



RESEARCH CENTER
Rennes - Bretagne-Atlantique

FIELD

Activity Report 2017

Section Scientific Foundations

Edition: 2018-02-19

1. ASAP Project-Team	4
2. ASCOLA Project-Team	6
3. ASPI Team	11
4. CAIRN Project-Team	16
5. CELTIQUE Project-Team (section vide)	20
6. CIDRE Project-Team	21
7. DIONYSOS Project-Team	24
8. DIVERSE Project-Team	26
9. DYLISS Project-Team	36
10. FLUMINANCE Project-Team	43
11. GENSCALE Project-Team	46
12. HYBRID Project-Team	48
13. HYCOMES Project-Team	51
14. I4S Project-Team	55
15. IPSO Project-Team	69
16. KERDATA Project-Team	75
17. LACODAM Project-Team	78
18. LAGADIC Project-Team	83
19. LINKMEDIA Project-Team	86
20. MIMETIC Project-Team	88
21. MYRIADS Project-Team	91
22. PACAP Project-Team	96
23. PANAMA Project-Team	104
24. SERPICO Project-Team	107
25. SIROCCO Project-Team	109
26. SUMO Project-Team	112
27. TACOMA Team	114
28. TAMIS Team	116
29. TEA Project-Team	119
30. VISAGES Project-Team	123

ASAP Project-Team

3. Research Program

3.1. Theory of distributed systems

Finding models for distributed computations prone to asynchrony and failures has received a lot of attention. A lot of research in this domain focuses on what can be computed in such models, and, when a problem can be solved, what are its best solutions in terms of relevant cost criteria. An important part of that research is focused on distributed computability: what can be computed when failure detectors are combined with conditions on process input values for example. Another part is devoted to model equivalence. What can be computed with a given class of failure detectors? Which synchronization primitives is a given failure class equivalent to? These are among the main topics addressed in the leading distributed computing community. A second fundamental issue related to distributed models is the definition of appropriate models suited to dynamic systems. Up to now, the researchers in that area consider that nodes can enter and leave the system, but do not provide a simple characterization, based on properties of computation instead of description of possible behaviors [46], [40], [41]. This shows that finding dynamic distributed computing models is today a "Holy Grail", whose discovery would allow a better understanding of the essential nature of dynamic systems.

3.2. Peer-to-peer overlay networks

A standard distributed system today is related to thousands or even millions of computing entities scattered all over the world and dealing with a huge amount of data. This major shift in scalability requirements has led to the emergence of novel computing paradigms. In particular, the peer-to-peer communication paradigm imposed itself as the prevalent model to cope with the requirements of large scale distributed systems. Peer-to-peer systems rely on a symmetric communication model where peers are potentially both clients and servers. They are fully decentralized, thus avoiding the bottleneck imposed by the presence of servers in traditional systems. They are highly resilient to peers arrivals and departures. Finally, individual peer behavior is based on a local knowledge of the system and yet the system converges toward global properties.

A peer-to-peer overlay network logically connects peers on top of IP. Two main classes of such overlays dominate, structured and unstructured. The differences relate to the choice of the neighbors in the overlay, and the presence of an underlying naming structure. Overlay networks represent the main approach to build large-scale distributed systems that we retained. An overlay network forms a logical structure connecting participating entities on top of the physical network, be it IP or a wireless network. Such an overlay might form a structured overlay network [47], [48], [49] following a specific topology or an unstructured network [45], [50] where participating entities are connected in a random or pseudo-random fashion. In between, lie weakly structured peer-to-peer overlays where nodes are linked depending on a proximity measure providing more flexibility than structured overlays and better performance than fully unstructured ones. Proximity-aware overlays connect participating entities so that they are connected to close neighbors according to a given proximity metric reflecting some degree of affinity (computation, interest, etc.) between peers. We extensively use this approach to provide algorithmic foundations of large-scale dynamic systems.

3.3. Epidemic protocols

Epidemic algorithms, also called gossip-based algorithms [44], [43], constitute a fundamental topic in our research. In the context of distributed systems, epidemic protocols are mainly used to create overlay networks and to ensure a reliable information dissemination in a large-scale distributed system. The principle underlying technique, in analogy with the spread of a rumor among humans via gossiping, is that participating entities continuously exchange information about the system in order to spread it gradually and reliably. Epidemic algorithms have proved efficient to build and maintain large-scale distributed systems in the context of many applications such as broadcasting [43], monitoring, resource management, search, and more generally in building unstructured peer-to-peer networks.

3.4. Malicious process behaviors

When assuming that processes fail by simply crashing, bounds on resiliency (maximum number of processes that may crash, number of exchanged messages, number of communication steps, etc.) are known both for synchronous and augmented asynchronous systems (recall that in purely asynchronous systems some problems are impossible to solve). If processes can exhibit malicious behaviors, these bounds are seldom the same. Sometimes, it is even necessary to change the specification of the problem. For example, the consensus problem for correct processes does not make sense if some processes can exhibit a Byzantine behavior and thus propose an arbitrary value. In this case, the validity property of consensus, which is normally "a decided value is a proposed value", must be changed to "if all correct processes propose the same value then only this value can be decided." Moreover, the resilience bound of less than half of faulty processes is at least lowered to "less than a third of Byzantine processes." These are some of the aspects that underlie our studies in the context of the classical model of distributed systems, in peer-to-peer systems and in sensor networks.

3.5. Online social networks and recommender systems

Social Networks have rapidly become a fundamental component of today's distributed applications. Web 2.0 applications have dramatically changed the way users interact with the Internet and with each other. The number of users of websites like Flickr, Delicious, Facebook, or MySpace is constantly growing, leading to significant technical challenges. On the one hand, these websites are called to handle enormous amounts of data. On the other hand, news continue to report the emergence of privacy threats to the personal data of social-network users. Our research aims to exploit our expertise in distributed systems to lead to a new generation of scalable, privacy-preserving, social applications.

We also investigate approaches to build implicit social networks, connecting users sharing similar interests. At the heart of the building of such similarity graphs lie k-nearest neighbor (KNN) algorithms. Our research in this area is to design and implement efficient KNN algorithms able to cope with a huge volume of data as well as a high level of dynamism. We investigate the use of such similarity graphs to build highly scalable infrastructures for recommendation systems.

ASCOLA Project-Team

3. Research Program

3.1. Overview

Since we mainly work on new concepts for the language-based definition and implementation of complex software systems, we first briefly introduce some basic notions and problems of software components (understood in a broad sense, that is, including modules, objects, architecture description languages and services), aspects, and domain-specific languages. We conclude by presenting the main issues related to distribution and concurrency, in particular related to capacity planning issues that are relevant to our work.

3.2. Software Composition

Modules and services. The idea that building *software components*, i.e., composable prefabricated and parameterized software parts, was key to create an effective software industry was realized very early [77]. At that time, the scope of a component was limited to a single procedure. In the seventies, the growing complexity of software made it necessary to consider a new level of structuring and programming and led to the notions of information hiding, *modules*, and module interconnection languages [84], [60]. Information hiding promotes a black-box model of program development whereby a module implementation, basically a collection of procedures, is strongly encapsulated behind an interface. This makes it possible to guarantee logical invariant *properties* of the data managed by the procedures and, more generally, makes *modular reasoning* possible.

In the context of today's Internet-based information society, components and modules have given rise to *software services* whose compositions are governed by explicit *orchestration or choreography* specifications that support notions of global properties of a service-oriented architecture. These horizontal compositions have, however, to be frequently adapted dynamically. Dynamic adaptations, in particular in the context of software evolution processes, often conflict with a black-box composition model either because of the need for invasive modifications, for instance, in order to optimize resource utilization or modifications to the vertical compositions implementing the high-level services.

Object-Oriented Programming. Classes and objects provide another kind of software component, which makes it necessary to distinguish between *component types* (classes) and *component instances* (objects). Indeed, unlike modules, objects can be created dynamically. Although it is also possible to talk about classes in terms of interfaces and implementations, the encapsulation provided by classes is not as strong as the one provided by modules. This is because, through the use of inheritance, object-oriented languages put the emphasis on *incremental programming* to the detriment of modular programming. This introduces a white-box model of software development and more flexibility is traded for safety as demonstrated by the *fragile base class* issue [80].

Architecture Description Languages. The advent of distributed applications made it necessary to consider more sophisticated connections between the various building blocks of a system. The *software architecture* [89] of a software system describes the system as a composition of *components* and *connectors*, where the connectors capture the *interaction protocols* between the components [48]. It also describes the rationale behind such a given architecture, linking the properties required from the system to its implementation. *Architecture Description Languages* (ADLs) are languages that support architecture-based development [78]. A number of these languages make it possible to generate executable systems from architectural descriptions, provided implementations for the primitive components are available. However, guaranteeing that the implementation conforms to the architecture is an issue.

Protocols. Today, protocols constitute a frequently used means to precisely define, implement, and analyze contracts, notably concerning communication and security properties, between two or more hardware or software entities. They have been used to define interactions between communication layers, security properties of distributed communications, interactions between objects and components, and business processes.

Object interactions [82], component interactions [95], [86] and service orchestrations [61] are most frequently expressed in terms of *regular interaction protocols* that enable basic properties, such as compatibility, substitutability, and deadlocks between components to be defined in terms of basic operations and closure properties of finite-state automata. Furthermore, such properties may be analyzed automatically using, e.g., model checking techniques [58], [67].

However, the limited expressive power of regular languages has led to a number of approaches using more expressive *non-regular* interaction protocols that often provide distribution-specific abstractions, e.g., session types [71], or context-free or turing-complete expressiveness [87], [55]. While these protocol types allow conformance between components to be defined (e.g., using unbounded counters), property verification can only be performed manually or semi-automatically.

3.3. Programming languages for advanced modularization

The main driving force for the structuring means, such as components and modules, is the quest for clean *separation of concerns* [62] on the architectural and programming levels. It has, however, early been noted that concern separation in the presence of crosscutting functionalities requires specific language and implementation level support. Techniques of so-called *computational reflection*, for instance, Smith's 3-Lisp or Kiczales's CLOS meta-object protocol [90], [74] as well as metaprogramming techniques have been developed to cope with this problem but proven unwieldy to use and not amenable to formalization and property analysis due to their generality. Methods and techniques from two fields have been particularly useful in addressing such advanced modularization problems: Aspect-Oriented Software Development as the field concerned with the systematic handling of modularization issues and domain-specific languages that provide declarative and efficient means for the definition of crosscutting functionalities.

Aspect-Oriented Software Development [73], [46] has emerged over the previous decade as the domain of systematic exploration of crosscutting concerns and corresponding support throughout the software development process. The corresponding research efforts have resulted, in particular, in the recognition of *crosscutting* as a fundamental problem of virtually any large-scale application, and the definition and implementation of a large number of aspect-oriented models and languages.

However, most current aspect-oriented models, notably AspectJ [72], rely on pointcuts and advice defined in terms of individual execution events. These models are subject to serious limitations concerning the modularization of crosscutting functionalities in distributed applications, the integration of aspects with other modularization mechanisms such as components, and the provision of correctness guarantees of the resulting AO applications. They do, in particular, only permit the manipulation of distributed applications on a per-host basis, that is, without direct expression of coordination properties relating different distributed entities [91]. Similarly, current approaches for the integration of aspects and (distributed) components do not directly express interaction properties between sets of components but rather seemingly unrelated modifications to individual components [59]. Finally, current formalizations of such aspect models are formulated in terms of low-level semantic abstractions (see, e.g., Wand's et al semantics for AspectJ [94]) and provide only limited support for the analysis of fundamental aspect properties.

Different approaches have been put forward to tackle these problems, in particular, in the context of so-called *stateful* or *history-based aspect languages* [63], [64], which provide pointcut and advice languages that directly express rich relationships between execution events. Such languages have been proposed to directly express coordination and synchronization issues of distributed and concurrent applications [83], [53], [66], provide more concise formal semantics for aspects and enable analysis of their properties [49], [65], [63], [47]. Furthermore, first approaches for the definition of *aspects over protocols* have been proposed, as well as over regular structures [63] and non-regular ones [93], [81], which are helpful for the modular definition and verification of protocols over crosscutting functionalities.

They represent, however, only first results and many important questions concerning these fundamental issues remain open, in particular, concerning the semantics foundations of AOP and the analysis and enforcement of correctness properties governing its, potentially highly invasive, modifications.

Domain-specific languages (DSLs) represent domain knowledge in terms of suitable basic language constructs and their compositions at the language level. By trading generality for abstraction, they enable complex relationships among domain concepts to be expressed concisely and their properties to be expressed and formally analyzed. DSLs have been applied to a large number of domains; they have been particularly popular in the domain of software generation and maintenance [79], [97].

Many modularization techniques and tasks can be naturally expressed by DSLs that are either specialized with respect to the type of modularization constructs, such as a specific brand of software component, or to the compositions that are admissible in the context of an application domain that is targeted by a modular implementation. Moreover, software development and evolution processes can frequently be expressed by transformations between applications implemented using different DSLs that represent an implementation at different abstraction levels or different parts of one application.

Functionalities that crosscut a component-based application, however, complicate such a DSL-based transformational software development process. Since such functionalities belong to another domain than that captured by the components, different DSLs should be composed. Such compositions (including their syntactic expression, semantics and property analysis) have only very partially been explored until now. Furthermore, restricted composition languages and many aspect languages that only match execution events of a specific domain (e.g., specific file accesses in the case of security functionality) and trigger only domain-specific actions clearly are quite similar to DSLs but remain to be explored.

3.4. Distribution and Concurrency

While ASCOLA does not investigate distribution and concurrency as research domains per se (but rather from a software engineering and modularization viewpoint), there are several specific problems and corresponding approaches in these domains that are directly related to its core interests that include the structuring and modularization of large-scale distributed infrastructures and applications. These problems include crosscutting functionalities of distributed and concurrent systems, support for the evolution of distributed software systems, and correctness guarantees for the resulting software systems.

Underlying our interest in these domains is the well-known observation that large-scale distributed applications are subject to *numerous crosscutting functionalities* (such as the transactional behavior in enterprise information systems, the implementation of security policies, and fault recovery strategies). These functionalities are typically partially encapsulated in distributed infrastructures and partially handled in an ad hoc manner by using infrastructure services at the application level. Support for a more principled approach to the development and evolution of distributed software systems in the presence of crosscutting functionalities has been investigated in the field of *open adaptable middleware* [54], [76]. Open middleware design exploits the concept of reflection to provide the desired level of configurability and openness. However, these approaches are subject to several fundamental problems. One important problem is their insufficient, framework-based support that only allows partial modularization of crosscutting functionalities.

There has been some *criticism* on the use of *AspectJ-like aspect models* (which middleware aspect models like that of JBoss AOP are an instance of) for the modularization of distribution and concurrency related concerns, in particular, for transaction concerns [75] and the modularization of the distribution concern itself [91]. Both criticisms are essentially grounded in AspectJ's inability to explicitly represent sophisticated relationships between execution events in a distributed system: such aspects therefore cannot capture the semantic relationships that are essential for the corresponding concerns. History-based aspects, as those proposed by the ASCOLA project-team provide a starting point that is not subject to this problem.

From a point of view of language design and implementation, aspect languages, as well as domain specific languages for distributed and concurrent environments share many characteristics with existing distributed languages: for instance, event monitoring is fundamental for pointcut matching, different synchronization strategies and strategies for code mobility [69] may be used in actions triggered by pointcuts. However, these relationships have only been explored to a small degree. Similarly, the formal semantics and formal properties of aspect languages have not been studied yet for the distributed case and only rudimentarily for the concurrent one [49], [66].

3.5. Security

Security properties and policies over complex service-oriented and standalone applications become ever more important in the context of asynchronous and decentralized communicating systems. Furthermore, they constitute prime examples of crosscutting functionalities that can only be modularized in highly insufficient ways with existing programming language and service models. Security properties and related properties, such as accountability properties, are therefore very frequently awkward to express and difficult to analyze and enforce (provided they can be made explicit in the first place).

Two main issues in this space are particularly problematic from a compositional point of view. First, information flow properties of programming languages, such as flow properties of Javascript [51], and service-based systems [57] are typically specially-tailored to specific properties, as well as difficult to express and analyze. Second, the enforcement of security properties and security policies, especially accountability-related properties [85], [92], is only supported using ad hoc means with rudimentary support for property verification.

The ASCOLA team has recently started to work on providing formal methods, language support and implementation techniques for the modular definition and implementation of information flow properties as well as policy enforcement in service-oriented systems as well as, mostly object-oriented, programming languages.

3.6. Green IT

With the emergence of the Future Internet and the dawn of new IT architecture and computation models such as cloud computing, the usage of data centers (DC) as well as their power consumption increase dramatically [56]. Besides the ecological impact [70], energy consumption is a predominant criterion for DC providers since it determines the daily cost of their infrastructure. As a consequence, power management becomes one of the main challenges for DC infrastructures and more generally for large-scale distributed systems.

To address this problem, we study two approaches: a workload-driven [52] and power-driven one [88]. As part of the workload-driven solution, we adapt the power consumption of the DC depending on the application workload, and evaluate whether this workload to be more reactive. We develop a distributed system from the system to the service-oriented level mainly based on hardware and virtualization capabilities that is managed in a user-transparent fashion. As part of the power-driven approach, we address energy consumption issues through a strong synergy inside the infrastructure software stack and more precisely between applications and resource management systems. This approach is characterized by adapting QoS properties aiming at the best trade-off between cost of energy (typically from the regular electric grid), its availability (for instance, from renewable energy), and service degradation caused, for instance, by application reconfigurations to jobs suspensions.

3.7. Capacity Planning for Large Scale Distributed System

Since the last decade, cloud computing has emerged as both a new economic model for software (provision) and as flexible tools for the management of computing capacity [50]. Nowadays, the major cloud features have become part of the mainstream (virtualization, storage and software image management) and the big market players offer effective cloud-based solutions for resource pooling. It is now possible to deploy virtual infrastructures that involve virtual machines (VMs), middleware, applications, and networks in such a simple manner that a new problem has emerged since 2010: VM sprawl (virtual machine proliferation) that consumes valuable computing, memory, storage and energy resources, thus menacing serious resource shortages. Scientific approaches that address VM sprawl are both based on classical administration techniques like the lifecycle management of a large number of VMs as well as the arbitration and the careful management of all resources consumed and provided by the hosting infrastructure (energy, power, computing, memory, network etc.) [68], [96].

The ASCOLA team investigates fundamental techniques for cloud computing and capacity planning, from infrastructures to the application level. Capacity planning is the process of planning for, analyzing, sizing, managing and optimizing capacity to satisfy demand in a timely manner and at a reasonable cost. Applied to distributed systems like clouds, a capacity planning solution must mainly provide the minimal set of resources necessary for the proper execution of the applications (i.e., to ensure SLA). The main challenges in this context are: scalability, fault tolerance and reactivity of the solution in a large-scale distributed system, the analysis and optimization of resources to minimize the cost (mainly costs related to the energy consumption of datacenters), as well as the profiling and adaptation of applications to ensure useful levels of quality of service (throughput, response time, availability etc.).

Our solutions are mainly based on virtualized infrastructures that we apply from the IaaS to the SaaS levels. We are mainly concerned by the management and the execution of the applications by harnessing virtualization capabilities, the investigation of alternative solutions that aim at optimizing the trade-off between performance and energy costs of both applications and cloud resources, as well as arbitration policies in the cloud in the presence of energy-constrained resources.

ASPI Team

3. Research Program

3.1. Interacting Monte Carlo methods and particle approximation of Feynman–Kac distributions

Monte Carlo methods are numerical methods that are widely used in situations where (i) a stochastic (usually Markovian) model is given for some underlying process, and (ii) some quantity of interest should be evaluated, that can be expressed in terms of the expected value of a functional of the process trajectory, which includes as an important special case the probability that a given event has occurred. Numerous examples can be found, e.g. in financial engineering (pricing of options and derivative securities) [36], in performance evaluation of communication networks (probability of buffer overflow), in statistics of hidden Markov models (state estimation, evaluation of contrast and score functions), etc. Very often in practice, no analytical expression is available for the quantity of interest, but it is possible to simulate trajectories of the underlying process. The idea behind Monte Carlo methods is to generate independent trajectories of this process or of an alternate instrumental process, and to build an approximation (estimator) of the quantity of interest in terms of the weighted empirical probability distribution associated with the resulting independent sample. By the law of large numbers, the above estimator converges as the size N of the sample goes to infinity, with rate $1/\sqrt{N}$ and the asymptotic variance can be estimated using an appropriate central limit theorem. To reduce the variance of the estimator, many variance reduction techniques have been proposed. Still, running independent Monte Carlo simulations can lead to very poor results, because trajectories are generated *blindly*, and only afterwards are the corresponding weights evaluated. Some of the weights can happen to be negligible, in which case the corresponding trajectories are not going to contribute to the estimator, i.e. computing power has been wasted.

A major breakthrough made in the mid 90's, has been the introduction of interacting Monte Carlo methods, also known as sequential Monte Carlo (SMC) methods, in which a whole (possibly weighted) sample, called *system of particles*, is propagated in time, where the particles

- *explore* the state space under the effect of a *mutation* mechanism which mimics the evolution of the underlying process,
- and are *replicated* or *terminated*, under the effect of a *selection* mechanism which automatically concentrates the particles, i.e. the available computing power, into regions of interest of the state space.

In full generality, the underlying process is a discrete–time Markov chain, whose state space can be finite, continuous, hybrid (continuous / discrete), graphical, constrained, time varying, pathwise, etc.,

the only condition being that it can easily be *simulated*.

In the special case of particle filtering, originally developed within the tracking community, the algorithms yield a numerical approximation of the optimal Bayesian filter, i.e. of the conditional probability distribution of the hidden state given the past observations, as a (possibly weighted) empirical probability distribution of the system of particles. In its simplest version, introduced in several different scientific communities under the name of *bootstrap filter* [38], *Monte Carlo filter* [43] or *condensation* (conditional density propagation) algorithm [42], and which historically has been the first algorithm to include a resampling step, the selection mechanism is governed by the likelihood function: at each time step, a particle is more likely to survive and to replicate at the next generation if it is consistent with the current observation. The algorithms also provide as a by–product a numerical approximation of the likelihood function, and of many other contrast functions for parameter estimation in hidden Markov models, such as the prediction error or the conditional least–squares criterion.

Particle methods are currently being used in many scientific and engineering areas

positioning, navigation, and tracking [39], [32], visual tracking [42], mobile robotics [33], [55], ubiquitous computing and ambient intelligence, sensor networks, risk evaluation and simulation of rare events [37], genetics, molecular simulation [34], etc.

Other examples of the many applications of particle filtering can be found in the contributed volume [22] and in the special issue of *IEEE Transactions on Signal Processing* devoted to *Monte Carlo Methods for Statistical Signal Processing* in February 2002, where the tutorial paper [23] can be found, and in the textbook [51] devoted to applications in target tracking. Applications of sequential Monte Carlo methods to other areas, beyond signal and image processing, e.g. to genetics, can be found in [48]. A recent overview can also be found in [25].

Particle methods are very easy to implement, since it is sufficient in principle to simulate independent trajectories of the underlying process. The whole problematic is multidisciplinary, not only because of the already mentioned diversity of the scientific and engineering areas in which particle methods are used, but also because of the diversity of the scientific communities which have contributed to establish the foundations of the field

target tracking, interacting particle systems, empirical processes, genetic algorithms (GA), hidden Markov models and nonlinear filtering, Bayesian statistics, Markov chain Monte Carlo (MCMC) methods.

These algorithms can be interpreted as numerical approximation schemes for Feynman–Kac distributions, a pathwise generalization of Gibbs–Boltzmann distributions, in terms of the weighted empirical probability distribution associated with a system of particles. This abstract point of view [30], [29], has proved to be extremely fruitful in providing a very general framework to the design and analysis of numerical approximation schemes, based on systems of branching and / or interacting particles, for nonlinear dynamical systems with values in the space of probability distributions, associated with Feynman–Kac distributions. Many asymptotic results have been proved as the number N of particles (sample size) goes to infinity, using techniques coming from applied probability (interacting particle systems, empirical processes [58]), see e.g. the survey article [30] or the textbooks [29], [28], and references therein

convergence in \mathbb{L}^p , convergence as empirical processes indexed by classes of functions, uniform convergence in time, see also [46], [47], central limit theorem, see also [44], [31], propagation of chaos, large deviations principle, etc.

The objective here is to systematically study the impact of the many algorithmic variants on the convergence results.

3.2. Multilevel splitting for rare event simulation

See 4.2, 5.1, and 5.2.

The estimation of the small probability of a rare but critical event, is a crucial issue in industrial areas such as

nuclear power plants, food industry, telecommunication networks, finance and insurance industry, air traffic management, etc.

In such complex systems, analytical methods cannot be used, and naive Monte Carlo methods are clearly inefficient to estimate accurately very small probabilities. Besides importance sampling, an alternate widespread technique consists in multilevel splitting [45], where trajectories going towards the critical set are given offsprings, thus increasing the number of trajectories that eventually reach the critical set. As shown in [6], the Feynman–Kac formalism of 3.1 is well suited for the design and analysis of splitting algorithms for rare event simulation.

Propagation of uncertainty Multilevel splitting can be used in static situations. Here, the objective is to learn the probability distribution of an output random variable $Y = F(X)$, where the function F is only defined pointwise for instance by a computer programme, and where the probability distribution of the input random variable X is known and easy to simulate from. More specifically, the objective could be to compute the probability of the output random variable exceeding a threshold, or more generally to evaluate the cumulative distribution function of the output random variable for different output values. This problem is characterized by the lack of an analytical expression for the function, the computational cost of a single pointwise evaluation of the function, which means that the number of calls to the function should be limited as much as possible, and finally the complexity and / or unavailability of the source code of the computer programme, which makes any modification very difficult or even impossible, for instance to change the model as in importance sampling methods.

The key issue is to learn as fast as possible regions of the input space which contribute most to the computation of the target quantity. The proposed splitting methods consists in (i) introducing a sequence of intermediate regions in the input space, implicitly defined by exceeding an increasing sequence of thresholds or levels, (ii) counting the fraction of samples that reach a level given that the previous level has been reached already, and (iii) improving the diversity of the selected samples, usually with an artificial Markovian dynamics for the input variable. In this way, the algorithm learns

- the transition probability between successive levels, hence the probability of reaching each intermediate level,
- and the probability distribution of the input random variable, conditioned on the output variable reaching each intermediate level.

A further remark, is that this conditional probability distribution is precisely the optimal (zero variance) importance distribution needed to compute the probability of reaching the considered intermediate level.

Rare event simulation To be specific, consider a complex dynamical system modelled as a Markov process, whose state can possibly contain continuous components and finite components (mode, regime, etc.), and the objective is to compute the probability, hopefully very small, that a critical region of the state space is reached by the Markov process before a final time T , which can be deterministic and fixed, or random (for instance the time of return to a recurrent set, corresponding to a nominal behaviour).

The proposed splitting method consists in (i) introducing a decreasing sequence of intermediate, more and more critical, regions in the state space, (ii) counting the fraction of trajectories that reach an intermediate region before time T , given that the previous intermediate region has been reached before time T , and (iii) regenerating the population at each stage, through resampling. In addition to the non-intrusive behaviour of the method, the splitting methods make it possible to learn the probability distribution of typical critical trajectories, which reach the critical region before final time T , an important feature that methods based on importance sampling usually miss. Many variants have been proposed, whether

- the branching rate (number of offsprings allocated to a successful trajectory) is fixed, which allows for depth-first exploration of the branching tree, but raises the issue of controlling the population size,
- the population size is fixed, which requires a breadth-first exploration of the branching tree, with random (multinomial) or deterministic allocation of offsprings, etc.

Just as in the static case, the algorithm learns

- the transition probability between successive levels, hence the probability of reaching each intermediate level,
- and the entrance probability distribution of the Markov process in each intermediate region.

Contributions have been given to

- minimizing the asymptotic variance, obtained through a central limit theorem, with respect to the shape of the intermediate regions (selection of the importance function), to the thresholds (levels), to the population size, etc.

- controlling the probability of extinction (when not even one trajectory reaches the next intermediate level),
- designing and studying variants suited for hybrid state space (resampling per mode, marginalization, mode aggregation),

and in the static case, to

- minimizing the asymptotic variance, obtained through a central limit theorem, with respect to intermediate levels, to the Metropolis kernel introduced in the mutation step, etc.

A related issue is global optimization. Indeed, the difficult problem of finding the set M of global minima of a real-valued function V can be replaced by the apparently simpler problem of sampling a population from a probability distribution depending on a small parameter, and asymptotically supported by the set M as the small parameter goes to zero. The usual approach here is to use the cross-entropy method [52], [27], which relies on learning the optimal importance distribution within a prescribed parametric family. On the other hand, multilevel splitting methods could provide an alternate nonparametric approach to this problem.

3.3. Statistical learning: pattern recognition and nonparametric regression

In pattern recognition and statistical learning, also known as machine learning, nearest neighbor (NN) algorithms are amongst the simplest but also very powerful algorithms available. Basically, given a training set of data, i.e. an N -sample of i.i.d. object-feature pairs, with real-valued features, the question is how to generalize, that is how to guess the feature associated with any new object. To achieve this, one chooses some integer k smaller than N , and takes the mean-value of the k features associated with the k objects that are nearest to the new object, for some given metric.

In general, there is no way to guess exactly the value of the feature associated with the new object, and the minimal error that can be done is that of the Bayes estimator, which cannot be computed by lack of knowledge of the distribution of the object-feature pair, but the Bayes estimator can be useful to characterize the strength of the method. So the best that can be expected is that the NN estimator converges, say when the sample size N grows, to the Bayes estimator. This is what has been proved in great generality by Stone [53] for the mean square convergence, provided that the object is a finite-dimensional random variable, the feature is a square-integrable random variable, and the ratio k/N goes to 0. Nearest neighbor estimator is not the only local averaging estimator with this property, but it is arguably the simplest.

The asymptotic behavior when the sample size grows is well understood in finite dimension, but the situation is radically different in general infinite dimensional spaces, when the objects to be classified are functions, images, etc.

Nearest neighbor classification in infinite dimension In finite dimension, the k -nearest neighbor classifier is universally consistent, i.e. its probability of error converges to the Bayes risk as N goes to infinity, whatever the joint probability distribution of the pair, provided that the ratio k/N goes to zero. Unfortunately, this result is no longer valid in general metric spaces, and the objective is to find out reasonable sufficient conditions for the weak consistency to hold. Even in finite dimension, there are exotic distances such that the nearest neighbor does not even get closer (in the sense of the distance) to the point of interest, and the state space needs to be complete for the metric, which is the first condition. Some regularity on the regression function is required next. Clearly, continuity is too strong because it is not required in finite dimension, and a weaker form of regularity is assumed. The following consistency result has been obtained: if the metric space is separable and if some Besicovich condition holds, then the nearest neighbor classifier is weakly consistent. Note that the Besicovich condition is always fulfilled in finite dimensional vector spaces (this result is called the Besicovich theorem), and that a counterexample [4] can be given in an infinite dimensional space with a Gaussian measure (in this case, the nearest neighbor classifier is clearly nonconsistent). Finally, a simple example has been found which verifies the Besicovich condition with a noncontinuous regression function.

Rates of convergence of the functional k -nearest neighbor estimator Motivated by a broad range of potential applications, such as regression on curves, rates of convergence of the k -nearest neighbor estimator of the regression function, based on N independent copies of the object–feature pair, have been investigated when the object is in a suitable ball in some functional space. Using compact embedding theory, explicit and general finite sample bounds can be obtained for the expected squared difference between the k -nearest neighbor estimator and the Bayes regression function, in a very general setting. The results have also been particularized to classical function spaces such as Sobolev spaces, Besov spaces and reproducing kernel Hilbert spaces. The rates obtained are genuine nonparametric convergence rates, and up to our knowledge the first of their kind for k -nearest neighbor regression.

This topic has produced several theoretical advances [1], [2] in collaboration with Gérard Biau (université Pierre et Marie Curie). A few possible target application domains have been identified in

- the statistical analysis of recommendation systems,
- the design of reduced–order models and analog samplers,

that would be a source of interesting problems.

CAIRN Project-Team

3. Research Program

3.1. Panorama

The development of complex applications is traditionally split in three stages: a theoretical study of the algorithms, an analysis of the target architecture and the implementation. When facing new emerging applications such as high-performance, low-power and low-cost mobile communication systems or smart sensor-based systems, it is mandatory to strengthen the design flow by a joint study of both algorithmic and architectural issues.

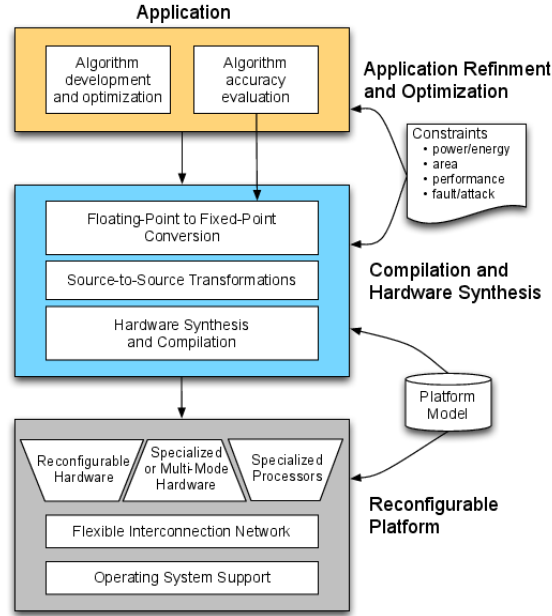


Figure 1. CAIRN's general design flow and related research themes

Figure 1 shows the global design flow we propose to develop. This flow is organized in levels which refer to our three research themes: application optimization (new algorithms, fixed-point arithmetic, advanced representations of numbers), architecture optimization (reconfigurable and specialized hardware, application-specific processors, arithmetic operators and functions), and stepwise refinement and code generation (code transformations, hardware synthesis, compilation).

In the rest of this part, we briefly describe the challenges concerning **new reconfigurable platforms** in Section 3.2 and the issues on **compiler and synthesis tools** related to these platforms in Section 3.3 .

3.2. Reconfigurable Architecture Design

Nowadays, FPGAs are not only suited for application specific algorithms, but also considered as fully-featured computing platforms, thanks to their ability to accelerate massively parallelizable algorithms much faster than their processor counterparts [75]. They also support to be dynamically reconfigured. At runtime, partially reconfigurable regions of the logic fabric can be reconfigured to implement a different task, which allows for a better resource usage and adaptation to the environment. Dynamically reconfigurable hardware can also cope with hardware errors by relocating some of its functionalities to another, sane, part of the logic fabric. It could also provide support for a multi-tasked computation flow where hardware tasks are loaded on-demand at runtime. Nevertheless, current design flows of FPGA vendors are still limited by the use of one partial bitstream for each reconfigurable region and for each design. These regions are defined at design time and it is not possible to use only one bitstream for multiple reconfigurable regions nor multiple chips. The multiplicity of such bitstreams leads to a significant increase in memory. Recent research has been conducted in the domain of task relocation on a reconfigurable fabric. All of the related work was conducted on architectures from commercial vendors (e.g., Xilinx, Altera) which share the same limitations: the inner details of the bitstream are not publicly known, which limits applicability of the techniques. To circumvent this issue, most dynamic reconfiguration techniques are either generating multiple bitstreams for each location [59] or implementing an online filter to relocate the tasks [69]. Both of these techniques still suffer from memory footprint and from the online complexity of task relocation.

Increasing the level and grain of reconfiguration is a solution to counterbalance the FPGA penalties. Coarse-grained reconfigurable architectures (CGRA) provide operator-level configurable functional blocks and word-level datapaths [76], [64], [74]. Compared to FPGA, they benefit from a massive reduction in configuration memory and configuration delay, as well as for routing and placement complexity. This in turns results in an improvement in the computation volume over energy cost ratio, although with a loss of flexibility compared to bit-level operations. Such constraints have been taken into account in the design of DART[7], Adres [72] or polymorphous computing fabrics[9]. These works have led to commercial products such as the PACT/XPP [58] or Montium from Recore systems, without however a real commercial success yet. Emerging platforms like Xilinx/Zynq or Intel/Altera are about to change the game.

In the context of emerging heterogenous multicore architecture, CAIRN advocates for associating general-purpose processors (GPP), flexible network-on-chip and coarse-grain or fine-grain dynamically reconfigurable accelerators. We leverage our skills on microarchitecture, reconfigurable computing, arithmetic, and low-power design, to discover and design such architectures with a focus on: -reduced energy per operation, - improved application performance through acceleration, - hardware flexibility and self-adaptive behavior, - tolerance to faults, computing errors, and process variation, - protections against side channel attacks, - limited silicon area overhead.

3.3. Compilation and Synthesis for Reconfigurable Platforms

In spite of their advantages, reconfigurable architectures, and more generally hardware accelerators, lack efficient and standardized compilation and design tools. As of today, this still makes the technology impractical for large-scale industrial use. Generating and optimizing the mapping from high-level specifications to reconfigurable hardware platforms are therefore key research issues, which have received considerable interest over the last years [62], [77], [73], [71], [70]. In the meantime, the complexity (and heterogeneity) of these platforms has also been increasing quite significantly, with complex heterogeneous multi-cores architectures becoming a *de facto* standard. As a consequence, the focus of designers is now geared toward optimizing overall system-level performance and efficiency [68]. Here again, existing tools are not well suited, as they fail at providing an unified programming view of the programmable and/or reconfigurable components implemented on the platform.

In this context, we have been pursuing our efforts to propose tools whose design principles are based on a tight coupling between the compiler and the target hardware architectures. We build on the expertise of the team members in High Level Synthesis (HLS) [4], ASIP optimizing compilers [10] and automatic parallelization for massively parallel specialized circuits [2]. We first study how to increase the efficiency of standard programmable processors by extending their instruction set to speed-up compute intensive kernels. Our focus is on efficient and exact algorithms for the identification, selection and scheduling of such instructions [5]. We address compilation challenges by borrowing techniques from high-level synthesis, optimizing compilers and automatic parallelization, especially when dealing with nested loop kernels. In addition, and independently of the scientific challenges mentioned above, proposing such flows also poses significant software engineering issues. As a consequence, we also study how leading edge software engineering techniques (Model Driven Engineering) can help the Computer Aided Design (CAD) and optimizing compiler communities prototyping new research ideas [3].

Efficient implementation of multimedia and signal processing applications (in software for DSP cores or as special-purpose hardware) often requires, for reasons related to cost, power consumption or silicon area constraints, the use of fixed-point arithmetic, whereas the algorithms are usually specified in floating-point arithmetic. Unfortunately, fixed-point conversion is very challenging and time-consuming, typically demanding up to 50% of the total design or implementation time. Thus, tools are required to automate this conversion. For hardware or software implementation, the aim is to optimize the fixed-point specification. The implementation cost is minimized under a numerical accuracy or an application performance constraint. For DSP-software implementation, methodologies have been proposed [6] to achieve fixed-point conversion. For hardware implementation, the best results are obtained when the word-length optimization process is coupled with the high-level synthesis [65]. Evaluating the effects of finite precision is one of the major and often the most time consuming step while performing fixed-point refinement. Indeed, in the word-length optimization process, the numerical accuracy is evaluated as soon as a new word-length is tested, thus, several times per iteration of the optimization process. Classical approaches are based on fixed-point simulations [66]. Leading to long evaluation times, they can hardly be used to explore the design space. Therefore, our aim is to propose closed-form expressions of errors due to fixed-point approximations that are used by a fast analytical framework for accuracy evaluation [8].

3.4. Software Frameworks Developed by the Team

With the ever raising complexity of embedded applications and platforms, the need for efficient and customizable compilation flows is stronger than ever. This need of flexibility is even stronger when it comes to research compiler infrastructures that are necessary to gather quantitative evidence of the performance/energy or cost benefits obtained through the use of reconfigurable platforms. From a compiler point of view, the challenges exposed by these complex reconfigurable platforms are quite significant, since they require the compiler to extract and to expose an important amount of coarse and/or fine grain parallelism, to take complex resource constraints into consideration while providing efficient memory hierarchy and power management.

Because they are geared toward industrial use, production compiler infrastructures do not offer the level of flexibility and productivity that is required for compiler and CAD tool prototyping. To address this issue, we designed an extensible source-to-source compiler infrastructure that takes advantage of leading edge model-driven object-oriented software engineering principles and technologies.

Figure 2 shows the global framework that is being developed in the group. Our compiler flow mixes several types of intermediate representations. The baseline representation is a simple tree-based model enriched with control flow information. This model is mainly used to support our source-to-source flow, and serves as the backbone for the infrastructure. We use the extensibility of the framework to provide more advanced representations along with their corresponding optimizations and code generation plug-ins. For example, for our pattern selection and accuracy estimation tools, we use a data dependence graph model in all basic blocks instead of the tree model. Similarly, to enable polyhedral based program transformations and analysis, we introduced a specific representation for affine control loops that we use to derive a Polyhedral Reduced Dependence Graph (PRDG). Our current flow assumes that the application is specified as a hierarchy of

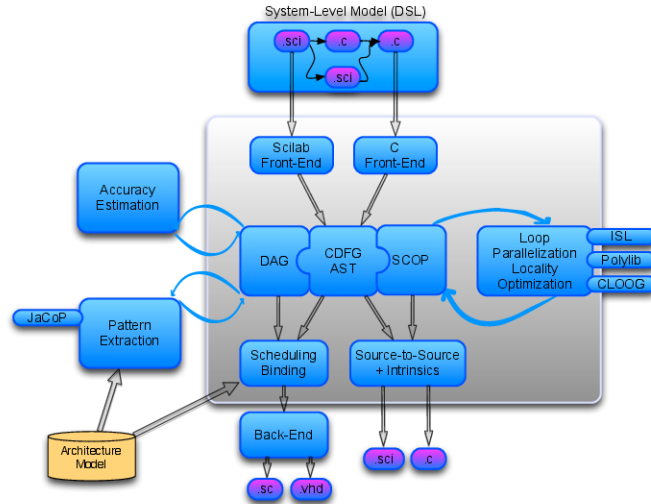


Figure 2. CAIRN's general software development framework.

communicating tasks, where each task is expressed using C or Matlab/Scilab, and where the system-level representation and the target platform model are often defined using Domain Specific Languages (DSL).

Gecos (Generic Compiler Suite) is the main backbone of CAIRN's flow. It is an open source Eclipse-based flexible compiler infrastructure developed for fast prototyping of complex compiler passes. Gecos is a 100% Java based implementation and is based on modern software engineering practices such as Eclipse plugin or model-driven software engineering with EMF (Eclipse Modeling Framework). As of today, our flow offers the following features:

- An automatic floating-point to fixed-point conversion flow (for ASIC/FPGA and embedded processors). **ID.Fix** is an infrastructure for the automatic transformation of software code aiming at the conversion of floating-point data types into a fixed-point representation.
- A polyhedral-based loop transformation and parallelization engine (mostly targeted at HLS).
- A custom instruction extraction flow (for ASIP and dynamically reconfigurable architectures). **Durase** is developed for the compilation and the synthesis targeting reconfigurable platforms and the automatic synthesis of application specific processor extensions. It uses advanced technologies, such as graph matching together with constraint programming methods.
- Several back-ends to enable the generation of VHDL for specialized or reconfigurable IPs, and SystemC for simulation purposes (e.g., fixed-point simulations).

Gecos, ID.Fix or Durase have been demonstrated during "University Booths" in various conference such as IEEE/ACM DAC or DATE.

CELTIQUE Project-Team (section vide)

CIDRE Project-Team

3. Research Program

3.1. Our perspective

For many aspects of our everyday life, we heavily rely on information systems, many of which are based on massively networked devices that support a population of interacting and cooperating entities. While these information systems become increasingly open and complex, accidental and intentional failures get considerably more frequent and severe.

Two research communities traditionally address the concern of accidental and intentional failures: the distributed computing community and the security community. While both communities are interested in the construction of systems that are correct and secure, an ideological gap and a lack of communication exist between them that is often explained by the incompatibility of the assumptions each of them traditionally makes. Furthermore, in terms of objectives, the distributed computing community has favored systems availability while the security community has focused on integrity and confidentiality, and more recently on privacy.

Our long term ambition is to contribute to the building of distributed systems that are trustworthy and respectful of privacy, even when some nodes⁰ in the system have been compromised. For that purpose, we are convinced that combining classical security approaches and distributed computing paradigms is an interesting way to enforce the security of large-scale distributed systems. More specifically, since a distributed system is composed of nodes, we assert that the security of large-scale distributed systems has to be addressed at three complementary levels:

- the level of each node: each standalone node has to enforce its own security;
- the level of an *identified* set of *trusted* nodes: the *trusted* nodes can *collaborate* to enforce together their security;
- the level of fully open large-scale distributed and dynamic systems: distributed computing paradigms such as consensus algorithms can be applied to cope with the possible presence of malicious nodes.

Notice that using a distributed architecture can also be an approach allowing the nodes to enforce their security without the need of a trusted third party.

The research activities of the CIDRE project-team focus mainly on the two following research axis:

- **Intrusion Detection System:** the objective is to detect any suspicious events with regard to the security by analyzing some data generated on the monitored system.
- **Privacy-preserving Services:** the objective is to ensure users' privacy even when this property seems incompatible with the provided services, like social networks or location-based services.

In all our studies, we consider a priori that the attacker is omnipotent. He can act as he wants. Nevertheless, since our team is not specialized in cryptography, we consider that we can rely on strong unbroken cryptosystems.

3.2. Intrusion Detection / Security Events Monitoring and Management

Today, we have not yet fully entered into a world of “security by design”. Security remains often a property that is considered a posteriori, when the system is deployed, which often results in applying patches when vulnerabilities are discovered (also called a “patch and pray” approach). Unfortunately, despite patching, the number of vulnerabilities remains high, as evidenced by the number of vulnerabilities published each year in the Common Vulnerabilities and Exposures (CVE) system. Thus, it is important to be able to early detect cyber-attacks, especially when they exploit vulnerabilities that are unknown. However, the efficiency of

⁰The term node either refers to a device that hosts a network client or service or to the process that runs this client or service.

security events monitoring and management systems (including the IDS - Intrusion Detection Systems) is still an open issue today. Indeed, they are often unable to effectively deal with huge numbers of security events, and they usually produce too many false alarms yet missing some attacks. So one of the main research challenges in IT security remains the definition of efficient security events monitoring systems, i.e., that enable both to process a huge number of security events and to detect any attacks without flooding the security analysts with false alarms.

By exploiting vulnerabilities in operating systems, applications, or network services, an attacker can defeat preventive security mechanisms and violate the security policy of the whole system. The goal of an Intrusion Detection Systems (IDS) is to detect such violations by analyzing some *security events* generated on a monitored system. Ideally, the IDS should produce an alert for any violation (no *false negative*), and only for violations (no *false positive*).

To produce alerts, two detection techniques exist: the misuse based detection and the anomaly based detection. A misuse based detection is actually a signature based detection approach : it allows to detect only the attacks whose signature is available. From our point of view, while useful in practice, misuse detection is intrinsically limited. Indeed, it requires to update in real-time the database of signatures, similarly to what has to be done for antivirus tools. The CIDRE project-team follows the alternative approach, namely the anomaly approach, which consists in detecting a deviation from a referenced behavior. Our contributions on anomaly-based IDS follow three axis:

- **Illegal Information Flow Detection:** our goal is to detect information flows in the monitored system (either a node or a set of trusted nodes) that are allowed by the access control mechanism, but are illegal from the security policy point of view. This approach is particularly appealing to detect intrusions in a standalone node, such as a smartphone.
- **Anomaly-Based Detection in Distributed Applications:** our goal is to specify the normal behavior based on either a formal specification of the distributed application, or previous executions. This approach is particularly appealing to detect intrusions in industrial control systems since these systems exhibit well-defined behaviors at different levels: network level (network communication patterns, protocol specifications, etc.), control level (continue and discrete process control laws), or even the state of the local resources (memory or CPU).
- **Online data analytics:** our goal is to estimate on the fly different statistics or metrics on distributed input streams to detect abnormal behavior with respect to a well-defined criterion such as the distance between different streams, their correlation or their entropy.

Beside the anomaly-based IDS, we have also led research work on alert correlation and visualisation of security events. Indeed, in large systems, multiple (host and network) IDS and many sensors are deployed and they continuously and independently generate notifications (event's observations, warnings and alerts). To cope with this huge amount of collected data, we have studied two different approaches, each with specific goal:

- **Alert Correlation System:** the alerts of *low level* IDSes can be viewed as *security events* of a *high level* IDS whose goal is to correlate these alerts. An alert correlation system aims at exploiting the known relationships between some elements that appear in the flow of low level notifications to generate high semantic meta-alerts. The main goal is to reduce the number of alerts (and especially, false positive) returned to the security analysts and to allow a higher level analysis of the situation (situational awareness).
- **Visualization Tools:** a visualization tools aims at relying on the capacity of human beings to detect patterns and outliers in datasets when these datasets are properly visually represented. Human beings also know pieces of contextual information that are very difficult to formalize so as to make them usable by a computer. Visualization is therefore a very useful complementary tool to detect abnormal events in real time (monitoring), to search for malicious events in log files (data exploration and forensics) and to communicate results (reporting).

3.3. Privacy

In a world of ubiquitous technologies, each individual constantly leaves digital traces related to his activities and interests. The current business plan of many web services such as social networks, is based on the sale of these digital traces. Of course, this is usually done in a legal way, the license of use clearly stating that the user gives the right to the service provider for using his personal data. However, on the one hand, users generally do not read these licenses, and on the other hand, these licenses are usually very vague on the use of personal data⁰. In addition these digital traces can potentially be stolen and maliciously used, they must therefore be protected. In this context, users' privacy is now recognized as a fundamental individual right. Any new IT service should thus follow the *privacy-by-design* approach: privacy issues have to be studied from the earliest phase of a project by taking into account the multi-stakeholders and transdisciplinary aspects in order to ensure proper, end-to-end private data protection properties.

In the CIDRE project, we mainly focus on domains in which privacy issues collide with provided services. Here are some concrete examples of such domains:

- **Location-based services:** the challenge is to design services that depend on the user's location while preserving the privacy of his location;
- **Social networks:** the challenge is to demonstrate that it is possible to design social networks respectful of users' privacy;
- **Mobile services:** given that such services are based on user's identity, the challenge is to design mobile services while preserving the users' anonymity;
- **Ad-hoc networks:** in ad-hoc networks, any participant can potentially know the relative location of the other participants. Thus, the issue is to allow nodes to forward messages while preserving the privacy of the communications.

For all of these domains, we have proposed new Privacy-Enhancing Techniques (PETs) based on a mix of different foundations such as cryptographic techniques, security policies and access control mechanisms, just to name a few. More generally, we think that a major option to protect users' privacy consists in using a decentralized architecture that enables to transfer control and services from the service providers to the users.

The concept of IDS seems to be in contradiction with the users' privacy. Indeed, an IDS is a monitoring system that needs to collect and analyze information coming from different levels such as network, applications and OS, this information being able to include users' personal data. However, we are confident that IDS and privacy are not completely antagonist. In particular, integrating some privacy features inside an IDS to build a privacy-preserving IDS may allow to limit the amount of information that can leak if one of the nodes within the system is compromised. On the other hand, enabling IDS to detect attacks against privacy as well as security violations can extend the range of their applicability.

⁰Besides, it has been shown that service providers do not necessarily comply with their own license.

DIONYSOS Project-Team

3. Research Program

3.1. Introduction

The scientific foundations of our work are those of network design and network analysis. Specifically, this concerns the principles of packet switching and in particular of IP networks (protocol design, protocol testing, routing, scheduling techniques), and the mathematical and algorithmic aspects of the associated problems, on which our methods and tools are based.

These foundations are described in the following paragraphs. We begin by a subsection dedicated to Quality of Service (QoS) and Quality of Experience (QoE), since they can be seen as unifying concepts in our activities. Then we briefly describe the specific sub-area of model evaluation and about the particular multidisciplinary domain of network economics.

3.2. Quality of Service and Quality of Experience

Since it is difficult to develop as many communication solutions as possible applications, the scientific and technological communities aim towards providing general *services* allowing to give to each application or user a set of properties nowadays called “Quality of Service” (QoS), a terminology lacking a precise definition. This QoS concept takes different forms according to the type of communication service and the aspects which matter for a given application: for performance it comes through specific metrics (delays, jitter, throughput, etc.), for dependability it also comes through appropriate metrics: reliability, availability, or vulnerability, in the case for instance of WAN (Wide Area Network) topologies, etc.

QoS is at the heart of our research activities: We look for methods to obtain specific “levels” of QoS and for techniques to evaluate the associated metrics. Our ultimate goal is to provide tools (mathematical tools and/or algorithms, under appropriate software “containers” or not) allowing users and/or applications to attain specific levels of QoS, or to improve the provided QoS, if we think of a particular system, with an optimal use of the resources available. Obtaining a good QoS level is a very general objective. It leads to many different areas, depending on the systems, applications and specific goals being considered. Our team works on several of these areas. We also investigate the impact of network QoS on multimedia payloads to reduce the impact of congestion.

Some important aspects of the behavior of modern communication systems have subjective components: the quality of a video stream or an audio signal, *as perceived by the user*, is related to some of the previous mentioned parameters (packet loss, delays, ...) but in an extremely complex way. We are interested in analyzing these types of flows from this user-oriented point of view. We focus on the *user perceived quality*, in short, PQ, the main component of what is nowadays called Quality of Experience (in short, QoE), to underline the fact that, in this case, we want to center the analysis on the user. In this context, we have a global project called PSQA, which stands for Pseudo-Subjective Quality Assessment, and which refers to a technology we have developed allowing to automatically measure this PQ.

Another special case to which we devote research efforts in the team is the analysis of qualitative properties related to interoperability assessment. This refers to the act of determining if end-to-end functionality between at least two communicating systems is as required by the base standards for those systems. Conformance is the act of determining to what extent a single component conforms to the individual requirements of the standard it is based on. Our purpose is to provide such a formal framework (methods, algorithms and tools) for interoperability assessment, in order to help in obtaining efficient interoperability test suites for new generation networks, mainly around IPv6-related protocols. The interoperability test suites generation is based on specifications (standards and/or RFCs) of network components and protocols to be tested.

3.3. Stochastic modeling

The scientific foundations of our modeling activities are composed of stochastic processes theory and, in particular, Markov processes, queuing theory, stochastic graphs theory, etc. The objectives are either to develop numerical solutions, or analytical ones, or possibly discrete event simulation or Monte Carlo (and Quasi-Monte Carlo) techniques. We are always interested in model evaluation techniques for dependability and performability analysis, both in static (network reliability) and dynamic contexts (depending on the fact that time plays an explicit role in the analysis or not). We look at systems from the classical so-called *call level*, leading to standard models (for instance, queues or networks of queues) and also at the *burst level*, leading to *fluid models*.

In recent years, our work on the design of the topologies of WANs led us to explore optimization techniques, in particular in the case of very large optimization problems, usually formulated in terms of graphs. The associated methods we are interested in are composed of simulated annealing, genetic algorithms, TABU search, etc. For the time being, we have obtained our best results with GRASP techniques.

Network pricing is a good example of a multi-disciplinary research activity half-way between applied mathematics, economy and networking, centered on stochastic modeling issues. Indeed, the Internet is facing a tremendous increase of its traffic volume. As a consequence, real users complain that large data transfers take too long, without any possibility to improve this by themselves (by paying more, for instance). A possible solution to cope with congestion is to increase the link capacities; however, many authors consider that this is not a viable solution as the network must respond to an increasing demand (and experience has shown that demand of bandwidth has always been ahead of supply), especially now that the Internet is becoming a commercial network. Furthermore, incentives for a fair utilization between customers are not included in the current Internet. For these reasons, it has been suggested that the current flat-rate fees, where customers pay a subscription and obtain an unlimited usage, should be replaced by usage-based fees. Besides, the future Internet will carry heterogeneous flows such as video, voice, email, web, file transfers and remote login among others. Each of these applications requires a different level of QoS: for example, video needs very small delays and packet losses, voice requires small delays but can afford some packet losses, email can afford delay (within a given bound) while file transfer needs a good average throughput and remote login requires small round-trip times. Some pricing incentives should exist so that each user does not always choose the best QoS for her application and so that the final result is a fair utilization of the bandwidth. On the other hand, we need to be aware of the trade-off between engineering efficiency and economic efficiency; for example, traffic measurements can help in improving the management of the network but is a costly option. These are some of the various aspects often present in the pricing problems we address in our work. More recently, we have switched to the more general field of network economics, dealing with the economic behavior of users, service providers and content providers, as well as their relations.

DIVERSE Project-Team

3. Research Program

3.1. Scientific background

3.1.1. Model-driven engineering

Model-Driven Engineering (MDE) aims at reducing the accidental complexity associated with developing complex software-intensive systems (e.g., use of abstractions of the problem space rather than abstractions of the solution space) [101]. It provides DIVERSE with solid foundations to specify, analyze and reason about the different forms of diversity that occur through the development lifecycle. A primary source of accidental complexity is the wide gap between the concepts used by domain experts and the low-level abstractions provided by general-purpose programming languages [72]. MDE approaches address this problem through modeling techniques that support separation of concerns and automated generation of major system artifacts from models (e.g., test cases, implementations, deployment and configuration scripts). In MDE, a model describes an aspect of a system and is typically created or derived for specific development purposes [54]. Separation of concerns is supported through the use of different modeling languages, each providing constructs based on abstractions that are specific to an aspect of a system. MDE technologies also provide support for manipulating models, for example, support for querying, slicing, transforming, merging, and analyzing (including executing) models. Modeling languages are thus at the core of MDE, which participates to the development of a sound *Software Language Engineering*⁰, including an unified typing theory that integrate models as first class entities [104].

Incorporating domain-specific concepts and high-quality development experience into MDE technologies can significantly improve developer productivity and system quality. Since the late nineties, this realization has led to work on MDE language workbenches that support the development of domain-specific modeling languages (DSMLs) and associated tools (e.g., model editors and code generators). A DSML provides a bridge between the field in which domain experts work and the implementation (programming) field. Domains in which DSMLs have been developed and used include, among others, automotive, avionics, and the emerging cyber-physical systems. A study performed by Hutchinson et al. [78] provides some indications that DSMLs can pave the way for wider industrial adoption of MDE.

More recently, the emergence of new classes of systems that are complex and operate in heterogeneous and rapidly changing environments raises new challenges for the software engineering community. These systems must be adaptable, flexible, reconfigurable and, increasingly, self-managing. Such characteristics make systems more prone to failure when running and thus the development and study of appropriate mechanisms for continuous design and run-time validation and monitoring are needed. In the MDE community, research is focused primarily on using models at design, implementation, and deployment stages of development. This work has been highly productive, with several techniques now entering a commercialization phase. As software systems are becoming more and more dynamic, the use of model-driven techniques for validating and monitoring run-time behavior is extremely promising [86].

3.1.2. Variability modeling

While the basic vision underlying *Software Product Lines* (SPL) can probably be traced back to David Parnas seminal article [94] on the Design and Development of Program Families, it is only quite recently that SPLs are emerging as a paradigm shift towards modeling and developing software system families rather than individual systems [92]. SPL engineering embraces the ideas of mass customization and software reuse. It focuses on the means of efficiently producing and maintaining multiple related software products, exploiting what they have in common and managing what varies among them.

⁰See <http://www.sleconf.org/>

Several definitions of the *software product line* concept can be found in the research literature. Clements *et al.* define it as *a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and are developed from a common set of core assets in a prescribed way* [91]. Bosch provides a different definition [60]: *A SPL consists of a product line architecture and a set of reusable components designed for incorporation into the product line architecture. In addition, the PL consists of the software products developed using the mentioned reusable assets.* In spite of the similarities, these definitions provide different perspectives of the concept: *market-driven*, as seen by Clements *et al.*, and *technology-oriented* for Bosch.

SPL engineering is a process focusing on capturing the *commonalities* (assumptions true for each family member) and *variability* (assumptions about how individual family members differ) between several software products [66]. Instead of describing a single software system, a SPL model describes a set of products in the same domain. This is accomplished by distinguishing between elements common to all SPL members, and those that may vary from one product to another. Reuse of core assets, which form the basis of the product line, is key to productivity and quality gains. These core assets extend beyond simple code reuse and may include the architecture, software components, domain models, requirements statements, documentation, test plans or test cases.

The SPL engineering process consists of two major steps:

1. **Domain Engineering**, or *development for reuse*, focuses on core assets development.
2. **Application Engineering**, or *development with reuse*, addresses the development of the final products using core assets and following customer requirements.

Central to both processes is the management of **variability** across the product line [74]. In common language use, the term *variability* refers to *the ability or the tendency to change*. Variability management is thus seen as the key feature that distinguishes SPL engineering from other software development approaches [61]. Variability management is thus growingly seen as the cornerstone of SPL development, covering the entire development life cycle, from requirements elicitation [106] to product derivation [111] to product testing [90], [89].

Halmans *et al.* [74] distinguish between *essential* and *technical* variability, especially at requirements level. Essential variability corresponds to the customer's viewpoint, defining what to implement, while technical variability relates to product family engineering, defining how to implement it. A classification based on the dimensions of variability is proposed by Pohl *et al.* [96]: beyond **variability in time** (existence of different versions of an artifact that are valid at different times) and **variability in space** (existence of an artifact in different shapes at the same time) Pohl *et al.* claim that variability is important to different stakeholders and thus has different levels of visibility: **external variability** is visible to the customers while **internal variability**, that of domain artifacts, is hidden from them. Other classification proposals come from Meekel *et al.* [84] (feature, hardware platform, performances and attributes variability) or Bass *et al.* [52] who discuss about variability at the architectural level.

Central to the modeling of variability is the notion of *feature*, originally defined by Kang *et al.* as: *a prominent or distinctive user-visible aspect, quality or characteristic of a software system or systems* [80]. Based on this notion of *feature*, they proposed to use a *feature model* to model the variability in a SPL. A feature model consists of a *feature diagram* and other associated information: *constraints* and *dependency rules*. Feature diagrams provide a *graphical tree-like notation depicting the hierarchical organization of high level product functionalities* represented as features. The root of the tree refers to the complete system and is progressively decomposed into more refined features (tree nodes). Relations between nodes (features) are materialized by *decomposition edges* and *textual constraints*. Variability can be expressed in several ways. Presence or absence of a feature from a product is modeled using *mandatory* or *optional features*. Features are graphically represented as rectangles while some graphical elements (e.g., unfilled circle) are used to describe the variability (e.g., a feature may be optional).

Features can be organized into *feature groups*. Boolean operators *exclusive alternative (XOR)*, *inclusive alternative (OR)* or *inclusive (AND)* are used to select one, several or all the features from a feature group.

Dependencies between features can be modeled using *textual constraints*: *requires* (presence of a feature requires the presence of another), *mutex* (presence of a feature automatically excludes another). Feature attributes can be also used for modeling quantitative (e.g., numerical) information. Constraints over attributes and features can be specified as well.

Modeling variability allows an organization to capture and select which version of which variant of any particular aspect is wanted in the system [61]. To implement it cheaply, quickly and safely, redoing by hand the tedious weaving of every aspect is not an option: some form of automation is needed to leverage the modeling of variability [56], [68]. Model Driven Engineering (MDE) makes it possible to automate this weaving process [79]. This requires that models are no longer informal, and that the weaving process is itself described as a program (which is as a matter of facts an executable meta-model [87]) manipulating these models to produce for instance a detailed design that can ultimately be transformed to code, or to test suites [95], or other software artifacts.

3.1.3. Component-based software development

Component-based software development [105] aims at providing reliable software architectures with a low cost of design. Components are now used routinely in many domains of software system designs: distributed systems, user interaction, product lines, embedded systems, etc. With respect to more traditional software artifacts (e.g., object oriented architectures), modern component models have the following distinctive features [67]: description of requirements on services required from the other components; indirect connections between components thanks to ports and connectors constructs [82]; hierarchical definition of components (assemblies of components can define new component types); connectors supporting various communication semantics [64]; quantitative properties on the services [59].

In recent years component-based architectures have evolved from static designs to dynamic, adaptive designs (e.g., SOFA [64], Palladio [57], Frascati [88]). Processes for building a system using a statically designed architecture are made of the following sequential lifecycle stages: requirements, modeling, implementation, packaging, deployment, system launch, system execution, system shutdown and system removal. If for any reason after design time architectural changes are needed after system launch (e.g., because requirements changed, or the implementation platform has evolved, etc) then the design process must be reexecuted from scratch (unless the changes are limited to parameter adjustment in the components deployed).

Dynamic designs allow for *on the fly* redesign of a component based system. A process for dynamic adaptation is able to reapply the design phases while the system is up and running, without stopping it (this is different from stop/redeploy/start). This kind of process supports *chosen adaptation*, when changes are planned and realized to maintain a good fit between the needs that the system must support and the way it supports them [81]. Dynamic component-based designs rely on a component meta-model that supports complex life cycles for components, connectors, service specification, etc. Advanced dynamic designs can also take platform changes into account at run-time, without human intervention, by adapting themselves [65], [108]. Platform changes and more generally environmental changes trigger *imposed adaptation*, when the system can no longer use its design to provide the services it must support. In order to support an eternal system [58], dynamic component based systems must separate architectural design and platform compatibility. This requires support for heterogeneity, since platform evolutions can be partial.

The Models@runtime paradigm denotes a model-driven approach aiming at taming the complexity of dynamic software systems. It basically pushes the idea of reflection one step further by considering the reflection layer as a real model “something simpler, safer or cheaper than reality to avoid the complexity, danger and irreversibility of reality [99]”. In practice, component-based (and/or service-based) platforms offer reflection APIs that make it possible to introspect the system (which components and bindings are currently in place in the system) and dynamic adaptation (by applying CRUD operations on these components and bindings). While some of these platforms offer rollback mechanisms to recover after an erroneous adaptation, the idea of Models@runtime is to prevent the system from actually enacting an erroneous adaptation. In other words, the “model at run-time” is a reflection model that can be uncoupled (for reasoning, validation, simulation purposes) and automatically resynchronized.

Heterogeneity is a key challenge for modern component based system. Until recently, component based techniques were designed to address a specific domain, such as embedded software for command and control, or distributed Web based service oriented architectures. The emergence of the Internet of Things paradigm calls for a unified approach in component based design techniques. By implementing an efficient separation of concern between platform independent architecture management and platform dependent implementations, *Models@runtime* is now established as a key technique to support dynamic component based designs. It provides DIVERSE with an essential foundation to explore an adaptation envelop at run-time.

Search Based Software Engineering [76] has been applied to various software engineering problems in order to support software developers in their daily work. The goal is to automatically explore a set of alternatives and assess their relevance with respect to the considered problem. These techniques have been applied to craft software architecture exhibiting high quality of services properties [73]. Multi Objectives Search based techniques [70] deal with optimization problem containing several (possibly conflicting) dimensions to optimize. These techniques provide DIVERSE with the scientific foundations for reasoning and efficiently exploring an envelope of software configurations at run-time.

3.1.4. Validation and verification

Validation and verification (V&V) theories and techniques provide the means to assess the validity of a software system with respect to a specific correctness envelop. As such, they form an essential element of DIVERSE's scientific background. In particular, we focus on model-based V&V in order to leverage the different models that specify the envelop at different moments of the software development lifecycle.

Model-based testing consists in analyzing a formal model of a system (*e.g.*, activity diagrams, which capture high-level requirements about the system, statecharts, which capture the expected behavior of a software module, or a feature model, which describes all possible variants of the system) in order to generate test cases that will be executed against the system. Model-based testing [107] mainly relies on model analysis, constraint solving [69] and search-based reasoning [83]. DIVERSE leverages in particular the applications of model-based testing in the context of highly-configurable systems and [109] interactive systems [85] as well as recent advances based on diversity for test cases selection [77].

Nowadays, it is possible to simulate various kinds of models. Existing tools range from industrial tools such as Simulink, Rhapsody or Telelogic to academic approaches like Omega [93], or Xholon⁰. All these simulation environments operate on homogeneous environment models. However, to handle diversity in software systems, we also leverage recent advances in heterogeneous simulation. Ptolemy [63] proposes a common abstract syntax, which represents the description of the model structure. These elements can be decorated using different directors that reflect the application of a specific model of computation on the model element. Metropolis [53] provides modeling elements amenable to semantically equivalent mathematical models. Metropolis offers a precise semantics flexible enough to support different models of computation. ModHel'X [75] studies the composition of multi-paradigm models relying on different models of computation.

Model-based testing and simulation are complemented by runtime fault-tolerance through the automatic generation of software variants that can run in parallel, to tackle the open nature of software-intensive systems. The foundations in this case are the seminal work about N-version programming [51], recovery blocks [97] and code randomization [55], which demonstrated the central role of diversity in software to ensure runtime resilience of complex systems. Such techniques rely on truly diverse software solutions in order to provide systems with the ability to react to events, which could not be predicted at design time and checked through testing or simulation.

3.1.5. Empirical software engineering

The rigorous, scientific evaluation of DIVERSE's contributions is an essential aspect of our research methodology. In addition to theoretical validation through formal analysis or complexity estimation, we also aim at applying state-of-the-art methodologies and principles of empirical software engineering. This approach encompasses a set of techniques for the sound validation contributions in the field of software engineering,

⁰<http://www.primordion.com/Xholon/>

ranging from statistically sound comparisons of techniques and large-scale data analysis to interviews and systematic literature reviews [102], [100]. Such methods have been used for example to understand the impact of new software development paradigms [62]. Experimental design and statistical tests represent another major aspect of empirical software engineering. Addressing large-scale software engineering problems often requires the application of heuristics, and it is important to understand their effects through sound statistical analyses [50].

3.2. Research axis

Figure 1 illustrates the four dimensions of software diversity, which form the core research axis of DIVERSE: the **diversity of languages** used by the stakeholders involved in the construction of these systems; the **diversity of features** required by the different customers; the **diversity of runtime environments** in which software has to run and adapt; the **diversity of implementations** that are necessary for resilience through redundancy. These four axis share and leverage the scientific and technological results developed in the area of model-driven engineering in the last decade. This means that all our research activities are founded on sound abstractions to reason about specific aspects of software systems, compose different perspectives and automatically generate parts of the system.

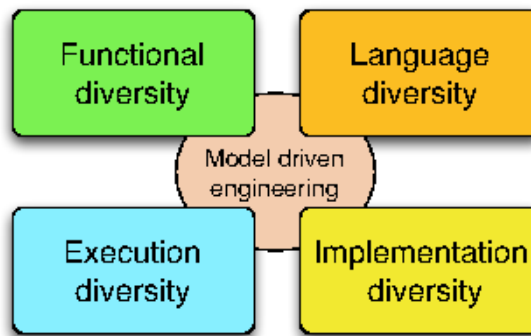


Figure 1. The four research axis of DIVERSE, which rely on a MDE scientific background

3.2.1. Software Language Engineering

The engineering of systems involves many different stakeholders, each with their own domain of expertise. Hence more and more organizations are adopting Domain Specific Modeling Languages (DSMLs) to allow domain experts to express solutions directly in terms of relevant domain concepts [101], [72]. This new trend raises new challenges about designing DSMLs, evolving a set of DSMLs and coordinating the use of multiple DSLs for both DSL designers and DSL users.

3.2.1.1. Challenges

Reusability of software artifacts is a central notion that has been thoroughly studied and used by both academics and industrials since the early days of software construction. Essentially, designing reusable artifacts allows the construction of large systems from smaller parts that have been separately developed and validated, thus reducing the development costs by capitalizing on previous engineering efforts. However, it is still hardly possible for language designers to design typical language artifacts (e.g. language constructs, grammars, editors or compilers) in a reusable way. The current state of the practice usually prevents the reusability of language artifacts from one language to another, consequently hindering the emergence of real engineering techniques around software languages. Conversely, concepts and mechanisms that enable artifacts reusability abound in the software engineering community.

Variability in modeling languages occur in the definition of the abstract and concrete syntax as well as in the specification of the language's semantics. The major challenges met when addressing the need for variability are: (i) set principles for modeling language units that support the modular specification of a modeling language; and (ii) design mechanisms to assemble these units in a complete language, according to the set of authorized variation points for the modeling language family.

A new generation of complex software-intensive systems (for example smart health support, smart grid, building energy management, and intelligent transportation systems) presents new opportunities for leveraging modeling languages. The development of these systems requires expertise in diverse domains. Consequently, different types of stakeholders (e.g., scientists, engineers and end-users) must work in a coordinated manner on various aspects of the system across multiple development phases. DSMLs can be used to support the work of domain experts who focus on a specific system aspect, but they can also provide the means for coordinating work across teams specializing in different aspects and across development phases. The support and integration of DSMLs leads to what we call **the globalization of modeling languages**, *i.e.* the use of multiple languages for the coordinated development of diverse aspects of a system. One can make an analogy with world globalization in which relationships are established between sovereign countries to regulate interactions (e.g., travel and commerce related interactions) while preserving each country's independent existence.

3.2.1.2. Scientific objectives

We address reuse and variability challenges through the investigation of the time-honored concepts of substitutability, inheritance and components, evaluate their relevance for language designers and provide tools and methods for their inclusion in software language engineering. We will develop novel techniques for the modular construction of language extensions with the support of model syntactical variability. From the semantics perspective, we investigate extension mechanisms for the specification of variability in operational semantics, focusing on static introduction and heterogeneous models of computation. The definition of variation points for the three aspects of the language definition provides the foundations for the novel concept Language Unit (LU) as well as suitable mechanisms to compose such units.

We explore the necessary breakthrough in software languages to support modeling and simulation of heterogeneous and open systems. This work relies on the specification of executable domain specific modeling languages (DSMLs) to formalize the various concerns of a software-intensive system, and of models of computation (MoCs) to explicitly model the concurrency, time and communication of such DSMLs. We develop a framework that integrates the necessary foundations and facilities for designing and implementing executable and concurrent domain-specific modeling languages. It also provides unique features to specify composition operators between (possibly heterogeneous) DSMLs. Such specifications are amenable to support the edition, execution, graphical animation and analysis of heterogeneous models. The objective is to provide both a significant improvement of MoCs and DSMLs design and implementation; and the simulation based validation and verification of complex systems.

We see an opportunity for the automatic diversification of programs' computation semantics, for example through the diversification of compilers or virtual machines. The main impact of this artificial diversity is to provide flexible computation and thus ease adaptation to different execution conditions. A combination of static and dynamic analysis could support the identification of what we call *plastic computation zones* in the code. We identify different categories of such zones: (i) areas in the code in which the order of computation can vary (e.g., the order in which a block of sequential statements is executed); (ii) areas that can be removed, keeping the essential functionality [103] (e.g., skip some loop iterations); (iii) areas that can be replaced by alternative code (e.g., replace a try-catch by a return statement). Once we know which zones in the code can be randomized, it is necessary to modify the model of computation to leverage the computation plasticity. This consists in introducing variation points in the interpreter to reflect the diversity of models of computation. Then, the choice of a given variation is performed randomly at run-time.

3.2.2. Variability Modeling and Engineering

The systematic modeling of variability in software systems has emerged as an effective approach to document and reason about software evolutions and heterogeneity (*cf.* Section 3.1.2). Variability modeling character-

izes an “envelope” of possible software variations. The industrial use of variability models and their relation to software artifact models require a complete engineering framework, including composition, decomposition, analysis, configuration and artifact derivation, refactoring, re-engineering, extraction, and testing. This framework can be used both to tame imposed diversity and to manage chosen diversity.

3.2.2.1. Challenges

A fundamental problem is that the **number of variants** can be exponential in the number of options (features). Already with 300 boolean configuration options, approximately 10^{90} configurations exist – more than estimated count of atoms in the universe. Domains like automotive or operating systems have to manage more than 10000 options (e.g., Linux). Practitioners face the challenge of developing billions of variants. It is easy to forget a necessary constraint, leading to the synthesis of unsafe variants, or to under-approximate the capabilities of the software platform. Scalable modelling techniques are therefore crucial to specify and reason about a very large set of variants.

Model-driven development supports two ways to deal with the increasing number of concerns in complex systems: (1) multi-view modeling, *i.e.* when modeling each concern separately, and variability modeling. However, there is little support to combine both approaches consistently. Techniques to integrate both approaches will enable the construction of a consistent set of views and variation points in each view.

The design, construction and maintenance of software families have a major impact on **software testing**. Among the existing challenges, we can cite: the selection of test cases for a specific variant; the evolution of test suites with integration of new variants; the combinatorial explosion of the number of software configurations to be tested. Novel model-based techniques for test generation and test management in a software product line context are needed to overcome state-of-the-art limits we already observed in some projects.

3.2.2.2. Scientific objectives

We aim at developing scalable techniques to automatically analyze variability models and their interactions with other views on the software intensive system (requirements, architecture, design). These techniques provide two major advancements in the state of the art: (1) an extension of the semantics of variability models in order to enable the definition of attributes (*e.g.*, cost, quality of service, effort) on features and to include these attributes in the reasoning; (2) an assessment of the consistent specification of variability models with respect to system views (since variability is orthogonal to system modeling, it is currently possible to specify the different models in ways that are semantically meaningless). The former aspect of analysis is tackled through constraint solving and finite-domain constraint programming, while the latter aspect is investigated through automatic search-based techniques (similar to genetic algorithms) for the exploration of the space of interaction between variability and view models.

We aim to develop procedures to reverse engineer dependencies and features’ sets from existing software artefacts – be it source code, configuration files, spreadsheets (*e.g.*, product comparison matrices) or requirements. We expect to scale up (*e.g.*, for extracting a very large number of variation points) and guarantee some properties (*e.g.*, soundness of configuration semantics, understandability of ontological semantics). For instance, when building complex software-intensive systems, textual requirements are captured in very large quantities of documents. In this context, adequate models to formalize the organization of requirements documents and automated techniques to support impact analysis (in case of changes in the requirements) have to be developed.

We aim at developing sound methods and tools to integrate variability management in model-based testing activities. In particular, we will leverage requirement models as an essential asset to establish formal relations between variation points and test models. These relations will form the basis for novel algorithms that drive the systematic selection of test configurations that satisfy well-defined test adequacy criteria as well as the generation of test cases for a specific product in the product line.

3.2.3. Heterogeneous and dynamic software architectures

Flexible yet dependable systems have to cope with heterogeneous hardware execution platforms ranging from smart sensors to huge computation infrastructures and data centers. Evolutions range from a mere change in the system configuration to a major architectural redesign, for instance to support addition of new features

or a change in the platform architecture (new hardware is made available, a running system switches to low bandwidth wireless communication, a computation node battery is running low, etc). In this context, we need to devise formalisms to reason about the impact of an evolution and about the transition from one configuration to another. It must be noted that this axis focuses on the use of models to drive the evolution from design time to run-time. Models will be used to (i) systematically define predictable configurations and variation points through which the system will evolve; (ii) develop behaviors necessary to handle unpredicted evolutions.

3.2.3.1. Challenges

The main challenge is to provide new homogeneous architectural modelling languages and efficient techniques that enable continuous software reconfiguration to react to changes. This work handles the challenges of handling the diversity of runtime infrastructures and managing the cooperation between different stakeholders. More specifically, the research developed in this axis targets the following dimensions of software diversity.

Platform architectural heterogeneity induces a first dimension of imposed diversity (type diversity). Platform reconfigurations driven by changing resources define another dimension of diversity (deployment diversity). To deal with these imposed diversity problems, we will rely on model based runtime support for adaptation, in the spirit of the dynamic distributed component framework developed by the Triskell team. Since the runtime environment composed of distributed, resource constrained hardware nodes cannot afford the overhead of traditional runtime adaptation techniques, we investigate the design of novel solutions relying on models@runtime and on specialized tiny virtual machines to offer resource provisioning and dynamic reconfigurations. In the next two years this research will be supported by the InfraJVM project.

Diversity can also be an asset to optimize software architecture. Architecture models must integrate multiple concerns in order to properly manage the deployment of software components over a physical platform. However, these concerns can contradict each other (*e.g.*, accuracy and energy). In this context, we investigate automatic solutions to explore the set of possible architecture models and to establish valid trade-offs between all concerns in case of changes.

3.2.3.2. Scientific objectives

Automatic synthesis of optimal software architectures. Implementing a service over a distributed platform (*e.g.*, a pervasive system or a cloud platform) consists in deploying multiple software components over distributed computation nodes. We aim at designing search-based solutions to (i) assist the software architect in establishing a good initial architecture (that balances between different factors such as cost of the nodes, latency, fault tolerance) and to automatically update the architecture when the environment or the system itself change. The choice of search-based techniques is motivated by the very large number of possible software deployment architectures that can be investigated and that all provide different trade-offs between qualitative factors. Another essential aspect that is supported by multi-objective search is to explore different architectural solutions that are not necessarily comparable. This is important when the qualitative factors are orthogonal to each other, such as security and usability for example.

Flexible software architecture for testing and data management. As the number of platforms on which software runs increases and different software versions coexist, the demand for testing environments also increases. For example, to test a software patch or upgrade, the number of testing environments is the product of the number of running environments the software supports and the number of coexisting versions of the software. Based on our first experiment on the synthesis of cloud environment using architectural models, our objective is to define a set of domain specific languages to catch the requirement and to design cloud environments for testing and data management of future internet systems from data centers to things. These languages will be interpreted to support dynamic synthesis and reconfiguration of a testing environment.

Runtime support for heterogeneous environments. Execution environments must provide a way to account or reserve resources for applications. However, current execution environments such as the Java Virtual Machine do not clearly define a notion of application: each framework has its own definition. For example, in OSGi, an application is a component, in JEE, an application is most of the time associated to a class loader, in the Multi-Tasking Virtual machine, an application is a process. The challenge consists in defining an execution environment that provides direct control over resources (CPU, Memory, Network I/O) independently from the

definition of an application. We propose to define abstract resource containers to account and reserve resources on a distributed network of heterogeneous devices.

3.2.4. Diverse implementations for resilience

Open software-intensive systems have to evolve over their lifetime in response to changes in their environment. Yet, most verification techniques assume a closed environment or the ability to predict all changes. Dynamic changes and evolutions thus represent a major challenge for these techniques that aim at assessing the correctness and robustness of the system. On the one hand, DIVERSE will adapt V&V techniques to handle diversity imposed by the requirements and the execution environment, on the other hand we leverage diversity to increase the robustness of software in face of unpredicted situations. More specifically, we address the following V&V challenges.

3.2.4.1. Challenges

One major challenge to build flexible and open yet dependable systems is that current software engineering techniques require architects to foresee all possible situations the system will have to face. However, openness and flexibility also mean unpredictability: unpredictable bugs, attacks, environmental evolutions, etc. Current fault-tolerance [97] and security [71] techniques provide software systems with the capacity of detecting accidental and deliberate faults. However, existing solutions assume that the set of bugs or vulnerabilities in a system does not evolve. This assumption does not hold for open systems, thus it is essential to revisit fault-tolerance and security solutions to account for diverse and unpredictable faults.

Diversity is known to be a major asset for the robustness of large, open, and complex systems (*e.g.*, economical or ecological systems). Following this observation, the software engineering literature provides a rich set of work that choose to implement diversity in software systems in order to improve robustness to attacks or to changes in quality of service. These works range from N-version programming to obfuscation of data structures or control flow, to randomization of instruction sets. An essential remaining challenge is to support the automatic synthesis and evolution of software diversity in open software-intensive systems. There is an opportunity to further enhance these techniques in order to cope with a wider diversity of faults, by multiplying the levels of diversity in the different software layers that are found in software-intensive systems (system, libraries, frameworks, application). This increased diversity must be based on artificial program transformations and code synthesis, which increase the chances of exploring novel solutions, better fitted at one point in time. The biological analogy also indicates that diversity should emerge as a side-effect of evolution, to prevent over-specialization towards one kind of diversity.

3.2.4.2. Scientific objectives

The main objective is to address one of the main limitations of N-version programming for fault-tolerant systems: the manual production and management of software diversity. Through automated injection of artificial diversity we aim at systematically increasing failure diversity and thus increasing the chances of early error detection at run-time. A fundamental assumption for this work is that software-intensive systems can be “good enough” [98], [110].

Proactive program diversification. We aim at establishing novel principles and techniques that favor the emergence of multiple forms of software diversity in software-intensive systems, in conjunction with the software adaptation mechanisms that leverage this diversity. The main expected outcome is a set of meta-design principles that maintain diversity in systems and the experimental demonstration of the effects of software diversity on the adaptive capacities of CASs. Higher levels of diversity in the system provide a pool of software solutions that can eventually be used to adapt to situations unforeseen at design time (bugs, crash, attacks, etc.). Principles of automated software diversification rely on the automated synthesis of variants in a software product line, as well as finer-grained program synthesis combining unsound transformations and genetic programming to explore the space of mutational robustness.

Multi-tier software diversification. We call multi-tier diversification the fact of diversifying several application software components simultaneously. The novelty of our proposal, with respect to the software diversity state of the art, is to diversify the application-level code (for example, diversify the business logics of the application), focusing on the technical layers found in web applications. The diversification of application software

code is expected to provide a diversity of failures and vulnerabilities in web server deployment. Web server deployment usually adopts a form of the Reactor architecture pattern, for scalability purposes: multiple copies of the server software stack, called request handlers, are deployed behind a load balancer. This architecture is very favorable for diversification, since by using the multiplicity of request handlers running in a web server we can simultaneously deploy multiple combinations of diverse software components. Then, if one handler is hacked or crashes the others should still be able to process client requests.

DYLISS Project-Team

3. Research Program

3.1. Modeling knowledge integration with combinatorial constraints

Biological networks are built with data-driven approaches aiming at translating genomic information into a functional map. Most methods are based on a probabilistic framework which defines a probability distribution over the set of models. The reconstructed network is then defined as the most likely model given the data.

Our team has investigated an alternative perspective where each data induces a set of constraints – related to the steady state response of the system dynamics – on the set of possible values in a network of fixed topology. The methods that we have developed complete the network with product states at the level of nodes and influence types at the level of edges, able to globally explain experimental data. In other words, the selection of relevant information in the model is no more performed by selecting *the* network with the highest score, but rather by exploring the complete space of models satisfying constraints on the possible dynamics supported by prior knowledge and observations. In the (common) case when there is no model satisfying all the constraints, we relax the problem by introducing new combinatorial optimization problems that introduce the possibility of correcting the data or the knowledge. Common properties to all solutions are considered as a robust information about the system, as they are independent from the choice of a single solution to the optimization problem [77].

Solving these computational issues requires addressing NP-hard qualitative (non-temporal) issues. We have developed a long-term collaboration with Potsdam University in order to use a logical paradigm named **Answer Set Programming** (ASP) [46], [73] to solve these constraint satisfiability and combinatorial optimization issues. Applied on transcriptomic or cancer networks, our methods identified which regions of a large-scale network shall be corrected [47], and proposed robust corrections [6]. This result suggested that this approach was compatible with efficiency, scale and expressivity needed by biological systems.

During the last years, our goal was to provide **formal models of queries on biological networks** with the focus of integrating dynamical information as explicit logical constraints in the modeling process. Using these technologies requires to revisit and reformulate constraint-satisfiability problems at hand in order both to decrease the search space size in the grounding part of the process and to improve the exploration of this search space in the solving part of the process. Concretely, getting logical encoding for the optimization problems forces to clarify the roles and dependencies between parameters involved in the problem. This paves the way to a refinement approach based on a fine investigation of the space of hypotheses in order to make it smaller and gain in the understanding of the system. Our studies confirmed that logical paradigms are a powerful approach to build and query reconstructed biological systems, in complement to discriminative ("black-box") approaches based on statistical machine-learning. Based on these technologies, we have developed a panel of methods allowing the integration of multi-scale data knowledge, linking genomics, metabolomics, expression data and protein measurement of several phenotypes.

Notice that our main issue is in the field of knowledge representation. More precisely, we do not wish to develop new solvers or grounders, a self-contained computational issue which is addressed by specialized teams such as our collaborator team in Potsdam. Our goal is rather to investigate how the constant progresses in the field of constraint logical programming, shown by the performance of ASP-solvers, are sufficient to address the complexity of constraint-satisfiability and combinatorial optimization issues explored in systems biology. In this direction, we work in close interaction with Potsdam university to feed their research activities with challenging issues from bioinformatics and, as a feed-back, take benefit of the prototypes they develop.

By exploring the complete space of models, our approach typically produces numerous candidate models compatible with the observations. We began investigating to what extent domain knowledge can further refine the analysis of the set of models by identifying classes of similar models, or by selecting a subset of models that satisfy an additional constraint (for instance, best fit with a set of experiments, or with a minimal size). We anticipate that this will be particularly relevant when studying non-model species for which little is known but valuable information from other species can be transposed or adapted. These efforts consist in developing reasoning methods based on ontologies as formal representation of symbolic knowledge. We use Semantic Web tools such as SPARQL for querying and integrating large sources of external knowledge, and measures of semantic similarity and particularity for analyzing data.

3.2. Modeling the dynamical response of biological systems with logical and (non)-linear constraints

As explained below, Answer Set Programming technologies enable the identification of key controllers based on the integration of static data. As a natural follow-up, we also develop optimization techniques to learn models of the dynamics of a biological system. As before, our strategy is not to select a single model fitting with experimental data but rather to decipher the complete set of families of models which are compatible with the observed response. Our main research line in this field is to decipher the appropriate level of expressivity (in terms of constraints) allowing both to properly report the nature of data and knowledge and to allow for an exhaustive study of the space of feasible models. To implement this strategy, we rely on several constraint programming frameworks, which depend on the model scale and the nature of time-points kinetic measurements. Logical programming (Answer Set Programming) is used to decipher the combinatorics of synchronic Boolean networks explaining static or dynamics response of signaling networks to perturbations (such as measured by phosphoproteomics technologies) [7]. SAT-based approaches are used to decipher the combinatorics of large-scale asynchronous boolean networks. In order to gain in expressivity, we model these networks as guarded-transition network, an extension of Petri nets [40]. Finally, classical learning methods are used to build ad-hoc parameterized numerical models that provide the most parsimonious explanations to experimental measurements.

3.3. Modeling sequences with formal grammars

Once groups of genome products involved in the answer of the species have been identified with integrative or dynamical methods, it remains to characterize the biological actors within genomes. To that goal, we both learn, model and parse formal patterns within DNA, RNA or protein sequences. More precisely, our research on modeling biomolecular sequences with expressive formal grammars focuses on learning such grammars from examples, helping biologists to design their own grammar and providing practical parsing tools.

On the development of **machine learning** algorithms for the induction of grammatical models [66], we have a strong expertise on learning finite state automata. We have proposed an algorithm that learns successfully automata modeling families of (non homologous) functional families of proteins [5], leading to a tool named Protomata-learner. The algorithm is based on a similar fragment merging heuristic approach which reports partial and local alignments contained in a family of sequences. As an example, this tool allowed us to properly model the TNF protein family, a difficult task for classical probabilistic-based approaches. It was also applied successfully to model important enzymatic families of proteins in cyanobacteria [4]. Our future goal is to further demonstrate the relevance of formal language modeling by addressing the question of a fully automatic prediction from the sequence of all the enzymatic families, aiming at improving even more the sensitivity and specificity of the models. As enzyme-substrate interactions are very specific central relations for integrated genome/metabolome studies and are characterized by faint signatures, we shall rely on models for active sites involved in cellular regulation or catalysis mechanisms. This requires to build models gathering both structural and sequence information in order to describe (potentially nested or crossing) long-term dependencies such as contacts of amino-acids that are far in the sequence but close in the 3D protein folding. Our current researches is focused on the inference of Context-Free Grammars including the topological information coming from the structural characterization of active sites.

Using context-free grammars instead of regular patterns increases the complexity of **parsing** issues. Indeed, efficient parsing tools have been developed to identify patterns within genomes but most of them are restricted to simple regular patterns. Definite Clause Grammars (DCG), a particular form of logical context-free grammars have been used in various works to model DNA sequence features [86]. An extended formalism, String Variable Grammars (SVGs), introduces variables that can be associated to a string during a pattern search [110], [109]. This increases the expressivity of the formalism towards mildly context sensitive grammars. Thus, those grammars model not only DNA/RNA sequence features but also structural features such as repeats, palindromes, stem/loop or pseudo-knots. A few years ago, we have designed a first tool, STAN (suffix-tree analyser), in order to make it possible to search for a subset of SVG patterns in full chromosome sequences [92]. This tool was used for the recognition of transposable elements in *Arabidopsis thaliana* [113]. We have enlarged this experience through a new modeling language, called Logol [1]. Generally, a suitable language for the search of particular components in languages has to meet several needs : expressing existing structures in a compact way, using existing databases of motifs, helping the description of interacting components. In other words, the difficulty is to find a good tradeoff between expressivity and complexity to allow the specification of realistic models at genome scale. The Logol language and associated framework have been built in this direction. The Logol specificity besides other SVG-like languages mainly lies in a systematic introduction of constraints on string variables.

3.4. Symbolic methods for model space exploration: Semantic web for life sciences and Formal Concepts Analysis

All the methods presented in the previous sections usually result in pools of candidates which equivalently explain the data and knowledge. These candidates can be dynamical systems, compounds, biological sequences, proteins... In any case, the output of our formal methods generally requires *a posteriori* investigation and filtering by domain experts. In order to assist them, we rely on two classes of symbolic technics: Semantic Web technologies and Formal Concept Analysis (FCA). They both aim at the formalization and management of knowledge, that is, the explicitation of relations occurring in structured data. These technics complement each other: the production of relevant concepts in FCA highly depends on the availability of semantic annotations using a controlled set of terms and conversely, building and exploiting ontologies is a complex process that can be made much easier with FCA.

Integrating heterogenous data with semantic web technologies The emergence of ontologies in biomedical informatics and bioinformatics happened in parallel with the development of the **Semantic Web** in the computer science community [108]. Let us recall that the Semantic Web is an extension of the current Web that provides an infrastructure integrating data and ontologies in order to support unified reasoning. Since the beginning, life sciences have been a major application domain for the Semantic Web [48]. This was motivated by the joint evolution of data acquisition capabilities in the biomedical field, and of the methods and infrastructures supporting data analysis (grids, the Internet...), resulting in an explosion of data production in complementary domains [58], [49]. Consequently, Semantic Web technologies have become an integral part of translational medicine and translational bioinformatics [63]. The Linked Open Data project promotes the integration of data sources in machine-processable formats compatible with the Semantic Web [57], with a strong involvement of life sciences in this initiative.

However, a specificity of life sciences “data deluge” is that the proportion of generated data is much higher than in the more general “big data phenomenon”, and that these data are highly connected [112]. **The bottleneck that once was data scarcity now lies in the lack of adequate methods supporting data integration, processing and analysis** [88]. Each of these steps typically hinges on domain knowledge, which is why they resist automation. This knowledge can be seen as the set of rules representing in what conditions data can be used or can be combined for inferring new data or new links between data.

In this setting, we are working on the integration of Semantic Web resources with our data analysis methods in order to take existing biological knowledge into account. We have introduced several methods to interpret semantic similarities and particularities [56], [55]. We now focus our attention on the semi-automated construction of RDF abstractions of heterogeneous datasets which can be handled by non-expert users. This

allows both to automatically prepare input datasets for the other methods developed in the team and to analyse the output of the methods in a wide knowledge context.

Using Formal concept analysis to explore the results of bioinformatics analyses Formal concept analysis aims at the development of conceptual structures which can be logically activated for the formation of judgments and conclusions [117]. It is used in various domains managing structured data such as knowledge processing, information retrieval or classification [90]. In its most simple form, one considers a binary relation between a set of objects and a set of attributes. In this setting, formal concept analysis formalizes the semantic notions of extension and intension. Concepts are related within a lattice structure (Galois connection) by subconcept-superconcept relations, and this allows drawing causality relations between attribute subsets. In bioinformatics, it has been used to derive phylogenetic relations among groups of organisms [87], a classification task that requires to take into account many-valued Galois connections. We have proposed in a similar way a classification scheme for the problem of protein assignment in a set of protein families [67].

One of the most important issue with concept analysis is due to the fact that current methods remain very sensitive to the presence of uncertainty or incompleteness in data. On the other hand, this apparent defect can be reversed to serve as a marker of incompleteness or inconsistency [68]. Following this inspiration, we have proposed a methodology to tackle the problem of uncertainty on biological networks where edges are mostly predicted links with a high level of false positives [118]. The general idea consists to look for a tradeoff between the simplicity of the conceptual representation and the need to manage exceptions. As a very prospective challenge, we are exploring the idea of using ontologies to help this or to help ontology refinement using concept analysis [93], [52], [102].

More generally, common difficult tasks in this context are visualization, search for local structures (graph mining) and network comparison. Network compression is a good solution for an efficient treatment of all these tasks. This has been used with success in power graphs, which are abstract graphs where nodes are clusters of nodes in the initial graph and edges represent bicliques between two sets of nodes [104]. In fact, concepts are maximal bicliques and we are currently developing the power graph idea in the framework of concept analysis.

3.5. Implementing methods in software and platforms

Seven platforms have been developped in the team for the last five years: Askomics, AuReMe, FinGoc, Caspo, Cadiom, Logol, Protomata. Indeed, one of the team's goals is to facilitate interplays between the tools for biological data analysis and integration. Improvements and novelties of these platforms are described in the "software" section. Our platforms aim at guiding the user to progressively reduce the space of models (families of sequences of genes or proteins, families of keys actors involved in a system response, dynamical models) which are compatible with both knowledge and experimental observations.

Most of our platforms are developed with the support of the GenOuest resource and data center hosted in the IRISA laboratory, including their computer facilities [more info]. It worths considering them into larger dedicated environments to benefit from the expertise of other research groups. The **BioShaddock** repository of the GenOuest platform allows one to share the different docker containers that we are developing [website]. The **GenOuest galaxy portal** of the GenOuest platform now provides access to most tools for integrative biology and sequence annotation (access on demand).

3.5.1. AskOmics platform

Goal Integration and interrogation software for linked biological data based on semantic web technologies [url].

Description *AskOmics* aims at bridging the gap between end user data and the Linked (Open) Data cloud. It allows heterogeneous bioinformatics data (formatted as tabular files or directly in RDF) to be loaded into a Triple Store system using a user-friendly web interface. *AskOmics* also provides an intuitive graph-based user interface supporting the creation of complex queries that currently require hours of manual searches across tens of spreadsheet files. The elements of interest selected in the graph are then automatically converted into a SPARQL query that is executed on the users's data.

Originality Our experience is that end users (i) do not benefit for all the information available in the LOD cloud repositories by lack of SPARQL expertise (understandably: they are biologists and most of them do not have an interest in either learning SPARQL nor in learning how to integrate data); (ii) do not contribute their data back to the LOD cloud. Again, they do not have the expertise nor the resources to produce and maintain datasets and the associated metadata as linked data, nor to maintain the underlying server infrastructure. Therefore there is a need for helping end users to (1) take advantage of the information readily available in the LOD cloud for analyzing their own data and (2) contribute back to the linked data by representing their data and the associated metadata in the proper format as well as by linking them to other resources. In this context, the main originality is the graphical interface that allows any SPARQL query to be built transparently and iteratively by a non-expert user.

Application This software was developed in the context of the MirnAdapt (pea-aphid) project in 2016. The tool has been presented to the agriculture communities in conferences [53], [84] and to the Galaxy community [29]. Up to now, more than 10 biological partners team are actually testing and using the prototype software (colza, pea-aphids, copper microbiology, marine biology), and SANOFI has shown its interest to co-develop the tool. Even if its current user base belongs to the bioinformatics community, the scope of *AskOmics* is domain-independent and has the potential to reach a wider audience related to the Semantic Web community.

3.5.2. *AuReMe workspace*

Goal Tractable reconstruction of metabolic networks [url].

Description The toolbox *AuReMe* allows for the Automatic Reconstruction of Metabolic networks based on the combination of multiple heterogeneous data and knowledge sources [64]. It is available as a Docker image. Five modules are composing *AuReMe*: 1) The Model-management PADmet module allows manipulating and tracing all metabolic data via a local database. [package] 2) The meneco python package allows the gaps of a metabolic network to be filled by using a topological approach that implements a logical programming approach to solve a combinatorial problem [107], [65] and [21] [python package] 3) The shogen python package allows genome and metabolic network to be aligned in order to identify genome units which contain a large density of genes coding for enzymes; it also implements a logical programming approach [60] [python package]. 4) The manual curation assistance PADmet module allows the reported metabolic networks and their metadata to be curated. 5) The Wiki-export PADmet module enables the export of the metabolic network and its functional genomic unit as a local wiki platform allowing a user-friendly investigation [package].

Originality The main added-values are the inclusion of graph-based gap-filling tools that are particularly relevant for the study of non-classical organisms, the possibility to trace the reconstruction and curation procedures, and the representation and exploration of reconstructed metabolic networks with wikis.

Application The tools included in *AuReMe* have been used for reconstructing metabolic networks of micro and macro-algae [97], extremophile bacteria [16] and communities of organisms [61] in the context of the Idealg, Ciric-omics and IPL Algae-In-Silico projects.

3.5.3. *FinGoc-tools*

Goal Filtering interaction networks with graph-based optimization criteria.

Description The goal is to offer a set of tools for the reconstruction of networks from genome, literature and large-scale observation data (expression data, metabolomics...) in order to elucidate the main regulators of an observed phenotype. Most of the optimization issues are addressed with Answer Set Programming. 1) The **lombarde package** enables the filtering of transcription-factor/binding-site regulatory networks with mutual information reported by the response to environmental perturbations. The high level of false-positive interactions is filtered according to graph-based criteria. Knowledge about regulatory modules such as operons or the output of the shogen package can be taken into account [39], [38] [web server]. 2) The **KeyRegulatorFinder package** allows searching key regulators of lists of molecules (like metabolites, enzymes or genes) by taking advantage of knowledge databases in cell metabolism and signaling. The complete information is transcribed into a large-scale interaction graph which is filtered to report the most significant upstream regulators of the considered list of molecules [59] [package]. 3) The **powerGrasp**

python package provides an implementation of graph compression methods oriented toward visualization, and based on power graph analysis. [package]. 4) The **iggy package** enables the repairing of an interaction graph with respect to expression data. It proposes a range of different operations for altering experimental data and/or a biological network in order to re-establish their mutual consistency, an indispensable prerequisite for automated prediction. For accomplishing repair and prediction, we take advantage of the distinguished modeling and reasoning capacities of Answer Set Programming. [6] [114] [Python package]

Originality The main added-value of these tools is to make explicit the criteria used to highlight the role of the main regulators: the underlying methods encode explicit graph-based criteria instead of relying on statistical approaches. This makes it possible to explain local relationships and patterns within interaction graphs by explicit biological relationships.

Application The tools have been used to figure out the main gene-regulators of the response of porks to several diets in [74], [76] and [18]. The tools were also used to decipher regulators of reproduction for the pea aphid, an insect that is a pest on plants [85], [119].

3.5.4. *Caspo* software

Participant: Anne Siegel.

Goal Studying synchronous boolean networks [url]

Description Cell ASP Optimizer (*Caspo*) constitutes a pipeline for automated reasoning on logical signaling networks. The main underlying issue is that inherent experimental noise is considered, many different logical networks can be compatible with a set of experimental observations (see [106] and [22]). It is available as a Docker container. Five modules are composing *Caspo*: 1) the *Caspo-learn* module performs an automated inference of logical networks from experimental data allows for identifying admissible large-scale logic models saving a lot of efforts and without any a priori bias [115] and [78]. 2) The *Caspo-classify*, predict and visualize modules allows for classifying a family of boolean networks with respect to their input-output predictions [78]. 3) The *Caspo-design* module designs experimental perturbations which would allow for an optimal discrimination of rival models in a family of boolean networks [116]. 4) The *Caspo-control* module identifies key-players of a family of networks: it computes robust intervention strategies that force a set of target species or compounds into a desired steady state [80]. 5) The *Caspo-timeseries* module to take into account time-series observation datasets in the learning procedure [94] [python package and docker container].

Originality The *Caspo* modules provide friendly and efficient solutions to problems that were previously addressed in theoretical papers with MILP programs. The main advantage is that it enables a complete study of logical network without requiring any linear constraint programs.

Application The *Caspo* tool was initiated in the framework of the BioTempo project. *Caspo-learn* has been included as a module to learn logical networks from early steady-state data in CellNopt, a generic platform which implements several methods for learning and studying signaling networks at different modeling levels (from logical models to numerical models).

3.5.5. *Cadbiom* package

Goal Building and analyzing the asynchronous dynamics of enriched logical networks [url]

Description Based on Guarded transition semantic, the *Cadbiom* software provides a formal framework to help the modeling of biological systems such as cell signaling network. It allows synchronization events to be investigated in biological networks [40]. It is available as a Docker image. Three modules are composing *Cadbiom*: 1) The *Cadbiom* graphical interface is useful to build and study moderate size models. It provides exploration, simulation and checking. For large-scale models, *Cadbiom* also allows to focus on specific nodes of interest. 2) The *Cadbiom* API allows a model to be loaded, performing static analysis and checking temporal properties on a finite horizon in the future or in the past. 3) Exploring large-scale knowledge repositories, the translations of the large-scale PID repository (about 10,000 curated interactions) have been translated into the *Cadbiom* formalism.

Originality Model-checking approaches applied to Boolean networks [81] or multivalued networks [91] allow the trajectories of the system to be entirely studied but they can only be applied to small-size networks. On the contrary, *Cadbiom* is able to handle large-scale knowledge databases.

Application The *Cadbiom* tool was applied to study the regulators of the TGF- β , a gene that controls liver fibrosis [40] in the framework of the *TGFSysBio* project. The study of its predictions also enabled large-scale knowledge databases (PID) to be curated [25].

3.5.6. *Logol* software

Goal Complex pattern modelling and matching [url]

Description The *Logol* toolbox is a swiss-army-knife for pattern matching on DNA/RNA/Protein sequences, using a high-level grammatical formalism to permit a large expressivity for patterns [50]. A *Logol* pattern can consist in a complex combination of motifs (such as degenerated strings) and structures (such as imperfect stem-loop ou repeats). *Logol* key features are the possibilities to divide a pattern description into several sub-patterns, to model long range dependencies, to enable the use of ambiguous models or to permit the inclusion of negative conditions in a pattern definition. The *LogolMatch* parser takes as input a biological sequence and a grammar file. It returns a XML file containing all the occurrences of the pattern in the sequence with their parsing details. The input sequences can be genomes from biological banks.

Originality Many pattern matching tools exist to efficiently model specific types of patterns: *vmatch* [82], *patmatch* [121], *cutadapt* [89], scoring matrix or profile HMMs [44], [71]. The main advantage of *Logol* is its very large expressivity. It encompasses most of the features of these specialized tools and enables interplays between several classes of patterns (motifs and structures).

Application The *Logol* tool was applied to the detection of mutated primers in a metabarcoding study [41], [42] or to stem-loop identification (e.g. in CRISPR ⁰ [103], [50]). Ongoing application is the search for transposable elements in the human genome in the context of a colorectal cancer study ⁰. *Logol* strongly supported the study of the LXR- α targets in the framework of the *FatInteger* project.

3.5.7. *Protomata-suite*

Goal Expressive pattern discovery on protein sequences [url]

Description *Protomata* is a machine learning suite for the inference of automata characterizing (functional) families of proteins from available sequences. Based on partial and local alignments, *Protomata* learns precise characterizations of the families of proteins, allowing new family members to be predicted with a high specificity. Three main modules are integrated in the *Protomata-learner* workflow are available as well as stand-alone programs: 1) *paloma* builds partial local multiple alignments, 2) *protobuild* infers automata from these alignements and 3) *protomatch* and *protoalign* scans, parses and aligns new sequences with learnt automata. The suite is completed by tools to handle or visualize data and can be used online by the biologists via a web interface on Genouest Platform. It is actively maintained (version v2.1 was released in April 2017) and we are scheduling a new major version with enhanced scoring schemes that we have proposed [105].

Originality The main specificity is that the power of characterization is beyond the scope of classical sequence patterns such as PSSM (e.g. MEME suite [43]), Profile HMM (e.g. HMMER package [71]), or Prosite Patterns [111] allowing new family members to be predicted with a high specificity.

Application The *Protomata* tool is used both to update automatically the Cyanolase database [62] and, when combined to Formal Concept Analysis, to automated enzyme classification, such as the HAD superfamily of proteins [67] in the framework of the *Idealg* project.

⁰<http://crispi.genouest.org/>, <https://hal.inria.fr/hal-00643408>

⁰Preprint: <http://www.biorxiv.org/content/early/2017/03/09/115030>

FLUMINANCE Project-Team

3. Research Program

3.1. Estimation of fluid characteristic features from images

The measurement of fluid representative features such as vector fields, potential functions or vorticity maps, enables physicists to have better understanding of experimental or geophysical fluid flows. Such measurements date back to one century and more but became an intensive subject of research since the emergence of correlation techniques [48] to track fluid movements in pairs of images of a particles laden fluid or by the way of clouds photometric pattern identification in meteorological images. In computer vision, the estimation of the projection of the apparent motion of a 3D scene onto the image plane, referred to in the literature as optical-flow, is an intensive subject of researches since the 80's and the seminal work of B. Horn and B. Schunk [61]. Unlike to dense optical flow estimators, the former approach provides techniques that supply only sparse velocity fields. These methods have demonstrated to be robust and to provide accurate measurements for flows seeded with particles. These restrictions and their inherent discrete local nature limit too much their use and prevent any evolutions of these techniques towards the devising of methods supplying physically consistent results and small scale velocity measurements. It does not authorize also the use of scalar images exploited in numerous situations to visualize flows (image showing the diffusion of a scalar such as dye, pollutant, light index refraction, fluorecein,...). At the opposite, variational techniques enable in a well-established mathematical framework to estimate spatially continuous velocity fields, which should allow more properly to go towards the measurement of smaller motion scales. As these methods are defined through PDE's systems they allow quite naturally constraints to be included such as kinematic properties or dynamic laws governing the observed fluid flows. Besides, within this framework it is also much easier to define characteristic features estimation procedures on the basis of physically grounded data model that describes the relation linking the observed luminance function and some state variables of the observed flow. The Fluminance group has allowed a substantial progress in this direction with the design of dedicated dense estimation techniques to estimate dense fluid motion fields. See [7] for a detailed review. More recently problems related to scale measurement and uncertainty estimation have been investigated [55]. Dynamically consistent and highly robust techniques have been also proposed for the recovery of surface oceanic streams from satellite images [51]. Very recently parameter-free approaches relying on uncertainty concept has been devised [52]. This technique outperforms the state of the art.

3.2. Data assimilation and Tracking of characteristic fluid features

Real flows have an extent of complexity, even in carefully controlled experimental conditions, which prevents any set of sensors from providing enough information to describe them completely. Even with the highest levels of accuracy, space-time coverage and grid refinement, there will always remain at least a lack of resolution and some missing input about the actual boundary conditions. This is obviously true for the complex flows encountered in industrial and natural conditions, but remains also an obstacle even for standard academic flows thoroughly investigated in research conditions.

This unavoidable deficiency of the experimental techniques is nevertheless more and more compensated by numerical simulations. The parallel advances in sensors, acquisition, treatment and computer efficiency allow the mixing of experimental and simulated data produced at compatible scales in space and time. The inclusion of dynamical models as constraints of the data analysis process brings a guaranty of coherency based on fundamental equations known to correctly represent the dynamics of the flow (e.g. Navier Stokes equations) [11]. Conversely, the injection of experimental data into simulations ensures some fitting of the model with reality.

To enable data and models coupling to achieve its potential, some difficulties have to be tackled. It is in particular important to outline the fact that the coupling of dynamical models and image data are far from being straightforward. The first difficulty is related to the space of the physical model. As a matter of fact, physical models describe generally the phenomenon evolution in a 3D Cartesian space whereas images provides generally only 2D tomographic views or projections of the 3D space on the 2D image plane. Furthermore, these views are sometimes incomplete because of partial occlusions and the relations between the model state variables and the image intensity function are otherwise often intricate and only partially known. Besides, the dynamical model and the image data may be related to spatio-temporal scale spaces of very different natures which increases the complexity of an eventual multiscale coupling. As a consequence of these difficulties, it is necessary generally to define simpler dynamical models in order to assimilate image data. This redefinition can be done for instance on an uncertainty analysis basis, through physical considerations or by the way of data based empirical specifications. Such modeling comes to define inexact evolution laws and leads to the handling of stochastic dynamical models. The necessity to make use and define sound approximate models, the dimension of the state variables of interest and the complex relations linking the state variables and the intensity function, together with the potential applications described earlier constitute very stimulating issues for the design of efficient data-model coupling techniques based on image sequences.

On top of the problems mentioned above, the models exploited in assimilation techniques often suffer from some uncertainties on the parameters which define them. Hence, a new emerging field of research focuses on the characterization of the set of achievable solutions as a function of these uncertainties. This sort of characterization indeed turns out to be crucial for the relevant analysis of any simulation outputs or the correct interpretation of operational forecasting schemes. In this context, stochastic modeling play a crucial role to model and process uncertainty evolution along time. As a consequence, stochastic parameterization of flow dynamics has already been present in many contributions of the Fluminance group in the last years and will remain a cornerstone of the new methodologies investigated by the team in the domain of uncertainty characterization.

This wide theme of research problems is a central topic in our research group. As a matter of fact, such a coupling may rely on adequate instantaneous motion descriptors extracted with the help of the techniques studied in the first research axis of the FLUMINANCE group. In the same time, this coupling is also essential with respect to visual flow control studies explored in the third theme. The coupling between a dynamics and data, designated in the literature as a Data Assimilation issue, can be either conducted with optimal control techniques [62], [63] or through stochastic filtering approaches [56], [59]. These two frameworks have their own advantages and deficiencies. We rely indifferently on both approaches.

3.3. Optimization and control of fluid flows with visual servoing

Fluid flow control is a recent and active research domain. A significant part of the work carried out so far in that field has been dedicated to the control of the transition from laminarity to turbulence. Delaying, accelerating or modifying this transition is of great economical interest for industrial applications. For instance, it has been shown that for an aircraft, a drag reduction can be obtained while enhancing the lift, leading consequently to limit fuel consumption. In contrast, in other application domains such as industrial chemistry, turbulence phenomena are encouraged to improve heat exchange, increase the mixing of chemical components and enhance chemical reactions. Similarly, in military and civilians applications where combustion is involved, the control of mixing by means of turbulence handling rouses a great interest, for example to limit infra-red signatures of fighter aircraft.

Flow control can be achieved in two different ways: passive or active control. Passive control provides a permanent action on a system. Most often it consists in optimizing shapes or in choosing suitable surfacing (see for example [54] where longitudinal riblets are used to reduce the drag caused by turbulence). The main problem with such an approach is that the control is, of course, inoperative when the system changes. Conversely, in active control the action is time varying and adapted to the current system's state. This approach requires an external energy to act on the system through actuators enabling a forcing on the flow through for instance blowing and suction actions [66], [58]. A closed-loop problem can be formulated as an optimal control

issue where a control law minimizing an objective cost function (minimization of the drag, minimization of the actuators power, etc.) must be applied to the actuators [50]. Most of the works of the literature indeed comes back to open-loop control approaches [65], [60], [64] or to forcing approaches [57] with control laws acting without any feedback information on the flow actual state. In order for these methods to be operative, the model used to derive the control law must describe as accurately as possible the flow and all the eventual perturbations of the surrounding environment, which is very unlikely in real situations. In addition, as such approaches rely on a perfect model, a high computational costs is usually required. This inescapable pitfall has motivated a strong interest on model reduction. Their key advantage being that they can be specified empirically from the data and represent quite accurately, with only few modes, complex flows' dynamics. This motivates an important research axis in the Fluminance group.

3.4. Numerical models applied to hydrogeology and geophysics

The team is strongly involved in numerical models for hydrogeology and geophysics. There are many scientific challenges in the area of groundwater simulations. This interdisciplinary research is very fruitful with cross-fertilizing subjects.

In geophysics, a main concern is to solve inverse problems in order to fit the measured data with the model. Generally, this amounts to solve a linear or nonlinear least-squares problem.

Models of geophysics are in general coupled and multi-physics. For example, reactive transport couples advection-diffusion with chemistry. Here, the mathematical model is a set of nonlinear Partial Differential Algebraic Equations. At each timestep of an implicit scheme, a large nonlinear system of equations arise. The challenge is to solve efficiently and accurately these large nonlinear systems.

3.5. Numerical algorithms and high performance computing

Linear algebra is at the kernel of most scientific applications, in particular in physical or chemical engineering. The objectives are to analyze the complexity of these different methods, to accelerate convergence of iterative methods, to measure and improve the efficiency on parallel architectures, to define criteria of choice.

GENSCALE Project-Team

3. Research Program

3.1. Axis 1: Data Structure

The aim of this axis is to develop efficient data structures for representing the mass of genomic data generated by the sequencing machines. This research is motivated by the fact that the treatments of large genomes, such as mammalian or plant genomes, require high computing resources, and more specifically very important memory configuration. For example, the ABYSS software used 4.3TB of memory to assemble the white spruce genome [45]. The main reason for such memory consumption is that the data structures used in ABYSS are far from optimal (and this is also the case for many assembly software).

Our research focuses on the de-Bruijn graph structure. This well-known data structure, directly built from raw sequencing data, have many properties matching perfectly well with NGS processing requirements (see next section). Here, the question we are interested in is how to provide a low memory footprint implementation of the de-Bruijn graph to process very large NGS datasets, including metagenomic ones.

Another research direction of this axis is the indexing of large sets of objects. A typical, but non exclusive, need is to annotate nodes of the de-Bruijn graph, that is potentially billions of items. Again, very low memory footprint indexing structures are mandatory to manage a very large quantity of objects.

3.2. Axis 2: Algorithms

The main goal of the GenScale team is to develop optimized tools dedicated to NGS processing. Optimization can be seen both in terms of space (low memory footprint) and in terms of time (fast execution time). The first point is mainly related to advanced data structures as presented in the previous section (axis 1). The second point relies on new algorithms and, when possible implementation on parallel structures (axis 3).

We do not have the ambition to cover the vast panel of software related to NGS needs. We particularly focused on the following areas:

- **NGS data Compression** De-Bruijn graphs are de facto a compressed representation of the NGS information from which very efficient and specific compressors can be designed. Furthermore, compressing the data using smart structures may speed up some downstream graph-based analyses since a graph structure is already built.
- **Genome assembly** This task remains very complicated, especially for large and complex genomes, such as plant genomes with polyploid and highly repeated structures. We worked both on the generation of contigs and on the scaffolding step.
- **Detection of variants** This is often the first information we want to extract from billions of reads. Variant structures range from SNPs or short indels to large insertions/deletions and long inversions over the chromosomes. We developed original methods to find variants without any reference genome.
- **Metagenomics** We focussed our research on comparative metagenomics by providing methods able to compare hundreds of metagenomic samples together. This is achieved by combining very low memory data structures and efficient implementation and parallelization on large clusters.
- **Genome Wide Association Study (GWAS)** We tackle this problem with algorithms commonly used in data mining. From two cohorts of individuals (case and control) we can exhibit statistically significant *patterns* spanning over full genomes.

3.3. Axis 3: Parallelism

This third axis is another lever to increase performances and scalability of NGS treatments. There are many levels of parallelism that can be used and/or combined to reduce the execution time of very time-consuming bioinformatics processes. A first level is the parallel nature of today processors that now house several cores. A second level is the grid structure that is present in all bioinformatics centers or in the cloud. These two levels are generally combined: a node of a grid is often a multicore system. Another possibility is to add hardware accelerators to a processor. A GPU board is a good example.

GenScale does not do explicit research on parallelism. It exploits the capacity of computing resources to support parallelism. The problem is addressed in two different directions. The first is an engineering approach that uses existing parallel tools to implement algorithms such as multithreading or MapReduce techniques. The second is a parallel algorithmic approach: during the development step, the algorithms are constrained by parallel criteria. This is particularly true for parallel algorithms targeting hardware accelerators.

HYBRID Project-Team

3. Research Program

3.1. Research Program

The scientific objective of Hybrid team is to improve 3D interaction of one or multiple users with virtual environments, by making full use of physical engagement of the body, and by incorporating the mental states by means of brain-computer interfaces. We intend to improve each component of this framework individually, but we also want to improve the subsequent combinations of these components.

The "hybrid" 3D interaction loop between one or multiple users and a virtual environment is depicted in Figure 1. Different kinds of 3D interaction situations are distinguished (red arrows, bottom): 1) body-based interaction, 2) mind-based interaction, 3) hybrid and/or 4) collaborative interaction (with at least two users). In each case, three scientific challenges arise which correspond to the three successive steps of the 3D interaction loop (blue squares, top): 1) the 3D interaction technique, 2) the modeling and simulation of the 3D scenario, and 3) the design of appropriate sensory feedback.

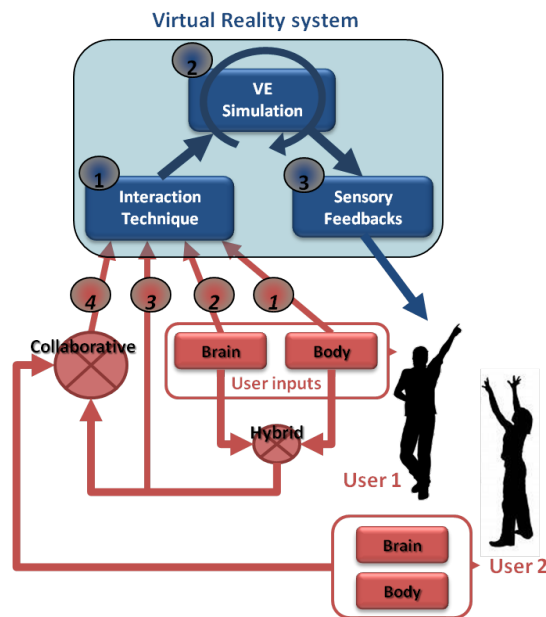


Figure 1. 3D hybrid interaction loop between one or multiple users and a virtual reality system. Top (in blue) three steps of 3D interaction with a virtual environment: (1-blue) interaction technique, (2-blue) simulation of the virtual environment, (3-blue) sensory feedbacks. Bottom (in red) different cases of interaction: (1-red) body-based, (2-red) mind-based, (3-red) hybrid, and (4-red) collaborative 3D interaction.

The 3D interaction loop involves various possible inputs from the user(s) and different kinds of output (or sensory feedback) from the simulated environment. Each user can involve his/her body and mind by means of corporal and/or brain-computer interfaces. A hybrid 3D interaction technique (1) mixes mental and motor inputs and translates them into a command for the virtual environment. The real-time simulation (2) of the

virtual environment is taking into account these commands to change and update the state of the virtual world and virtual objects. The state changes are sent back to the user and perceived by means of different sensory feedbacks (e.g., visual, haptic and/or auditory) (3). The sensory feedbacks are closing the 3D interaction loop. Other users can also interact with the virtual environment using the same procedure, and can eventually “collaborate” by means of “collaborative interactive techniques” (4).

This description is stressing three major challenges which correspond to three mandatory steps when designing 3D interaction with virtual environments:

- **3D interaction techniques:** This first step consists in translating the actions or intentions of the user (inputs) into an explicit command for the virtual environment. In virtual reality, the classical tasks that require such kinds of user command were early categorized in four [33]: navigating the virtual world, selecting a virtual object, manipulating it, or controlling the application (entering text, activating options, etc). The addition of a third dimension, the use of stereoscopic rendering and the use of advanced VR interfaces make however inappropriate many techniques that proved efficient in 2D, and make it necessary to design specific interaction techniques and adapted tools. This challenge is here renewed by the various kinds of 3D interaction which are targeted. In our case, we consider various cases, with motor and/or cerebral inputs, and potentially multiple users.
- **Modeling and simulation of complex 3D scenarios:** This second step corresponds to the update of the state of the virtual environment, in real-time, in response to all the potential commands or actions sent by the user. The complexity of the data and phenomena involved in 3D scenarios is constantly increasing. It corresponds for instance to the multiple states of the entities present in the simulation (rigid, articulated, deformable, fluids, which can constitute both the user’s virtual body and the different manipulated objects), and the multiple physical phenomena implied by natural human interactions (squeezing, breaking, melting, etc). The challenge consists here in modeling and simulating these complex 3D scenarios and meeting, at the same time, two strong constraints of virtual reality systems: performance (real-time and interactivity) and genericity (e.g., multi-resolution, multi-modal, multi-platform, etc).
- **Immersive sensory feedbacks:** This third step corresponds to the display of the multiple sensory feedbacks (output) coming from the various VR interfaces. These feedbacks enable the user to perceive the changes occurring in the virtual environment. They are closing the 3D interaction loop, making the user immersed, and potentially generating a subsequent feeling of presence. Among the various VR interfaces which have been developed so far we can stress two kinds of sensory feedback: visual feedback (3D stereoscopic images using projection-based systems such as CAVE systems or Head Mounted Displays); and haptic feedback (related to the sense of touch and to tactile or force-feedback devices). The Hybrid team has a strong expertise in haptic feedback, and in the design of haptic and “pseudo-haptic” rendering [34]. Note that a major trend in the community, which is strongly supported by the Hybrid team, relates to a “perception-based” approach, which aims at designing sensory feedbacks which are well in line with human perceptual capacities.

These three scientific challenges are addressed differently according to the context and the user inputs involved. We propose to consider three different contexts, which correspond to the three different research axes of the Hybrid research team, namely: 1) body-based interaction (motor input only), 2) mind-based interaction (cerebral input only), and then 3) hybrid and collaborative interaction (i.e., the mixing of body and brain inputs from one or multiple users).

3.2. Research Axes

The scientific activity of Hybrid team follows three main axes of research:

- **Body-based interaction in virtual reality.** Our first research axis concerns the design of immersive and effective “body-based” 3D interactions, i.e., relying on a physical engagement of the user’s body. This trend is probably the most popular one in VR research at the moment. Most VR setups make use of tracking systems which measure specific positions or actions of the user in order to interact with a virtual environment. However, in recent years, novel options have emerged for measuring

“full-body” movements or other, even less conventional, inputs (e.g. body equilibrium). In this first research axis we are thus concerned by the emergence of new kinds of “body-based interaction” with virtual environments. This implies the design of novel 3D user interfaces and novel 3D interactive techniques, novel simulation models and techniques, and novel sensory feedbacks for body-based interaction with virtual worlds. It involves real-time physical simulation of complex interactive phenomena, and the design of corresponding haptic and pseudo-haptic feedback.

- **Mind-based interaction in virtual reality.** Our second research axis concerns the design of immersive and effective “mind-based” 3D interactions in Virtual Reality. Mind-based interaction with virtual environments is making use of Brain-Computer Interface technology. This technology corresponds to the direct use of brain signals to send “mental commands” to an automated system such as a robot, a prosthesis, or a virtual environment. BCI is a rapidly growing area of research and several impressive prototypes are already available. However, the emergence of such a novel user input is also calling for novel and dedicated 3D user interfaces. This implies to study the extension of the mental vocabulary available for 3D interaction with VE, then the design of specific 3D interaction techniques “driven by the mind” and, last, the design of immersive sensory feedbacks that could help improving the learning of brain control in VR.
- **Hybrid and collaborative 3D interaction.** Our third research axis intends to study the combination of motor and mental inputs in VR, for one or multiple users. This concerns the design of mixed systems, with potentially collaborative scenarios involving multiple users, and thus, multiple bodies and multiple brains sharing the same VE. This research axis therefore involves two interdependent topics: 1) collaborative virtual environments, and 2) hybrid interaction. It should end up with collaborative virtual environments with multiple users, and shared systems with body and mind inputs.

HYCOMES Project-Team

3. Research Program

3.1. Hybrid Systems Modeling

Systems industries today make extensive use of mathematical modeling tools to design computer controlled physical systems. This class of tools addresses the modeling of physical systems with models that are simpler than usual scientific computing problems by using only Ordinary Differential Equations (ODE) and Difference Equations but not Partial Differential Equations (PDE). This family of tools first emerged in the 1980's with SystemBuild by MatrixX (now distributed by National Instruments) followed soon by Simulink by Mathworks, with an impressive subsequent development.

In the early 90's control scientists from the University of Lund (Sweden) realized that the above approach did not support component based modeling of physical systems with reuse⁰. For instance, it was not easy to draw an electrical or hydraulic circuit by assembling component models of the various devices. The development of the Omola language by Hilding Elmqvist was a first attempt to bridge this gap by supporting some form of Differential Algebraic Equations (DAE) in the models. Modelica quickly emerged from this first attempt and became in the 2000's a major international concerted effort with the Modelica Consortium⁰. A wider set of tools, both industrial and academic, now exists in this segment⁰. In the EDA sector, VHDL-AMS was developed as a standard [13].

Despite these tools are now widely used by a number of engineers, they raise a number of technical difficulties. The meaning of some programs, their mathematical semantics, can be tainted with uncertainty. A main source of difficulty lies in the failure to properly handle the discrete and the continuous parts of systems, and their interaction. How the propagation of mode changes and resets should be handled? How to avoid artifacts due to the use of a global ODE solver causing unwanted coupling between seemingly non interacting subsystems? Also, the mixed use of an equational style for the continuous dynamics with an imperative style for the mode changes and resets is a source of difficulty when handling parallel composition. It is therefore not uncommon that tools return complex warnings for programs with many different suggested hints for fixing them. Yet, these "pathological" programs can still be executed, if wanted so, giving surprising results — See for instance the Simulink examples in [20], [1] and [16].

Indeed this area suffers from the same difficulties that led to the development of the theory of synchronous languages as an effort to fix obscure compilation schemes for discrete time equation based languages in the 1980's. Our vision is that hybrid systems modeling tools deserve similar efforts in theory as synchronous languages did for the programming of embedded systems.

3.2. Background on non-standard analysis

Non-Standard analysis plays a central role in our research on hybrid systems modeling [1], [20], [17], [16]. The following text provides a brief summary of this theory and gives some hints on its usefulness in the context of hybrid systems modeling. This presentation is based on our paper [1], a chapter of Simon Bliudze's PhD thesis [25], and a recent presentation of non-standard analysis, not axiomatic in style, due to the mathematician Lindström [47].

⁰<http://www.lccc.lth.se/media/LCCC2012/WorkshopSeptember/slides/Astrom.pdf>

⁰<https://www.modelica.org/>

⁰SimScape by Mathworks, Amesim by LMS International, now Siemens PLM, and more.

Non-standard numbers allowed us to reconsider the semantics of hybrid systems and propose a radical alternative to the *super-dense time semantics* developed by Edward Lee and his team as part of the Ptolemy II project, where cascades of successive instants can occur in zero time by using $\mathbb{R}_+ \times \mathbb{N}$ as a time index. In the non-standard semantics, the time index is defined as a set $\mathbb{T} = \{n\partial \mid n \in {}^*\mathbb{N}\}$, where ∂ is an *infinitesimal* and ${}^*\mathbb{N}$ is the set of *non-standard integers*. Remark that (1) \mathbb{T} is dense in \mathbb{R}_+ , making it “continuous”, and (2) every $t \in \mathbb{T}$ has a predecessor in \mathbb{T} and a successor in \mathbb{T} , making it “discrete”. Although it is not effective from a computability point of view, the *non-standard semantics* provides a framework that is familiar to the computer scientist and at the same time efficient as a symbolic abstraction. This makes it an excellent candidate for the development of provably correct compilation schemes and type systems for hybrid systems modeling languages.

Non-standard analysis was proposed by Abraham Robinson in the 1960s to allow the explicit manipulation of “infinitesimals” in analysis [53], [41], [12]. Robinson’s approach is axiomatic; he proposes adding three new axioms to the basic Zermelo-Fraenkel (ZFC) framework. There has been much debate in the mathematical community as to whether it is worth considering non-standard analysis instead of staying with the traditional one. We do not enter this debate. The important thing for us is that non-standard analysis allows the use of the non-standard discretization of continuous dynamics “as if” it was operational.

Not surprisingly, such an idea is quite ancient. Iwasaki et al. [43] first proposed using non-standard analysis to discuss the nature of time in hybrid systems. Bliudze and Krob [26], [25] have also used non-standard analysis as a mathematical support for defining a system theory for hybrid systems. They discuss in detail the notion of “system” and investigate computability issues. The formalization they propose closely follows that of Turing machines, with a memory tape and a control mechanism.

3.3. Contract-Based Design, Interfaces Theories, and Requirements Engineering

System companies such as automotive and aeronautic companies are facing significant difficulties due to the exponentially raising complexity of their products coupled with increasingly tight demands on functionality, correctness, and time-to-market. The cost of being late to market or of imperfections in the products is staggering as witnessed by the recent recalls and delivery delays that many major car and airplane manufacturers had to bear in the recent years. The specific root causes of these design problems are complex and relate to a number of issues ranging from design processes and relationships with different departments of the same company and with suppliers, to incomplete requirement specification and testing.

We believe the most promising means to address the challenges in systems engineering is to employ structured and formal design methodologies that seamlessly and coherently combine the various viewpoints of the design space (behavior, space, time, energy, reliability, ...), that provide the appropriate abstractions to manage the inherent complexity, and that can provide correct-by-construction implementations. The following technology issues must be addressed when developing new approaches to the design of complex systems:

- The overall design flows for heterogeneous systems and the associated use of models across traditional boundaries are not well developed and understood. Relationships between different teams inside a same company, or between different stake-holders in the supplier chain, are not well supported by solid technical descriptions for the mutual obligations.
- System requirements capture and analysis is in large part a heuristic process, where the informal text and natural language-based techniques in use today are facing significant challenges. Formal requirements engineering is in its infancy: mathematical models, formal analysis techniques and links to system implementation must be developed.
- Dealing with variability, uncertainty, and life-cycle issues, such as extensibility of a product family, are not well-addressed using available systems engineering methodologies and tools.

The challenge is to address the entire process and not to consider only local solutions of methodology, tools, and models that ease part of the design.

Contract-based design has been proposed as a new approach to the system design problem that is rigorous and effective in dealing with the problems and challenges described before, and that, at the same time, does not require a radical change in the way industrial designers carry out their task as it cuts across design flows of different type. Indeed, contracts can be used almost everywhere and at nearly all stages of system design, from early requirements capture, to embedded computing infrastructure and detailed design involving circuits and other hardware. Contracts explicitly handle pairs of properties, respectively representing the assumptions on the environment and the guarantees of the system under these assumptions. Intuitively, a contract is a pair $C = (A, G)$ of assumptions and guarantees characterizing in a formal way 1) under which context the design is assumed to operate, and 2) what its obligations are. Assume/Guarantee reasoning has been known for a long time, and has been used mostly as verification mean for the design of software [51]. However, contract based design with explicit assumptions is a philosophy that should be followed all along the design, with all kinds of models, whenever necessary. Here, specifications are not limited to profiles, types, or taxonomy of data, but also describe the functions, performances of various kinds (time and energy), and reliability. This amounts to enrich a component's interface with, on one hand, formal specifications of the behavior of the environment in which the component may be instantiated and, on the other hand, of the expected behavior of the component itself. The consideration of rich interfaces is still in its infancy. So far, academic researchers have addressed the mathematics and algorithmics of interfaces theories and contract-based reasoning. To make them a technique of choice for system engineers, we must develop:

- Mathematical foundations for interfaces and requirements engineering that enable the design of frameworks and tools;
- A system engineering framework and associated methodologies and tool sets that focus on system requirements modeling, contract specification, and verification at multiple abstraction layers.

A detailed bibliography on contract and interface theories for embedded system design can be found in [2]. In a nutshell, contract and interface theories fall into two main categories:

Assume/guarantee contracts. By explicitly relying on the notions of assumptions and guarantees, A/G-contracts are intuitive, which makes them appealing for the engineer. In A/G-contracts, assumptions and guarantees are just properties regarding the behavior of a component and of its environment. The typical case is when these properties are formal languages or sets of traces, which includes the class of safety properties [44], [33], [50], [15], [34]. Contract theories were initially developed as specification formalisms able to refuse some inputs from the environment [42]. A/G-contracts were advocated by the SPEEDS project [19]. They were further experimented in the framework of the CESAR project [37], with the additional consideration of *weak* and *strong* assumptions. This is still a very active research topic, with several recent contributions dealing with the timed [24] and probabilistic [29], [30] viewpoints in system design, and even mixed-analog circuit design [52].

Automata theoretic interfaces. Interfaces combine assumptions and guarantees in a single, automata theoretic specification. Most interface theories are based on Lynch Input/Output Automata [49], [48]. Interface Automata [56], [55], [57], [31] focus primarily on parallel composition and compatibility: Two interfaces can be composed and are compatible if there is at least one environment where they can work together. The idea is that the resulting composition exposes as an interface the needed information to ensure that incompatible pairs of states cannot be reached. This can be achieved by using the possibility, for an Interface Automaton, to refuse selected inputs from the environment in a given state, which amounts to the implicit assumption that the environment will never produce any of the refused inputs, when the interface is in this state. Modal Interfaces [3] inherit from both Interface Automata and the originally unrelated notion of Modal Transition System [46], [14], [27], [45]. Modal Interfaces are strictly more expressive than Interface Automata by decoupling the I/O orientation of an event and its deontic modalities (mandatory, allowed or forbidden). Informally, a *must* transition is available in every component that realizes the modal interface, while a *may* transition needs not be. Research on interface theories is still very active. For instance, timed [58], [21], [23], [39], [38], [22], probabilistic [29], [40] and energy-aware [32] interface theories have been proposed recently.

Requirements Engineering is one of the major concerns in large systems industries today, particularly so in sectors where certification prevails [54]. DOORS projects collecting requirements are poorly structured and cannot be considered a formal modeling framework today. They are nothing more than an informal documentation enriched with hyperlinks. As examples, medium size sub-systems may have a few thousands requirements and the Rafale fighter aircraft has above 250,000 of them. For the Boeing 787, requirements were not stable while subcontractors performed the development of the fly-by-wire and of the landing gear subsystems.

We see Contract-Based Design and Interfaces Theories as innovative tools in support of Requirements Engineering. The Software Engineering community has extensively covered several aspects of Requirements Engineering, in particular:

- the development and use of large and rich *ontologies*; and
- the use of Model Driven Engineering technology for the structural aspects of requirements and resulting hyperlinks (to tests, documentation, PLM, architecture, and so on).

Behavioral models and properties, however, are not properly encompassed by the above approaches. This is the cause of a remaining gap between this phase of systems design and later phases where formal model based methods involving behavior have become prevalent—see the success of Matlab/Simulink/Scade technologies. We believe that our work on contract based design and interface theories is best suited to bridge this gap.

I4S Project-Team

3. Research Program

3.1. Vibration analysis

In this section, the main features for the key monitoring issues, namely identification, detection, and diagnostics, are provided, and a particular instantiation relevant for vibration monitoring is described.

It should be stressed that the foundations for identification, detection, and diagnostics, are fairly general, if not generic. Handling high order linear dynamical systems, in connection with finite elements models, which call for using subspace-based methods, is specific to vibration-based SHM. Actually, one particular feature of model-based sensor information data processing as exercised in I4S, is the combined use of black-box or semi-physical models together with physical ones. Black-box and semi-physical models are, for example, eigenstructure parameterizations of linear MIMO systems, of interest for modal analysis and vibration-based SHM. Such models are intended to be identifiable. However, due to the large model orders that need to be considered, the issue of model order selection is really a challenge. Traditional advanced techniques from statistics such as the various forms of Akaike criteria (AIC, BIC, MDL, ...) do not work at all. This gives rise to new research activities specific to handling high order models.

Our approach to monitoring assumes that a model of the monitored system is available. This is a reasonable assumption, especially within the SHM areas. The main feature of our monitoring method is its intrinsic ability to the early warning of small deviations of a system with respect to a reference (safe) behavior under usual operating conditions, namely without any artificial excitation or other external action. Such a normal behavior is summarized in a reference parameter vector θ_0 , for example a collection of modes and mode-shapes.

3.1.1. Identification

The behavior of the monitored continuous system is assumed to be described by a parametric model $\{\mathbf{P}_\theta, \theta \in \Theta\}$, where the distribution of the observations (Z_0, \dots, Z_N) is characterized by the parameter vector $\theta \in \Theta$.

For reasons closely related to the vibrations monitoring applications, we have been investigating subspace-based methods, for both the identification and the monitoring of the eigenstructure (λ, ϕ_λ) of the state transition matrix F of a linear dynamical state-space system :

$$\begin{cases} X_{k+1} &= F X_k + V_{k+1} \\ Y_k &= H X_k + W_k \end{cases}, \quad (1)$$

namely the $(\lambda, \varphi_\lambda)$ defined by :

$$\det (F - \lambda I) = 0, \quad (F - \lambda I) \phi_\lambda = 0, \quad \varphi_\lambda \triangleq H \phi_\lambda \quad (2)$$

The (canonical) parameter vector in that case is :

$$\theta \triangleq \begin{pmatrix} \Lambda \\ \text{vec} \Phi \end{pmatrix} \quad (3)$$

where Λ is the vector whose elements are the eigenvalues λ , Φ is the matrix whose columns are the φ_λ 's, and vec is the column stacking operator.

Subspace-based methods is the generic name for linear systems identification algorithms based on either time domain measurements or output covariance matrices, in which different subspaces of Gaussian random vectors play a key role [62].

Let $R_i \triangleq \mathbf{E} (Y_k Y_{k-i}^T)$ and:

$$\mathcal{H}_{p+1,q} \triangleq \begin{pmatrix} R_1 & R_2 & \vdots & R_q \\ R_2 & R_3 & \vdots & R_{q+1} \\ \vdots & \vdots & \vdots & \vdots \\ R_{p+1} & R_{p+2} & \vdots & R_{p+q} \end{pmatrix} \triangleq \text{Hank}(R_i) \quad (4)$$

be the output covariance and Hankel matrices, respectively; and: $G \triangleq \mathbf{E} (X_k Y_{k-1}^T)$. Direct computations of the R_i 's from the equations (4) lead to the well known key factorizations :

$$\begin{aligned} R_i &= H F^{i-1} G \\ \mathcal{H}_{p+1,q} &= \mathcal{O}_{p+1}(H, F) \mathcal{C}_q(F, G) \end{aligned} \quad (5)$$

where:

$$\mathcal{O}_{p+1}(H, F) \triangleq \begin{pmatrix} H \\ HF \\ \vdots \\ HF^p \end{pmatrix} \quad \text{and} \quad \mathcal{C}_q(F, G) \triangleq (G \ FG \ \dots \ F^{q-1}G) \quad (6)$$

are the observability and controllability matrices, respectively. The observation matrix H is then found in the first block-row of the observability matrix \mathcal{O} . The state-transition matrix F is obtained from the shift invariance property of \mathcal{O} . The eigenstructure (λ, ϕ_λ) then results from (5).

Since the actual model order is generally not known, this procedure is run with increasing model orders.

3.1.2. Detection

Our approach to on-board detection is based on the so-called asymptotic statistical local approach. It is worth noticing that these investigations of ours have been initially motivated by a vibration monitoring application example. It should also be stressed that, as opposite to many monitoring approaches, our method does not require repeated identification for each newly collected data sample.

For achieving the early detection of small deviations with respect to the normal behavior, our approach generates, on the basis of the reference parameter vector θ_0 and a new data record, indicators which automatically perform :

- The early detection of a slight mismatch between the model and the data;
- A preliminary diagnostics and localization of the deviation(s);
- The tradeoff between the magnitude of the detected changes and the uncertainty resulting from the estimation error in the reference model and the measurement noise level.

These indicators are computationally cheap, and thus can be embedded. This is of particular interest in some applications, such as flutter monitoring.

Choosing the eigenvectors of matrix F as a basis for the state space of model (4) yields the following representation of the observability matrix:

$$\mathcal{O}_{p+1}(\theta) = \begin{pmatrix} \Phi \\ \Phi \Delta \\ \vdots \\ \Phi \Delta^p \end{pmatrix} \quad (7)$$

where $\Delta \triangleq \text{diag}(\Lambda)$, and Λ and Φ are as in (6). Whether a nominal parameter θ_0 fits a given output covariance sequence $(R_j)_j$ is characterized by:

$$\mathcal{O}_{p+1}(\theta_0) \text{ and } \mathcal{H}_{p+1,q} \text{ have the same left kernel space.} \quad (8)$$

This property can be checked as follows. From the nominal θ_0 , compute $\mathcal{O}_{p+1}(\theta_0)$ using (10), and perform e.g. a singular value decomposition (SVD) of $\mathcal{O}_{p+1}(\theta_0)$ for extracting a matrix U such that:

$$U^T U = I_s \text{ and } U^T \mathcal{O}_{p+1}(\theta_0) = 0 \quad (9)$$

Matrix U is not unique (two such matrices relate through a post-multiplication with an orthonormal matrix), but can be regarded as a function of θ_0 . Then the characterization writes:

$$U(\theta_0)^T \mathcal{H}_{p+1,q} = 0 \quad (10)$$

3.1.2.1. Residual associated with subspace identification.

Assume now that a reference θ_0 and a new sample Y_1, \dots, Y_N are available. For checking whether the data agree with θ_0 , the idea is to compute the empirical Hankel matrix $\hat{\mathcal{H}}_{p+1,q}$:

$$\hat{\mathcal{H}}_{p+1,q} \triangleq \text{Hank}(\hat{R}_i), \quad \hat{R}_i \triangleq 1/(N-i) \sum_{k=i+1}^N Y_k Y_{k-i}^T \quad (11)$$

and to define the residual vector:

$$\zeta_N(\theta_0) \triangleq \sqrt{N} \text{vec} \left(U(\theta_0)^T \hat{\mathcal{H}}_{p+1,q} \right) \quad (12)$$

Let θ be the actual parameter value for the system which generated the new data sample, and \mathbf{E}_θ be the expectation when the actual system parameter is θ . From (13), we know that $\zeta_N(\theta_0)$ has zero mean when no change occurs in θ , and nonzero mean if a change occurs. Thus $\zeta_N(\theta_0)$ plays the role of a residual.

As in most fault detection approaches, the key issue is to design a *residual*, which is ideally close to zero under normal operation, and has low sensitivity to noises and other nuisance perturbations, but high sensitivity to small deviations, before they develop into events to be avoided (damages, faults, ...). The originality of our approach is to :

- *Design* the residual basically as a *parameter estimating function*,
- *Evaluate* the residual thanks to a kind of central limit theorem, stating that the residual is asymptotically Gaussian and reflects the presence of a deviation in the parameter vector through a change in its own mean vector, which switches from zero in the reference situation to a non-zero value.

The central limit theorem shows [56] that the residual is asymptotically Gaussian :

$$\zeta_N \xrightarrow{N \rightarrow \infty} \begin{cases} \mathcal{N}(0, \Sigma) & \text{under } \mathbf{P}_{\theta_0} , \\ \mathcal{N}(\mathcal{J} \eta, \Sigma) & \text{under } \mathbf{P}_{\theta_0 + \eta/\sqrt{N}} , \end{cases} \quad (13)$$

where the asymptotic covariance matrix Σ can be estimated, and manifests the deviation in the parameter vector by a change in its own mean value. Then, deciding between $\eta = 0$ and $\eta \neq 0$ amounts to compute the following χ^2 -test, provided that \mathcal{J} is full rank and Σ is invertible :

$$\chi^2 = \bar{\zeta}^T \mathbf{F}^{-1} \bar{\zeta} \geq \lambda . \quad (14)$$

where

$$\bar{\zeta} \triangleq \mathcal{J}^T \Sigma^{-1} \zeta_N \quad \text{and} \quad \mathbf{F} \triangleq \mathcal{J}^T \Sigma^{-1} \mathcal{J} \quad (15)$$

3.1.3. Diagnostics

A further monitoring step, often called *fault isolation*, consists in determining which (subsets of) components of the parameter vector θ have been affected by the change. Solutions for that are now described. How this relates to diagnostics is addressed afterwards.

The question: *which (subsets of) components of θ have changed ?*, can be addressed using either nuisance parameters elimination methods or a multiple hypotheses testing approach [55].

In most SHM applications, a complex physical system, characterized by a generally non identifiable parameter vector Φ has to be monitored using a simple (black-box) model characterized by an identifiable parameter vector θ . A typical example is the vibration monitoring problem for which complex finite elements models are often available but not identifiable, whereas the small number of existing sensors calls for identifying only simplified input-output (black-box) representations. In such a situation, two different diagnosis problems may arise, namely diagnosis in terms of the black-box parameter θ and diagnosis in terms of the parameter vector Φ of the underlying physical model.

The isolation methods sketched above are possible solutions to the former. Our approach to the latter diagnosis problem is basically a detection approach again, and not a (generally ill-posed) inverse problem estimation approach.

The basic idea is to note that the physical sensitivity matrix writes $\mathcal{J} \mathcal{J}_{\Phi\theta}$, where $\mathcal{J}_{\Phi\theta}$ is the Jacobian matrix at Φ_0 of the application $\Phi \mapsto \theta(\Phi)$, and to use the sensitivity test for the components of the parameter vector Φ . Typically this results in the following type of directional test :

$$\chi_{\Phi}^2 = \zeta^T \Sigma^{-1} \mathcal{J} \mathcal{J}_{\Phi\theta} (\mathcal{J}_{\Phi\theta}^T \mathcal{J}^T \Sigma^{-1} \mathcal{J} \mathcal{J}_{\Phi\theta})^{-1} \mathcal{J}_{\Phi\theta}^T \mathcal{J}^T \Sigma^{-1} \zeta \geq \lambda . \quad (16)$$

It should be clear that the selection of a particular parameterization Φ for the physical model may have a non negligible influence on such type of tests, according to the numerical conditioning of the Jacobian matrices $\mathcal{J}_{\Phi\theta}$.

3.2. Thermal methods

3.2.1. Infrared thermography and heat transfer

This section introduce the infrared radiation and its link with the temperature, in the next part different measurement methods based on that principle are presented.

3.2.1.1. Infrared radiation

Infrared is an electromagnetic radiation having a wavelength between $0.2\mu\text{m}$ and 1 mm , this range begin in uv spectrum and it ends on the microwaves domain, see Figure 1 .

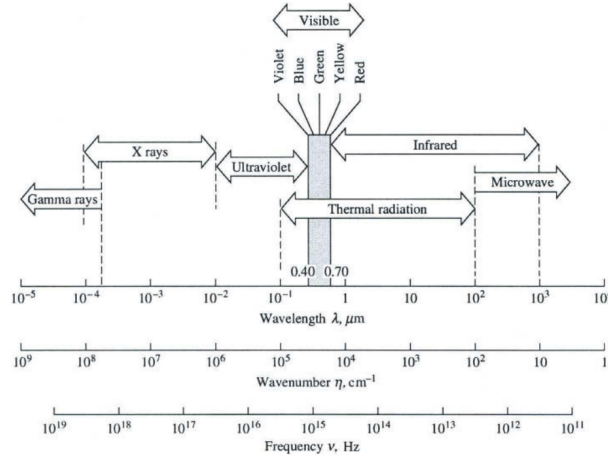


Figure 1. Electromagnetic spectrum - Credit MODEST, M.F. (1993). Radiative Heat Transfer. Academic Press.

For scientific purpose infrared can be divided in three ranges of wavelength in which the application varies, see Table 1 .

Table 1. Wavelength bands in the infrared according to ISO 20473:2007

Band name	wavelength	Uses \ definition
Near infrared (PIR, IR-A, NIR)	$0.7 - 3\mu\text{m}$	Reflected solar heat flux
Mid infrared (MIR, IR-B)	$3 - 50\mu\text{m}$	Thermal infrared
Far infrared (LIR, IR-C, FIR)	$50 - 1000\mu\text{m}$	Astronomy

Our work is concentrated in the mid infrared spectral band. Keep in mind that Table 1 represents the ISO 20473 division scheme, in the literature boundaries between bands can move slightly.

The Plank's law, proposed by Max Planck in 1901, allows to compute the black body emission spectrum for various temperatures (and only temperatures), see Figure 2 left. The black body is a theoretical construction, it represents perfect energy emitter at a given temperature, cf Equation (20).

$$M_{\lambda,T}^o = \frac{C_1 \lambda^{-5}}{\exp \frac{C_2}{\lambda T} - 1} \quad (17)$$

With λ the wavelength in m and T as the temperature in Kelvin. The C_1 and C_2 constant, respectively in W.m^2 and m.K are defined as follow:

$$\begin{aligned} C_1 &= 2hc^2\pi \\ C_2 &= h\frac{c}{k} \end{aligned} \quad (18)$$

with

- c The electromagnetic wave speed (in vacuum c is the light speed in m.s^{-1}).
- $k = 1.381e^{-23} \text{ J.K}^{-1}$ The Boltzmann (Entropy definition from Ludwig Boltzmann 1873). It can be seen as a proportionality factor between the temperature and the energy of a system.
- $h \approx 6,62606957e^{-34} \text{ J.s}$ The Plank constant. It is the link between the photons energy and their frequency.

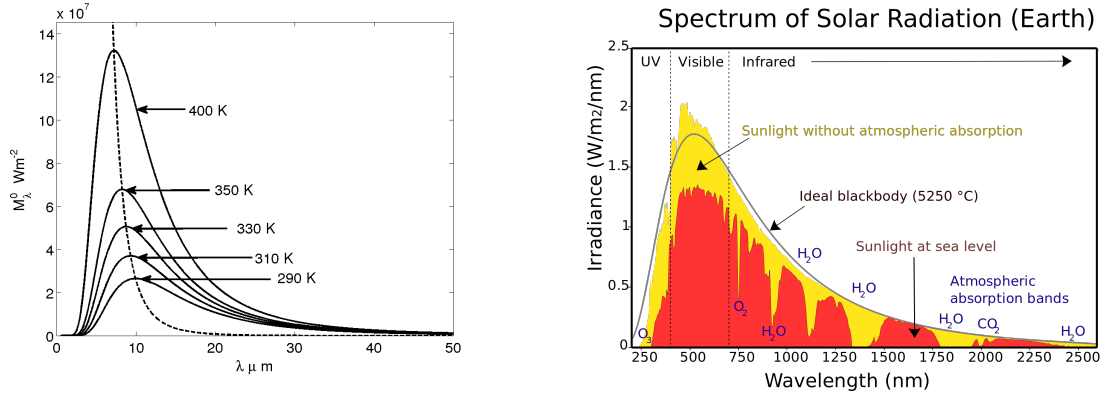


Figure 2. Left: Plank's law at various temperatures - Right: Energy spectrum of the atmosphere

By generalizing the Plank's law with the Stefan Boltzmann law (proposed first in 1879 and then in 1884 by Joseph Stefan and Ludwig Boltzmann) it is possible to address mathematically the energy spectrum of real body at each wavelength dependent of the temperature, the optical condition and the real body properties, which is the base of the infrared thermography.

For example, Figure 2 right presents the energy spectrum of the atmosphere at various levels, it can be seen that the various properties of the atmosphere affect the spectrum at various wavelengths. Other important point is that the infrared solar heat flux can be approximated by a black body at 5523,15 K.

3.2.1.2. Infrared Thermography

The infrared thermography is a way to measure the thermal radiation received from a medium. With that information about the electromagnetic flux it is possible to estimate the surface temperature of the body, see section 3.2.1.1. Various types of detector can assure the measure of the electromagnetic radiation.

Those different detectors can take various forms and/or manufacturing process. For our research purpose we use uncooled infrared camera using a matrix of microbolometers detectors. A microbolometer, as a lot of transducers, converts a radiation in electric current used to represent the physical quantity (here the heat flux).

This field of activity includes the use and the improvement of vision system, like in [3].

3.2.2. Heat transfer theory

Once the acquisition process is done, it is useful to model the heat conduction inside the cartesian domain Ω . Note that in opaque solid medium the heat conduction is the only mode of heat transfer. Proposed by Jean Baptiste Biot in 1804 and experimentally demonstrated by Joseph Fourier in 1821, the Fourier Law describes the heat flux inside a solid, cf Equation (22).

$$\varphi = k \nabla T \quad X \in \Omega \quad (19)$$

Where k is the thermal conductivity in $\text{W.m}^{-1}.\text{K}^{-1}$, ∇ is the gradient operator and φ is the heat flux density in W.m^{-2} . This law illustrates the first principle of thermodynamic (law of conservation of energy) and implies the second principle (irreversibility of the phenomenon), from this law it can be seen that the heat flux always goes from hot area to cold area.

An energy balance with respect to the first principle drives to the expression of the heat conduction in all point of the domain Ω , cf Equation (23). This equation has been proposed by Joseph Fourier in 1811.

$$\rho C \frac{\partial T(X, t)}{\partial t} = \nabla \cdot (k \nabla T) + P \quad X \in \Omega \quad (20)$$

With $\nabla \cdot ()$ the divergence operator, C the specific heat capacity in $\text{J.kg}^{-1}.\text{K}^{-1}$, ρ the volumetric mass density in kg.m^{-3} , X the space variable $X = \{x, y, z\}$ and P a possible internal heat production in W.m^{-3} .

To solve the system (23), it is necessary to express the boundaries conditions of the system. With the developments presented in section 3.2.1.1 and the Fourier's law it is possible, for example, to express the thermal radiation and the convection phenomenon which can occur at $\partial\Omega$ the system boundaries, cf Equation (24).

$$\varphi = k \nabla T \cdot n = \underbrace{h (T_{\text{fluid}} - T_{\text{Boundary}})}_{\text{Convection}} + \underbrace{\epsilon \sigma_s (T_{\text{environnement}}^4 - T_{\text{Boundary}}^4)}_{\text{Radiation}} + \varphi_0 \quad X \in \partial\Omega \quad (21)$$

Equation (24) is the so called Robin condition on the boundary $\partial\Omega$, where n is the normal, h the convective heat transfer coefficient in $\text{W.m}^{-2}.\text{K}^{-1}$ and φ_0 an external energy contribution W.m^{-2} , in cases where the external energy contribution is artificial and controlled we call it active thermography (spotlight etc...) in the contrary it is called passive thermography (direct solar heat flux).

The systems presented in the different sections above (3.2.1 to 3.2.2) are useful to build physical models in order to represents the measured quantity. To estimate key parameters, as the conductivity, one way to do is the model inversion, the next section will introduce that principle.

3.2.3. Inverse model for parameters estimation

Lets take any model A which can for example represent the conductive heat transfer in a medium, the model is solved for a parameter vector P and it results another vector b , cf Equation (25). For example if A represents the heat transfer, b can be the temperature evolution.

$$AP = b \quad (22)$$

With A a matrix of size $n \times m$, P a vector of size m and b of size n , preferentially $n \gg m$. This model is called direct model, the inverse model consist to find a vector P which satisfy the results b of the direct model. For that we need to inverse the matrix A , cf Equation (26).

$$P = A^{-1}b \quad (23)$$

Here we want find the solution AP which is closest to the acquired measures M , Equation (27).

$$AP \approx M \quad (24)$$

To do that it is important to respect the well posed condition established by Jacques Hadamard in 1902

- A solution exists.
- The solution is unique.
- The solution's behavior changes continuously with the initial conditions.

Unfortunately those condition are rarely respected in our field of study. That is why we don't solve directly the system (27) but we minimise the quadratic cost function (28) which represents the Legendre-Gauss least square algorithm for linear problems.

$$\min_P (\|AP - \mathcal{M}\|^2) = \min_P (\mathcal{F}) \quad (25)$$

Where \mathcal{F} can be a product of matrix.

$$\mathcal{F} = [AP - \mathcal{M}]^T [AP - \mathcal{M}]$$

In some case the problem is still ill-posed and need to be regularized for example using the Tikhonov regularization. An elegant way to minimize the cost function \mathcal{F} is compute the gradient, Equation (29) and find where it is equal to zero.

$$\nabla \mathcal{F}(P) = 2 \left[-\frac{\partial AP^T}{\partial P} \right] [AP - \mathcal{M}] = 2J(P)^T [AP - \mathcal{M}] \quad (26)$$

Where J is the sensitivity matrix of the model A to its parameter vector P .

Until now the inverse method proposed is valid only when the model A is linearly dependent of its parameter P , for the heat equation it is the case when you want to estimate the external heat flux, φ_0 in equation 24. For all the other parameters, like the conductivity k the model is non-linearly dependant of its parameter P . For such case the use of iterative algorithm is needed, for example the Levenberg-Marquardt algorithm, cf Equation (30).

$$P^{k+1} = P^k + [(J^k)^T J^k + \mu^k \Omega^k]^{-1} (J^k)^T [\mathcal{M} - A(P^k)] \quad (27)$$

Equation (30) is solved iteratively at each loop k . Some of our results with such linear or non linear method can be seen in [4] or [2], more specifically [1] is a custom implementation of the Levenberg-Marquardt algorithm based on the adjoint method (developed by Jacques Louis Lions in 1968) coupled to the conjugate gradient algorithm to estimate wide properties field in a medium.

3.3. Reflectometry-based methods for electrical engineering and for civil engineering

The fast development of electronic devices in modern engineering systems involves more and more connections through cables, and consequently, with an increasing number of connection failures. Wires and connectors are subject to ageing and degradation, sometimes under severe environmental conditions. In many applications, the reliability of electrical connexions is related to the quality of production or service, whereas in critical applications reliability becomes also a safety issue. It is thus important to design smart diagnosis systems able to detect connection defects in real time. This fact has motivated research projects on methods for fault diagnosis in this field. Some of these projects are based on techniques of reflectometry, which consist in injecting waves into a cable or a network and in analyzing the reflections. Depending on the injected waveforms and on the methods of analysis, various techniques of reflectometry are available. They all have the common advantage of being non destructive.

At Inria the research activities on reflectometry started within the SISYPHE EPI several years ago and now continue in the I4S EPI. Our most notable contribution in this area is a method based on the *inverse scattering* theory for the computation of *distributed characteristic impedance* along a cable from reflectometry measurements [14], [11], [61]. It provides an efficient solution for the diagnosis of *soft* faults in electrical cables, like in the example illustrated in Figure 3. While most reflectometry methods for fault diagnosis are based on the detection and localization of impedance discontinuity, our method yielding the spatial profile of the characteristic impedance is particularly suitable for the diagnosis of soft faults *with no or weak impedance discontinuities*.

Fault diagnosis for wired networks have also been studied in Inria [63], [59]. The main results concern, on the one hand, simple star-shaped networks from measurements made at a single node, on the other hand, complex networks of arbitrary topological structure with complete node observations.

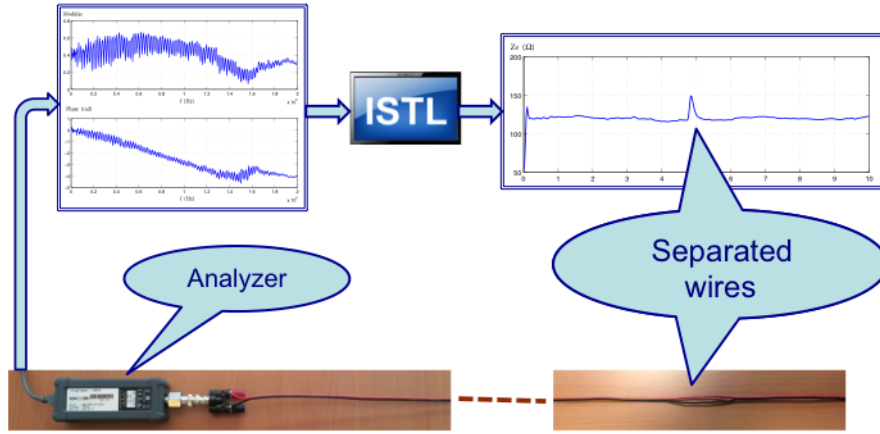


Figure 3. Inverse scattering software (ISTL) for cable soft fault diagnosis.

Though initially our studies on reflectometry were aiming at applications in electrical engineering, since the creation of the I4S team, we are also investigating applications in the field of civil engineering, by using electrical cables as sensors for monitoring changes in mechanical structures.

What follows is about some basic elements on mathematical equations of electric cables and networks, the main approach we follow in our study, and our future research directions.

3.3.1. Mathematical model of electric cables and networks

A cable excited by a signal generator can be characterized by the telegrapher's equations [60]

$$\begin{aligned} \frac{\partial}{\partial z} V(t, z) + L(z) \frac{\partial}{\partial t} I(t, z) + R(z) I(t, z) &= 0 \\ \frac{\partial}{\partial z} I(t, z) + C(z) \frac{\partial}{\partial t} V(t, z) + G(z) V(t, z) &= 0 \end{aligned} \quad (28)$$

where t represents the time, z is the longitudinal coordinate along the cable, $V(t, z)$ and $I(t, z)$ are respectively the voltage and the current in the cable at the time instant t and at the position z , $R(z)$, $L(z)$, $C(z)$ and $G(z)$ denote respectively the series resistance, the inductance, the capacitance and the shunt conductance per unit length of the cable at the position z . The left end of the cable (corresponding to $z = a$) is connected to a voltage source $V_s(t)$ with internal impedance R_s . The quantities $V_s(t)$, R_s , $V(t, a)$ and $I(t, a)$ are related by

$$V(t, a) = V_s(t) - R_s I(t, a). \quad (29)$$

At the right end of the cable (corresponding to $z = b$), the cable is connected to a load of impedance R_L , such that

$$V(t, b) = R_L I(t, b). \quad (30)$$

One way for deriving the above model is to spatially discretize the cable and to characterize each small segment with 4 basic lumped parameter elements for the j -th segment: a resistance ΔR_j , an inductance ΔL_j , a capacitance ΔC_j and a conductance ΔG_j . The entire circuit is described by a system of ordinary differential equations. When the spatial discretization step size tends to zero, the limiting model leads to the telegrapher's equations.

A wired network is a set of cables connected at some nodes, where loads and sources can also be connected. Within each cable the current and voltage satisfy the telegrapher's equations, whereas at each node the current and voltage satisfy the Kirchhoff's laws, unless in case of connector failures.

3.3.2. The inverse scattering theory applied to cables

The inverse scattering transform was developed during the 1970s-1980s for the analysis of some nonlinear partial differential equations [58]. The visionary idea of applying this theory to solving the cable inverse problem goes also back to the 1980s [57]. After having completed some theoretic results directly linked to practice [14], [61], we started to successfully apply the inverse scattering theory to cable soft fault diagnosis, in collaboration with GEEPS-SUPELEC [11].

To link electric cables to the inverse scattering theory, the telegrapher's equations are transformed in a few steps to fit into a particular form studied in the inverse scattering theory. The Fourier transform is first applied to obtain a frequency domain model, the spatial coordinate z is then replaced by the propagation time

$$x(z) = \int_0^z \sqrt{L(s)C(s)} ds$$

and the frequency domain variables $V(\omega, x)$, $I(\omega, x)$ are replaced by the pair

$$\begin{aligned} \nu_1(\omega, x) &= \frac{1}{2} \left[Z_0^{-\frac{1}{2}}(x) U(\omega, x) - Z_0^{\frac{1}{2}}(x) I(\omega, x) \right] \\ \nu_2(\omega, x) &= \frac{1}{2} \left[Z_0^{-\frac{1}{2}}(x) U(\omega, x) + Z_0^{\frac{1}{2}}(x) I(\omega, x) \right] \end{aligned} \quad (31)$$

with

$$Z_0(x) = \sqrt{\frac{L(x)}{C(x)}}. \quad (32)$$

These transformations lead to the Zakharov-Shabat equations

$$\begin{aligned} \frac{d\nu_1(\omega, x)}{dx} + ik\nu_1(\omega, x) &= q^*(x)\nu_1(\omega, x) + q^+(x)\nu_2(\omega, x) \\ \frac{d\nu_2(\omega, x)}{dx} - ik\nu_2(\omega, x) &= q^-(x)\nu_1(\omega, x) - q^*(x)\nu_2(\omega, x) \end{aligned} \quad (33)$$

with

$$\begin{aligned}
 q^{\pm}(x) &= -\frac{1}{4} \frac{d}{dx} \left[\ln \frac{L(x)}{C(x)} \right] \mp \frac{1}{2} \left[\frac{R(x)}{L(x)} - \frac{G(x)}{C(x)} \right] \\
 &= -\frac{1}{2Z_0(x)} \frac{d}{dx} Z_0(x) \mp \frac{1}{2} \left[\frac{R(x)}{L(x)} - \frac{G(x)}{C(x)} \right] \\
 q^*(x) &= \frac{1}{2} \left[\frac{R(x)}{L(x)} + \frac{G(x)}{C(x)} \right].
 \end{aligned} \tag{34}$$

These equations have been well studied in the inverse scattering theory, for the purpose of determining partly the “potential functions” $q^{\pm}(x)$ and $q^*(x)$ from the scattering data matrix, which turns out to correspond to the data typically collected with reflectometry instruments. For instance, it is possible to compute the function $Z_0(x)$ defined in (35), often known as the characteristic impedance, from the reflection coefficient measured at one end of the cable. Such an example is illustrated in Figure 3. Any fault affecting the characteristic impedance, like in the example of Figure 3 caused by a slight geometric deformation, can thus be efficiently detected, localized and characterized.

3.4. Research Program

The research will first focus on the extension and implementation of current techniques as developed in I4S and IFSTTAR. Before doing any temperature rejection on large scale structures as planned, we need to develop good and accurate models of thermal fields. We also need to develop robust and efficient versions of our algorithms, mainly the subspace algorithms before envisioning linking them with physical models. Briefly, we need to mature our statistical toolset as well as our physical modeling before mixing them together later on.

3.4.1. Vibration analysis and monitoring

3.4.1.1. Direct vibration modeling under temperature changes

This task builds upon what has been achieved in the CONSTRUCTIF project, where a simple formulation of the temperature effect has been exhibited, based on relatively simple assumptions. The next step is to generalize this modeling to a realistic large structure under complex thermal changes. Practically, temperature and resulting structural prestress and pre strains of thermal origin are not uniform and civil structures are complex. This leads to a fully 3D temperature field, not just a single value. Inertia effects also forbid a trivial prediction of the temperature based on current sensor outputs while ignoring past data. On the other side, the temperature is seen as a nuisance. That implies that any damage detection procedure has first to correct the temperature effect prior to any detection.

Modeling vibrations of structures under thermal prestress does and will play an important role in the static correction of kinematic measurements, in health monitoring methods based on vibration analysis as well as in durability and in the active or semi-active control of civil structures that by nature are operated under changing environmental conditions. As a matter of fact, using temperature and dynamic models the project aims at correcting the current vibration state from induced temperature effects, such that damage detection algorithms rely on a comparison of this thermally corrected current vibration state with a reference state computed or measured at a reference temperature. This approach is expected to cure damage detection algorithms from the environmental variations.

I4S will explore various ways of implementing this concept, notably within the FUI SIPRIS project.

3.4.1.2. Damage localization algorithms (in the case of localized damages such as cracks)

During the CONSTRUCTIF project, both feasibility and efficiency of some damage detection and localization algorithms were proved. Those methods are based on the tight coupling of statistical algorithms with finite element models. It has been shown that effective localization of some damaged elements was possible, and this was validated on a numerical simulated bridge deck model. Still, this approach has to be validated on real structures.

On the other side, new localization algorithms are currently investigated such as the one developed conjointly with University of Boston and tested within the framework of FP7 ISMS project. These algorithms will be implemented and tested on the PEGASE platform as well as all our toolset.

When possible, link with temperature rejection will be done along the lines of what has been achieved in the CONSTRUCTIF project.

3.4.1.3. Uncertainty quantification for system identification algorithms

Some emphasis will be put on expressing confidence intervals for system identification. It is a primary goal to take into account the uncertainty within the identification procedure, using either identification algorithms derivations or damage detection principles. Such algorithms are critical for both civil and aeronautical structures monitoring. It has been shown that confidence intervals for estimation parameters can theoretically be related to the damage detection techniques and should be computed as a function of the Fisher information matrix associated to the damage detection test. Based on those assumptions, it should be possible to obtain confidence intervals for a large class of estimates, from damping to finite elements models. Uncertainty considerations are also deeply investigated in collaboration with Dassault Aviation in Mellinger PhD thesis or with Northeastern University, Boston, within Gallegos PhD thesis.

3.4.2. Reflectometry-based methods for civil engineering structure health monitoring

The inverse scattering method we developed is efficient for the diagnosis of all soft faults affecting the characteristic impedance, the major parameter of a cable. In some particular applications, however, faults would rather affect the series resistance (ohmic loss) or shunt conductance (leakage loss) than the characteristic impedance. The first method we developed for the diagnosis of such losses had some numerical stability problems. The new method is much more reliable and efficient. It is also important to develop efficient solutions for long cables, up to a few kilometers.

For wired networks, the methods we already developed cover either the case of simple networks with a single node measurement or the case of complex networks with complete node measurements. Further developments are still necessary for intermediate situations.

In terms of applications, the use of electric cables as sensors for the monitoring of various structures is still at its beginning. We believe that this new technology has a strong potential in different fields, notably in civil engineering and in materials engineering.

3.4.3. Non Destructive testing of CFRP bonded on concrete through active thermography

Strengthening or retrofitting of reinforced concrete structures by externally bonded fibre-reinforced polymer (FRP) systems is now a commonly accepted and widespread technique. However, the use of bonding techniques always implies following rigorous installation procedures. The number of carbon fibre-reinforced polymer (CFRP) sheets and the glue layer thickness are designed by civil engineers to address strengthening objectives. Moreover, professional crews have to be trained accordingly in order to ensure the durability and long-term performance of the FRP reinforcements. Conformity checking through an ‘in situ’ verification of the bonded FRP systems is then highly desirable. The quality control programme should involve a set of adequate inspections and tests. Visual inspection and acoustic sounding (hammer tap) are commonly used to detect delaminations (disbonds). Nevertheless, these techniques are unable to provide sufficient information about the depth (in case of multilayered composite) and width of the disbanded areas. They are also incapable of evaluating the degree of adhesion between the FRP and the substrate (partial delamination, damage of the resin and poor mechanical properties of the resin). Consequently, rapid and efficient inspection methods are required. Among the non-destructive (NDT) methods currently under study, active infrared thermography is investigated due to its ability to be used in the field. In such context and to reach the aim of having an in situ efficient NDT method, we carried out experiments and subsequent data analysis using thermal excitation. Image processing, inverse thermal modelling and 3D numerical simulations are used and then applied to experimental data obtained in laboratory conditions.

3.4.4. IRSHM: Multi-Sensing system for outdoor thermal monitoring

Ageing of transport infrastructures combined with traffic and climatic solicitations contribute to the reduction of their performances. To address and quantify the resilience of civil engineering structure, investigations on robust, fast and efficient methods are required. Among research works carried out at IFSTTAR, methods for long term monitoring face an increasing demand. Such works take benefits of this last decade technological progresses in ICT domain.

Thanks to IFSTTAR years of experience in large scale civil engineering experiment, I4S is able to perform very long term thermal monitoring of structures exposed to environmental condition, as the solar heat flux, natural convection or seasonal perturbation. Informations system are developed to asses the data acquisition and researchers work on the quantification of the data to detect flaws emergence on structure, those techniques are also used to diagnose thermal insulation of buildings or monitoring of guided transport infrastructures, Figure 4 left. Experiments are carried out on a real transport infrastructure open to traffic and buildings. The detection of the inner structure of the deck is achieved by image processing techniques (as FFT), principal component thermography (PCT), Figure 4 right, or characterization of the inner structure thanks to an original image processing approach.

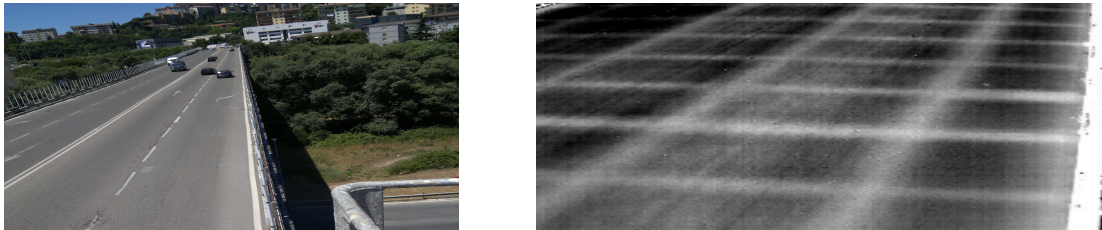


Figure 4. Left: Image in the visible spectrum of the deck surface - Right: PCT result on a bridge deck

For the next few years, I4S is actively implied in the SenseCity EQUIPEX (<http://sense-city.ifsttar.fr/>) where our informations systems are used to monitor a mini-city replica, Figure 5 .

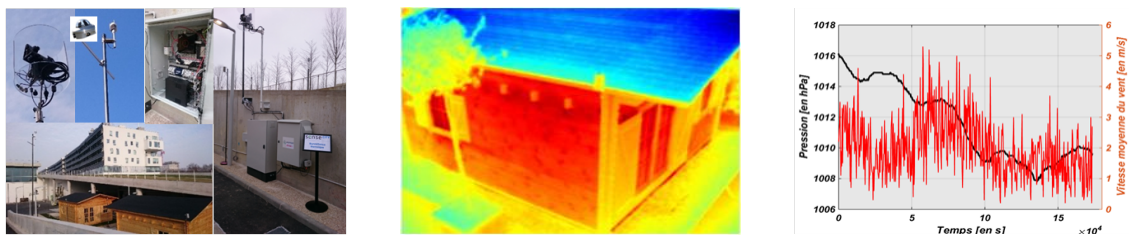


Figure 5. Various view and results of the SenseCity experimentation site - (site and hardware view, IR imaging, Environmental Monitoring)

3.4.5. R5G: The 5th Generation Road

The road has to reinvent itself periodically in response to innovations, societal issues and rising user expectations. The 5th Generation Road (R5G) focuses firmly on the future and sets out to be automated, safe, sustainable and suited to travel needs. Several research teams are involved in work related to this flagship

project for IFSTTAR, which is a stakeholder in the Forever Open Road. Through its partnership with the COSYS (IFSTTAR) department, I4S is fully implicated in the development of the 5th Generation Road.

Most of the innovations featured in R5G are now mature, for example communication and few solutions for energy exchange between the infrastructure, the vehicle and the network manager; recyclable materials with the potential for self-diagnosis and repair, a pavement surface that remains permanently optimal irrespective of climatic variations... Nevertheless, implementing them on an industrial scale at a reasonable cost still represents a real challenge. Consultation with the stakeholders (researchers, industry, road network owners and users) has already established the priorities for the creation of full-scale demonstrators. The next stages are to achieve synergy between the technologies tested by the demonstrators, to manage the interfaces and get society to adopt R5G.

IPSO Project-Team

3. Research Program

3.1. Structure-preserving numerical schemes for solving ordinary differential equations

Participants: Francois Castella, Philippe Chartier, Erwan Faou.

ordinary differential equation, numerical integrator, invariant, Hamiltonian system, reversible system, Lie-group system

In many physical situations, the time-evolution of certain quantities may be written as a Cauchy problem for a differential equation of the form

$$\begin{aligned} y'(t) &= f(y(t)), \\ y(0) &= y_0. \end{aligned} \tag{35}$$

For a given y_0 , the solution $y(t)$ at time t is denoted $\varphi_t(y_0)$. For fixed t , φ_t becomes a function of y_0 called the *flow* of (1). From this point of view, a numerical scheme with step size h for solving (1) may be regarded as an approximation Φ_h of φ_h . One of the main questions of *geometric integration* is whether *intrinsic* properties of φ_t may be passed on to Φ_h .

This question can be more specifically addressed in the following situations:

3.1.1. Reversible ODEs

The system (1) is said to be ρ -reversible if there exists an involutive linear map ρ such that

$$\rho \circ \varphi_t = \varphi_t^{-1} \circ \rho = \varphi_{-t} \circ \rho. \tag{36}$$

It is then natural to require that Φ_h satisfies the same relation. If this is so, Φ_h is said to be *symmetric*. Symmetric methods for reversible systems of ODEs are just as much important as *symplectic* methods for Hamiltonian systems and offer an interesting alternative to symplectic methods.

3.1.2. ODEs with an invariant manifold

The system (1) is said to have an invariant manifold g whenever

$$\mathcal{M} = \{y \in \mathbb{R}^n; g(y) = 0\} \tag{37}$$

is kept *globally* invariant by φ_t . In terms of derivatives and for sufficiently differentiable functions f and g , this means that

$$\forall y \in \mathcal{M}, g'(y)f(y) = 0.$$

As an example, we mention Lie-group equations, for which the manifold has an additional group structure. This could possibly be exploited for the space-discretisation. Numerical methods amenable to this sort of problems have been reviewed in a recent paper [37] and divided into two classes, according to whether they use g explicitly or through a projection step. In both cases, the numerical solution is forced to live on the manifold at the expense of some Newton's iterations.

3.1.3. Hamiltonian systems

Hamiltonian problems are ordinary differential equations of the form:

$$\begin{aligned}\dot{p}(t) &= -\nabla_q H(p(t), q(t)) \in \mathbb{R}^d \\ \dot{q}(t) &= \nabla_p H(p(t), q(t)) \in \mathbb{R}^d\end{aligned}\tag{38}$$

with some prescribed initial values $(p(0), q(0)) = (p_0, q_0)$ and for some scalar function H , called the Hamiltonian. In this situation, H is an invariant of the problem. The evolution equation (4) can thus be regarded as a differential equation on the manifold

$$\mathcal{M} = \{(p, q) \in \mathbb{R}^d \times \mathbb{R}^d; H(p, q) = H(p_0, q_0)\}.$$

Besides the Hamiltonian function, there might exist other invariants for such systems: when there exist d invariants in involution, the system (4) is said to be *integrable*. Consider now the parallelogram P originating from the point $(p, q) \in \mathbb{R}^{2d}$ and spanned by the two vectors $\xi \in \mathbb{R}^{2d}$ and $\eta \in \mathbb{R}^{2d}$, and let $\omega(\xi, \eta)$ be the sum of the *oriented* areas of the projections over the planes (p_i, q_i) of P ,

$$\omega(\xi, \eta) = \xi^T J \eta,$$

where J is the *canonical symplectic matrix*

$$J = \begin{bmatrix} 0 & I_d \\ -I_d & 0 \end{bmatrix}.$$

A continuously differentiable map g from \mathbb{R}^{2d} to itself is called *symplectic* if it preserves ω , i.e. if

$$\omega(g'(p, q)\xi, g'(p, q)\eta) = \omega(\xi, \eta).$$

A fundamental property of Hamiltonian systems is that their exact flow is symplectic. Integrable Hamiltonian systems behave in a very remarkable way: as a matter of fact, their invariants persist under small perturbations, as shown in the celebrated theory of Kolmogorov, Arnold and Moser. This behavior motivates the introduction of *symplectic* numerical flows that share most of the properties of the exact flow. For practical simulations of Hamiltonian systems, symplectic methods possess an important advantage: the error-growth as a function of time is indeed linear, whereas it would typically be quadratic for non-symplectic methods.

3.1.4. Differential-algebraic equations

Whenever the number of differential equations is insufficient to determine the solution of the system, it may become necessary to solve the differential part and the constraint part altogether. Systems of this sort are called differential-algebraic systems. They can be classified according to their index, yet for the purpose of this expository section, it is enough to present the so-called index-2 systems

$$\begin{aligned}\dot{y}(t) &= f(y(t), z(t)), \\ 0 &= g(y(t)),\end{aligned}\tag{39}$$

where initial values $(y(0), z(0)) = (y_0, z_0)$ are given and assumed to be consistent with the constraint manifold. By constraint manifold, we imply the intersection of the manifold

$$\mathcal{M}_1 = \{y \in \mathbb{R}^n, g(y) = 0\}$$

and of the so-called hidden manifold

$$\mathcal{M}_2 = \{(y, z) \in \mathbb{R}^n \times \mathbb{R}^m, \frac{\partial g}{\partial y}(y)f(y, z) = 0\}.$$

This manifold $\mathcal{M} = \mathcal{M}_1 \cap \mathcal{M}_2$ is the manifold on which the exact solution $(y(t), z(t))$ of (5) lives.

There exists a whole set of schemes which provide a numerical approximation lying on \mathcal{M}_1 . Furthermore, this solution can be projected on the manifold \mathcal{M} by standard projection techniques. However, it is worth mentioning that a projection destroys the symmetry of the underlying scheme, so that the construction of a symmetric numerical scheme preserving \mathcal{M} requires a more sophisticated approach.

3.2. Highly-oscillatory systems

Participants: Francois Castella, Philippe Chartier, Nicolas Crouseilles, Erwan Faou, Florian Mehats, Mohammed Lemou.

second-order ODEs, oscillatory solutions, Schrödinger and wave equations, step size restrictions.

In applications to molecular dynamics or quantum dynamics for instance, the right-hand side of (1) involves *fast* forces (short-range interactions) and *slow* forces (long-range interactions). Since *fast* forces are much cheaper to evaluate than *slow* forces, it seems highly desirable to design numerical methods for which the number of evaluations of slow forces is not (at least not too much) affected by the presence of fast forces.

A typical model of highly-oscillatory systems is the second-order differential equations

$$\ddot{q} = -\nabla V(q) \quad (40)$$

where the potential $V(q)$ is a sum of potentials $V = W + U$ acting on different time-scales, with $\nabla^2 W$ positive definite and $\|\nabla^2 W\| \gg \|\nabla^2 U\|$. In order to get a bounded error propagation in the linearized equations for an explicit numerical method, the step size must be restricted according to

$$h\omega < C,$$

where C is a constant depending on the numerical method and where ω is the highest frequency of the problem, i.e. in this situation the square root of the largest eigenvalue of $\nabla^2 W$. In applications to molecular dynamics for instance, *fast* forces deriving from W (short-range interactions) are much cheaper to evaluate than *slow* forces deriving from U (long-range interactions). In this case, it thus seems highly desirable to design numerical methods for which the number of evaluations of slow forces is not (at least not too much) affected by the presence of fast forces.

Another prominent example of highly-oscillatory systems is encountered in quantum dynamics where the Schrödinger equation is the model to be used. Assuming that the Laplacian has been discretized in space, one indeed gets the *time*-dependent Schrödinger equation:

$$i\dot{\psi}(t) = \frac{1}{\varepsilon} H(t)\psi(t), \quad (41)$$

where $H(t)$ is finite-dimensional matrix and where ε typically is the square-root of a mass-ratio (say electron/ion for instance) and is small ($\varepsilon \approx 10^{-2}$ or smaller). Through the coupling with classical mechanics ($H(t)$ is obtained by solving some equations from classical mechanics), we are faced once again with two different time-scales, 1 and ε . In this situation also, it is thus desirable to devise a numerical method able to advance the solution by a time-step $h > \varepsilon$.

3.3. Geometric schemes for the Schrödinger equation

Participants: Francois Castella, Philippe Chartier, Erwan Faou, Florian Mehats.

Schrödinger equation, variational splitting, energy conservation.

Given the Hamiltonian structure of the Schrödinger equation, we are led to consider the question of energy preservation for time-discretization schemes.

At a higher level, the Schrödinger equation is a partial differential equation which may exhibit Hamiltonian structures. This is the case of the time-dependent Schrödinger equation, which we may write as

$$i\varepsilon \frac{\partial \psi}{\partial t} = H\psi, \quad (42)$$

where $\psi = \psi(x, t)$ is the wave function depending on the spatial variables $x = (x_1, \dots, x_N)$ with $x_k \in \mathbb{R}^d$ (e.g., with $d = 1$ or 3 in the partition) and the time $t \in \mathbb{R}$. Here, ε is a (small) positive number representing the scaled Planck constant and i is the complex imaginary unit. The Hamiltonian operator H is written

$$H = T + V$$

with the kinetic and potential energy operators

$$T = - \sum_{k=1}^N \frac{\varepsilon^2}{2m_k} \Delta_{x_k} \quad \text{and} \quad V = V(x),$$

where $m_k > 0$ is a particle mass and Δ_{x_k} the Laplacian in the variable $x_k \in \mathbb{R}^d$, and where the real-valued potential V acts as a multiplication operator on ψ .

The multiplication by i in (8) plays the role of the multiplication by J in classical mechanics, and the energy $\langle \psi | H | \psi \rangle$ is conserved along the solution of (8), using the physicists' notations $\langle u | A | u \rangle = \langle u, Au \rangle$ where $\langle \cdot, \cdot \rangle$ denotes the Hermitian L^2 -product over the phase space. In quantum mechanics, the number N of particles is very large making the direct approximation of (8) very difficult.

The numerical approximation of (8) can be obtained using projections onto submanifolds of the phase space, leading to various PDEs or ODEs: see [41], [40] for reviews. However the long-time behavior of these approximated solutions is well understood only in this latter case, where the dynamics turns out to be finite dimensional. In the general case, it is very difficult to prove the preservation of qualitative properties of (8) such as energy conservation or growth in time of Sobolev norms. The reason for this is that backward error analysis is not directly applicable for PDEs. Overwhelming these difficulties is thus a very interesting challenge.

A particularly interesting case of study is given by symmetric splitting methods, such as the Strang splitting:

$$\psi_1 = \exp(-i(\delta t)V/2) \exp(i(\delta t)\Delta) \exp(-i(\delta t)V/2) \psi_0 \quad (43)$$

where δt is the time increment (we have set all the parameters to 1 in the equation). As the Laplace operator is unbounded, we cannot apply the standard methods used in ODEs to derive long-time properties of these schemes. However, its projection onto finite dimensional submanifolds (such as Gaussian wave packets space or FEM finite dimensional space of functions in x) may exhibit Hamiltonian or Poisson structure, whose long-time properties turn out to be more tractable.

3.4. High-frequency limit of the Helmholtz equation

Participant: Francois Castella.

waves, Helmholtz equation, high oscillations.

The Helmholtz equation models the propagation of waves in a medium with variable refraction index. It is a simplified version of the Maxwell system for electro-magnetic waves.

The high-frequency regime is characterized by the fact that the typical wavelength of the signals under consideration is much smaller than the typical distance of observation of those signals. Hence, in the high-frequency regime, the Helmholtz equation at once involves highly oscillatory phenomena that are to be described in some asymptotic way. Quantitatively, the Helmholtz equation reads

$$i\alpha_\varepsilon u_\varepsilon(x) + \varepsilon^2 \Delta_x u_\varepsilon + n^2(x)u_\varepsilon = f_\varepsilon(x). \quad (44)$$

Here, ε is the small adimensional parameter that measures the typical wavelength of the signal, $n(x)$ is the space-dependent refraction index, and $f_\varepsilon(x)$ is a given (possibly dependent on ε) source term. The unknown is $u_\varepsilon(x)$. One may think of an antenna emitting waves in the whole space (this is the $f_\varepsilon(x)$), thus creating at any point x the signal $u_\varepsilon(x)$ along the propagation. The small $\alpha_\varepsilon > 0$ term takes into account damping of the waves as they propagate.

One important scientific objective typically is to describe the high-frequency regime in terms of *rays* propagating in the medium, that are possibly refracted at interfaces, or bounce on boundaries, etc. Ultimately, one would like to replace the true numerical resolution of the Helmholtz equation by that of a simpler, asymptotic model, formulated in terms of rays.

In some sense, and in comparison with, say, the wave equation, the specificity of the Helmholtz equation is the following. While the wave equation typically describes the evolution of waves between some initial time and some given observation time, the Helmholtz equation takes into account at once the propagation of waves over *infinitely long* time intervals. Qualitatively, in order to have a good understanding of the signal observed in some bounded region of space, one readily needs to be able to describe the propagative phenomena in the whole space, up to infinity. In other words, the “rays” we refer to above need to be understood from the initial time up to infinity. This is a central difficulty in the analysis of the high-frequency behaviour of the Helmholtz equation.

3.5. From the Schrödinger equation to Boltzmann-like equations

Participant: Francois Castella.

Schrödinger equation, asymptotic model, Boltzmann equation.

The Schrödinger equation is the appropriate way to describe transport phenomena at the scale of electrons. However, for real devices, it is important to derive models valid at a larger scale.

In semi-conductors, the Schrödinger equation is the ultimate model that allows to obtain quantitative information about electronic transport in crystals. It reads, in convenient adimensional units,

$$i\partial_t \psi(t, x) = -\frac{1}{2} \Delta_x \psi + V(x)\psi, \quad (45)$$

where $V(x)$ is the potential and $\psi(t, x)$ is the time- and space-dependent wave function. However, the size of real devices makes it important to derive simplified models that are valid at a larger scale. Typically, one wishes to have kinetic transport equations. As is well-known, this requirement needs one to be able to describe “collisions” between electrons in these devices, a concept that makes sense at the macroscopic level, while it does not at the microscopic (electronic) level. Quantitatively, the question is the following: can one obtain the Boltzmann equation (an equation that describes collisional phenomena) as an asymptotic model for the Schrödinger equation, along the physically relevant micro-macro asymptotics? From the point of view of modelling, one wishes here to understand what are the “good objects”, or, in more technical words, what are the relevant “cross-sections”, that describe the elementary collisional phenomena. Quantitatively, the Boltzmann equation reads, in a simplified, linearized, form :

$$\partial_t f(t, x, v) = \int_{\mathbf{R}^3} \sigma(v, v') [f(t, x, v') - f(t, x, v)] dv'. \quad (46)$$

Here, the unknown is $f(x, v, t)$, the probability that a particle sits at position x , with a velocity v , at time t . Also, $\sigma(v, v')$ is called the cross-section, and it describes the probability that a particle “jumps” from velocity v to velocity v' (or the converse) after a collision process.

KERDATA Project-Team

3. Research Program

3.1. Research axis 1: Convergence of Extreme-Scale Computing and Big Data Infrastructures

The tools and cultures of High Performance Computing and Big Data Analytics have evolved in divergent ways. This is to the detriment of both. However, big computations still generate and are needed to analyze Big Data. As scientific research increasingly depends on both high-speed computing and data analytics, the potential interoperability and scaling convergence of these two eco-systems is crucial to the future. Our objective for the next years is premised on the idea that we must begin to systematically map out and account for the ways in which the major issues associated with Big Data intersect with, impinge upon, and potentially change the plans that are now being laid for achieving Exascale computing.

Collaboration. *This axis is addressed in close collaboration with **María Pérez** (UPM), **Rob Ross** (ANL), **Toni Cortes** (BSC), **Bogdan Nicolae** (formerly at IBM Research, now at Huawei Research).*

Relevant groups with similar interests are the following ones.

- *The group of **Jack Dongarra**, Innovative Computing Laboratory at University of Tennessee/Oak Ridge National Laboratory, working on joint tools Exascale Computing and Big Data.*
- *The group of **Satoshi Matsuoka**, Tokyo Institute of Technology, working on system software for Clouds and HPC.*
- *The group of **Franck Cappello** at Argonne National Laboratory/NCSA working on on-demand data analytics and storage for extreme-scale simulations and experiments.*

3.1.1. High-performance storage for concurrent Big Data applications

We argue that storage is a plausible pathway to convergence. In this context, we plan to focus on the needs of concurrent Big Data applications that require high-performance storage, as well as transaction support. Although blobs (binary large objects) are an increasingly popular storage model for such applications, state-of-the-art blob storage systems offer no transaction semantics. This demands users to coordinate data access carefully in order to avoid race conditions, inconsistent writes, overwrites and other problems that cause erratic behavior.

We argue there is a gap between existing storage solutions and application requirements, which limits the design of transaction-oriented applications. In this context, one idea on which we plan to focus our efforts is exploring how blob storage systems could provide built-in, multi-blob transactions, while retaining sequential consistency and high throughput under heavy access concurrency.

The early principles of this research direction have already raised interest from our partners at ANL (Rob Ross) and UPM (María Pérez) for potential collaborations. In this direction, the acceptance of our paper on the Týr transactional blob storage system as a Best Student Paper Award Finalist at the SC16 conference [10] is a very encouraging step.

3.1.2. Big Data analytics on Exascale HPC machines

Big Data analytics is another interesting direction that we plan to explore, building on top of these converged storage architectures. Specifically, we will examine the ways in which Exascale infrastructures can be leveraged not only by HPC-centric, but also by scientific, cloud-centric applications. Many of the current state-of-the-art Big Data processing approaches, including Hadoop and Spark [41] are optimized to run on commodity machines. This impacts the mechanisms used to deal with failures and the limited network bandwidth.

A blind adoption of these systems on extreme-scale platforms would result in high overheads. It would therefore prevent users from fully benefiting from the high performance infrastructure. The objective that we set here is to explore design and implementation options for new data analytics systems that can exploit the features of extreme-scale HPC machines: multi-core nodes, multiple memory and storage technologies including a large memory, NVRAM, SSDs, etc.

3.2. Research axis 2: Advanced data processing on Clouds

The recent evolutions in the area of Big Data processing have pointed out some limitations of the initial Map-Reduce model. It is well suited for batch data processing, but less suited for real-time processing of dynamic data streams. New types of data-intensive applications emerge, e.g., for enterprises who need to perform analysis on their stream data in ways that can give fast results (i.e., in real time) at scale (e.g., click-stream analysis and network-monitoring log analysis). Similarly, scientists require fast and accurate data processing techniques in order to analyze their experimental data correctly at scale (e.g., collectively analysis of large data sets distributed in multiple geographically distributed locations).

Our plan is to revisit current data management techniques to cope with the volatile requirements of data-intensive applications on large-scale dynamic clouds in a cost-efficient way.

Collaboration. *This axis is addressed in close collaboration with **María Pérez** (UPM), **Kate Keahey** (ANL) and **Toni Cortes** (BSC).*

Relevant groups with similar interests include the following ones.

- *The **AMPLab**, UC Berkeley, USA, working on scheduling stream data applications in heterogeneous clouds.*
- *The group of **Ewa Deelman**, USC Information Sciences Institute, working on resource management for workflows in Clouds.*
- *The **XTRA** group, Nanyang Technological University, Singapore, working on resource provisioning for workflows in the cloud.*

3.2.1. Stream-oriented, Big Data processing on clouds

The state-of-the-art Hadoop Map-Reduce framework cannot deal with stream data applications, as it requires the data to be initially stored in a distributed file system in order to process them. To better cope with the above-mentioned requirements, several systems have been introduced for stream data processing such as Flink [36], Spark [41], Storm [42], and Google MillWheel [44]. These systems keep computation in memory to decrease latency, and preserve scalability by using data-partitioning or dividing the streams into a set of deterministic batch computations.

However, they are designed to work in dedicated environments and they do not consider the performance variability (i.e., network, I/O, etc.) caused by resource contention in the cloud. This variability may in turn cause high and unpredictable latency when output streams are transmitted to further analysis. Moreover, they overlook the dynamic nature of data streams and the volatility in their computation requirements. Finally, they still address failures in a best-effort manner.

Our objective is to investigate new approaches for reliable, stream Big Data processing on clouds. We will explore new mechanisms that expose resource heterogeneity (observed variability in resource utilization at runtime) when scheduling stream data applications. We will also investigate how to adapt to node failures automatically, and to adapt the failure handling techniques to the characteristics of the running application and to the root cause of failures.

3.2.2. Geographically distributed workflows on multi-site clouds

Many data processing jobs in data-intensive applications are modeled as workflows (i.e., as sets of tasks linked according to their data and computation dependencies) to facilitate the management and analysis of large volumes of data. With the fast growth of volumes of data to be handled at larger and larger scales, geographically distributed workflows are emerging as a natural data processing paradigm. This may bring several benefits: resilience to failures, distribution across partitions (e.g., moving computation close to data or vice versa), elastic scaling to support usage bursts, user proximity, etc.

In this context, sharing, disseminating and analyzing the data sets results in frequent large-scale data movements across widely distributed sites. Studies show that the inter-datacenter traffic is expected to triple in the following years. Our objective is to investigate approaches to data management enabling an efficient execution of such geographically distributed workflows running on multi-site clouds.

While in the past years we have addressed some data management issues in this area, mainly in support to efficient task scheduling of scientific workflows running on multisite clouds, we will now focus on an increasingly common scenario where workflows generate and process a huge number of small files, which is particularly challenging. As such workloads generate a deluge of small and independent I/O operations, efficient data and metadata handling is critical. We will explore specific means to better hide latency for data and metadata access in such scenarios, as a way to improve global performance.

3.3. Research axis 3: I/O management, in situ visualization and analysis on HPC systems at extreme scales

Over the past few years, the increasing amounts of data produced by large-scale simulations have motivated a shift from traditional offline data analysis to in situ analysis and visualization. In situ processing started by coupling a parallel simulation with an analysis or visualization library, to avoid the cost of writing data on storage and reading it back. Going beyond this simple pairwise tight coupling, complex analysis workflows today are graphs with one or more data sources and several interconnected analysis components.

Collaboration. *This axis is worked out in close collaboration with Rob Ross (ANL), Tom Peterka (ANL), Matthieu Dorier (ANL), Toni Cortes (BSC), Bruno Raffin (Inria). Some additional collaborations are in discussion with other members of JLESC, and with CEA and Total.*

Relevant groups with similar interests include the following ones.

- *The group of Manish Parashar at Rutgers University, USA (I/O management for HPC systems, in situ processing).*
- *The group of Scott Klasky at Oak Ridge National Lab, USA (I/O management for HPC systems, in situ processing).*
- *The CNRS IPSL laboratory (Sébastien Denvil, Pôle de modélisation du climat) in Paris, France (in situ data analytics).*

3.3.1. Toward a joint optimized architecture for in situ visualization and advanced processing

From Inria and ANL, four tools at least have emerged to address some challenges of coupling simulations with visualization packages or analysis workflows. Each of them focused on some particular aspect:

Damaris (Inria, [5], [4]) exploits dedicated cores to enable jitter-free I/O and in situ visualization;

Decaf (ANL, [31]) implements a coupling service for workflows;

FlowVR (Inria, [43]) connects workflow components for in situ processing;

Swift (ANL, [46]) focuses on implicitly parallel data flows and was optimized for Big Data processing.

Our plan is to explore how these tools could best leverage their respective strengths in a *joint optimized architecture for in situ visualization and advanced processing* in the HPC area. We published a preliminary study describing the lessons learned from using these tools in production environments with real applications [7]. Such a joint architecture will contribute to address the data volume and velocity challenges raised by data-intensive workflows, including complex data-intensive analytics phases. It may also impact, in a subsequent step, future data analysis pipelines for converged Big Data and HPC architectures.

LACODAM Project-Team

3. Research Program

3.1. Introduction

The three research axes of the LACODAM project-team are the following. First, we briefly introduce these axes, as well as their interplay:

- The first research axis is dedicated to the design of *novel pattern mining methods*. Pattern mining is one of the most important approaches to discover novel knowledge in data, and one of our strongest areas of expertise. The work on this axis will serve as foundations for work on the other two axes. Thus, this axis will have the strongest impact on our goals overall.
- The second axis tackles another aspect of knowledge discovery in data: the *interaction between the user and the system* in order to co-discover novel knowledge. Our team has plenty of experience collaborating with domain experts, and is therefore aware of the need to improve such interaction.
- The third axis concerns *decision support*. With the help of methods from the two previous axes, our goal here is to design systems that can either assist humans with making decisions, or make relevant decisions in situations where extremely fast reaction is required.

The following figure sums up the detailed work presented in the next few pages: we show the three research axes of the team (X-axis) on the left and our main applications areas (Y-axis) below. In the middle there are colored squares that represent the precise research topics of the team aligned with their axis and main application area. These research topics will be described in this section. Lines represent projects that can link several topics, and that are also connected to their main application area.

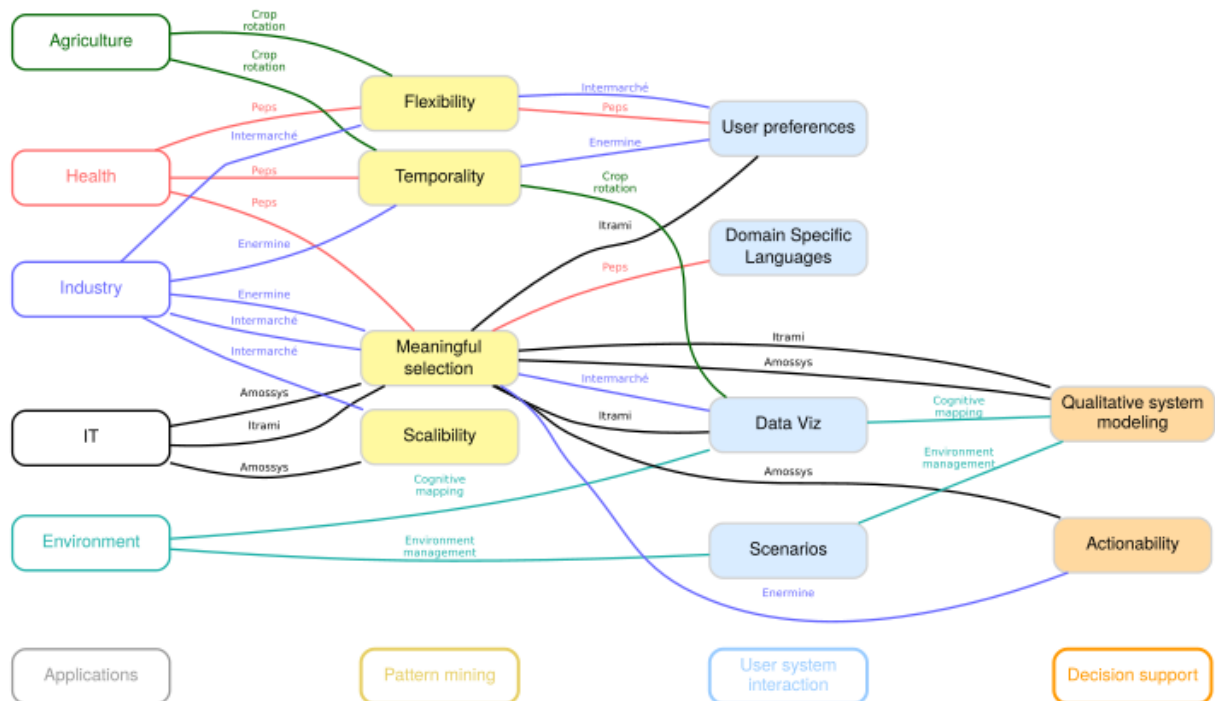
3.2. Pattern mining algorithms

Twenty years of research in pattern mining have resulted in efficient approaches to handle the algorithmic complexity of the problem. Existing algorithms are now able to efficiently extract patterns with complex structures (ex: sequences, graphs, co-variations) from large datasets. However, when dealing with large, real-world datasets, these methods still output a huge set of patterns, which is impractical for human analysis. This problem is called *pattern explosion*. The ongoing challenge of pattern mining research is to extract fewer but more meaningful patterns. The LACODAM team is committed to solve the pattern explosion problem by pursuing the following four research topics:

1. the design of dedicated algorithms for mining temporal patterns
2. the design of flexible pattern mining approaches
3. the automatic selection of interesting data mining results
4. the design of parallel pattern algorithms to ensure scalability

The originality of our contributions relies on the exploration of knowledge-based approaches whose principle is to incorporate dedicated domain knowledge (aka application background knowledge) deep into the mining process. While most of the data mining approaches are based on agnostic approaches that are designed to cope with the pattern explosion, we propose to develop data mining techniques relying on knowledge-based artificial intelligence techniques. This entails the use of structured knowledge representations, as well as reasoning methods, in combination with mining.

The first topic concerns classical pattern mining in conjunction with expert knowledge in order to define new pattern types (and related algorithms) that can solve applicative issues. In particular, we investigate how to handle temporality in pattern representations which turns out to be important in many real world applications (in particular for decision support) and deserves particular attention.



Lacodam research focus seen through its short term thematic applications

Figure 1. Lacodam research topics organized by axis and application

The next two topics aim at proposing alternative pattern mining methods to let the user incorporate, on her own, knowledge that will help define her pattern domain of interest. Flexible pattern mining approaches enable analysts to easily incorporate extra knowledge, for example domain related constraints, in order to extract only the most relevant patterns. On the other hand, the selection of interesting data mining results aims at devising strategies to filter out the results that are useless for the data analyst. Besides the challenge of algorithmic efficiency, we are interested in formalizing the foundations of interestingness, according to background knowledge modeled with logical knowledge representation paradigms.

Last but not least, pattern mining algorithms are compute-intensive. It is thus important to exploit all the available computing power. Parallelism is for a foreseeable future one of the main ways to speed up computations, and we have a strong competence on the design of parallel pattern mining algorithms. We will exploit this competence in order to guarantee that our approaches scale up to the data provided by our partners.

3.3. User/system interaction

As we pointed out before, there is a strong need to present relevant patterns to the user. This can be done by using more specific constraints, background knowledge and/or tailor-made optimization functions. Due to the difficulty of determining these elements beforehand, one of the most promising solutions is that the system and the user co-construct the definition of relevance, i.e., to have a human in the loop. This requires to have means to present intermediate results to the user, and to get user feedback in order to guide the search space exploration process in the right direction. This is an important research axis for LACODAM, which will be tackled in several complementary ways:

- *Domain Specific Languages*: One way to interact with the user is to propose a Domain Specific Language (DSL) tailored to the domain at hand and to the analysis tasks. The challenge is to propose a DSL allowing the users to easily express the required processing workflows, to deploy those workflows for mining on large volumes of data and to offer as much automation as possible.
- *What if / What for scenarios*: We also investigate the use of scenarios to query results from data mining processes, as well as other complex processes such as complex system simulations or model predictions. Such scenarios are answers to questions of the type “what if [situation]?” or “what [should be done] for [expected outcome]?”.
- *User preferences*: In exploratory analysis, users often do not have a precise idea of what they want, and are not able to formulate such queries. Hence, in LACODAM we investigate simple ways for users to express their interests and preferences, either during the mining process – to guide the search space exploration –, or afterwards during the filtering and interpretation of the most relevant results.
- *Data visualization*: Most of the research directions presented in this document require users to examine patterns at some point. The output of most pattern mining algorithms is usually a (long) list of patterns. While this presentation can be sufficient for some applications, often it does not provide a complete understanding, especially for non-experts in pattern mining. A transversal research topic that we want to explore in LACODAM is to propose data visualization techniques that are adequate for understanding output results. Numerous (failed) experiments have shown that data mining and data visualization are fields, which require distinct skills, thus researchers in one field usually do not make significant advances in the other field (this is detailed in [Keim 2010]). Thus, our strategy is to establish collaborations with prominent data visualization teams for this line of research, with a long term goal to recruit a specialist in data visualization if the opportunity arises.

3.4. Decision support

Predictive sequential patterns have a direct application in diagnosis. LACODAM inherits a strong background in decision support systems with internationally recognized expertise in diagnosis from the former DREAM team. This AI subfield is concerned with determining whether a system is operating normally or not, and the cause of faulty behaviors. The studied system can be an agro- or eco-system, a software system, a living being, etc.

The increasing volumes of data coming from a range of different systems (ex: sensor data from agro-environmental systems, log data from software systems, biological data coming from health monitoring systems) can help human and software agents make better decisions. Hence, LACODAM builds upon the idea that decision support systems (an interest bequeathed from DREAM) should take advantage of the available data. This third and last research axis is thus a meeting point for all members of the team, as it requires the integration of AI techniques for traditional decision support systems with results from data mining techniques.

Two main research sub-axes are investigated in LACODAM:

- *Diagnosis-based approaches.* We are exploring how to integrate knowledge found from pattern mining approaches, possibly with the help of interactive methods, into the qualitative models. The goal of such work is to automate as much as possible the construction of prediction models, which can require a lot of human effort.
- *Actionable patterns and rules.* In many settings of “exploratory data mining”, the actual interestingness of a pattern is hard to assess, as it may be subjective. However, for some applications there are well defined measures of interestingness and applicability for patterns. Patterns and rules that can lead to actual actions –that are relevant to the user– are called “actionable patterns” and are of vital importance to industrial settings.

3.5. Long-term goals

The following perspectives are at the convergence of the three aforementioned research axes and can be seen as ideal towards our goals:

- *Automating data science workflow discovery.* The current methods for knowledge extraction and construction of decision support systems require a lot of human effort. Our three research axes aim at alleviating this effort, by devising methods that are more generic and by improving the interaction between the user and the system. An ideal solution would be that the user could forget completely about the existence of pattern mining or decision support methods. Instead the user would only loosely specify her problem, while the system constructs various data science / decision support workflows, possibly further refined via interactions.

We consider that this is a second order AI task, where AI techniques such as planning are used to explore the workflow search space, the workflow itself being composed of data mining and/or decision support components. This is a strategic evolution for data science endeavors, where the demand far exceeds the available human skilled manpower.

- *Logic argumentation based on epistemic interest.* Having increasingly automated approaches will require better and better ways to handle the interactions with the user. Our second long term goal is to explore the use of logic argumentation, i.e., the formalisation of human strategies for reasoning and arguing, in the interaction between users and data analysis tools. Alongside visualization and interactive data mining tools, logic argumentation can be a way for users to query both the results and the way they are obtained. Such querying can also help the expert to reformulate her query in an interactive analysis setting.

This research direction aims at exploiting principles of interactive data analysis in the context of epistemic interestingness measures. Logic argumentation can be a natural tool for interactions between the user and the system: display of possibly exhaustive list of arguments, relationships between arguments (e.g., reinforcement, compatibility or conflict), possible solutions for argument conflicts, etc.

The first step is to define a formal argumentation framework for explaining data mining results. This implies to continue theoretical work on the foundations of argumentation in order to identify the most adapted framework (either existing or a new one to be defined). Logic argumentation may be implemented and deeply explored in ASP, allowing us to build on our expertise in this logic language.

- *Collaborative feedback and knowledge management.* We are convinced that improving the data science process, and possibly automating it, will rely on high-quality feedback from communities on the web. Consider for example what has been achieved by collaborative platforms such as StackOverflow: it has become the reference site for any programming question.

Data science is a more complex problem than programming, as in order to get help from the community, the user has to share her data and workflow, or at least some parts of them. This raises obvious privacy issues that may prevent this idea to succeed. As our research on automating the production of data science workflows should enable more people to have access to data science results, we are interested in the design of collaborative platforms to exchange expert advices over data, workflows and analysis results. This aims at exploiting human feedback to improve the automation of data science system via machine learning methods.

LAGADIC Project-Team

3. Research Program

3.1. Visual Servoing

Basically, visual servoing techniques consist in using the data provided by one or several cameras in order to control the motions of a dynamic system [1]. Such systems are usually robot arms, or mobile robots, but can also be virtual robots, or even a virtual camera. A large variety of positioning tasks, or mobile target tracking, can be implemented by controlling from one to all the degrees of freedom (DoF) of the system. Whatever the sensor configuration, which can vary from one on-board camera on the robot end-effector to several free-standing cameras, a set of visual features has to be selected at best from the image measurements available, allowing to control the desired DoF. A control law has also to be designed so that these visual features $s(t)$ reach a desired value s^* , defining a correct realization of the task. A desired planned trajectory $s^*(t)$ can also be tracked. The control principle is thus to regulate the error vector $s(t) - s^*(t)$ to zero. With a vision sensor providing 2D measurements, potential visual features are numerous, since 2D data (coordinates of feature points in the image, moments, ...) as well as 3D data provided by a localization algorithm exploiting the extracted 2D features can be considered. It is also possible to combine 2D and 3D visual features to take the advantages of each approach while avoiding their respective drawbacks.

More precisely, a set s of k visual features can be taken into account in a visual servoing scheme if it can be written:

$$s = s(x(p(t)), a) \quad (47)$$

where $p(t)$ describes the pose at the instant t between the camera frame and the target frame, x the image measurements, and a a set of parameters encoding a potential additional knowledge, if available (such as for instance a coarse approximation of the camera calibration parameters, or the 3D model of the target in some cases).

The time variation of s can be linked to the relative instantaneous velocity v between the camera and the scene:

$$\dot{s} = \frac{\partial s}{\partial p} \dot{p} = L_s v \quad (48)$$

where L_s is the interaction matrix related to s . This interaction matrix plays an essential role. Indeed, if we consider for instance an eye-in-hand system and the camera velocity as input of the robot controller, we obtain when the control law is designed to try to obtain an exponential decoupled decrease of the error:

$$v_c = -\lambda \widehat{L}_s^+ (s - s^*) - \widehat{L}_s^+ \frac{\partial s}{\partial t} \quad (49)$$

where λ is a proportional gain that has to be tuned to minimize the time-to-convergence, \widehat{L}_s^+ is the pseudo-inverse of a model or an approximation of the interaction matrix, and $\frac{\partial s}{\partial t}$ an estimation of the features velocity due to a possible own object motion.

From the selected visual features and the corresponding interaction matrix, the behavior of the system will have particular properties as for stability, robustness with respect to noise or to calibration errors, robot 3D trajectory, etc. Usually, the interaction matrix is composed of highly non linear terms and does not present any decoupling properties. This is generally the case when s is directly chosen as x . In some cases, it may lead to inadequate robot trajectories or even motions impossible to realize, local minimum, tasks singularities, etc. It is thus extremely important to design adequate visual features for each robot task or application, the ideal case (very difficult to obtain) being when the corresponding interaction matrix is constant, leading to a simple linear control system. To conclude in a few words, **visual servoing is basically a non linear control problem. Our Holy Grail quest is to transform it into a linear control problem.**

Furthermore, embedding visual servoing in the task function approach allows solving efficiently the redundancy problems that appear when the visual task does not constrain all the DoF of the system. It is then possible to realize simultaneously the visual task and secondary tasks such as visual inspection, or joint limits or singularities avoidance. This formalism can also be used for tasks sequencing purposes in order to deal with high level complex applications.

3.2. Visual Tracking

Elaboration of object tracking algorithms in image sequences is an important issue for researches and applications related to visual servoing and more generally for robot vision. A robust extraction and real time spatio-temporal tracking process of visual cues is indeed one of the keys to success of a visual servoing task. If fiducial markers may still be useful to validate theoretical aspects in modeling and control, natural scenes with non-cooperative objects and subject to various illumination conditions have to be considered for addressing large scale realistic applications.

Most of the available tracking methods can be divided into two main classes: feature-based and model-based. The former approach focuses on tracking 2D features such as geometrical primitives (points, segments, circles,...), object contours, regions of interest, etc. The latter explicitly uses a model of the tracked objects. This can be either a 3D model or a 2D template of the object. This second class of methods usually provides a more robust solution. Indeed, the main advantage of the model-based methods is that the knowledge about the scene allows improving tracking robustness and performance, by being able to predict hidden movements of the object, detect partial occlusions and acts to reduce the effects of outliers. The challenge is to build algorithms that are fast and robust enough to meet our application requirements. Therefore, even if we still consider 2D feature tracking in some cases, our researches mainly focus on real-time 3D model-based tracking, since these approaches are very accurate, robust, and well adapted to any class of visual servoing schemes. Furthermore, they also meet the requirements of other classes of application, such as augmented reality.

3.3. SLAM

Most of the applications involving mobile robotic systems (ground vehicles, aerial robots, automated submarines,...) require a reliable localization of the robot in its environment. A challenging problem is when neither the robot localization nor the map is known. Localization and mapping must then be considered concurrently. This problem is known as Simultaneous Localization And Mapping (SLAM). In this case, the robot moves from an unknown location in an unknown environment and proceeds to incrementally build up a navigation map of the environment, while simultaneously using this map to update its estimated position.

Nevertheless, solving the SLAM problem is not sufficient for guaranteeing an autonomous and safe navigation. The choice of the representation of the map is, of course, essential. The representation has to support the different levels of the navigation process: motion planning, motion execution and collision avoidance and, at the global level, the definition of an optimal strategy of displacement. The original formulation of the SLAM problem is purely metric (since it basically consists in estimating the Cartesian situations of the robot and a set of landmarks), and it does not involve complex representations of the environment. However, it is now well recognized that **several complementary representations are needed to perform exploration, navigation, mapping, and control tasks successfully. We propose to use composite models of the environment that**

mix topological, metric, and grid-based representations. Each type of representation is well adapted to a particular aspect of autonomous navigation [7]: the metric model allows one to locate the robot precisely and plan Cartesian paths, the topological model captures the accessibility of different sites in the environment and allows a coarse localization, and finally the grid representation is useful to characterize the free space and design potential functions used for reactive obstacle avoidance. However, ensuring the consistency of these various representations during the robot exploration, and merging observations acquired from different viewpoints by several cooperative robots, are difficult problems. This is particularly true when different sensing modalities are involved. New studies to derive efficient algorithms for manipulating the hybrid representations (merging, updating, filtering...) while preserving their consistency are needed.

3.4. Scene Modeling and Understanding

Long-term mapping has received an increasing amount of attention during last years, largely motivated by the growing need to integrate robots into the real world wherein dynamic objects constantly change the appearance of the scene. A mobile robot evolving in such a dynamic world should not only be able to build a map of the observed environment at a specific moment, but also to maintain this map consistent over a long period of time. It has to deal with dynamic changes that can cause the navigation process to fail. However updating the map is particularly challenging in large-scale environments. To identify changes, robots have to keep a memory of the previous states of the environment and the more dynamic it is, the higher will be the number of states to manage and the more computationally intensive will be the updating process. Mapping large-scale dynamic environments is then particularly difficult as the map size can be arbitrary large. Additionally, mapping many times the whole environment is not always possible or convenient and it is useful to take advantages of methods using only a small number of observations.

A recent trend in robotic mapping is to augment low-level maps with semantic interpretation of their content, which allows to improve the robot's environmental awareness through the use of high-level concepts. In mobile robot navigation, the so-called semantic maps have already been used to improve path planning methods, mainly by providing the robot with the ability to deal with human-understandable targets.

LINKMEDIA Project-Team

3. Research Program

3.1. Scientific background

LINKMEDIA is a multidisciplinary research team, with multimedia data as the main object of study. We are guided by the data and their specificity—semantically interpretable, heterogeneous and multimodal, available in large amounts, unstructured and disconnected—, as well as by the related problems and applications.

With multimedia data at the center, orienting our choices of methods and algorithms and serving as a basis for experimental validation, the team is directly contributing to the following scientific fields:

- multimedia: content-based analysis; multimodal processing and fusion; multimedia applications;
- computer vision: compact description of images; object and event detection;
- natural language processing: topic segmentation; information extraction;
- information retrieval: high-dimensional indexing; approximate k-nn search; efficient set comparison.

LINKMEDIA also takes advantage of advances in the following fields, adapting recent developments to the multimedia area:

- signal processing: image processing; compression;
- machine learning: deep architectures; structured learning; adversarial learning;
- security: data encryption; differential privacy;
- data mining: time series mining and alignment; pattern discovery; knowledge extraction.

3.2. Workplan

Research activities in LINKMEDIA are organized along three major lines of research which build upon the scientific domains already mentioned.

3.2.1. *Unsupervised motif discovery*

As an alternative to supervised learning techniques, unsupervised approaches have emerged recently in multimedia with the goal of discovering directly patterns and events of interest from the data, in a totally unsupervised manner. In the absence of prior knowledge on what we are interested in, meaningfulness can be judged based on one of three main criteria: unexpectedness, saliency and recurrence. This last case posits that repeating patterns, known as motifs, are potentially meaningful, leading to recent work on the unsupervised discovery of motifs in multimedia data [56], [54], [55].

LINKMEDIA seeks to *develop unsupervised motif discovery approaches which are both accurate and scalable*. In particular, we consider the discovery of repeating objects in image collections and the discovery of repeated sequences in video and audio streams. Research activities are organized along the following lines:

- developing the scientific basis for scalable motif discovery: sparse histogram representations; efficient co-occurrence counting; geometry and time aware indexing schemes;
- designing and evaluating accurate and scalable motif discovery algorithms applied to a variety of multimedia content: exploiting efficient geometry or time aware matching functions; fast approximate dynamic time warping; symbolic representations of multimedia data, in conjunction with existing symbolic data mining approaches;
- developing methodology for the interpretation, exploitation and evaluation of motif discovery algorithms in various use-cases: image classification; video stream monitoring; transcript-free natural language processing (NLP) for spoken document.

3.2.2. Description and structuring

Content-based analysis has received a lot of attention from the early days of multimedia, with an extensive use of supervised machine learning for all modalities [57], [51]. Progress in large scale entity and event recognition in multimedia content has made available general purpose approaches able to learn from very large data sets and performing fairly decently in a large number of cases. Current solutions are however limited to simple, homogeneous, information and can hardly handle structured information such as hierarchical descriptions, tree-structured or nested concepts.

LINKMEDIA aims at *expanding techniques for multimedia content modeling, event detection and structure analysis*. The main transverse research lines that LINKMEDIA will develop are as follows:

- context-aware content description targeting (homogeneous) collections of multimedia data: latent variable discovery; deep feature learning; motif discovery;
- secure description to enable privacy and security aware multimedia content processing: leveraging encryption and obfuscation; exploring adversarial machine learning in a multimedia context; privacy-oriented image processing;
- multilevel modeling with a focus on probabilistic modeling of structured multimodal data: multiple kernels; structured machine learning; conditional random fields.

3.2.3. Linking and collection data model

Creating explicit links between media content items has been considered on different occasions, with the goal of seeking and discovering information by browsing, as opposed to information retrieval via ranked lists of relevant documents. Content-based link creation has been initially addressed in the hypertext community for well-structured texts [50] and was recently extended to multimedia content [58], [53], [52]. The problem of organizing collections with links remains mainly unsolved for large heterogeneous collections of unstructured documents, with many issues deserving attention: linking at a fine semantic grain; selecting relevant links; characterizing links; evaluating links; etc.

LINKMEDIA targets pioneering research on media linking by *developing scientific ground, methodology and technology for content-based media linking* directed to applications exploiting rich linked content such as navigation or recommendation. Contributions are concentrated along the following lines:

- algorithmic of linked media for content-based link authoring in multimedia collections: time-aware graph construction; multimodal hypergraphs; large scale k-nn graphs;
- link interpretation and characterization to provide links semantics for interpretability: text alignment; entity linking; intention vs. extension;
- linked media usage and evaluation: information retrieval; summarization; data models for navigation; link prediction.

MIMETIC Project-Team

3. Research Program

3.1. Biomechanics and Motion Control

Human motion control is a highly complex phenomenon that involves several layered systems, as shown in Figure 3. Each layer of this controller is responsible for dealing with perceptual stimuli in order to decide the actions that should be applied to the human body and his environment. Due to the intrinsic complexity of the information (internal representation of the body and mental state, external representation of the environment) used to perform this task, it is almost impossible to model all the possible states of the system. Even for simple problems, there generally exists an infinity of solutions. For example, from the biomechanical point of view, there are much more actuators (i.e. muscles) than degrees of freedom leading to an infinity of muscle activation patterns for a unique joint rotation. From the reactive point of view there exists an infinity of paths to avoid a given obstacle in navigation tasks. At each layer, the key problem is to understand how people select one solution among these infinite state spaces. Several scientific domains have addressed this problem with specific points of view, such as physiology, biomechanics, neurosciences and psychology.

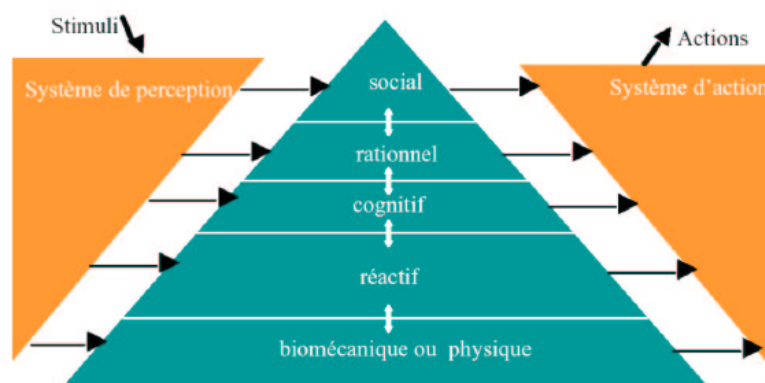


Figure 3. Layers of the motion control natural system in humans.

In biomechanics and physiology, researchers have proposed hypotheses based on accurate joint modeling (to identify the real anatomical rotational axes), energy minimization, force and torques minimization, comfort maximization (i.e. avoiding joint limits), and physiological limitations in muscle force production. All these constraints have been used in optimal controllers to simulate natural motions. The main problem is thus to define how these constraints are composed altogether such as searching the weights used to linearly combine these criteria in order to generate a natural motion. Musculoskeletal models are stereotyped examples for which there exists an infinity of muscle activation patterns, especially when dealing with antagonist muscles. An unresolved problem is to define how to use the above criteria to retrieve the actual activation patterns, while optimization approaches still leads to unrealistic ones. It is still an open problem that will require multidisciplinary skills including computer simulation, constraint solving, biomechanics, optimal control, physiology and neuroscience.

In neuroscience, researchers have proposed other theories, such as coordination patterns between joints driven by simplifications of the variables used to control the motion. The key idea is to assume that instead of controlling all the degrees of freedom, people control higher level variables which correspond to combinations of joint angles. In walking, data reduction techniques such as Principal Component Analysis have shown that lower-limb joint angles are generally projected on a unique plane whose angle in the state space is associated with energy expenditure. Although knowledge exists for specific motions, such as locomotion or grasping, this type of approach is still difficult to generalize. The key problem is that many variables are coupled and it is very difficult to objectively study the behavior of a unique variable in various motor tasks. Computer simulation is a promising method to evaluate such type of assumptions as it enables to accurately control all the variables and to check if it leads to natural movements.

Neuroscience also addresses the problem of coupling perception and action by providing control laws based on visual cues (or any other senses), such as determining how the optical flow is used to control direction in navigation tasks, while dealing with collision avoidance or interception. Coupling of the control variables is enhanced in this case as the state of the body is enriched by the large amount of external information that the subject can use. Virtual environments inhabited with autonomous characters whose behavior is driven by motion control assumptions is a promising approach to solve this problem. For example, an interesting problem in this field is navigation in an environment inhabited with other people. Typically, avoiding static obstacles together with other people displacing into the environment is a combinatory problem that strongly relies on the coupling between perception and action.

One of the main objectives of MimeTIC is to enhance knowledge on human motion control by developing innovative experiments based on computer simulation and immersive environments. To this end, designing experimental protocols is a key point and some of the researchers in MimeTIC have developed this skill in biomechanics and perception-action coupling. Associating these researchers to experts in virtual human simulation, computational geometry and constraints solving enable us to contribute to enhance fundamental knowledge in human motion control.

3.2. Experiments in Virtual Reality

Understanding interactions between humans is challenging because it addresses many complex phenomena including perception, decision-making, cognition and social behaviors. Moreover, all these phenomena are difficult to isolate in real situations, and it is therefore highly complex to understand their individual influence on these human interactions. It is then necessary to find an alternative solution that can standardize the experiments and that allows the modification of only one parameter at a time. Video was first used since the displayed experiment is perfectly repeatable and cut-offs (stop the video at a specific time before its end) allow having temporal information. Nevertheless, the absence of adapted viewpoint and stereoscopic vision does not provide depth information that are very meaningful. Moreover, during video recording session, the real human is acting in front of a camera and not of an opponent. The interaction is then not a real interaction between humans.

Virtual Reality (VR) systems allow full standardization of the experimental situations and the complete control of the virtual environment. It is then possible to modify only one parameter at a time and to observe its influence on the perception of the immersed subject. VR can then be used to understand what information is picked up to make a decision. Moreover, cut-offs can also be used to obtain temporal information about when information is picked up. When the subject can moreover react as in a real situation, his movement (captured in real time) provides information about his reactions to the modified parameter. Not only is the perception studied, but the complete perception-action loop. Perception and action are indeed coupled and influence each other as suggested by Gibson in 1979.

Finally, VR allows the validation of virtual human models. Some models are indeed based on the interaction between the virtual character and the other humans, such as a walking model. In that case, there are two ways to validate it. First, they can be compared to real data (e.g. real trajectories of pedestrians). But such data are not always available and are difficult to get. The alternative solution is then to use VR. The validation of the realism of the model is then done by immersing a real subject in a virtual environment in which a virtual

character is controlled by the model. Its evaluation is then deduced from how the immersed subject reacts when interacting with the model and how realistic he feels the virtual character is.

3.3. Computational Geometry

Computational geometry is a branch of computer science devoted to the study of algorithms which can be stated in terms of geometry. It aims at studying algorithms for combinatorial, topological and metric problems concerning sets of points in Euclidian spaces. Combinatorial computational geometry focuses on three main problem classes: static problems, geometric query problems and dynamic problems.

In static problems, some inputs are given and the corresponding outputs need to be constructed or found. Such problems include linear programming, Delaunay triangulations, and Euclidian shortest paths for instance. In geometric query problems, commonly known as geometric search problems, the input consists of two parts: the search space part and the query part, which varies over the problem instances. The search space typically needs to be preprocessed, in a way that multiple queries can be answered efficiently. Some typical problems are range searching, point location in a portioned space, or nearest neighbor queries. In dynamic problems, the goal is to find an efficient algorithm for finding a solution repeatedly after each incremental modification of the input data (addition, deletion or motion of input geometric elements). Algorithms for problems of this type typically involve dynamic data structures. Both of previous problem types can be converted into a dynamic problem, for instance, maintaining a Delaunay triangulation between moving points.

In this context, distance geometry relies solely on distances, instead of points and lines, as in classical geometry. Various applications lead to the definition of problems that can be formulated as a distance geometry, including sensor network localization, robot coordination, the identification of molecular conformations, or as in the context of MimeTIC relations between objects in virtual scenes (e.g., distances between body segments, agents, or cameras). In recent years, scientific research has been oriented to the assumptions allowing for discretizing the search space of a given distance geometry problem. The discretization (which is exact in some situations) allows to conceive ad-hoc and efficient algorithms, and for enumerating the entire solution set of a given instance.

The MimeTIC team works on problems such as crowd simulation, spatial analysis, path and motion planning in static and dynamic environments, camera planning with visibility constraints for instance. The core of those problems, by nature, relies on problems and techniques belonging to computational geometry. Proposed models pay attention to algorithms complexity to be compatible with performance constraints imposed by interactive applications.

MYRIADS Project-Team

3. Research Program

3.1. Introduction

In this section, we present our research challenges along four work directions: resource and application management in distributed cloud architectures for scaling clouds in Section 3.2, energy management strategies for greening clouds in Section 3.3, security and data protection aspects for securing cloud-based information systems and applications in Section 3.4, and methods for experimenting with clouds in Section 3.5.

3.2. Scaling clouds

3.2.1. Resource management in hierarchical clouds

The next generation of utility computing appears to be an evolution from highly centralized clouds towards more decentralized platforms. Today, cloud computing platforms mostly rely on large data centers servicing a multitude of clients from the edge of the Internet. Servicing cloud clients in this manner suggests that locality patterns are ignored: wherever the client issues his/her request from, the request will have to go through the backbone of the Internet provider to the other side of the network where the data center relies. Besides this extra network traffic and this latency overhead that could be avoided, other common centralization drawbacks in this context stand in limitations in terms of security/legal issues and resilience.

At the same time, it appears that network backbones are over-provisioned for most of their usage. This advocates for placing computing resources directly within the backbone network. The general challenge of resource management for such clouds stands in trying to be locality-aware: for the needs of an application, several virtual machines may exchange data. Placing them *close* to each others can significantly improve the performance of the application they compose. More generally, building an overlay network which takes the hierarchical aspects of the platform without being a hierarchical overlay – which comes with load balancing and resilience issues is a challenge by itself.

The results of these works are planned to be integrated into the Discovery initiative [60] which aims at revisiting OpenStack to offer a cloud stack able to manage utility computing platforms where computing resources are located in small computing centers in the backbone's PoPs (Point of Presence) and interconnected through the backbone's internal links.

3.2.2. Resource management in mobile edge clouds

Mobile edge cloud (MEC) infrastructures are composed of compute, storage and networking resources located at the edge of wide-area networks, in immediate proximity to the end users. Instead of treating the mobile operator's network as a high-latency dumb pipe between the end users and the external service providers, MEC platforms aim at deploying cloud functionalities *within* the mobile phone network, inside or close to the mobile access points. Doing so is expected to deliver added value to the content providers and the end users by enabling new types of applications ranging from Internet-of-Things applications to extremely interactive systems (e.g., augmented reality). Simultaneously, it will generate extra revenue streams for the mobile network operators, by allowing them to position themselves as cloud computing operators and to rent their already-deployed infrastructure to content and application providers.

Mobile edge clouds have very different geographical distribution compared to traditional clouds. While traditional clouds are composed of many reliable and powerful machines located in a very small number of data centers and interconnected by very high-speed networks, mobile edge cloud are composed of a very large number of points-of-presence with a couple of weak and potentially unreliable servers, interconnected with each other by commodity long-distance networks. This creates new demands for the organization of a scalable mobile edge computing infrastructure, and opens new directions for research.

The main challenges that we plan to address are:

- How should an edge cloud infrastructure be designed such that it remains scalable, fault-tolerant, controllable, energy-efficient, etc.?
- How should applications making use of edge clouds be organized? One promising direction is to explore the extent to which stream-data processing platforms such as Apache Spark and Apache Flink can be adapted to become one of the main application programming paradigms in such environments.

3.2.3. *Self-optimizing applications in multi-cloud environments*

As the use of cloud computing becomes pervasive, the ability to deploy an application on a multi-cloud infrastructure becomes increasingly important. Potential benefits include avoiding dependence on a single vendor, taking advantage of lower resource prices or resource proximity, and enhancing application availability. Supporting multi-cloud application management involves two tasks. First, it involves selecting an initial multi-cloud application deployment that best satisfies application objectives and optimizes performance and cost. Second, it involves dynamically adapting the application deployment in order to react to changes in execution conditions, application objectives, cloud provider offerings, or resource prices. Handling price changes in particular is becoming increasingly complex. The reason is the growing trend of providers offering sophisticated, dynamic pricing models that allow buying and selling resources of finer granularities for shorter time durations with varying prices.

Although multi-cloud platforms are starting to emerge, these platforms impose a considerable amount of effort on developers and operations engineers, provide no support for dynamic pricing, and lack the responsiveness and scalability necessary for handling highly-distributed, dynamic applications with strict quality requirements. The goal of this work is to develop techniques and mechanisms for automating application management, enabling applications to cope with and take advantage of the dynamic, diverse, multi-cloud environment in which they operate.

The main challenges arising in this context are:

- selecting effective decision-making approaches for application adaptation,
- supporting scalable monitoring and adaptation across multiple clouds,
- performing adaptation actions in a cost-efficient and safe manner.

3.3. Greening clouds

ICT (Information and Communications Technologies) ecosystem now approaches 5% of world electricity consumption and this ICT energy use will continue grow fast because of the information appetite of Big Data, big networks and big infrastructures as Clouds that unavoidably leads to big power.

3.3.1. *Smart grids and clouds*

We propose exploiting Smart Grid technologies to come to the rescue of energy-hungry Clouds. Unlike in traditional electrical distribution networks, where power can only be moved and scheduled in very limited ways, Smart Grids dynamically and effectively adapt supply to demand and limit electricity losses (currently 10% of produced energy is lost during transmission and distribution).

For instance, when a user submits a Cloud request (such as a Google search for instance), it is routed to a data center that processes it, computes the answer and sends it back to the user. Google owns several data centers spread across the world and for performance reasons, the center answering the user's request is more likely to be the one closest to the user. However, this data center may be less energy efficient. This request may have consumed less energy, or a different kind of energy (renewable or not), if it had been sent to this further data center. In this case, the response time would have been increased but maybe not noticeably: a different trade-off between quality of service (QoS) and energy-efficiency could have been adopted.

While Clouds come naturally to the rescue of Smart Grids for dealing with this big data issue, little attention has been paid to the benefits that Smart Grids could bring to distributed Clouds. To our knowledge, no previous work has exploited the Smart Grids potential to obtain and control the energy consumption of entire Cloud infrastructures from underlying facilities such as air conditioning equipment (which accounts for 30% to 50% of a data center's electricity bill) to network resources (which are often operated by several actors) and to computing resources (with their heterogeneity and distribution across multiple data centers). We aim at taking advantage of the opportunity brought by the Smart Grids to exploit renewable energy availability and to optimize energy management in distributed Clouds.

3.3.2. Energy cost models

Cloud computing allows users to outsource the computer resources required for their applications instead of using a local installation. It offers on-demand access to the resources through the Internet with a pay-as-you-go pricing model. However, this model hides the electricity cost of running these infrastructures.

The costs of current data centers are mostly driven by their energy consumption (specifically by the air conditioning, computing and networking infrastructure). Yet, current pricing models are usually static and rarely consider the facilities' energy consumption per user. The challenge is to provide a fair and predictable model to attribute the overall energy costs per virtual machine and to increase energy-awareness of users.

Another goal consists in better understanding the energy consumption of computing and networking resources of Clouds in order to provide energy cost models for the entire infrastructure including incentivizing cost models for both Cloud providers and energy suppliers. These models will be based on experimental measurement campaigns on heterogeneous devices. Inferring a cost model from energy measurements is an arduous task since simple models are not convincing, as shown in our previous work. We aim at proposing and validating energy cost models for the heterogeneous Cloud infrastructures in one hand, and the energy distribution grid on the other hand. These models will be integrated into simulation frameworks in order to validate our energy-efficient algorithms at larger scale.

3.3.3. Energy-aware users

In a Cloud moderately loaded, some servers may be turned off when not used for energy saving purpose. Cloud providers can apply resource management strategies to favor idle servers. Some of the existing solutions propose mechanisms to optimize VM scheduling in the Cloud. A common solution is to consolidate the mapping of the VMs in the Cloud by grouping them in a fewer number of servers. The unused servers can then be turned off in order to lower the global electricity consumption.

Indeed, current work focuses on possible levers at the virtual machine suppliers and/or services. However, users are not involved in the choice of using these levers while significant energy savings could be achieved with their help. For example, they might agree to delay slightly the calculation of the response to their applications on the Cloud or accept that it is supported by a remote data center, to save energy or wait for the availability of renewable energy. The VMs are black boxes from the Cloud provider point of view. So, the user is the only one to know the applications running on her VMs.

We plan to explore possible collaborations between virtual machine suppliers, service providers and users of Clouds in order to provide users with ways of participating in the reduction of the Clouds energy consumption. This work will follow two directions: 1) to investigate compromises between power and performance/service quality that cloud providers can offer to their users and to propose them a variety of options adapted to their workload; and 2) to develop mechanisms for each layer of the Cloud software stack to provide users with a quantification of the energy consumed by each of their options as an incentive to become greener.

3.4. Securing clouds

3.4.1. Security monitoring SLO

While the trend for companies to outsource their information system in clouds is confirmed, the problem of securing an information system becomes more difficult. Indeed, in the case of infrastructure clouds, physical

resources are shared between companies (also called tenants) but each tenant controls only parts of the shared resources, and, thanks to virtualization, the information system can be dynamically and automatically reconfigured with added or removed resources (for example starting or stopping virtual machines), or even moved between physical resources (for example using virtual machine migration). Partial control of shared resources brings new classes of attacks between tenants, and security monitoring mechanisms to detect such attacks are better placed out of the tenant-controlled virtual information systems, that is under control of the cloud provider. Dynamic and automatic reconfigurations of the information system make it unfeasible for a tenant's security administrator to setup the security monitoring components to detect attacks, and thus an automated self-adaptable security monitoring service is required.

Combining the two previous statements, there is a need for a dependable, automatic security monitoring service provided to tenants by the cloud provider. Our goal is to address the following challenges to design such a security monitoring service:

1. to define relevant Service-Level Objectives (SLOs) of a security monitoring service, that can figure in the Service-Level Agreement (SLA) signed between a cloud provider and a tenant;
2. to design heuristics to automatically configure provider-controlled security monitoring software components and devices so that SLOs are reached, even during automatic reconfigurations of tenants' information systems;
3. to design evaluation methods for tenants to check that SLOs are reached.

Moreover in challenges 2 and 3 the following sub-challenges must be addressed:

- although SLAs are bi-lateral contracts between the provider and each tenant, the implementation of the contracts is based on shared resources, and thus we must study methods to combine the SLOs;
- the designed methods should have a minimal impact on performance.

3.4.2. Data Protection in Cloud-based IoT Services

The Internet of Things is becoming a reality. Individuals have their own swarm of connected devices (e.g. smartphone, wearables, and home connected objects) continually collecting personal data. A novel generation of services is emerging exploiting data streams produced by the devices' sensors. People are deprived of control of their personal data as they don't know precisely what data are collected by service providers operating on Internet (oISP), for which purpose they could be used, for how long they are stored, and to whom they are disclosed. In response to privacy concerns the European Union has introduced, with the Global Data Protection Regulation (GDPR), new rules aimed at enforcing the people's rights to personal data protection. The GDPR also gives strong incentives to oISPs to comply. However, today, oISPs can't make their systems GDPR-compliant since they don't have the required technologies. We argue that a new generation of system is mandatory for enabling oISPs to conform to the GDPR. We plan to design an open source distributed operating system for native implementation of new GDPR rules and ease the programming of compliant cloud-based IoT services. Among the new rules, transparency, right of erasure, and accountability are the most challenging ones to be implemented in IoT environments but could fundamentally increase people's confidence in oISPs. Deployed on individuals' swarms of devices and oISPs' cloud-hosted servers, it will enforce detailed data protection agreements and accountability of oISPs' data processing activities. Ultimately we will show to what extent the new GDPR rules can be implemented for cloud-based IoT services.

3.5. Experimenting with Clouds

Cloud platforms are challenging to evaluate and study with a sound scientific methodology. As with any distributed platform, it is very difficult to gather a global and precise view of the system state. Experiments are not reproducible by default since these systems are shared between several stakeholder. This is even worsened by the fact that microscopic differences in the experimental conditions can lead to drastic changes since typical Cloud applications continuously adapt their behavior to the system conditions.

3.5.1. Experimentation methodologies for clouds

We propose to combine two complementary experimental approaches: direct execution on testbeds such as Grid'5000, that are eminently believable but rather labor intensive, and simulations (using *e.g.* SimGrid) that are much more light-weighted, but requires careful assessment. One specificity of the Myriads team is that we are working on these experimental methodologies *per se*, raising the standards of *good experiments* in our community.

We plan to make SimGrid widely usable beyond research laboratories, in order to evaluate industrial systems and to teach the future generations of cloud practitioners. This requires to frame the specific concepts of Cloud systems and platforms in actionable interfaces. The challenge is to make the framework both easy to use for simple studies in educational settings while modular and extensible to suit the specific needs of every advanced industrial-class users.

We aim at leveraging the convergence opportunities between methodologies by further bridging simulation and real testbeds. The predictions obtained from the simulator should be validated against some real-world experiments obtained on the target production platform, or on a similar platform. This (in)validation of the predicted results often improves the understanding of the modeled system. On the other side, it may even happen that the measured discrepancies are due to some mis-configuration of the real platform that would have been undetected without this (in)validation study. In that sense, the simulator constitutes a precious tool for the quality assurance of real testbeds such as Grid'5000.

Scientists need more help to make their Cloud experiments fully reproducible, in the spirit of Open Science exemplified by the HAL Open Archive, actively backed by Inria. Users still need practical solutions to archive, share and compare the whole experimental settings, including the raw data production (particularly in the case of real testbeds) and their statistical analysis. This is a long lasting task to which we plan to collaborate through the research communities gathered around the Grid'5000 and SimGrid scientific instruments.

Finally, since correction and performance can constitute contradictory goals, it is particularly important to study them jointly. To that extend, we want to bridge the performance studies, that constitute our main scientific heritage, to correction studies leveraging formal techniques. SimGrid already includes to exhaustively explore the possible executions. We plan to continue this work to ease the use of the relevant formal methods to the experimenter studying Cloud systems.

3.5.2. Use cases

In system research it is important to work on real-world use cases from which we extract requirements inspiring new research directions and with which we can validate the system services and mechanisms we propose. In the framework of our close collaboration with the Data Science Technology department of the LBNL, we will investigate cloud usage for scientific data management. Next-generation scientific discoveries are at the boundaries of datasets, *e.g.*, across multiple science disciplines, institutions and spatial and temporal scales. Today, data integration processes and methods are largely adhoc or manual. A generalized resource infrastructure that integrates knowledge of the data and the processing tasks being performed by the user in the context of the data and resource lifecycle is needed. Clouds provide an important infrastructure platform that can be leveraged by including knowledge for distributed data integration.

PACAP Project-Team

3. Research Program

3.1. Motivation

Our research program is naturally driven by the evolution of our ecosystem. Relevant recent changes can be classified in the following categories: technological constraints, evolving community, and domain constraints. We hereby summarize these evolutions.

3.1.1. Technological constraints

Until recently, binary compatibility guaranteed portability of programs, while increased clock frequency and improved micro-architecture provided increased performance. However, in the last decade, advances in technology and micro-architecture started translating into more parallelism instead. Technology roadmaps even predict the feasibility of thousands of cores on a chip by 2020. Hundreds are already commercially available. Since the vast majority of applications are still sequential, or contain significant sequential sections, such a trend put an end to the automatic performance improvement enjoyed by developers and users. Many research groups consequently focused on parallel architectures and compiling for parallelism.

Still, the performance of applications will ultimately be driven by the performance of the sequential part. Despite a number of advances (some of them contributed by members of the team), sequential tasks are still a major performance bottleneck. Addressing it is still on the agenda of the proposed PACAP project-team.

In addition, due to power constraints, only part of the billions of transistors of a microprocessor can be operated at any given time (the *dark silicon* paradigm). A sensible approach consists in specializing parts of the silicon area to provide dedicated accelerators (not run simultaneously). This results in diverse and heterogeneous processor cores. Application and compiler designers are now confronted with a moving target, challenging portability and jeopardizing performance.

Finally, we live in a world where billions of sensors, actuators, and computers play a crucial role in our life: flight control, nuclear plant management, defense systems, banking, or health care. These systems must be reliable, despite the fact that they are subject to faults (for example due to aging, charged particle hit, or random noise). Faults will soon become the new *de facto* standard. The evolutions of the semiconductor industry predict an exponential growth of the number of permanent faults [56]. Reliability considerations usually degrade performance. We will propose solutions to mitigate this impact (for example by limiting overheads to critical sections).

Note on technology.

Technology also progresses at a fast pace. We do not propose to pursue any research on technology *per se*. Recently proposed paradigms (non-Si, brain-inspired) have received lots of attention from the research community. We do *not* intend to invest in those paradigms, but we will continue to investigate compilation and architecture for more conventional programming paradigms. Still, several technological shifts may have consequences for us, and we will closely monitor their developments, they include for example non-volatile memory (impacts security, makes writes longer than loads), 3D-stacking (impacts bandwidth), and photonics (impacts latencies and connection network).

3.1.2. Evolving community

The PACAP project-team tackles performance-related issues, for conventional programming paradigms. In fact, programming complex environments is no longer the exclusive domain of experts in compilation and architecture. A large community now develops applications for a wide range of targets, including mobile “apps”, cloud, multicore or heterogeneous processors.

This also includes domain scientists (in biology, medicine, but also social sciences) who started relying heavily on computational resources, gathering huge amounts of data, and requiring considerable amount of processing to analyze them. Our research is motivated by the growing discrepancy between on the one hand the complexity of the workloads and the computing systems, and on the other hand the expanding community of developers at large, with limited expertise to optimize and to map efficiently computations to compute nodes.

3.1.3. Domain constraints

Mobile, embedded systems have become ubiquitous. Many of them have real-time constraints. For this class of systems, correctness implies not only producing the correct result, but also doing so within specified deadlines. In the presence of heterogeneous, complex and highly dynamic systems, producing *tight* (i.e. useful) upper bound to the worst-case execution time has become extremely challenging. Our research will aim at improving the tightness as well as enlarging the set of features that can be safely analyzed.

The ever growing dependence of our economy on computing systems also implies that security has become of utmost importance. Many systems are under constant attacks from intruders. Protection has a cost also in terms of performance. We plan to leverage our background to contribute solutions that minimize this impact.

Note on Applications Domains.

PACAP works on fundamental technologies for computer science: processor architecture, performance-oriented compilation and guaranteed response time for real-time. The research results may have impacts on any application domain that requires high performance execution (telecommunication, multimedia, biology, health, engineering, environment...), but also on many embedded applications that exhibit other constraints such as power consumption, code size and guaranteed response time.

We strive to extract from active domains the fundamental characteristics that are relevant to our research. For example, *big data* is of interest to PACAP because it relates to the study of hardware/software mechanisms to efficiently transfer huge amounts of data to the computing nodes. Similarly, the *Internet of Things* is of interest because it has implications in terms of ultra low power consumption.

3.2. Research Objectives

Processor micro-architecture and compilation have been at the core of the research carried by the members of the project teams for two decades, with undeniable contributions. They continue to be the foundation of PACAP.

Heterogeneity and diversity of processor architectures now require new techniques to guarantee that the hardware is satisfactorily exploited by the software. One of our goals is to devise new static compilation techniques (cf. Section 3.2.1), but also build upon iterative [1] and split [2] compilation to continuously adapt software to its environment (Section 3.2.2). Dynamic binary optimization will also play a key role in delivering adapting software and delivering performance.

The end of Moore's law and Dennard's scaling⁰ offer an exciting window of opportunity, where performance improvements will no longer derive from additional transistor budget or increased clock frequency, but rather come from breakthroughs in microarchitecture (Section 3.2.3). Reconciling CPU and GPU designs (Section 3.2.4) is one of our objectives.

Heterogeneity and multicores are also major obstacles to determining tight worst-case execution times of real-time systems (Section 3.2.5), which we plan to tackle.

Finally, we also describe how we plan to address transversal aspects such as reliability (Section 3.2.6), power efficiency (Section 3.2.7), and security (Section 3.2.8).

⁰According to Dennard scaling, as transistors get smaller the power density remains constant, and the consumed power remains proportional to the area.

3.2.1. Static Compilation

Static compilation techniques continues to be relevant to address the characteristics of emerging hardware technologies, such as non-volatile memories, 3D-stacking, or novel communication technologies. These techniques expose new characteristics to the software layers. As an example, non-volatile memories typically have asymmetric read-write latencies (writes are much longer than reads) and different power consumption profiles. PACAP studies the new optimization opportunities and develop tailored compilation techniques for the upcoming compute nodes. New technologies may also be coupled with traditional solutions to offer new trade-offs. We study how programs can adequately exploit the specific features of the proposed heterogeneous compute nodes.

We propose to build upon iterative compilation [1] to explore how applications perform on different configurations. When possible, Pareto points are related to application characteristics. The best configuration, however, may actually depend on runtime information, such as input data, dynamic events, or properties that are available only at runtime. Unfortunately a runtime system has little time and means to determine the best configuration. For these reasons, we also leverage split-compilation [2]: the idea consists in pre-computing alternatives, and embedding in the program enough information to assist and drive a runtime system towards to the best solution.

3.2.2. Software Adaptation

More than ever, software needs to adapt to its environment. In most cases, this environment remains unknown until runtime. This is already the case when one deploys an application to a cloud, or an “app” to mobile devices. The dilemma is the following: for maximum portability, developers should target the most general device; but for performance they would like to exploit the most recent and advanced hardware features. JIT compilers can handle the situation to some extent, but binary deployment requires dynamic binary rewriting. Our work has shown how SIMD instructions can be upgraded from SSE to AVX [3]. Many more opportunities will appear with diverse and heterogeneous processors, featuring various kinds of accelerators.

On shared hardware, the environment is also defined by other applications competing for the same computational resources. It becomes increasingly important to adapt to changing runtime conditions, such as the contention of the cache memories, available bandwidth, or hardware faults. Fortunately, optimizing at runtime is also an opportunity, because this is the first time the program is visible as a whole: executable and libraries (including library versions). Optimizers may also rely on dynamic information, such as actual input data, parameter values, etc. We have already developed a software platform [16] to analyze and optimize programs at runtime, and we started working on automatic dynamic parallelization of sequential code, and dynamic specialization.

We started addressing some of these challenges in ongoing projects such as Nano2017 PSAIC Collaborative research program with STMicroelectronics, as well as within the Inria Project Lab MULTICORE. The starting H2020 FET HPC project ANTAREX also addresses these challenges from the energy perspective. We further leverage our platform and initial results to address other adaptation opportunities. Efficient software adaptation requires expertise from all domains tackled by PACAP, and strong interaction between all team members is expected.

3.2.3. Research directions in uniprocessor microarchitecture

Achieving high single-thread performance remains a major challenge even in the multicore era (Amdahl’s law). The members of the PACAP project-team have been conducting research in uniprocessor micro-architecture research for about 20 years covering major topics including caches, instruction front-end, branch prediction, out-of-order core pipeline, and value prediction. In particular, in the recent years they have been recognized as world leaders in branch prediction [22][11] and in cache prefetching [9] and they have revived the forgotten concept of value prediction [14][13]. This research was supported by the ERC Advanced grant DAL (2011-2016) and also by Intel. We pursue research on achieving ultimate uncore performance. Below are several non-orthogonal directions that we have identified for mid-term research:

1. management of the memory hierarchy (particularly the hardware prefetching);

2. practical design of very wide issue execution core;
3. speculative execution.

Memory design issues:

Performance of many applications is highly impacted by the memory hierarchy behavior. The interactions between the different components in the memory hierarchy and the out-of-order execution engine have high impact on performance.

The last *Data Prefetching Contest* held with ISCA 2015 has illustrated that achieving high prefetching efficiency is still a challenge for wide-issue superscalar processors, particularly those featuring a very large instruction window. The large instruction window enables an implicit data prefetcher. The interaction between this implicit hardware prefetcher and the explicit hardware prefetcher is still relatively mysterious as illustrated by Pierre Michaud's BO prefetcher (winner of DPC2) [9]. The first objective of the research is to better understand how the implicit prefetching enabled by the large instruction window interacts with the L2 prefetcher and then to understand how explicit prefetching on the L1 also interacts with the L2 prefetcher.

The second objective of the research is related to the interaction of prefetching and virtual/physical memory. On real hardware, prefetching is stopped by page frontiers. The interaction between TLB prefetching (and on which level) and cache prefetching must be analyzed.

The prefetcher is not the only actor in the hierarchy that must be carefully controlled. Significant benefit can also be achieved through careful management of memory access bandwidth, particularly the management of spatial locality on memory accesses, both for reads and writes. The exploitation of this locality is traditionally handled in the memory controller. However, it could be better handled if larger temporal granularity was available. Finally, we also intend to continue to explore the promising avenue of compressed caches. In particular we recently proposed the skewed compressed cache [17]. It offers new possibility for efficient compression schemes.

Ultra wide-issue superscalar:

To effectively leverage memory level parallelism, one requires huge out-of-order execution structures as well as very wide issue superscalar processor. For the two past decades, implementing ever wider issue superscalar processor has been challenging. The objective of our research on the execution core is to explore (and revisit) directions to allow the design of a very wide-issue (8-to-16 way) out-of-order execution core while mastering its complexity (silicon area, hardware logic complexity, power/energy consumption).

The first direction that we are exploring is the use of clustered architecture [10]. Symmetric clustered organization allows to benefit from simpler bypass network, but induce large complexity on the issue queue. One remarkable finding of our study [10] is that, when considering two large clusters (e.g. 8-wide) steering large groups of consecutive instructions (e.g. 64 μ ops) to the same cluster is quite efficient. This opens opportunities to limit the complexity of the issue queues (monitoring fewer buses) and register files (fewer ports and physical registers) in the clusters, since not all results have to be forwarded to the other cluster.

The second direction that we are exploring is associated with the approach that we developed with Sembrant et al. [18]. It reduces the number of instructions waiting in the instruction queues for the applications benefiting from very large instruction windows. Instructions are dynamically classified as ready (independent from any long latency instruction) or non-ready, and as urgent (part of a dependency chain leading to a long latency instruction) or non-urgent. Non-ready non-urgent instructions can be delayed until the long latency instruction has been executed; this allows to reduce the pressure on the issue queue. This proposition opens the opportunity to consider an asymmetric microarchitecture with a cluster dedicated to the execution of urgent instructions and a second cluster executing the non-urgent instructions. The microarchitecture of this second cluster could be optimized to reduce complexity and power consumption (smaller instruction queue, less aggressive scheduling...)

Speculative execution.

Out-of-order (OoO) execution relies on speculative execution that requires predictions of all sorts: branch, memory dependency, value...

The PACAP members have been major actors of the branch prediction research for the last 20 years; and their proposals have influenced the design of most of the hardware branch predictors in current microprocessors. We will continue to steadily explore new branch predictor designs as for instance [20].

In speculative execution, we have recently revisited value prediction (VP) which was a hot research topic between 1996 and 2002. However it was considered until recently that value prediction would lead to a huge increase in complexity and power consumption in every stage of the pipeline. Fortunately, we have recently shown that complexity usually introduced by value prediction in the OoO engine can be overcome [14][13][22][11]. First, very high accuracy can be enforced at reasonable cost in coverage and minimal complexity [14]. Thus, both prediction validation and recovery by squashing can be done outside the out-of-order engine, at commit time. Furthermore, we propose a new pipeline organization, EOLE ({Early | Out-of-order | Late} Execution), that leverages VP with validation at commit to execute many instructions outside the OoO core, in-order [13]. With EOLE, the issue-width in OoO core can be reduced without sacrificing performance, thus benefiting the performance of VP without a significant cost in silicon area and/or energy. In the near future, we will explore new avenues related to value prediction. These directions include register equality prediction and compatibility of value prediction with weak memory models in multiprocessors.

3.2.4. *Towards heterogeneous single-ISA CPU-GPU architectures*

Heterogeneous single-ISA architectures have been proposed in the literature during the 2000's [55] and are now widely used in the industry (ARM big.LITTLE, NVIDIA 4+1...) as a way to improve power-efficiency in mobile processors. These architectures include multiple cores whose respective microarchitectures offer different trade-offs between performance and energy efficiency, or between latency and throughput, while offering the same interface to software. Dynamic task migration policies leverage the heterogeneity of the platform by using the most suitable core for each application, or even each phase of processing. However, these works only tune cores by changing their complexity. Energy-optimized cores are either identical cores implemented in a low-power process technology, or simplified in-order superscalar cores, which are far from state-of-the-art throughput-oriented architectures such as GPUs.

We investigate the convergence of CPU and GPU at both architecture and compilation levels.

Architecture.

The architecture convergence between Single Instruction Multiple Threads (SIMT) GPUs and multicore processors that we have been pursuing [7] opens the way for heterogeneous architectures including latency-optimized superscalar cores and throughput-optimized GPU-style cores, which all share the same instruction set. Using SIMT cores in place of superscalar cores will enable the highest energy efficiency on regular sections of applications. As with existing single-ISA heterogeneous architectures, task migration will not necessitate any software rewrite and will accelerate existing applications.

Compilers for emerging heterogeneous architectures.

Single-ISA CPU+GPU architectures will provide the necessary substrate to enable efficient heterogeneous processing. However, it will also introduce substantial challenges at the software and firmware level. Task placement and migration will require advanced policies that leverage both static information at compile time and dynamic information at run-time. We are tackling the heterogeneous task scheduling problem at the compiler level. As a first step, we are prototyping scheduling algorithms on existing multiple-ISA CPU+GPU architectures like NVIDIA Tegra X1.

3.2.5. *Real-time systems*

Safety-critical systems (e.g. avionics, medical devices, automotive...) have so far used simple uncore hardware systems as a way to control their predictability, in order to meet timing constraints. Still, many critical embedded systems have increasing demand in computing power, and simple uncore processors are not sufficient anymore. General-purpose multicore processors are not suitable for safety-critical real-time systems, because they include complex micro-architectural elements (cache hierarchies, branch, stride and value predictors) meant to improve average-case performance, and for which worst-case performance is difficult to predict. The prerequisite for calculating tight WCET is a deterministic hardware system that avoids dynamic, time-unpredictable calculations at run-time.

Even for multi and manycore systems designed with time-predictability in mind (Kalray MPPA manycore architecture ⁰, or the Recore manycore hardware ⁰) calculating WCETs is still challenging. The following two challenges will be addressed in the mid-term:

1. definition of methods to estimate WCETs tightly on manycores, that smartly analyzes and/or controls shared resources such as buses, NoCs or caches;
2. methods to improve the programmability of real-time applications through automatic parallelization and optimizations from model-based designs.

3.2.6. Fault Tolerance

Technology trends suggest that, in tomorrow's computing world, failures will become commonplace due to many factors, and the expected probability of failure will increase with scaling. While well-known approaches, such as error correcting codes, exist to recover from failures and provide fault-free chips, the exponential growth of the number of faults will make them unaffordable in the future. Consequently, other approaches such as fine-grained disabling and reconfiguration of hardware elements (e.g. individual functional units or cache blocks) will become economically necessary. We are going to enter a new era: functionally correct chips with variable performance among chips and throughout their lifetime [56].

Transient and permanent faults may be detected by similar techniques, but correcting them generally involves different approaches. We are primarily interested in permanent faults, even though we do not necessarily disregard transient faults (e.g. the TMR approach in the next paragraph addresses both kind of faults).

CPU.

Permanent faults can occur anywhere in the processor. The performance implications of faulty cells vary depending on how the array is used in a processor. Most of micro-architectural work aiming at assessing the performance implications of permanently faulty cells relies on simulations with random fault-maps. These studies are, therefore, limited by the fault-maps they use that may not be representative for the average and distributed performance. They also do not consider aging effect.

Considering the memory hierarchy, we have already studied [5] the impact of permanent faults on the average and worst-case performance based on analytical models. We will extend these models to cover other components and other designs, and to analyze the interaction between faulty components.

For identified critical hardware structures, such as the memory hierarchy, we will propose protection mechanisms by for instance using larger cells, or even by selecting a different array organization to mitigate the impact of faults.

Another approach to deal with faults is to introduce redundancy at the code level. We propose to consider static compilation techniques focusing on existing hardware. As an example, we plan to leverage SIMD extensions of current instruction sets to introduce redundancy in scalar code at minimum cost. With these instructions, it will be possible to protect the execution from both soft errors by using TMR (triple modular redundancy) with voters in the code itself, and permanent faults without the need of extra hardware support to deconfigure faulty functional units.

Reconfigurable Computing.

In collaboration with the CAIRN project-team, we propose to construct Coarse Grain Reconfigurable Architectures (CGRA) from a sea of basic arithmetic and memory elements organized into clusters and connected through a hierarchical interconnection network. These clusters of basic arithmetic operators (e.g. 8-bit arithmetic and logic units) would be able to be seamlessly configured to various accuracy and data types to adapt the consumed energy to application requirements taking advantage of approximate computations. We propose to add new kinds of error detection (and sometimes correction) directly at the operator level by taking advantage of the massive redundancy of the array. As an example, errors can be tracked and detected in a complex sequence of double floating-point operations by using a reduced-precision version of the same processing.

⁰<http://www.kalrayinc.com>

⁰<http://www.recoresystems.com/>

Such reconfigurable blocks will be driven by compilation techniques, in charge of computing checkpoints, detecting faults, and replaying computations when needed.

Dynamic compilation techniques will help better exploit faulty hardware, by allocating data and computations on correct resources. In case of permanent faults, we will provide a mechanism to reconfigure the hardware, for example by reducing the issue width of VLIW processors implemented in CGRA. Dynamic code generation (JIT compiler) will re-generate code for the new configuration, guaranteeing portability and optimal exploitation of the hardware.

3.2.7. Power efficiency

PACAP addresses power-efficiency at several levels. First, we design static and split compilation techniques to contribute to the race for Exascale computing (the general goal is to reach 10^{18} FLOP/s at less than 20 MW). Second, we focus on high-performance low-power embedded compute nodes. Within the ANR project Continuum, in collaboration with architecture and technology experts from LIRMM and the SME Cortus, we research new static and dynamic compilation techniques that fully exploit emerging memory and NoC technologies. Finally, in collaboration with the CAIRN project-team, we investigate the synergy of reconfigurable computing and dynamic code generation.

Green and heterogeneous high-performance computing.

Concerning HPC systems, our approach consists in mapping, runtime managing and autotuning applications for green and heterogeneous High-Performance Computing systems up to the Exascale level. One key innovation of the proposed approach consists of introducing a separation of concerns (where self-adaptivity and energy efficient strategies are specified aside to application functionalities) promoted by the definition of a Domain Specific Language (DSL) inspired by aspect-oriented programming concepts for heterogeneous systems. The new DSL will be introduced for expressing adaptivity/energy/performance strategies and to enforce at runtime application autotuning and resource and power management. The goal is to support the parallelism, scalability and adaptability of a dynamic workload by exploiting the full system capabilities (including energy management) for emerging large-scale and extreme-scale systems, while reducing the Total Cost of Ownership (TCO) for companies and public organizations.

High-performance low-power embedded compute nodes.

We will address the design of next generation energy-efficient high-performance embedded compute nodes. It focuses at the same time on software, architecture and emerging memory and communication technologies in order to synergistically exploit their corresponding features. The approach of the project is organized around three complementary topics: 1) compilation techniques; 2) multicore architectures; 3) emerging memory and communication technologies. PACAP will focus on the compilation aspects, taking as input the software-visible characteristics of the proposed emerging technology, and making the best possible use of the new features (non-volatility, density, endurance, low-power).

Hardware Accelerated JIT Compilation.

Reconfigurable hardware offers the opportunity to limit power consumption by dynamically adjusting the number of available resources to the requirements of the running software. In particular, VLIW processors can adjust the number of available issue lanes. Unfortunately, changing the processor width often requires recompiling the application, and VLIW processors are highly dependent of the quality of the compilation, mainly because of the instruction scheduling phase performed by the compiler. Another challenge lies in the high constraints of the embedded system: the energy and execution time overhead due to the JIT compilation must be carefully kept under control.

We started exploring ways to reduce the cost of JIT compilation targeting VLIW-based heterogeneous many-core systems. Our approach relies on a hardware/software JIT compiler framework. While basic optimizations and JIT management are performed in software, the compilation back-end is implemented by means of specialized hardware. This back-end involves both instruction scheduling and register allocation, which are known to be the most time-consuming stages of such a compiler.

3.2.8. Security

Security is a mandatory concern of any modern computing system. Various threat models have led to a multitude of protection solutions. Members of PACAP already contributed, thanks to the HAVEGE [59] random number generator, and code obfuscating techniques (the obfuscating just-in-time compiler [54], or thread-based control flow mangling [58]).

We partner with security experts who can provide intuition, know-how and expertise, in particular in defining threat models, and assessing the quality of the solutions. Our background in compilation and architecture helps design more efficient and less expensive protection mechanisms.

We already have ongoing research directions related to security. SECURE (Secure Codes to Thwart Cyber-physical Attacks) is a project started January 2016, in collaboration with security experts from Télécom Paris Tech, Paris 8, Université Catholique de Louvain (Belgium), and University of Sabancı (Turkey). We also plan to partner with the Inria/CentraleSupélec CIDRE project-team to design a tainting technique based on a just-in-time compiler.

Compiler-based data protection.

We specify and design error correction codes suitable for an efficient protection of sensitive information in the context of Internet of Things (IoT) and connected objects. We partner with experts in security and codes to prototype a platform that demonstrates resilient software. PACAP's expertise is key to select and tune the protection mechanisms developed within the project, and to propose safe, yet cost-effective solutions from an implementation point of view.

JIT-based tainting.

Dynamic information flow control (DIFC, also known as *tainting*) is used to detect intrusions and to identify vulnerabilities. It consists in attaching metadata (called *taints* or *labels*) to information containers, and to propagate the taints when particular operations are applied to the containers: reads, writes, etc. The goal is then to guarantee that confidential information is never used to generate data sent to an untrusted container; conversely, data produced by untrusted entities cannot be used to update sensitive data.

The containers can be of various granularities: fine-grain approaches can deal with single variables, coarser-grain approaches consider a file as a whole. The CIDRE project-team has developed several DIFC monitors. kBlare is coarse-grain monitor in the Linux kernel. JBlare is a fine-grain monitor for Java applications. Fine-grain monitors provide a better precision at the cost of a significant overhead in execution time.

Combining the expertise of CIDRE in DIFC with our expertise in JIT compilation will help design hybrid approaches. An initial static analysis of the program prior to installation or execution will feed information to a dynamic analyzer that propagates taints during just-in-time compilation.

PANAMA Project-Team

3. Research Program

3.1. Axis 1: Sparse Models and Representations

3.1.1. *Efficient Sparse Models and Dictionary Design for Large-scale Data*

Sparse models are at the core of many research domains where the large amount and high-dimensionality of digital data requires concise data descriptions for efficient information processing. Recent breakthroughs have demonstrated the ability of these models to provide concise descriptions of complex data collections, together with algorithms of provable performance and bounded complexity.

A crucial prerequisite for the success of today's methods is the knowledge of a "dictionary" characterizing how to concisely describe the data of interest. Choosing a dictionary is currently something of an "art", relying on expert knowledge and heuristics.

Pre-chosen dictionaries such as wavelets, curvelets or Gabor dictionaries, are based upon stylized signal models and benefit from fast transform algorithms, but they fail to fully describe the content of natural signals and their variability. They do not address the huge diversity underlying modern data much beyond time series and images: data defined on graphs (social networks, internet routing, brain connectivity), vector valued data (diffusion tensor imaging of the brain), multichannel or multi-stream data (audiovisual streams, surveillance networks, multimodal biomedical monitoring).

The alternative to a pre-chosen dictionary is a trained dictionary learned from signal instances. While such representations exhibit good performance on small-scale problems, they are currently limited to low-dimensional signal processing due to the necessary training data, memory requirements and computational complexity. Whether designed or learned from a training corpus, dictionary-based sparse models and the associated methodology fail to scale up to the volume and resolution of modern digital data, for they intrinsically involve difficult linear inverse problems. To overcome this bottleneck, a new generation of efficient sparse models is needed, beyond dictionaries, encompassing the ability to provide sparse and structured data representations as well as computational efficiency. For example, while dictionaries describe low-dimensional signal models in terms of their "synthesis" using few elementary building blocks called atoms, in "analysis" alternatives the low-dimensional structure of the signal is rather "carved out" by a set of equations satisfied by the signal. Linear as well as nonlinear models can be envisioned.

3.1.2. *Compressive Learning*

A flagship emerging application of sparsity is the paradigm of compressive sensing, which exploits sparse models at the analog and digital levels for the acquisition, compression and transmission of data using limited resources (fewer/less expensive sensors, limited energy consumption and transmission bandwidth, etc.). Besides sparsity, a key pillar of compressive sensing is the use of random low-dimensional projections. Through compressive sensing, random projections have shown their potential to allow drastic dimension reduction with controlled information loss, provided that the projected signal vector admits a sparse representation in some transformed domain. A related scientific domain, where sparsity has been recognized as a key enabling factor, is Machine Learning, where the overall goal is to design statistically founded principles and efficient algorithms in order to infer general properties of large data collections through the observation of a limited number of representative examples. Marrying sparsity and random low-dimensional projections with machine learning shall allow the development of techniques able to efficiently capture and process the information content of large data collections. The expected outcome is a dramatic increase of the impact of sparse models in machine learning, as well as an integrated framework from the signal level (signals and their acquisition) to the semantic level (information and its manipulation), and applications to data sizes and volumes of collections that cannot be handled by current technologies.

3.2. Axis 2: Robust Acoustic Scene Analysis

3.2.1. *Compressive Acquisition and Processing of Acoustic Scenes*

Acoustic imaging and scene analysis involve acquiring the information content from acoustic fields with a limited number of acoustic sensors. A full 3D+t field at CD quality and Nyquist spatial sampling represents roughly 10^6 microphones/ m^3 . Dealing with such high-dimensional data requires to drastically reduce the data flow by positioning appropriate sensors, and selecting from all spatial locations the few spots where acoustic sources are active. The main goal is to develop a theoretical and practical understanding of the conditions under which compressive acoustic sensing is both feasible and robust to inaccurate modeling, noisy measures, and partially failing or uncalibrated sensing devices, in various acoustic sensing scenarios. This requires the development of adequate algorithmic tools, numerical simulations, and experimental data in simple settings where hardware prototypes can be implemented.

3.2.2. *Robust Audio Source Separation*

Audio signal separation consists in extracting the individual sound of different instruments or speakers that were mixed on a recording. It is now successfully addressed in the academic setting of linear instantaneous mixtures. Yet, real-life recordings, generally associated to reverberant environments, remain an unsolved difficult challenge, especially with many sources and few audio channels. Much of the difficulty comes from the combination of (i) complex source characteristics, (ii) sophisticated underlying mixing model and (iii) adverse recording environments. Moreover, as opposed to the “academic” blind source separation task, most applicative contexts and new interaction paradigms offer a variety of situations in which prior knowledge and adequate interfaces enable the design and the use of informed and/or manually assisted source separation methods.

The former METISS team has developed a generic and flexible probabilistic audio source separation framework that has the ability to combine various acoustic models such as spatial and spectral source models. Building on this existing framework, a first objective of PANAMA is to instantiate and validate specific instances of this framework targeted to real-world industrial applications, such as 5.1 movie re-mastering, interactive music soloist control and outdoor speech enhancement. Extensions of the framework are needed to achieve real-time online processing, and advanced constraints or probabilistic priors for the sources at hand need to be designed, while paying attention to computational scalability issues.

In parallel to these efforts, expected progress in sparse modeling for inverse problems shall bring new approaches to source separation and modeling, as well as to source localization, which is often an important first step in a source separation workflow.

3.2.3. *Robust Audio Source Localization*

Audio source localization consists in estimating the position of one or several sound sources given the signals received by a microphone array. Knowing the geometry of an audio scene is often a pre-requisite to perform higher-level tasks such as speaker identification and tracking, speech enhancement and recognition or audio source separation. It can be decomposed into two sub-tasks : (i) compute spatial auditory features from raw audio input and (ii) map these features to the desired spatial information. Robustly addressing both these aspects with a limited number of microphones, in the presence of noise, reverberation, multiple and possibly moving sources remains a key challenge in audio signal processing. The first aspect will be tackled by both advanced statistical and acoustical modeling of spatial auditory features. The second one will be addressed by two complementary approaches. *Physics-driven* approaches cast sound source localization as an inverse problem given the known physics of sound propagation within the considered system. *Data-driven* approaches aim at learning the desired feature-to-source-position mapping using real-world or synthetic training datasets adapted to the problem at hand. Combining these approaches should allow a widening of the notion of source localization, considering problems such as the identification of the directivity or diffuseness of the source as well as some of the boundary conditions of the room. A general perspective is to investigate the relations between the physical structure of the source and the particular structures that can be discovered or enforced in the representations and models used for characterization, localization and separation.

3.3. Axis 3: Large-scale Audio Content Processing and Self-organization

3.3.1. Motif Discovery in Audio Data

Facing the ever-growing quantity of multimedia content, the topic of motif discovery and mining has become an emerging trend in multimedia data processing with the ultimate goal of developing weakly supervised paradigms for content-based analysis and indexing. In this context, speech, audio and music content, offers a particularly relevant information stream from which meaningful information can be extracted to create some form of “audio icons” (key-sounds, jingles, recurrent locutions, musical choruses, etc ...) without resorting to comprehensive inventories of expected patterns.

This challenge raises several fundamental questions that will be among our core preoccupations over the next few years. The first question is the deployment of motif discovery on a large scale, a task that requires extending audio motif discovery approaches to incorporate efficient time series pattern matching methods (fingerprinting, similarity search indexing algorithms, stochastic modeling, etc.). The second question is that of the use and interpretation of the motifs discovered. Linking motif discovery and symbolic learning techniques, exploiting motif discovery in machine learning are key research directions to enable the interpretation of recurring motifs.

On the application side, several use cases can be envisioned which will benefit from motif discovery deployed on a large scale. For example, in spoken content, word-like repeating fragments can be used for several spoken document-processing tasks such as language-independent topic segmentation or summarization. Recurring motifs can also be used for audio summarization of audio content. More fundamentally, motif discovery paves the way for a shift from supervised learning approaches for content description to unsupervised paradigms where concepts emerge from the data.

3.3.2. Structure Modeling and Inference in Audio and Musical Contents

Structuring information is a key step for the efficient description and learning of all types of contents, and in particular audio and musical contents. Indeed, structure modeling and inference can be understood as the task of detecting dependencies (and thus establishing relationships) between different fragments, parts or sections of information content.

A stake of structure modeling is to enable more robust descriptions of the properties of the content and better model generalization abilities that can be inferred from a particular content, for instance via cache models, trigger models or more general graphical models designed to render the information gained from structural inference. Moreover, the structure itself can become a robust descriptor of the content, which is likely to be more resistant than surface information to a number of operations such as transmission, transduction, copyright infringement or illegal use.

In this context, information theory concepts need to be investigated to provide criteria and paradigms for detecting and modeling structural properties of audio contents, covering potentially a wide range of application domains in speech content mining, music modeling or audio scene monitoring.

SERPICO Project-Team

3. Research Program

3.1. Statistics and algorithms for computational microscopy

Fluorescence microscopy limitations are due to the optical aberrations, the resolution of the microscopy system, and the photon budget available for the biological specimen. Hence, new concepts have been defined to address challenging image restoration and molecule detection problems while preserving the integrity of samples. Accordingly, the main stream regarding denoising, deconvolution, registration and detection algorithms advocates appropriate signal processing framework to improve spatial resolution, while at the same time pushing the illumination to extreme low levels in order to limit photo-damages and phototoxicity [7], [6]. As a consequence, the question of adapting cutting-edge signal denoising and deconvolution, object detection, and image registration methods to 3D fluorescence microscopy imaging has retained the attention of several teams over the world.

In this area, the Serpico team has developed a strong expertise in key topics in computational imaging including image denoising and deconvolution, object detection and multimodal image registration. Several algorithms proposed by the team outperformed the state-of-the-art results, and some developments are compatible with “high-throughput microscopy” and the processing of several hundreds of cells. We especially promoted non local, non-parametric and patch-based methods to solve well-known inverse problems or more original reconstruction problems. A recent research direction consists in adapting the deep learning concept to solve challenging detection and reconstruction problems in microscopy. We have investigated convolution neural networks to detect small macromolecules in 3D noisy electron images with promising results. The next step consists in proposing smart paradigms and architectures to save memory and computations.

More generally, many inverse problems and image processing become intractable with modern 3D microscopy, because very large temporal series of volumes (200 to 1000 images per second for one 3D stack) are acquired for several hours. Novel strategies are needed for 3D image denoising, deconvolution and reconstruction since computation is extremely heavy. Accordingly, we will adapt the estimator aggregation approach developed for optical flow computation to meet the requirements of 3D image processing. We plan to investigate regularization-based aggregation energy over super-voxels to reduce complexity, combined to modern optimization algorithms. Finally, we will design parallelized algorithms that fast process 3D images, perform energy minimization in few seconds per image, and run on low-cost graphics processor boards (GPU).

3.2. From image data to motion descriptors: trajectory computation and dynamics analysis

Several particle tracking methods for intracellular analysis have been tailored to cope with different types of cellular and subcellular motion down to Brownian single molecule behavior. Many algorithms were carefully evaluated on the particle tracking challenge dataset published in the Nature Methods journal in 2014 [8]. Actually, there is no definitive solution to the particle tracking problem which remains application-dependent in most cases. The work of Serpico in particle motion analysis is significant in multiple ways, and inserts within a very active international context. One of the remaining key open issues is the tracking of objects with heterogeneous movements in crowded configurations. Moreover, particle tracking methods are not always adapted for motion analysis, especially when the density of moving features hampers the individual extraction of objects of interest undergoing complex motion. Estimating flow fields can be more appropriate to capture the complex dynamics observed in biological sequences. The existing optical flow methods can be classified into two main categories: i/ local methods impose a parametric motion model (e.g. local translation) in a given neighborhood; ii/ global methods estimate the dense motion field by minimizing a global energy functional composed of a data term and a regularization term.

The Serpico team has developed a strong expertise in key topics, especially in object tracking for fluorescence microscopy, optical flow computation and high-level analysis of motion descriptors and trajectories. Several algorithms proposed by the team are very competitive when compared to the state-of-the-art results, and our new paradigms offer promising ways for molecule traffic quantification and analysis. Amongst the problems that we currently address, we can mention: computation of 3D optical flow for large-size images, combination of two frame-based differential methods and sparse sets of trajectories, detection and analysis of unexpected local motion patterns in global coherent collective motion. Development of efficient numerical schemes will be central in the future but visualization methods are also crucial for evaluation and quality assessment. Another direction of research consists in exploiting deep learning to 3D optical flow so as to develop efficient numerical schemes that naturally capture complex motion patterns. Investigation in machine learning and statistics will be actually conducted in the team in the two first research axes to address a large range of inverse problems in bioimaging. Deep learning is an appealing approach since expertise of biologists, via iterative annotation of training data, will be included in the design of image analysis schemes.

3.3. Biological and biophysical models and spatial statistics for quantitative bioimaging

A number of stochastic mathematical models were proposed to describe various intracellular trafficking, where molecules and proteins are transported to their destinations via free diffusion, subdiffusion and ballistic motion representing movements along the cytoskeleton networks assisted by molecular motors. Accordingly, the study of diffusion and stochastic dynamics has known a growing interest in bio-mathematics, biophysics and cell biology with the popularization of fluorescence dynamical microscopy and super-resolution imaging. In this area, the competing teams mainly studied MSD and fluorescence correlation spectroscopy methods.

In the recent period, the Serpico team achieved important results for diffusion-related dynamics involved in exocytosis mechanisms. Robustness to noise has been well investigated, but robustness to environmental effects has yet to be effectively achieved. Particular attention has been given to the estimation of particle motion regime changes, but the available results are still limited for analysing short tracks. The analysis of spatiotemporal molecular interactions from set of 3D computed trajectories or motion vector fields (e.g., co-alignment) must be investigated to fully quantify specific molecular machineries. We have already made efforts in that directions this year (e.g., for colocalization) but important experiments are required to make our preliminary algorithms reliable enough and well adapted to specific transport mechanisms.

Accordingly, we will study quantification methods to represent interactions between molecules and trafficking around three lines of research. First, we will focus on 3D space-time global and local object-based co-orientation and co-alignment methods, in the line of previous work on colocalization, to quantify interactions between molecular species. In addition, given N tracks associated to N molecular species, interaction descriptors, dynamics models and stochastic graphical models representing molecular machines will be studied in the statistical data assimilation framework. Second, we will analyse approaches to estimate molecular mobility, active transport and motion regime changes from computed trajectories in the Lagrangian and Eulerian settings. We will focus on the concept of super-resolution to provide spatially high-resolved maps of diffusion and active transport parameters based on stochastic biophysical models and sparse image representation. Third, we plan to extend the aggregation framework dedicated to optical flow to the problem of diffusion-transport estimation. Finally, we will investigate data assimilation methods to better combine algorithms, models, and experiments in an iterative and virtuous circle. The overview of ultrastructural organization will be achieved by additional 3D electron microscopy technologies.

SIROCCO Project-Team

3. Research Program

3.1. Introduction

The research activities on analysis, compression and communication of visual data mostly rely on tools and formalisms from the areas of statistical image modelling, of signal processing, of coding and information theory. However, the objective of better exploiting the Human Visual System (HVS) properties in the above goals also pertains to the areas of perceptual modelling and cognitive science. Some of the proposed research axes are also based on scientific foundations of computer vision (e.g. multi-view modelling and coding). We have limited this section to some tools which are central to the proposed research axes, but the design of complete compression and communication solutions obviously rely on a large number of other results in the areas of motion analysis, transform design, entropy code design, etc which cannot be all described here.

3.2. Parameter Estimation and Inference

Bayesian estimation, Expectation-Maximization, stochastic modelling

Parameter estimation is at the core of the processing tools studied and developed in the team. Applications range from the prediction of missing data or future data, to extracting some information about the data in order to perform efficient compression. More precisely, the data are assumed to be generated by a given stochastic data model, which is partially known. The set of possible models translates the a priori knowledge we have on the data and the best model has to be selected in this set. When the set of models or equivalently the set of probability laws is indexed by a parameter (scalar or vectorial), the model is said parametric and the model selection resorts to estimating the parameter. Estimation algorithms are therefore widely used at the encoder to analyze the data. In order to achieve high compression rates, the parameters are usually not sent and the decoder has to jointly select the model (i.e. estimate the model parameters) and extract the information of interest.

3.3. Data Dimensionality Reduction

Manifolds, locally linear embedding, non-negative matrix factorization, principal component analysis

A fundamental problem in many data processing tasks (compression, classification, indexing) is to find a suitable representation of the data. It often aims at reducing the dimensionality of the input data so that tractable processing methods can then be applied. Well-known methods for data dimensionality reduction include principal component analysis (PCA) and independent component analysis (ICA). The methodologies which will be central to several proposed research problems will instead be based on sparse representations, on locally linear embedding (LLE) and on the “non negative matrix factorization” (NMF) framework.

The objective of *sparse representations* is to find a sparse approximation of a given input data. In theory, given $A \in \mathbb{R}^{m \times n}$, $m < n$, and $\mathbf{b} \in \mathbb{R}^m$ with $m \ll n$ and A is of full rank, one seeks the solution of $\min\{\|\mathbf{x}\|_0 : A\mathbf{x} = \mathbf{b}\}$, where $\|\mathbf{x}\|_0$ denotes the L_0 norm of x , i.e. the number of non-zero components in z . There exist many solutions x to $Ax = b$. The problem is to find the sparsest, the one for which x has the fewest non zero components. In practice, one actually seeks an approximate and thus even sparser solution which satisfies $\min\{\|\mathbf{x}\|_0 : \|A\mathbf{x} - \mathbf{b}\|_p \leq \rho\}$, for some $\rho \geq 0$, characterizing an admissible reconstruction error. The norm p is usually 2, but could be 1 or ∞ as well. Except for the exhaustive combinatorial approach, there is no known method to find the exact solution under general conditions on the dictionary A . Searching for this sparsest representation is hence unfeasible and both problems are computationally intractable. Pursuit algorithms have been introduced as heuristic methods which aim at finding approximate solutions to the above problem with tractable complexity.

Non negative matrix factorization (NMF) is a non-negative approximate data representation⁰. NMF aims at finding an approximate factorization of a non-negative input data matrix V into non-negative matrices W and H , where the columns of W can be seen as *basis vectors* and those of H as coefficients of the linear approximation of the input data. Unlike other linear representations like PCA and ICA, the non-negativity constraint makes the representation purely additive. Classical data representation methods like PCA or Vector Quantization (VQ) can be placed in an NMF framework, the differences arising from different constraints being placed on the W and H matrices. In VQ, each column of H is constrained to be unitary with only one non-zero coefficient which is equal to 1. In PCA, the columns of W are constrained to be orthonormal and the rows of H to be orthogonal to each other. These methods of data-dependent dimensionality reduction will be at the core of our visual data analysis and compression activities.

3.4. Perceptual Modelling

Saliency, visual attention, cognition

The human visual system (HVS) is not able to process all visual information of our visual field at once. To cope with this problem, our visual system must filter out irrelevant information and reduce redundant information. This feature of our visual system is driven by a selective sensing and analysis process. For instance, it is well known that the greatest visual acuity is provided by the fovea (center of the retina). Beyond this area, the acuity drops down with the eccentricity. Another example concerns the light that impinges on our retina. Only the visible light spectrum lying between 380 nm (violet) and 760 nm (red) is processed. To conclude on the selective sensing, it is important to mention that our sensitivity depends on a number of factors such as the spatial frequency, the orientation or the depth. These properties are modeled by a sensitivity function such as the Contrast Sensitivity Function (CSF).

Our capacity of analysis is also related to our visual attention. Visual attention which is closely linked to eye movement (note that this attention is called *overt* while the covert attention does not involve eye movement) allows us to focus our biological resources on a particular area. It can be controlled by both top-down (i.e. goal-directed, intention) and bottom-up (stimulus-driven, data-dependent) sources of information⁰. This detection is also influenced by prior knowledge about the environment of the scene⁰. Implicit assumptions related to prior knowledge or beliefs play an important role in our perception (see the example concerning the assumption that light comes from above-left). Our perception results from the combination of prior beliefs with data we gather from the environment. A Bayesian framework is an elegant solution to model these interactions⁰. We define a vector \vec{v}_l of local measurements (contrast of color, orientation, etc.) and vector \vec{v}_c of global and contextual features (global features, prior locations, type of the scene, etc.). The salient locations S for a spatial position \vec{x} are then given by:

$$S(\vec{x}) = \frac{1}{p(\vec{v}_l | \vec{v}_c)} \times p(s, \vec{x} | \vec{v}_c) \quad (50)$$

The first term represents the bottom-up salience. It is based on a kind of contrast detection, following the assumption that rare image features are more salient than frequent ones. Most of existing computational models of visual attention rely on this term. However, different approaches exist to extract the local visual features as well as the global ones. The second term is the contextual priors. For instance, given a scene, it indicates which parts of the scene are likely the most salient.

⁰D.D. Lee and H.S. Seung, "Algorithms for non-negative matrix factorization", Nature 401, 6755, (Oct. 1999), pp. 788-791.

⁰L. Itti and C. Koch, "Computational Modelling of Visual Attention", Nature Reviews Neuroscience, Vol. 2, No. 3, pp. 194-203, 2001.

⁰J. Henderson, "Regarding scenes", Directions in Psychological Science, vol. 16, pp. 219-222, 2007.

⁰L. Zhang, M. Tong, T. Marks, H. Shan, H. and G.W. Cottrell, "SUN: a Bayesian framework for saliency using natural statistics", Journal of Vision, vol. 8, pp. 1-20, 2008.

3.5. Coding theory

OPTA limit (Optimum Performance Theoretically Attainable), Rate allocation, Rate-Distortion optimization, lossy coding, joint source-channel coding multiple description coding, channel modelization, oversampled frame expansions, error correcting codes.

Source coding and channel coding theory⁰ is central to our compression and communication activities, in particular to the design of entropy codes and of error correcting codes. Another field in coding theory which has emerged in the context of sensor networks is Distributed Source Coding (DSC). It refers to the compression of correlated signals captured by different sensors which do not communicate between themselves. All the signals captured are compressed independently and transmitted to a central base station which has the capability to decode them jointly. DSC finds its foundation in the seminal Slepian-Wolf⁰ (SW) and Wyner-Ziv⁰ (WZ) theorems. Let us consider two binary correlated sources X and Y . If the two coders communicate, it is well known from Shannon's theory that the minimum lossless rate for X and Y is given by the joint entropy $H(X, Y)$. Slepian and Wolf have established in 1973 that this lossless compression rate bound can be approached with a vanishing error probability for long sequences, even if the two sources are coded separately, provided that they are decoded jointly and that their correlation is known to both the encoder and the decoder.

In 1976, Wyner and Ziv considered the problem of coding of two correlated sources X and Y , with respect to a fidelity criterion. They have established the rate-distortion function $R_{X|Y}^*(D)$ for the case where the side information Y is perfectly known to the decoder only. For a given target distortion D , $R_{X|Y}^*(D)$ in general verifies $R_{X|Y}(D) \leq R_{X|Y}^*(D) \leq R_X(D)$, where $R_{X|Y}(D)$ is the rate required to encode X if Y is available to both the encoder and the decoder, and R_X is the minimal rate for encoding X without SI. These results give achievable rate bounds, however the design of codes and practical solutions for compression and communication applications remain a widely open issue.

⁰T. M. Cover and J. A. Thomas, Elements of Information Theory, Second Edition, July 2006.

⁰D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources." IEEE Transactions on Information Theory, 19(4), pp. 471-480, July 1973.

⁰A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder." IEEE Transactions on Information Theory, pp. 1-10, January 1976.

SUMO Project-Team

3. Research Program

3.1. Analysis and verification of quantitative systems

The overall objective of this axis is to develop the quantitative aspects of formal methods while maintaining the tractability of verification objectives and progressing toward the management of large systems. This covers the development of relevant modeling formalisms, to nicely weave time, costs and probabilities with existing models for concurrency. We plan to further study time(d) Petri nets, networks of timed automata (with synchronous or asynchronous communications), stochastic automata, partially-observed Markov decision processes, etc. A second objective is to develop verification methods for such quantitative systems. This covers several aspects: quantitative verification questions (e.g. computing an optimal scheduling policy), Boolean questions on quantitative features (deciding whether some probability is greater than a threshold), robustness issues (will a system have the same behaviors if some parameter is slightly altered?), etc. Our goal is to explore the frontier between decidable and undecidable problems, or more pragmatically tractable and untractable problems. Of course, there is a tradeoff between the expressivity and the tractability of a model. Models that incorporate distributed aspects, probabilities, time, etc., are typically untractable. In such a case, abstraction or approximation techniques are a workaround that we will explore.

Here are some precise topics that we place in our agenda:

- analysis of diagnosability and opacity properties for stochastic systems;
- verification of time(d) Petri nets;
- robustness analysis for timed and/or stochastic systems;
- abstraction techniques for quantitative systems.

3.2. Control of quantitative systems

The main objective of this research axis is to explore the quantitative and/or distributed extensions of classical control problems. We envision control in its widest meaning of driving a system in order to guarantee or enforce some extra property (i.e. not guaranteed by the system alone), in a partially- or totally-observed setting. This property can either be logical (e.g. reachability or safety) or quantitative (e.g. reach some performance level). These problems have of course an offline facet (e.g. controller design, existence of a policy/strategy) and an online facet (e.g. algorithm to select some optimal action at runtime).

Our objectives comprise classical controller synthesis for discrete-event systems, with extensions to temporal/stochastic/reward settings. They also cover maintaining or maximizing extra properties such as diagnosability or opacity, for example in stochastic systems. We also target further analysis of POMDPs (partially-observed Markov decision processes), and multi-agent versions of policy synthesis relying on tools from game theory. We aim at addressing some control problems motivated by industrial applications, that raise issues like the optimal control of timed and stochastic discrete-event systems, with concerns like robustness to perturbations and multicriteria optimization. Finally, we also plan to work on modular testing, and on runtime enforcement techniques, in order to guarantee extra logical and temporal properties to event flows.

3.3. Management of large or distributed systems

The generic terms of “supervision” or “management” of distributed systems cover problems like control, diagnosis, sensor placement, planning, optimization, (state) estimation, parameter identification, testing, etc. This research axis examines how classical settings for such problems can scale up to large or distributed systems. Our work will be driven by considerations like: how to take advantage of modularity, how to design approximate management algorithms, how to design relevant abstractions to make large systems more tractable, how to deal with models of unknown size, how to design mechanisms to obtain relevant models, etc.

As more specific objectives, let us mention:

- Parametric-size systems: how to verify properties of distributed systems with an unknown number of components;
- Approximate management methods: we will explore the extension of ideas developed for Bayesian inference in large-scale stochastic systems (such as turbo-algorithms) to the field of modular dynamic systems. When component interactions are sparse, even if exact management methods are unaccessible (for diagnosis, planning, control, etc.), good approximations based on local computations may be accessible;
- Model abstraction: we will explore techniques to design more tractable abstractions of stochastic dynamic systems defined on large sets of variables;
- Self-modelling, which consists in managing large-scale systems that are known by their building rules, but where the specific instance is only discovered on-the-fly at runtime. The model of the managed system is built on-line, following the needs of the management algorithms;
- Distributed control: we will tackle issues related to asynchronous communications between local controllers, and to abstraction techniques allowing to address large systems;
- Test and enforcement: we will tackle coverage issues for the test of large systems, and the test and enforcement of properties for timed models, or for systems handling data.

3.4. Data driven systems

Data-driven systems are systems whose behaviour depends both on explicit workflows (scheduling and durations of tasks, calls to possibly distant services, ...) and on the data processed by the system (stored data, parameters of a request, results of a request, ...). This family of systems covers workflows that convey data (business processes or information systems), transactional systems (web stores), large databases managed with rules (banking systems), collaborative environments (crowds, health systems), etc. These systems are distributed, modular, and open: they integrate components and sub-services distributed over the web, and accept requests from clients. Our objective is to provide validation and supervision tools for such systems. To achieve this goal, we have to solve several challenging tasks:

- provide realistic models, and sound automated abstraction techniques, to reason on models that are reasonable abstractions of real systems. These models should be able to encompass modularity, distribution, in a context where workflows and data aspects are tightly connected;
- address design of data driven systems in a declarative way: declarative models are another way to handle data-driven systems. Rather than defining the explicit workflows and their effects on data, rule-based models state how actions are enacted in terms of the shape (pattern matching) or value of the current data. We think that distributed rewriting rules or attributed grammars can provide a practical yet formal framework for maintenance, by providing a solution to update mandatory documentation during the lifetime of an artifact.
- provide tractable solutions for validation of models: frequent issues are safety questions (can a system reach some bad configuration?), but also liveness (workflows progress), ... These questions should not only remain decidable on our models, but also with efficient computational methods.
- address QoS management in large reconfigurable systems: data-driven distributed systems often have constraints in terms of QoS. This QoS questions adress performance issues, but also data quality. This calls for an analysis of quantitative features and for reconfiguration techniques to meet desired QoS.

TACOMA Team

3. Research Program

3.1. Collecting pertinent information

In our model, applications adapt their behavior (for instance, the level of automation) to the quality of their perception of the environment. This is important to alleviate the development constraint we usually have on automated system. We "just" have to be sure a given process will always operate at the right automation level given the precision, the completeness or the confidence it has on its own perception. For instance, a car passing through a cross would choose its speed depending on the confidence it has gained during perception data gathering. When it has not enough information or when it could not trust it, it should reduce the automation level, therefore the speed, to only rely on its own sensors. Such adaptation capability shift requirements from the design and deployment (availability, robustness, accuracy, etc.) to the **assessment of the environment perception** we aim to facilitate in this first research axis.

Data characterization. The quality (freshness, accuracy, confidence, reliability, confidentiality, etc.) of the data are of crucial importance to assess the quality of the perception and therefore to ensure proper behavior. The way data is produced, consolidated, and aggregated while flowing to the consumer has an impact on its quality. Moreover part of these quality attributes requires to gather information at several communication layers from various entities. For this purpose, we want to design **lightweight cross-layer interactions** to collect relevant data. As a "frugality" principle should guide our approach, it is not appropriate to build all attributes we can imagine. It is therefore necessary to identify attributes relevant to the application and to have mechanisms to activate/deactivate at run-time the process to collect them.

Data fusion. Raw data should be directly used only to determine low-level abstraction. Further help in abstracting from low-level details can be provided by **data fusion** mechanisms. A good (re)construction of a meaningful information for the application reduces the complexity of the pervasive applications and helps the developers to concentrate on the application logic rather on the management of raw data. Moreover, the reactivity required in pervasive systems and the aggregation of large amounts of data (and its processing) are antagonists. We study **software services that can be deployed closer to the edge of the network**. The exploration of data fusion technics will be guided by different criteria: relevance of abstractions produced for pervasive applications, anonymization of exploited raw data, processing time, etc.

Assessing the correctness of the behavior. To ease the design of new applications and to align the development of new products with the ever faster standard developments, continuous integration could be used in parallel with continuous conformance and interoperability testing. We already participate in the design of new shared platforms that aims at facilitating this providing remote testing tools. Unfortunately, it is not possible to be sure that all potential peers in the surrounding have a conform behavior. Moreover, upon failure or security breach, a piece of equipment could stop to operate properly and lead to global mis-behavior. We want to propose conceptual tools for **testing at runtime devices in the environment**. The result of such conformance or interoperability tests could be stored safely in the environment by authoritative testing entity. Then application could interact with the device with a higher confidence. The confidence level of a device could be part of the quality attribute of the information it contributed to generate. The same set of tools could be used to identify misbehaving device for maintenance purpose or to trigger further testing.

3.2. Building relevant abstraction for new interactions

The pervasive applications are often designed in an ad hoc manner depending on the targeted application area. Ressources (sensors / actuators, connected objects etc.) are often used in silos which complexify the implementation of rich pervasive computing scenarios. In the second research axis, we want to get away from technical aspects identifying **common and reusable system mechanisms** that could be used in various applications.

Tagging the environment. Information relative to environment could be stored by the application itself, but it could be complex to manage for mobile application since it could cross a large number of places with various features. Moreover the developer has to build its own representation of information especially when he wants to share information with other instances of the same application or with other applications. A promising approach is to store and to maintain this information associated to an object or to a place, in the environment itself. The infrastructure should provide services to application developers: add/retrieve information in the environment, share information and control who can access it, add computed properties to object for further usage. We want to study an **extensible model to describe and augment the environment**. Beyond a simple distributed storage, we have in mind a new kind of interaction between pervasive applications and changing environment and between applications themselves.

Taking advantages of the spatial and temporal relationships. To understand the world they have to interact with, pervasive applications often have to (re)build a model of it from the exchange they have with others or from their own observations. A part of the programmer's task consists in building a model of the spatial layout of the objects in the surrounding. The term *layout* can be understood in several ways: the co-location of multiple objects in the same vicinity, the physical arrangement of two objects relative to each other, or even the crossing of an object of a physical area to another, etc. Determining remotely these spatial properties (see figure 1 -a) is difficult without exchanging a lot of information. Properties related to the spatial layout are far easier to characterize locally. They could be abstracted from interaction pattern without any complex virtual representation of the environment (see figure 1 -b). We want to be able to rely on this type of spatial layout in a pervasive environment. In the prior years, the members of TACOMA already worked on **models for processing object interactions** in the physical world to automatically trigger processing. This was the case in particular of the spatial programming principle: physical space is treated as a tuple-space in which objects are automatically synchronized according to their spatial arrangement. We want to follow this approach by considering **richer and more expressive programming models**.

3.3. Acting on the environment

The conceptual tools we aim to study must be *frugal*: they use as less as possible resources, while having the possibility to use much more when it is required. Data needed by an application are not made available for "free"; for example, it costs energy to measure a characteristic of the environment, or to transmit it. So this "design frugality" requires a **fine-grained control** on how data is actually collected from the environment. The third research axis aims at designing solutions that give this control to application developers by **acting on the environment**.

Acting on the data collection. We want to be able to identify which information are really needed during the perception elaboration process. If a piece of data is missing to build a given information with the appropriate quality level, the data collection mechanism should find relevant information in the environment or modify the way it aggregate it. These could lead to a modification of the behavior of the network layer and the path the piece of data use in the aggregation process.

Acting on object interactions. Object in the environment could adapt their behavior in a way that strongly depend on the object itself and that is difficult to generalize. Beyond the specific behaviors of actuators triggered through specialized or standard interfaces, the production of information required by an application could necessitate an adaptation at the object level (eg. calibration, sampling). The environment should then be able to initiate such adaption transparently to the application, which may not know all objects it passes by.

Adapting object behaviors. The radio communication layers become more flexible and able to adapt the way they use energy to what is really required for a given transmission. We already study how beamforming technics could be used to adapt multicast strategy for video services. We want to show how playing with these new parameters of transmissions (eg. beamforming, power, ...) allows to control spatial relationships objects could have. There is a tradeoff to find between the capacity of the medium, the electromagnetic pollution and the reactivity of the environment. We plan to expend our previous on interface selection and more generally on what we call **opportunistic networking**.

TAMIS Team

3. Research Program

3.1. Axis 1: Vulnerability analysis

This axis proposes different techniques to discover vulnerabilities in systems. The outcomes of this axis are (a) new techniques to discover system vulnerabilities as well as to analyze them, and (b) to understand the importance of the hardware support.

Most existing approaches used at the engineering level rely on testing and fuzzing. Such techniques consist in simulating the system for various input values, and then checking that the result conforms to a given standard. The problem being the large set of inputs to be potentially tested. Existing solutions propose to extract significant sets by mutating a finite set of inputs. Other solutions, especially concolic testing developed at Microsoft, propose to exploit symbolic executions to extract constraints on new values. We build on those existing work, and extend them with recent techniques based on dissimilarity distances and learning. We also account for the execution environment, and study techniques based on the combination of timing attacks with fuzzing techniques to discover and classify classes of behavior of the system under test.

Techniques such as model checking and static analysis have been used for verifying several types of requirements such as safety and reliability. Recently, several works have attempted to adapt model checking to the detection of security issues. It has clearly been identified that this required to work at the level of binary code. Applying formal techniques to such code requires the development of disassembly techniques to obtain a semantically well-defined model. One of the biggest issues faced with formal analysis is the state space explosion problem. This problem is amplified in our context as representations of data (such as stack content) definitively blow up the state space. We propose to use statistical model checking (SMC) of rare events to efficiently identify problematic behaviors.

We also seek to understand vulnerabilities at the architecture and hardware levels. Particularly, we evaluate vulnerabilities of the interfaces and how an adversary could use them to get access to core assets in the system. One particular mechanism to be investigated is the DMA and the so-called Trustzone. An ad-hoc technique to defend against adversarial DMA-access to memory is to keep key material exclusively in registers. This implies co-analyzing machine code and an accurate hardware model.

3.2. Axis 2: Malware analysis

Axis 1 is concerned with vulnerabilities. Such vulnerabilities can be exploited by an attacker in order to introduce malicious behaviors in a system. Another method to identify vulnerabilities is to analyze malware that exploits them. However, modern malware has a wide variety of analysis avoidance techniques. In particular, attackers obfuscate the code leading to a security exploit. For doing so, recent black hat research suggests hiding constants in program choices via polynomials. Such techniques hinder forensic analysis by making detailed analysis labor intensive and time consuming. The objective of research axis 2 is to obtain a full tool chain for malware analysis starting from (a) the observability of the malware via deobfuscation, and (b) the analysis of the resulting binary file. A complementary objective is to understand how hardware attacks can be exploited by malwares.

We first investigate obfuscation techniques. Several solutions exist to mitigate the packer problem. As an example, we try to reverse the packer and remove the environment evaluation in such a way that it performs the same actions and outputs the resulting binary for further analysis. There is a wide range of techniques to obfuscate malware, which includes flattening and virtualization. We will produce a taxonomy of both techniques and tools. We will first give a particular focus to control flow obfuscation via mixed Boolean algebra, which is highly deployed for malware obfuscation. We recently showed that a subset of them can be broken via SAT-solving and synthesis. Then, we will expand our research to other obfuscation techniques.

Once the malware code has been unpacked/deobfuscated, the resulting binary still needs to be fully understood. Advanced malware often contains multiple stages, multiple exploits and may unpack additional features based on its environment. Ensuring that one understands all interesting execution paths of a malware sample is related to enumerating all of the possible execution paths when checking a system for vulnerabilities. The main difference is that in one case we are interested in finding vulnerabilities and in the other in finding exploitative behavior that may mutate. Still, some of the techniques of Axis 1 can be helpful in analyzing malware. The main challenge for axis 2 is thus to adapt the tools and techniques to deal with binary programs as inputs, as well as the logic used to specify malware behavior, including behavior with potentially rare occurrences. Another challenge is to take mutation into account, which we plan to do by exploiting mining algorithms.

Most recent attacks against hardware are based on fault injection which dynamically modifies the semantics of the code. We demonstrated the possibility to obfuscate code using constraint solver in such a way that the code becomes intentionally hostile while hit by a laser beam. This new form of obfuscation opens a new challenge for secure devices where malicious programs can be designed and uploaded that defeat comprehensive static analysis tools or code reviews, due to their multi-semantic nature. We have shown on several products that such an attack cannot be mitigated with the current defenses embedded in Java cards. In this research, we first aim at extending the work on fault injection, then at developing new techniques to analyze such hostile code. This is done by proposing formal models of fault injection, and then reusing results from our work on obfuscation/deobfuscation.

3.3. Axis 3: Building a secure network stack

To evaluate the techniques developed in Axes 1 and 2, we analyze concrete systems developed not only with industry partners, but also within the team. By using our own systems, we can co-evolve best-practices, while externally developed systems provide realistic challenges especially with respect to analyzing obfuscated malware in the hardware or complex vulnerabilities. In this context, Christian Grothoff (ARP Inria) is currently developing a new Internet, which is supposed to be more secure. This introduces interesting challenges both in terms of vulnerability and malware analysis, and hence should be a great opportunity to mix the competences of all the members of the team.

More precisely, this system intends to challenge the idea that network security is an administrative task, where network administrators shield users with passwords, firewalls, intrusion detection systems and policies. Instead, we want to eliminate administrators that have power over user's data, and as such administrators themselves are liabilities, and because a network design that permits administrative intrusion inherently adds vulnerabilities. Instead, the system should ensure secure communication mechanisms without trusted third parties.

Key challenges we work on include (a) improving scalable secure ad-hoc decentralized routing, including key-value lookup, unicast and multicast communication, (b) protecting meta-data in the overlay using advanced decentralized onion routing, (c) a unified public-key infrastructure and identity management solution that is suitable to replace the Web-of-Trust, X.509, DNSSEC and other legacy methods for naming and identifying services, (d) secure synchronous and asynchronous messaging at scale, providing decentralized alternatives to common online social applications and addressing challenges in protocol evolution and compatibility. Finally, we are currently working on GNU Taler, a new secure privacy-preserving payment system where users never have to authenticate. This system in particular can be used as a concrete test case for the methods developed in the team.

To support this research work, we develop a framework named GNUnet. It provides a clear separation into layers, which facilitates testing and verifying the various components. However, we see that often existing formal verification techniques still do not scale to typical subsystems encountered in practice. Our objective is thus to exploit efficient and scalable formal techniques proposed in Axis 1 together with engineering skills in order to guide the validation (message synchronization, data protection, ...) and reach the best compromise. An additional complication is that we need a validation process that not merely covers the software itself, but also all of its dependencies (such as database, cryptographic libraries and networking libraries). For the Taler-specific hardware, we are envisioning an NFC-powered device, which creates new

challenges in terms of securing cryptographic computations in a setting where the adversary has control over the power supply. In such a case, the attacker can drive the environment and modify the behavior of the system as we have shown in Axis 2. Providing the control of the environment is a new vector for attackers.

Christian Grothoff, who leads this axis, got a position in Bern in 2017. This axis is expected to follow him in the future, although Tamis still holds expertise and members to finish ongoing work with the team. Cooperations with Bern are expected in the future.

TEA Project-Team

3. Research Program

3.1. Previous Works

The challenges of team TEA support the claim that sound Cyber-Physical System design (including embedded, reactive, and concurrent systems altogether) should consider multi-form time models as a central aspect. In this aim, architectural specifications found in software engineering are a natural focal point to start from. Architecture descriptions organize a system model into manageable components, establish clear interfaces between them, collect domain-specific constraints and properties to help correct integration of components during system design. The definition of a formal design methodology to support heterogeneous or multi-form models of time in architecture descriptions demands the elaboration of sound mathematical foundations and the development of formal calculi and methods to instrument them. This constitutes the research program of team TEA.

System design based on the “synchronous paradigm” has focused the attention of many academic and industrial actors on abstracting non-functional implementation details from system design. This elegant design abstraction focuses on the logic of interaction in reactive programs rather than their timed behavior, allowing to secure functional correctness while remaining an intuitive programming model for embedded systems. Yet, it corresponds to embedded technologies of single cores and synchronous buses from the 90s, and may hardly cover the semantic diversity of distribution, parallelism, heterogeneity, of cyber-physical systems found in 21st century Internet-connected, true-timeTM-synchronized clouds, of tomorrow’s grids.

By contrast with a synchronous hypothesis yet from the same era, the polychronous MoCC implemented in the data-flow specification language Signal, available in the Eclipse project POP⁰ and in the CCSL standard.⁰, are inherently capable of describing multi-clock abstractions of GALS systems. The POP and TimeSquare projects provide toolled infrastructures to refine high-level specifications into real-time streaming applications or locally synchronous and globally asynchronous systems, through a series of model analysis, verification, and synthesis services. These tool-supported refinement and transformation techniques can assist the system engineer from the earliest design stages of requirement specification to the latest stages of synthesis, scheduling and deployment. These characteristics make polychrony much closer to the required semantic for compositional, refinement-based, architecture-driven, system design.

While polychrony was a step ahead of the traditional synchronous hypothesis, CCSL is a leap forward from synchrony and polychrony. The essence of CCSL is “multi-form time” toward addressing all of the domain-specific physical, electronic and logical aspects of cyber-physical system design.

3.2. Modeling Times

To make a sense and eventually formalize the semantics of time in system design, we should most certainly rely on algebraic representations of time found in previous works and introduce the paradigm of “time systems” (type systems to represent time) in a way reminiscent to CCSL. Just as a type system abstracts data carried along operations in a program, a time system abstracts the causal interaction of that program module or hardware element with its environment, its pre and post conditions, its assumptions and guarantees, either logical or numerical, discrete or continuous. Some fundamental concepts of the time systems we envision are present in the clock calculi found in data-flow synchronous languages like Signal or Lustre, yet bound to a particular model of concurrency, hence time.

⁰ Polychrony on Polarsys, <https://www.polarsys.org/projects/polarsys.pop>

⁰ Clock Constraints in UML/MARTE CCSL. C. André, F. Mallet. RR-6540. Inria, 2008. <http://hal.inria.fr/inria-00280941>

In particular, the principle of refinement type systems⁰, is to associate information (data-types) inferred from programs and models with properties pertaining, for instance, to the algebraic domain on their value, or any algebraic property related to its computation: effect, memory usage, pre-post condition, value-range, cost, speed, time, temporal logic⁰. Being grounded on type and domain theories, a time system should naturally be equipped with program analysis techniques based on type inference (for data-type inference) or abstract interpretation (for program properties inference) to help establish formal relations between heterogeneous component “types”. Just as a time calculus may formally abstract timed concurrent behaviors of system components, timed relations (abstraction and refinement) represent interaction among components.

Scalability and compositionality requires the use of assume-guarantee reasoning to represent them, and to facilitate composition by behavioral sub-typing, in the spirit of the (static) contract-based formalism proposed by Passerone et al.⁰. Verification problems encompassing heterogeneously timed specifications are common and of great variety: checking correctness between abstract and concrete time models relates to desynchronisation (from synchrony to asynchrony) and scheduling analysis (from synchrony to hardware). More generally, they can be perceived from heterogeneous timing viewpoints (e.g. mapping a synchronous-time software on a real-time middle-ware or hardware).

This perspective demands capabilities not only to inject time models one into the other (by abstract interpretation, using refinement calculi), to compare time abstractions one another (using simulation, refinement, bi-simulation, equivalence relations) but also to prove more specific properties (synchronization, determinism, endochrony). All this formalization effort will allow to effectively perform the tooled validation of common cross-domain properties (e.g. cost v.s. power v.s. performance v.s. software mapping) and tackle equally common yet though case studies such as these linking battery capacity, to on-board CPU performance, to static software schedulability, to logical software correctness and plant controllability: the choice of the right sampling period across the system components.

3.3. Modeling Architectures

To address the formalization of such cross-domain case studies, modeling the architecture formally plays an essential role. An architectural model represents components in a distributed system as boxes with well-defined interfaces, connections between ports on component interfaces, and specifies component properties that can be used in analytical reasoning about the model. Several architectural modeling languages for embedded systems have emerged in recent years, including the SAE AADL⁰, SysML⁰, UML MARTE⁰.

In system design, an architectural specification serves several important purposes. First, it breaks down a system model into manageable components to establish clear interfaces between components. In this way, complexity becomes manageable by hiding details that are not relevant at a given level of abstraction. Clear, formally defined, component interfaces allow us to avoid integration problems at the implementation phase. Connections between components, which specify how components affect each other, help propagate the effects of a change in one component to the linked components.

Most importantly, an architectural model is a repository to share knowledge about the system being designed. This knowledge can be represented as requirements, design artifacts, component implementations, held together by a structural backbone. Such a repository enables automatic generation of analytical models for different aspects of the system, such as timing, reliability, security, performance, energy, etc. Since all the models are generated from the same source, the consistency of assumptions w.r.t. guarantees, of abstractions w.r.t. refinements, used for different analyses becomes easier, and can be properly ensured in a design methodology based on formal verification and synthesis methods.

⁰Abstract Refinement Types. N. Vazou, P. Rondon, and R. Jhala. European Symposium on Programming. Springer, 2013.

⁰LTL types FRP. A. Jeffrey. Programming Languages meets Program Verification.

⁰A contract-based formalism for the specification of heterogeneous systems. L. Benvenistu, et al. FDL, 2008

⁰Architecture Analysis and Design Language, AS-5506. SAE, 2004. <http://standards.sae.org/as5506b>

⁰System modeling Language. OMG, 2007. <http://www.omg.org/spec/SysML>

⁰UML Profile for MARTE. OMG, 2009. <http://www.omg.org/spec/MARTE>

Related works in this aim, and closer in spirit to our approach (to focus on modeling time) are domain-specific languages such as Prelude⁰ to model the real-time characteristics of embedded software architectures. Conversely, standard architecture description languages could be based on algebraic modeling tools, such as interface theories with the ECDAR tool⁰.

In project TEA, it takes form by the normalization of the AADL standard's formal semantics and the proposal of a time specification annex in the form of related standards, such as CCSL, to model concurrency time and physical properties, and PSL, to model timed traces.

3.4. Scheduling Theory

Based on sound formalization of time and CPS architectures, real-time scheduling theory provides tools for predicting the timing behavior of a CPS which consists of many interacting software and hardware components. Expressing parallelism among software components is a crucial aspect of the design process of a CPS. It allows for efficient partition and exploitation of available resources.

The literature about real-time scheduling⁰ provides very mature schedulability tests regarding many scheduling strategies, preemptive or non-preemptive scheduling, uniprocessor or multiprocessor scheduling, etc. Scheduling of data-flow graphs has also been extensively studied in the past decades.

A milestone in this prospect is the development of abstract affine scheduling techniques⁰. It consists, first, of approximating task communication patterns (e.g. between Safety-Critical Java threads) using cyclo-static data-flow graphs and affine functions. Then, it uses state of the art ILP techniques to find optimal schedules and to concretize them as real-time schedules in the program implementations⁰⁰.

Abstract scheduling, or the use of abstraction and refinement techniques in scheduling borrowed to the theory of abstract interpretation⁰ is a promising development toward tooling methodologies to orchestrate thousands of heterogeneous hardware/software blocks on modern CPS architectures (just consider modern cars or aircrafts). It is an issue that simply defies the state of the art and known bounds of complexity theory in the field, and consequently requires a particular address.

To develop the underlying theory of this promising research topic, we first need to deepen the theoretical foundation to establish links between scheduling analysis and abstract interpretation. A theory of time systems would offer the ideal framework to pursue this development. It amounts to representing scheduling constraints, inferred from programs, as types or contract properties. It allows to formalize the target time model of the scheduler (the architecture, its middle-ware, its real-time system) and defines the basic concepts to verify assumptions made in one with promises offered by the other: contract verification or, in this case, synthesis.

3.5. Virtual Prototyping

Virtual Prototyping is the technology of developing realistic simulators from models of a system under design; that is, an emulated device that captures most, if not all, of the required properties of the real system, based on its specifications. A virtual prototype should be run and tested like the real device. Ideally, the real application software would be run on the virtual prototyping platform and produce the same results as the real device with the same sequence of outputs and reported performance measurements. This may be true to some extent only. Some trade-offs have often to be made between the accuracy of the virtual prototype, and time to develop accurate models.

⁰The Prelude language. LIFL and ONERA, 2012. <http://www.lifl.fr/~forget/prelude.html>

⁰PyECDAR, timed games for timed specifications. Inria, 2013. <https://project.inria.fr/pyecdar>

⁰A survey of hard real-time scheduling for multiprocessor systems. R. I. Davis and A. Burns. *ACM Computing Survey* 43(4), 2011.

⁰Buffer minimization in EDF scheduling of data-flow graphs. A. Bouakaz and J.-P. Talpin. *LCTES*, ACM, 2013.

⁰ADFG for the synthesis of hard real-time applications. A. Bouakaz, J.-P. Talpin, J. Vitek. *ACSD*, IEEE, June 2012.

⁰Design of SCJ Level 1 Applications Using Affine Abstract Clocks. A. Bouakaz and J.-P. Talpin. *SCOPES*, ACM, 2013.

⁰La vérification de programmes par interprétation abstraite. P. Cousot. Séminaire au Collège de France, 2008.

In order to speed-up simulation time, the virtual prototype must trade-off with something. Depending upon the application designer's goals, one may be interested in trading some loss of accuracy in exchange for simulation speed, which leads to constructing simulation models that focus on some design aspects and provide abstraction of others. A simulation model can provide an abstraction of the simulated hardware in three directions:

- *Computation abstraction.* A hardware component computes a high level function by carrying out a series of small steps executed by composing logical gates. In a virtual prototyping environment, it is often possible to compute the high level function directly by using the available computing resources on the simulation host machine, thus abstracting the hardware function.
- *Communication abstraction.* Hardware components communicate together using some wiring, and some protocol to transmit the data. Simulation of the communication and the particular protocol may be irrelevant for the purpose of virtual prototyping: communication can be abstracted into higher level data transmission functions.
- *Timing Abstraction.* In a cycle accurate simulator, there are multiple simulation tasks, and each task makes some progress on each clock cycle, but this slows down the simulation. In a virtual prototyping experiment, one may not need such precise timing information: coarser time abstractions can be defined allowing for faster simulation.

The cornerstone of a virtual prototyping platform is the component that simulates the processor(s) of the platform, and its associated peripherals. Such simulation can be *static* or *dynamic*.

A solution usually adopted to handle time in virtual prototyping is to manage hierarchical time scales, use component abstractions where possible to gain performance, use refinement to gain accuracy where needed. Localized time abstraction may not only yield faster simulation, but facilitate also verification and synthesis (e.g. synchronous abstractions of physically distributed systems). Such an approach requires computations and communications to be harmoniously discretized and abstracted from originally heterogeneous viewpoints onto a structuring, articulating, pivot model, for concerted reasoning about time and scheduling of events in a way that ensures global system specification correctness.

In the short term these component models could be based on libraries of predefined models of different levels of abstractions. Such abstractions are common in large programming workbench for hardware modeling, such as SystemC, but less so, because of the engineering required, for virtual prototyping platforms.

The approach of team TEA provides an additional ingredient in the form of rich component interfaces. It therefore dictates to further investigate the combined use of conventional virtual prototyping libraries, defined as executable abstractions of real hardware, with executable component simulators synthesised from rich interface specifications (using, e.g., conventional compiling techniques used for synchronous programs).

VISAGES Project-Team

3. Research Program

3.1. Research Program

The scientific foundations of our team concern the development of new processing algorithms in the field of medical image computing : image fusion (registration and visualization), image segmentation and analysis, management of image related information. Since this is a very large domain, which can be applied on numerous types of application; for seek of efficiency, the purpose of our methodological work primarily focuses on clinical aspects and for the most part on head and neck related diseases. In addition, we emphasize our research efforts on the neuroimaging domain. Concerning the scientific foundations, we have pushed our research efforts:

- In the field of image fusion and image registration (rigid and deformable transformations) with a special emphasis on new challenging registration issues, especially when statistical approaches based on joint histogram cannot be used or when the registration stage has to cope with loss or appearance of material (like in surgery or in tumor imaging for instance).
- In the field of image analysis and statistical modeling with a new focus on image feature and group analysis problems. A special attention was also to develop advanced frameworks for the construction of atlases and for automatic and supervised labeling of brain structures.
- In the field of image segmentation and structure recognition, with a special emphasis on the difficult problems of *i*) image restoration for new imaging sequences (new Magnetic Resonance Imaging protocols, 3D ultrasound sequences...), and *ii*) structure segmentation and labelling based on shape, multimodal and statistical information.
- Following past national projects where we had leading roles (e.g., Neurobase, NeuroLog, . . .), we wanted to enhance the development of distributed and heterogeneous medical image processing systems.

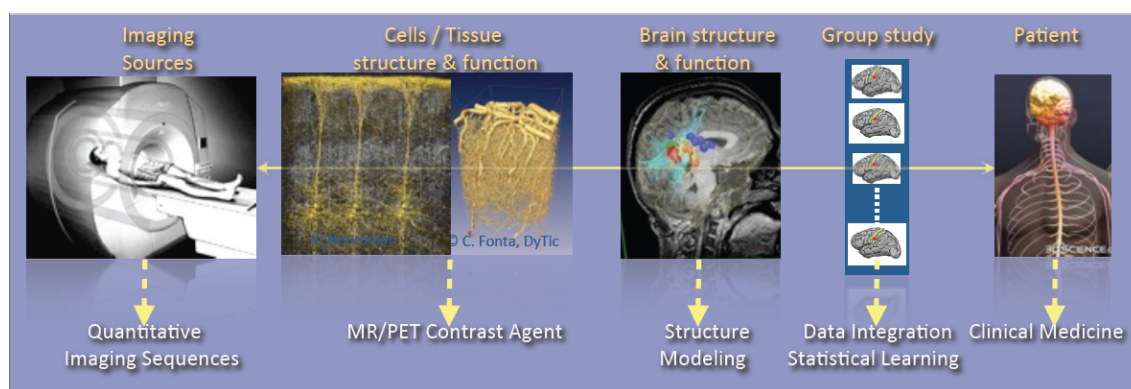


Figure 1. The major overall scientific foundation of the team concerns the integration of data from the Imaging source to the patient at different scales: from the cellular or molecular level describing the structure and function, to the functional and structural level of brain structures and regions, to the population level for the modelling of group patterns and the learning of group or individual imaging markers.

As shown in Fig. 1, research activities of the VISAGES U1228 team are tightly coupling observations and models through integration of clinical and multi-scale data, phenotypes (cellular, molecular or structural patterns). We work on personalized models of central nervous system organs and pathologies, and intend to confront these models to clinical investigation studies for quantitative diagnosis, prevention of diseases, therapy planning and validation. These approaches are developed in a translational framework where the data integration process to build the models inherits from specific clinical studies, and where the models are assessed on prospective clinical trials for diagnosis and therapy planning. All of this research activity is conducted in tight links with the **Neurinfo** imaging platform environments and the engineering staff of the platform. In this context, some of our major challenges in this domain concern:

- The elaboration of new descriptors to study the brain structure and function (e.g., variation of brain perfusion with and without contrast agent, evolution in shape and size of an anatomical structure in relation with normal, pathological or functional patterns, computation of asymmetries from shapes and volumes).
- The integration of additional spatio-temporal imaging sequences covering a larger range of observation, from the molecular level to the organ through the cell (Arterial Spin Labeling, diffusion MRI, MR relaxometry, MR cell labeling imaging, PET molecular imaging, ...). This includes the elaboration of new image descriptors coming from spatio-temporal quantitative or contrast-enhanced MRI.
- The creation of computational models through data fusion of molecular, cellular, structural and functional image descriptors from group studies of normal and/or pathological subjects.
- The evaluation of these models on acute pathologies especially for the study of degenerative, psychiatric or developmental brain diseases (e.g., Multiple Sclerosis, Epilepsy, Parkinson, Dementia, Strokes, Depression, Schizophrenia, ...) in a translational framework.

In terms of methodological developments, we are particularly working on statistical methods for multidimensional image analysis, and feature selection and discovery, which include:

- The development of specific shape and appearance models, construction of atlases better adapted to a patient or a group of patients in order to better characterize the pathology;
- The development of advanced segmentation and modeling methods dealing with longitudinal and multidimensional data (vector or tensor fields), especially with the integration of new prior models to control the integration of multiscale data and aggregation of models;
- The development of new models and probabilistic methods to create water diffusion maps from MRI;
- The integration of machine learning procedures for classification and labeling of multidimensional features (from scalar to tensor fields and/or geometric features): pattern and rule inference and knowledge extraction are key techniques to help in the elaboration of knowledge in the complex domains we address;
- The development of new dimensionality reduction techniques for problems with massive data, which includes dictionary learning for sparse model discovery. Efficient techniques have still to be developed to properly extract from a raw mass of images derived data that are easier to analyze.