing, other domains in MANETs such as medium access, multi-carrier transmission, quality of service, and quality of experience have received less attention. The establishment of research contracts for EVA in the field of MANETs is expected to remain substantial. MANETs will remain a key application domain for EVA with users such as the military, firefighters, emergency services and NGOs.

Vehicular Ad hoc Networks (VANETs) are arguably one of the most promising applications for MANETs. These networks primarily aim at improving road safety. Radio spectrum has been ring-fenced for VANETs worldwide, especially for safety applications. International standardization bodies are working on building efficient standards to govern vehicle-to-vehicle or vehicle-to-infrastructure communication.

4.6.4. Cellular and Device-to-Device Networks

We propose to initially focus this activity on spectrum sensing. For efficient spectrum sensing, the first step is to discover the links (sub-carriers) on which nodes may initiate communications. In Device-to-Device (D2D) networks, one difficulty is scalability.

For link sensing, we will study and design new random access schemes for D2D networks, starting from active signaling. This will assume the availability of a control channel devoted to D2D neighbor discovery. It is therefore naturally coupled with cognitive radio algorithms (allocating such resources): coordination of link discovery through eNode-B information exchanges can yield further spectrum usage optimization.

FUN Project-Team (section vide)

GANG Project-Team

4. Application Domains

4.1. Large scale networks

Application domains include evaluating Internet performances, the design of new peer-to-peer applications, enabling large scale networks, and developing tools for transportation networks.

GANG Project-Team

4. Application Domains

4.1. Large scale networks

Application domains include evaluating Internet performances, the design of new peer-to-peer applications, enabling large scale networks, and developing tools for transportation networks.

MARACAS Team

4. Application Domains

4.1. 5G, 6G, and beyond

The fifth generation (5G) broadens the usage of cellular networks but requires new features, typically very high rates, high reliability, ultra low latency, for immersive applications, tactile internet, M2M communications.

From the technical side, new elements such as millimeter waves, massive MIMO, massive access are under evaluation. The initial 5G standard validated in 2019, is finally not really disruptive with respect to the 4G and the clear breakthrough is not there yet. The ideal network architecture for billions of devices in the general context of Internet of Things, is not well established and the debate still exists between several proposals such as NB-IoT, Sigfox, Lora. We are developing a deep understanding of these techniques, in collaboration with major actors (Orange Labs, Nokia Bell Labs, Sequans, Sigfox) and we want to be able to evaluate, to compare and to propose evolutions of these standards with an independent point of view.

This is why we are interested in developing partnerships with major industries, access providers but also with service providers to position our research in a joint optimization of the network infrastructure and the data services, from a theoretical perspective as well as from experimentation.

4.2. Energy sustainability

The energy footprint and from a more general perspective, the sustainability of wireless cellular networks and wireless connectivity is somehow questionable.

We develop our models and analysis with a careful consideration of the energy footprint : sleeping modes, power adaptation, interference reduction, energy gathering, ... many techniques can be optimized to reduce the energetic impact of wireless connectivity. In a *computing networks* approach, considering simultaneously transmission, storage and computation constraints may help to reduce drastically the overall energy footprint.

4.3. Smart building, smart cities, smart environments

Smart environments rely on the deployment of many sensors and actuators allowing to create interactions between the twinned virtual and real worlds. These smart environments (e.g. smart building) are for us an ideal playground to develop new models based on information theory and estimation theory to optimize the network architecture including storage, transmission, computation at the right place.

Our work can be seen as the dark side of cloud/edge computing. In collaboration with other teams expert in distributed computing or middleware (typically at CITIlab, with the team Dynamid of Frédéric Le Mouel) and in the framework of the chaire SPIE/ICS-INSA Lyon, we want to optimize the mechanisms associated to these technologies : in a multi-constrained approach, we want to design new distributed algorithms appropriate for large scale smart environments.

4.4. Machine learning based radio

During the first 6G wireless meeting which was held in Lapland, Finland in March 2019, machine learning (ML) was clearly identified as one of the most promising breakthroughs for future 6G wireless systems expected to be in use around 2030 (<https://www.6gsummit.com/>). The research community is entirely leveraging the international ML tsunami. We strongly believe that the paradigm of wireless networks is moving toward to a new era. Our view is supported by the fact that artificial Intelligence (AI) in wireless communications is not new at all. The telecommunications industry has been seeking for 20 years to reduce the operational complexity of communication networks in order to simplify constraints and to reduce costs on deployments. This obviously relies on data-driven techniques allowing the network to self-tune its own parameters. Over the successive 3GPP standard releases, more and more sophisticated network control has been introduced. This has supported increasing flexibility and further self-optimization capabilities for radio resource management (RRM) as well as for network parameters optimization.

We target the following key elements :

- Obtaining data from experimental scenarios, at the lowest level (baseband I/Q signals) in multi-user scenarios (based upon FIT/CorteXlab).
- Developing a framework and algorithms for deep learning based radio.
- Developing new reinforcement learning techniques in high dimensional state-action spaces.
- Embedding NN structures on radio devices (FPGA or m-controllers) and in FIT/CorteXlab.
- Evaluating the gap between these algorithms and fundamental limits from information theory.
- Building an application scenario in a smart environment to experiment a fully cross-layer design (e.g. within a smart-building context, how could a set of object could learn their protocols efficiently ?)

4.5. Molecular communications

Many communication mechanisms are based on acoustic or electromagnetic propagation; however, the general theory of communication is much more widely applicable. One recent proposal is molecular communication, where information is encoded in the type, quantity, or time or release of molecules. This perspective has interesting implications for the understanding of biochemical processes and also chemical-based communication where other signaling schemes are not easy to use (e.g., in mines). Our work in this area focuses on two aspects: (i) the fundamental limits of communication (i.e., how much data can be transmitted within a given period of time); and (ii) signal processing strategies which can be implemented by circuits built from chemical reaction-diffusion systems.

A novel perspective introduced within our work is the incorporation of coexistence constraints. That is, we consider molecular communication in a crowded biochemical environment where communication should not impact pre-existing behavior of the environment. This has lead to new connections with communication subject to security constraints as well as the stability theory of stochastic chemical reaction-diffusion systems and systems of partial differential equations which provide deterministic approximations.

NEO Project-Team

4. Application Domains

4.1. Network Science

Network Science is a multidisciplinary body of knowledge, principally concerned with the emergence of global properties in a network of individual agents, from the “local” properties of this network, namely, the way agents interact with each other. The central model of “networks” is the graph (of Graph Theory/Operations Research), with nodes representing the different entities managing information and taking decisions, and the links representing the fact that entities interact, or not. Links are usually equipped with a “weight” that measures the intensity of interaction. Adding evolution rules to this quite elementary representation leads to dynamic network models, the properties of which Network Science tries to analyze.

A classical example of properties sought in networks is the famous “six degrees of separation” (or “small world”) property: how and why does it happen so frequently? Another ubiquitous property of real-life networks is the Zipf or “scale-free” distribution for degrees. Some of these properties, when properly exploited, lead to successful business opportunities: just consider the PageRank algorithm of Google, which miraculously connects the relevance of some Web information with the relevance of the other information that points to it.

4.2. Network Engineering

In its primary acceptance, Network Science involves little or no engineering: phenomena are assumed to be “natural” and emerge without intervention. However, the idea comes fast to intervene in order to modify the outcome of the phenomenon. This is where NEO is positioned. Beyond the mostly descriptive approach of Network Science, we aim at using the techniques of Operations Research so as to engineer complex networks.

To quote just two examples: controlling the spread of diseases through a “network” of people is of primarily interest for mankind. Similarly, controlling the spread of information or reputation through a social network is of great interest in the Internet. Precisely: given the impact of web visibility on business income, it is tempting (and quite common) to manipulate the graph of the web by adding links so as to drive the PageRank algorithm to a desired outcome.

Another interesting example is the engineering of community structures. Recently, thousands of papers have been written on the topic of community *detection* problem. In most of the works, the researchers propose methods, most of the time, heuristics, for detecting communities or dense subgraphs inside a large network. Much less effort has been put in the understanding of community formation process and even much less effort has been dedicated to the question of how one can influence the process of community formation, e.g. in order to increase overlap among communities and reverse the fragmentation of the society.

Our ambition for the medium term is to reach an understanding of the behavior of complex networks that will make us capable of influencing or producing a certain property in said network. For this purpose, we will develop families of models to capture the essential structure, dynamics, and uncertainty of complex networks. The “solution” of these models will provide the correspondence between metrics of interest and model parameters, thus opening the way to the synthesis of effective control techniques.

In the process of tackling real, very large size networks, we increasingly deal with large graph data analysis and the development of decision techniques with low algorithmic complexity, apt at providing answers from large datasets in reasonable time.

RESIST Team

4. Application Domains

4.1. Internet

Among the different network types, the Internet is the one to link them all and is consequently our most prominent subject, not to mention its prime importance in today's society. The Internet also exhibits its own challenges due to the scale and diversity of stakeholders, applications and network technologies in use.

From a security perspective, **monitoring and analysing Internet traffic is an important part of threat prevention and predictive security**. Indeed, large network telescopes like the one we use in the High Security Laboratory⁰ allow detecting world-wide campaigns of attacks which target a specific exploit in some applications. Moreover the monitoring of the Internet traffic at the **edge** is the best way to quickly detect distributed attacks like DDoS and to mitigate them before they become effective. However, the Internet traffic analysis is made much more complicated since the **massive shift towards encryption** that happened few years ago, which requires new traffic classification methods.

The performance and resilience of services running over the Internet is also a major topic of Resist. In particular, it is very difficult to **diagnose the cause of a degradation of performance among the different actors and technologies** that are used to deliver a service over the Internet (access medium, ISP, CDN, web-browser, etc.). Networked systems deployed at Internet scale are also a natural research subject for Resist. Indeed **decentralized systems** like P2P networks or blockchains are known to be robust and scalable. However, their security and performance have to be carefully assessed because a single flaw in their design can endanger the whole system.

4.2. SDN and Data-Center Networks

As the SDN paradigm and its implementations bring new opportunities that can be leveraged in different contexts, in particular for security and performance, programmable networks are also part of the research scope of Resist. This includes data-plane **programming models and hardware offloading** that enable very flexible programming at the network level. While OpenFlow was initially designed for academic research, SDN in general has then been adopted by industrial players, above all in **data-center networks**. It supports innovations to better share load and optimize resources among processes, in particular for virtualization platforms. Contributing to the development of these technologies is primordial for us as they are key elements for monitoring and enhancing the performance and security of future data-center networks.

When defining or extending SDN technologies, the strongest constraint is to guarantee a satisfactory level of performance, i.e. enabling high flexibility in programming with a **reduced footprint of network throughput**. However, as it may also break isolation principles between multiple tenants, security has to carefully considered, either by adding safeguard mechanisms at run-time or through a priori verification and testing.

4.3. Fog and Cloud computing

Cloud computing has largely evolved in the last years including new networking capabilities as highlighted in the previous section towards the model of XaaS or **everything-as-a-service**. Moreover, cloud computing continues to be more distributed and aims at integrating more heterogeneous resources. One particular example is **fog computing** that consists of a massively distributed number of different resources, including low-performance ones. Large network operators have a great interest in fog computing because they already operate such an infrastructure (e.g. a national operator with regional clouds and setup boxes in end users' homes). Softwarization or virtualization of all functions and services will help them to be competitive by reducing their costs. In general, intelligent orchestration of massively distributed resources will be investigated in various application domains, including **federated cloud infrastructures, fog computing, 5G networks, IoT and big data infrastructures**.

⁰<https://lhs.loria.fr>

The manageability of such largely distributed systems is a core topic with questions related to monitoring, security and orchestration of resources. Major changes and errors can have dramatic effects on a real system, that actually lead to only minor changes being carried out and slow down innovation and adoption of new propositions. Hence, **controlled and reproducible experiments are vital**.

As shown by our past work, we are able to quickly adjust to experimental needs in most areas of distributed computing and networking, such as *High Performance Computing (HPC)*, *Big Data*, *Peer-to-peer systems*, *Grid computing*, etc. However, in the context of Resist, **we will focus mainly on Software-Defined Infrastructures**, gathering *cloud computing* for compute and storage resources, *software-defined networking* and *network function virtualization* for networking. Those infrastructures share many common features: need for performance, for scalability, for resilience, all implemented using flexible software components.

Worth mentioning here is our involvement in the international testbed community (FIRE, GENI). We plan to strengthen our existing links with the Chameleon and CloudLab US projects, to leverage the recently accepted Fed4FIRE+ project on a testbed federation, and, at the national level, to contribute to the SILECS initiative for a new large-scale experimental computer science infrastructure.

4.4. Cyber-Physical Systems

Cyber-Physical Systems (CPSs) used to be well isolated and so designed accordingly. In the last decade, they have become **integrated within larger systems** and so accessible through the Internet. This is the case with **industrial systems**, like SCADA, that have been unfortunately exposed to major threats. Furthermore, the **Internet-of-Things (IoT)** has become a reality with numerous protocols, platforms and devices being developed and used to support the growing deployment of smart* services: smart home, transport, health, city... and even rather usual rigid systems such as industry 4.0.

From an academic perspective, the IoT can be seen as an evolution of sensor networks. It thus inherits from the same problems regarding security and scalability, but with a higher order of magnitude both in terms of number of devices and their capabilities, which can be exploited by attackers. Research in this area has focused on developing dedicated protocols or operating systems to guarantee security and performance, Resist aims to tackle identical problems but **assuming a more practical deployment of IoT systems composed of heterogeneous and uncontrolled devices**. Indeed, this ecosystem is very rich and **cannot be controlled by a unique entity**, e.g. services are often developed by third parties, manufacturers of embed devices are different from those providing connectivity.

As a result, managing an IoT system (monitoring, changing configuration, etc.) is very hard to achieve as most of the devices or applications cannot be directly controlled. For instance, many IoT providers rely on their own cloud services, with their own unknown **proprietary protocols** and most of the time through **encrypted channels**. Above all, the use of middle-boxes like gateways hides the IoT end-devices and applications. We will thus need to infer knowledge from **indirect and partial observations**. Likewise, control will be also indirect for example through filtering or altering communications.

SOCRATE Project-Team (section vide)

TRIBE Project-Team (section vide)