IN PARTNERSHIP WITH:
CNRS

Université Denis Diderot (Paris 7)

Activity Report 2015

Project-Team GANG

Networks, Graphs and Algorithms

IN COLLABORATION WITH: Laboratoire d'Informatique Algorithmique Fondamentale et Appliquée (LIAFA)

RESEARCH CENTER
Paris - Rocquencourt

THEME
Networks and Telecommunications

Table of contents

Project-Team GANG

Creation of the Project-Team: 2007 July 01

Keywords:

Computer Science and Digital Science:

1.2. - Networks
1.2.3. - Routing
1.2.9. - Social Networks
1.3. - Distributed Systems
6.1.3. - Discrete Modeling (multi-agent, people centered)
7.1. - Parallel and distributed algorithms
7.10. - Network science
7.13. - Quantum algorithms
7.2. - Discrete mathematics, combinatorics
7.3. - Operations research, optimization, game theory
7.9. - Graph theory

Other Research Topics and Application Domains:

1.1.1. - Structural biology
1.1.11. - Systems biology
1.1.6. - Genomics
6.3.2. - Network protocols
6.3.4. - Social Networks
7.2.2. - Smart road

1. Members

Research Scientists

Laurent Viennot [Team leader, Inria, Senior Researcher, HdR]
Pierre Fraigniaud [CNRS, Senior Researcher, HdR]
Amos Korman [CNRS, Researcher, HdR]
Adrian Kosowski [Inria, Researcher, HdR]

Faculty Members

Yacine Boufkhad [Univ. Paris VII, Associate Professor]
Pierre Charbit [Univ. Paris VII, Associate Professor]
Fabien de Montgolfier [Univ. Paris VII, Associate Professor]
Hugues Fauconnier [Univ. Paris VII, Professor]
Carole Gallet Delporte [Univ. Paris VII, Professor]
Michel Habib [Univ. Paris VII, Professor]

Engineer

François Durand [Inria, until Aug 2015]

PhD Students

The-Dang Huynh [Bell Labs (Alcatel), until Jul 2015]
Leonardo Linguaglossa [Inria, financed by Alcatel-Lucent Bell Labs]
Simon Collet [CNRS, financed by ERC]

Lucas Boczkowski [CNRS, financed by ERC]
Laurent Feuilloley [ENS Cachan]

Post-Doctoral Fellow

Paolo Penna [Univ. Paris VII, Post-doctoral Researcher, until Sep 2015]

Administrative Assistant

Christine Anocq [Inria]

Others

Ricardo de La Paz Guala [Inria, internship student, from Feb 2015 until May 2015]
Nishant Rai [Inria, internship student, from May 2015 until Jul 2015]
Parth Shah [Inria, internship student, from May 2015 until Jul 2015]
Marc Heinrich [ENS Paris, internship student, from March 2015 until June 2015]

2. Overall Objectives

2.1. Overall Objectives

GANG focuses on algorithm design for large scale networks using structural properties of these networks. Application domains include the development of optimized protocols for large dynamic networks such as mobile networks or overlay networks over Internet. This includes for instance peer-to-peer applications, or the navigability of social networks. GANG tools come from recent advances in the field of graph algorithms, both in centralized and distributed settings. In particular, this includes graph decomposition and geometric properties (such as low doubling dimension, low dimension embedding, etc.). Today, the management of large networks, Internet being the reference, is best effort. However, the demand for mobility (ad hoc networks, wireless connectivity, etc.) and for dynamicity (node churn, fault tolerance, etc.) is increasing. In this distributed setting, it becomes necessary to design a new generation of algorithms and protocols to face the challenge of large scale mobility and dynamicity. In the mean time, recent and sophisticated theoretical results have emerged, offering interesting new tracks for managing large networks. These results concern centralized and decentralized algorithms for solving key problems in communication networks, including routing, but also information retrieval, localization, or load balancing. They are mainly based on structural properties observed in most of real networks: approximate topology with low dimension metric spaces, low treewidth, low doubling dimension, graph minor freeness, etc. In addition, graph decomposition techniques have recently progressed. The scientific community has now tools for optimizing network management. First striking results include designing overlay networks for peer-to-peer systems and understanding the navigability of large social networks.

3. Research Program

3.1. Graph and Combinatorial Algorithms

We focus on two approaches for designing algorithms for large graphs: decomposing the graph and relying on simple graph traversals.

3.1.1. Graph Decompositions

We study new decompositions schemes such as 2-join, skew partitions and others partition problems. These graph decompositions appeared in the structural graph theory and are the basis of some well-known theorems such as the Perfect Graph Theorem. For these decompositions there is a lack of efficient algorithms. We aim at designing algorithms working in $O(nm)$ since we think that this could be a lower bound for these decompositions.

3.1.2. Graph Search

We more deeply study multi-sweep graph searches. In this domain a graph search only yields a total ordering of the vertices which can be used by the subsequent graph searches. This technique can be used on huge graphs and do not need extra memory. We already have obtained preliminary results in this direction and many well-known graph algorithms can be put in this framework. The idea behind this approach is that each sweep discovers some structure of the graph. At the end of the process either we have found the underlying structure (for example an interval representation for an interval graph) or an approximation of it (for example in hard discrete optimization problems). We envision applications to exact computations of centers in huge graphs, to underlied combinatorial optimization problems, but also to networks arising in Biology.

3.1.3. Graph Exploration

In the course of graph exploration, a mobile agent is expected to regularly visit all the nodes of an unknown network, trying to discover all its nodes as quickly as possible. Our research focuses on the design and analysis of agent-based algorithms for exploration-type problems, which operate efficiently in a dynamic network environment, and satisfy imposed constraints on local computational resources, performance, and resilience. Our recent contributions in this area concern the design of fast deterministic algorithms for teams of agents operating in parallel in a graph, with limited or no persistent state information available at nodes. We plan further studies to better understand the impact of memory constraints and of the availability of true randomness on efficiency of the graph exploration process.

3.2. Distributed Computing

The distributed community can be viewed as the union of two sub-communities. This is true even in our team. Even though they are not completely disjoint, they are disjoint enough not to leverage each other's results. At a high level, one is mostly interested in timing issues (clock drifts, link delays, crashes, etc.) while the other one is mostly interested in spatial issues (network structure, memory requirements, etc.). Indeed, one sub-community is mostly focusing on the combined impact of asynchronism and faults on distributed computation, while the other addresses the impact of network structural properties on distributed computation. Both communities address various forms of computational complexities, through the analysis of different concepts. This includes, e.g., failure detectors and wait-free hierarchy for the former community, and compact labeling schemes and computing with advice for the latter community. We have the ambitious project to achieve the reconciliation between the two communities by focusing on the same class of problems, the yes/no-problems, and establishing the scientific foundations for building up a consistent theory of computability and complexity for distributed computing. The main question addressed is therefore: is the absence of globally coherent computational complexity theories covering more than fragments of distributed computing, inherent to the field? One issue is obviously the types of problems located at the core of distributed computing. Tasks like consensus, leader election, and broadcasting are of very different nature. They are not *yes-no* problems, neither are they minimization problems. Coloring and Minimal Spanning Tree are optimization problems but we are often more interested in constructing an optimal solution than in verifying the correctness of a given solution. Still, it makes full sense to analyze the *yes-no* problems corresponding to checking the validity of the output of tasks. Another issue is the power of individual computation. The FLP impossibility result as well as Linial's lower bound hold independently from the individual computational power of the involved computing entities. For instance, the individual power of solving NP-hard problems in constant time would not help overcoming these limits which are inherent to the fact that computation is distributed. A third issue is the abundance of models for distributed computing frameworks, from shared memory to message passing, spanning all kinds of specific network structures (complete graphs, unit-disk graphs, etc.) and or timing constraints (from complete synchronism to full asynchronism). There are however models, typically the wait-free model and the LOCAL model, which, though they do not claim to reflect accurately real distributed computing systems, enable focusing on some core issues. Our research program is ongoing to carry many important notions of Distributed Computing into a *standard* computational complexity.

3.3. Network Algorithms and Analysis

Based on our scientific foundation on both graph algorithms and distributed algorithms, we plan to analyze the behavior of various networks such as future Internet, social networks, overlay networks resulting from distributed applications or online social networks.

3.3.1. Information Dissemination

One of the key aspects of networks resides in the dissemination of information among the nodes. We aim at analyzing various procedures of information propagation from dedicated algorithms to simple distributed schemes such as flooding. We also consider various models, where noise can alter information as it propagates or where memory of nodes is limited for example.

3.3.2. Routing Paradigms

We try to explore new routing paradigms such as greedy routing in social networks for example. We are also interested in content centric networking where routing is based on content name rather than content address. One of our target is multiple path routing: how to design forwarding tables providing multiple disjoint paths to a destination?

3.3.3. Beyond Peer-to-Peer

Based on our past experience of peer-to-peer application design, we would like to broaden the spectrum of distributed applications where new efficient algorithms and analysis can be performed. We especially target online social networks if we see them as collaborative tools for exchanging information. A basic question resides in making the right connections for gathering filtered and accurate information with sufficient coverage.

3.3.4. SAT and Forwarding Information Verification

As forwarding tables of networks grow and are sometimes manually modified, the problem of verifying forwarding information becomes critical and has recently gained in interest. Some problems that arise in network verification such as loop detection for example, may be naturally encoded as Boolean Satisfiability problems. Beside the theoretical interest of this encoding in complexity proofs, it has also a practical value for solving these problems by taking advantage of the many efficient Satisfiability testing solvers. Indeed, SAT solvers have proved to be very efficient in solving problems coming from various areas (Circuit Verification, Dependency and Conflicts in Software distributions...) and encoded in Conjunctive Normal Form. To test an approach using SAT solvers in network verification, one need to collect data sets from real network and to develop good models for generating realistic networks. The technique of encoding and the solvers themselves need to be adapted to this kind of problems. All this represent a rich experimental field of future research.

3.3.5. Network Analysis

Finally, we are interested in analyzing the structural properties of practical networks. This can include diameter computation or ranking of nodes. As we mostly consider large networks, we are often interested in efficient heuristics. Ideally, we target heuristics that give exact answer although fast computation time is not guaranteed for all networks. We already have designed such heuristics for diameter computation; understanding the structural properties that enable fast computation time in practice is still an open question.

4. Application Domains

4.1. Application Domains

Application domains include evaluating Internet performances, the design of new peer-to-peer applications, enabling large scale ad hoc networks and mapping the web.

- The application of measuring and modeling Internet metrics such as latencies and bandwidth is to provide tools for optimizing Internet applications. This concerns especially large scale applications such as web site mirroring and peer-to-peer applications.

- Peer-to-peer protocols are based on a all equal paradigm that allows to design highly reliable and scalable applications. Besides the file sharing application, peer-to-peer solutions could take over in web content dissemination resistant to high demand bursts or in mobility management. Envisioned peer-to-peer applications include video on demand, streaming, exchange of classified ads,...
- Wifi networks have entered our every day life. However, enabling them at large scale is still a challenge. Algorithmic breakthrough in large ad hoc networks would allow to use them in fast and economic deployment of new radio communication systems.
- The main application of the web graph structure consists in ranking pages. Enabling site level indexing and ranking is a possible application of such studies.

5. Highlights of the Year

5.1. Highlights of the Year

Roads

COMPUTATION OF ROAD NETWORK DIAMETER

Based on the algorithms presented in [5], Laurent Viennot has computed the diameter and radius of the worldwide road network. The diameter of a graph is the distance between two points that are furthest apart one from another. The interesting distance notion in a road network is often travel time. Finding the worldwide road network diameter thus amounts to find two points such that the travel time from one to another is maximal. Once such a pair of points is identified, we can compute the shortest path between them to obtain somehow the longest road trip in the world. Computing the diameter of a general graph usually requires to compute all pairwise distances, which is impractical for such a big graph. However, the team has developed heuristics that appear to work fast on many practical graphs including road networks. Thanks to OpenStreetMap data, the team has thus been able to compute the world road diameter (and the diameter of various restricted parts of the network). The results can be visualized on https://who.rocq.inria.fr/Laurent.Viennot/road/.

Erc

NEW ERC CONSOLIDATOR GRANT

Amos Korman has received an ERC Consolidator Grant, entitled “Distributed Biological Algorithms (DBA)”, which started in May 2015. The goal of this interdisciplinary project is to demonstrate the usefulness of an algorithmic perspective in studies of complex biological systems. It focuses on the aspect of collective behavior, demonstrating the benefits of applying distributed computing techniques to establish algorithmic insights into the behavior of biological ensembles.

Highpapers

WORK ON DISTRIBUTED COMPUTING

The team has published a number of papers on Distributed Computing theory at high-profile venues. A subjective selection of these results includes: an almost-tight bound on the space complexity of set agreement [29], a study of the power of randomization in proof-labeling schemes [22] (both published at PODC’15), and a characterization of convergence in an important class of population protocols [28] (published at ICALP’15 track A).

6. New Software and Platforms

6.1. Svvamp: Simulator of Various Voting Algorithms in Manipulating Populations

Svvamp is a Python package dedicated to the study of voting systems with an emphasis on manipulation analysis. Svvamp can generate datasets based on a large library of artificial models, or use any kind of real dataset as input. It currently implements more than 20 voting systems. Using state of the art algorithms, it can analyze multiple variants of tactical voting (e.g. absence of weak/strong Nash equilibrium). Svvamp is free software, under the GNU General Public License version 3. Its documentation includes installation procedure, tutorials, reference guide and instructions for new contributors.

Svvamp represents about ten thousands lines of code, and according to the Python Software Foundation, 2568 downloads have been reported in the last month (as of December, 10th, 2015). It is available at https://svvamp.readthedocs.org.

Svvamp [39] will be demonstrated in The Thirtieth Conference on Artificial Intelligence.

6.1.1. Svvamp self-assesment

A3: ambitious software, usable by people inside and outside the team but without a clear and strong dissemination and support action plan. So3up4: original software reusing known ideas and introducing a few new ideas / original software implementing a fair number of original ideas.

SM3: well-developed software, fairly extensive documentation, reasonable software engineering and testing, attention to usability, dissemination, bug fixes, and user feedback;

EM2: basic maintenance to keep the software alive;

SDL4: public source or binary distribution on the web, organized by the development team;

François Durand is the main contributor (4) in: a) design and architecture (DA) b) coding and debugging (CD) c) maintenance and support (MS) d) team/project management (TPM)

6.2. Big Graph Tools

The team is starting a software development activity around big graph manipulation. A preliminary library offering diameter and skeleton computation is available at https://who.rocq.inria.fr/Laurent.Viennot/dev/big-graph-tools/. This library was used to compute the diameters of the worldwide road network (200M edges) and the Twitter follower-followee graph (23G edges).

7. New Results

7.1. Graph and Combinatorial Algorithms

7.1.1. Rainbow matchings in hypergraphs

A rainbow matching for (not necessarily distinct) sets $F_1, ..., F_k$ of hypergraph edges is a matching consisting of k edges, one from each F_i. In [8], we give some order to the multitude of conjectures that relate to this concept, as well as introduce some new conjectures. We also present some partial results on one of these conjectures, that seems central among them – the so-called Ryser-Brualdi-Stein conjecture.

7.1.2. A graph formulation of the union-closed sets conjecture

In 1979, Frankl conjectured that in a finite non-trivial union-closed collection of sets there has to be an element that belongs to at least half the sets. In [7], we show that this is equivalent to the conjecture that in a finite non-trivial graph there are two adjacent vertices, each belonging to at most half of the maximal stable sets. In this graph formulation other special cases become natural. The conjecture is trivially true for non-bipartite graphs and we show that it also holds for the classes of chordal bipartite graphs, subcubic bipartite graphs, bipartite series-parallel graphs and bipartitioned circular interval graphs.

7.1.3. Cops-and-robber games on k-chordal graphs

The cops-and-robber games, introduced by Winkler and Nowakowski (in Discrete Math. 43, 1983) and independently defined by Quilliot (in J. Comb. Theory, Ser. B 38, 1985), concern a team of cops that must capture a robber moving in a graph. In [20], we consider the class of k-chordal graphs, i.e., graphs with no induced (chordless) cycle of length greater than k, $k \geq 3$. We prove that $k-1$ cops are always sufficient to capture a robber in k-chordal graphs. This leads us to our main result, a new structural decomposition for a graph class including k-chordal graphs.

We present a polynomial-time algorithm that, given a graph G and $k \geq 3$, either returns an induced cycle larger than k in G, or computes a tree-decomposition of G, each bag of which contains a dominating path with at most $k-1$ vertices. This allows us to prove that any k-chordal graph with maximum degree Δ has treewidth at most $(k-1)(\Delta-1)+2$, improving the $O(\Delta(\Delta-1)k-3)$ bound of Bodlaender and Thilikos (Discrete Appl. Math. 79, 1997). Moreover, any graph admitting such a tree-decomposition has small hyperbolicity). As an application, for any n-vertex graph admitting such a tree-decomposition, we propose a compact routing scheme using routing tables, addresses and headers of size $O(k \log \Delta + \log n)$ bits and achieving an additive stretch of $O(k \ \ \log \Delta)$. As far as we know, this is the first routing scheme with $O(k \log \Delta + \log n)$-routing tables and small additive stretch for k-chordal graphs.

7.1.4. Distinguishing views in symmetric networks

The view of a node in a port-labeled network is an infinite tree encoding all walks in the network originating from this node. In [16], we prove that for any integers $n \geq D \geq 1$, there exists a port-labeled network with at most n nodes and diameter at most D, which contains a pair of nodes whose (infinite) views are different, but whose views truncated to depth $\Omega(D \log (n/D))$ are identical.

7.1.5. Vertex elimination orderings for hereditary graph classes

In [3], we provide a general method to prove the existence and compute efficiently elimination orderings in graphs. This method relies on several tools that were known before, but that were not put together so far: the algorithm LexBFS due to Rose, Tarjan and Lueker, its additional properties discovered by Berry and Bordat, and a local decomposition property of graphs discovered by Maffray, Trotignon and Vušković. We use this method to prove the existence of elimination orderings in several classes of graphs, and to compute them in linear time. Some of the classes have already been studied, namely even-hole-free graphs, square-theta-free Berge graphs, universally signable graphs and wheel-free graphs. Some other classes are new. It turns out that all the classes that we consider can be defined by excluding some of the so-called Truemper configurations. For several classes of graphs, we obtain directly bounds on the chromatic number, or fast algorithms for the maximum clique problem or the coloring problem.

7.1.6. Fast collaborative graph exploration

In [14], we study the following scenario of online graph exploration. A team of k agents is initially located at a distinguished vertex r of an undirected graph. We ask how many time steps are required to complete exploration, i.e., to make sure that every vertex has been visited by some agent. As our main result, we provide the first strategy which performs exploration of a graph with n vertices at a distance of at most D from r in time $O(D)$, using a team of agents of polynomial size $k = Dn^{1+\epsilon} < n^{2+\epsilon}$, for any $\epsilon > 0$. Our strategy works in the local communication model, in which agents can only exchange information when located at a vertex, without knowledge of global parameters such as n or D.

We also obtain almost-tight bounds on the asymptotic relation between exploration time and team size, for large k, in both the local and the global communication model.

7.1.7. Position discovery for a system of bouncing robots

In [11], we consider a scenario in which a collection of n anonymous mobile robots is deployed on a unit-perimeter ring or a unit-length line segment. Every robot starts moving at constant speed, and bounces each time it meets any other robot or segment endpoint, changing its walk direction. We study the problem of position discovery, in which the task of each robot is to detect the presence and the initial positions of all other robots. The robots cannot communicate or perceive information about the environment in any way other than by bouncing nor they have control over their walks which are determined by their initial positions and their starting directions. Each robot has a clock allowing it to observe the times of its bounces. We give complete characterizations of all initial configurations for both the ring and the segment in which no position detection algorithm exists and we design optimal position detection algorithms for all feasible configurations.

7.1.8. Rendezvous of mobile agents in edge-weighted networks

In [15], we introduce a variant of the deterministic rendezvous problem for a pair of heterogeneous agents operating in an undirected graph, which differ in the time they require to traverse particular edges of the graph. Each agent knows the complete topology of the graph and the initial positions of both agents. The agent also knows its own traversal times for all of the edges of the graph, but is unaware of the corresponding traversal times for the other agent. The goal of the agents is to meet on an edge or a node of the graph. In this scenario, we study the time required by the agents to meet, compared to the meeting time T_{OPT} in the offline scenario in which the agents have complete knowledge about each others' speed characteristics. When no additional assumptions are made, we show that rendezvous in our model can be achieved after time $O(nT_{OPT})$ in a n-node graph, and that such time is essentially in some cases the best possible. However, we prove that the rendezvous time can be reduced to $\Theta(T_{OPT})$ when the agents are allowed to exchange $\Theta(n)$ bits of information at the start of the rendezvous process. We then show that under some natural assumption about the traversal times of edges, the hardness of the heterogeneous rendezvous problem can be substantially decreased, both in terms of time required for rendezvous without communication, and the communication complexity of achieving rendezvous in time $\Theta(T_{OPT})$.

7.1.9. Monitoring a graph using faulty mobile robots

In the scenario studied in [27], a team of k mobile robots is deployed on a weighted graph whose edge weights represent distances. The robots perpetually move along the domain, represented by all points belonging to the graph edges, not exceeding their maximal speed. The robots need to patrol the graph by regularly visiting all points of the domain. Here, we consider a team of robots (patrolmen), at most f of which may be unreliable, i.e. they fail to comply with their patrolling duties.

What algorithm should be followed so as to minimize the maximum time between successive visits of every edge point by a reliable patrolmen? The corresponding measure of efficiency of patrolling called idleness has been widely accepted in the robotics literature. We extend it to the case of untrusted patrolmen; we denote by $I_k^f(G)$ the maximum time that a point of the domain may remain unvisited by reliable patrolmen. The objective is to find patrolling strategies minimizing $I_k^f(G)$.

We investigate this problem for various classes of graphs. We design optimal algorithms for line segments, which turn out to be surprisingly different from strategies for related patrolling problems proposed in the literature. We then use these results to study the case of general graphs. For Eulerian graphs G, we give an optimal patrolling strategy with idleness $I_k^f(G) = (f+1)|E|/k$, where $|E|$ is the sum of the lengths of the edges of G. Further, we show the hardness of the problem of computing the idle time for three robots, at most one of which is faulty, by reduction from 3-edge-coloring of cubic graphs — a known NP-hard problem. A byproduct of our proof is the investigation of classes of graphs minimizing idle time (with respect to the total length of edges); an example of such a class is known in the literature under the name of Kotzig graphs.

7.1.10. Limit behavior of the rotor-router system

The rotor-router model, also called the Propp machine, was introduced as a deterministic alternative to the random walk. In this model, a group of identical tokens are initially placed at nodes of the graph. Each node maintains a cyclic ordering of the outgoing arcs, and during consecutive turns the tokens are propagated along arcs chosen according to this ordering in round-robin fashion. The behavior of the model is fully deterministic. Yanovski et al. (Algorithmica, 2003) proved that a single rotor-router walk on any graph with m edges and diameter D stabilizes to a traversal of an Eulerian circuit on the set of all $2m$ directed arcs on the edge set of the graph, and that such periodic behaviour of the system is achieved after an initial transient phase of at most $2mD$ steps.

The case of multiple parallel rotor-routers was studied experimentally, leading Yanovski et al. to the experimental observation that a system of $k > 1$ parallel walks also stabilizes with a period of length at most $2m$ steps. In our work [26] we disprove this observation, showing that the period of parallel rotor-router walks can in fact, be superpolynomial in the size of graph. On the positive side, we provide a characterization of the periodic behavior of parallel router walks, in terms of a structural property of stable states called a subcycle decomposition. This property provides us the tools to efficiently detect whether a given system configuration corresponds to the transient or to the limit behavior of the system. Moreover, we provide polynomial upper bounds of $O(m^4D^2 + mD \log k)$ and $O(m^5k^2)$ on the number of steps it takes for the system to stabilize. Thus, we are able to predict any future behavior of the system using an algorithm that takes polynomial time and space. In addition, we show that there exists a separation between the stabilization time of the single-walk and multiple-walk rotor-router systems, and that for some graphs the latter can be asymptotically larger even for the case of $k = 2$ walks.

7.2. Distributed Computing

7.2.1. Self-stabilizing verification and computation of an MST

In the work [19], we demonstrate the usefulness of distributed local verification of proofs, as a tool for the design of self-stabilizing algorithms. In particular, it introduces a somewhat generalized notion of distributed local proofs, and utilizes it for improving the time complexity significantly, while maintaining space optimality. As a result, we show that optimizing the memory size carries at most a small cost in terms of time, in the context of Minimum Spanning Tree (MST). That is, we present algorithms that are both time and space efficient for both constructing an MST and for verifying it. This involves several parts that may be considered contributions in themselves.

First, we generalize the notion of local proofs, trading off the time complexity for memory efficiency. This adds a dimension to the study of distributed local proofs, which has been gaining attention recently. Specifically, we design a (self-stabilizing) proof labeling scheme which is memory optimal (i.e., $O(\log n)$ bits per node), and whose time complexity is $O(\log^2 n)$ in synchronous networks, or $O(\Delta \log^3 n)$ time in asynchronous ones, where Δ is the maximum degree of nodes. This answers an open problem posed by Awerbuch and Varghese (FOCS 1991). We also show that $\Omega(\log n)$ time is necessary, even in synchronous networks. Another property is that if f faults occurred, then, within the required detection time above, they are detected by some node in the O(f log n) locality of each of the faults. Second, we show how to enhance a known transformer that makes input/output algorithms self-stabilizing. It now takes as input an efficient construction algorithm and an efficient self-stabilizing proof labeling scheme, and produces an efficient self-stabilizing algorithm. When used for MST, the transformer produces a memory optimal self-stabilizing algorithm, whose time complexity, namely, $O(n)$, is significantly better even than that of previous algorithms. (The time complexity of previous MST algorithms that used $O(\log^2 n)$ memory bits per node was $O(n^2)$, and the time for optimal space algorithms was $O(n|E|)$.) Inherited from our proof labelling scheme, our self-stabilising MST construction algorithm also has the following two properties: (1) if faults occur after the construction ended, then they are detected by some nodes within $O(\log^2 n)$ time in synchronous networks, or within $O(\Delta \log^3 n)$ time in asynchronous ones, and (2) if f faults occurred, then, within the required detection time above, they are detected within the $O(f \log n)$ locality of each of the faults. We also show how to improve the above two properties, at the expense of some increase in the memory.

7.2.2. Clock synchronization and distributed estimation in highly dynamic networks

In [21], we consider the External Clock Synchronization problem in dynamic sensor networks. Initially, sensors obtain inaccurate estimations of an external time reference and subsequently collaborate in order to synchronize their internal clocks with the external time. For simplicity, we adopt the drift-free assumption, where internal clocks are assumed to tick at the same pace. Hence, the problem is reduced to an estimation problem, in which the sensors need to estimate the initial external time. In this context of distributed estimation, this work is further relevant to the problem of collective approximation of environmental values by biological groups.

Unlike most works on clock synchronization that assume static networks, the setting considered here is an extreme case of highly dynamic networks. We do however impose a restriction on the dynamicity of the network. Specifically, we assume a non-adaptive scheduler adversary that dictates an arbitrary, yet independent, meeting pattern. Such meeting patterns fit, for example, with short-time scenarios in highly dynamic settings, where each sensor interacts with only few other arbitrary sensors.

We propose an extremely simple clock synchronization (or an estimation) algorithm that is based on weighted averages, and prove that its performance on any given independent meeting pattern is highly competitive with that of the best possible algorithm, which operates without any resource or computational restrictions, and further knows the whole meeting pattern in advance. In particular, when all distributions involved are Gaussian, the performances of our scheme coincide with the optimal performances. Our proofs rely on an extensive use of the concept of Fisher information. We use the Cramér-Rao bound and our definition of a Fisher Channel Capacity to quantify information flows and to obtain lower bounds on collective performance. This opens the door for further rigorous quantifications of information flows within collaborative sensors.

7.2.3. Wait-freedom with advice

In [13], we motivate and propose a new way of thinking about failure detectors which allows us to define what it means to solve a distributed task wait-free using a failure detector. In our model, the system is composed of computation processes that obtain inputs and are supposed to produce outputs and synchronization processes that are subject to failures and can query a failure detector. Under the condition that correct (never failing) synchronization processes take sufficiently many steps, they provide the computation processes with enough advice to solve the given task wait-free: every computation process outputs in a finite number of its own steps, regardless of the behavior of other computation processes. Every task can thus be characterized by the weakest failure detector that allows for solving it, and we show that every such failure detector captures a form of set agreement. We then obtain a complete classification of tasks, including ones that evaded comprehensible characterization so far, such as renaming or weak symmetry breaking.

7.2.4. Linear-space bootstrap communication schemes

In [12], we consider a system of n processes with ids that are drawn from a large space. How can these n processes communicate to solve a problem? It is shown that linear number of Multi-Writer Multi-Reader (MWMR) registers are sufficient to solve any read-write wait-free solvable problem and needed to solve some read-write wait-free solvable problem. This contrasts with the existing possible solution borrowed from adaptive algorithms that require $\Theta(n^{3/2})$ MWMR registers.

To obtain the sufficiency result, we show how the processes can non-blockingly emulate a system of n Single-Writer Multi-Reader (SWMR) registers on top of n Multi-Writer Multi-Reader (MWMR) registers. For the necessity result, we show it is impossible to do such an emulation with $n-1$ MWMR registers.

We also presents a wait-free emulation, using $2n-1$ rather than just n registers. The emulation can be used to solve an infinite sequence of tasks that are sequentially dependent (processes need the previous task's outputs in order to proceed to the next task). A non-blocking emulation cannot be used in this case, because it might starve a process forever.

7.2.5. Space complexity of set agreement

The k-set agreement problem is a generalization of the classical consensus problem in which processes are permitted to output up to k different input values. In a system of n processes, an m-obstruction-free solution to the problem requires termination only in executions where the number of processes taking steps is eventually bounded by m. This family of progress conditions generalizes wait-freedom ($m = n$) and obstruction-freedom ($m = 1$). In [29], we prove upper and lower bounds on the number of registers required to solve m-obstruction-free k-set agreement, considering both one-shot and repeated formulations. In particular, we show that repeated k set agreement can be solved using $n + 2m - k$ registers and establish a nearly matching lower bound of $n + m - k$.

7.2.6. Consensus capability of distributed systems

A fundamental research theme in distributed computing is the comparison of systems in terms of their ability to solve basic problems such as consensus that cannot be solved in completely asynchronous systems. In particular, in a seminal work (ACM Trans. Program. Lang. Syst. 13, 1991), Herlihy compares shared-memory systems in terms of the shared objects that they have: he proved that there are shared objects that are powerful enough to solve consensus for n processes, but are too weak to solve consensus for $n + 1$ processes; such objects are placed at level n of a wait-free hierarchy.

Similarly as in that work, in [30] we compare shared-memory systems with respect to their ability to solve consensus for n processes. But instead of comparing systems defined by the shared objects that they have, we compare read-write systems defined by the set of process schedules that can occur in these systems. Defining systems this way can capture many types of systems, e.g., systems whose synchrony ranges from fully synchronous to completely asynchronous, several systems with failure detectors, and "obstruction-free" systems. Here, we consider read-write systems defined in terms of sets of process schedules, and investigate the following fundamental question: Is there a system of $n + 1$ processes such that consensus can be solved for every subset of n processes in the system, but consensus cannot be solved for the $n + 1$ processes of the system? We show that the answer to the above question is "yes", and so these systems can be classified into a hierarchy akin to Herlihy's hierarchy.

7.2.7. Shared whiteboard models of distributed systems

In [4], we study distributed algorithms on massive graphs where links represent a particular relationship between nodes (for instance, nodes may represent phone numbers and links may indicate telephone calls). Since such graphs are massive they need to be processed in a distributed way. When computing graph-theoretic properties, nodes become natural units for distributed computation. Links do not necessarily represent communication channels between the computing units and therefore do not restrict the communication flow. Our goal is to model and analyze the computational power of such distributed systems where one computing unit is assigned to each node. Communication takes place on a whiteboard where each node is allowed to write at most one message. Every node can read the contents of the whiteboard and, when activated, can write one small message based on its local knowledge. When the protocol terminates its output is computed from the final contents of the whiteboard. We describe four synchronization models for accessing the whiteboard. We show that message size and synchronization power constitute two orthogonal hierarchies for these systems. We exhibit problems that separate these models, i.e., that can be solved in one model but not in a weaker one, even with increased message size. These problems are related to maximal independent set and connectivity. We also exhibit problems that require a given message size independently of the synchronization model.

7.2.8. Discrete Lotka-Volterra population protocols

In [28], we focus on a natural class of population protocols whose dynamics are modeled by the discrete version of Lotka-Volterra equations with no linear term. In such protocols, when an agent a of type (species) i interacts with an agent b of type (species) j with a as the initiator, then b's type becomes i with probability P_{ij}. In such an interaction, we think of a as the predator, b as the prey, and the type of the prey is either converted to that of the predator or stays as is. Such protocols capture the dynamics of some opinion spreading models and generalize the well-known Rock-Paper-Scissors discrete dynamics. We consider the pairwise interactions among agents that are scheduled uniformly at random.

We start by considering the convergence time and show that any Lotka-Volterra-type protocol on a n-agent population converges to some absorbing state in time polynomial in n, w.h.p., when any pair of agents is allowed to interact. By contrast, when the interaction graph is a star, there exist protocols of the considered type, such as Rock-Paper-Scissors, which require exponential time to converge. We then study threshold effects exhibited by Lotka-Volterra-type protocols with 3 and more species under interactions between any pair of agents. We present a simple 4-type protocol in which the probability difference of reaching the two possible absorbing states is strongly amplified by the ratio of the initial populations of the two other types, which are transient, but "control" convergence. We then prove that the Rock-Paper-Scissors protocol reaches each of its three possible absorbing states with almost equal probability, starting from any configuration satisfying some sub-linear lower bound on the initial size of each species. That is, Rock-Paper-Scissors is a realization of a “coin-flip consensus” in a distributed system. Some of our techniques may be of independent value.

7.2.9. Deterministic load-balancing

In [23], we consider the problem of deterministic load balancing of tokens in the discrete model. A set of n processors is connected into a d-regular undirected network. In every time step, each processor exchanges some of its tokens with each of its neighbors in the network. The goal is to minimize the discrepancy between the number of tokens on the most-loaded and the least-loaded processor as quickly as possible. Rabani et al. (FOCS 1998) present a general technique for the analysis of a wide class of discrete load balancing algorithms. Their approach is to characterize the deviation between the actual loads of a discrete balancing algorithm with the distribution generated by a related Markov chain. The Markov chain can also be regarded as the underlying model of a continuous diffusion algorithm. Rabani et al. showed that after time $T = O(\log(Kn)/\mu)$, any algorithm of their class achieves a discrepancy of $O(d \log n/\mu)$, where μ is the spectral gap of the transition matrix of the graph, and K is the initial load discrepancy in the system.

In this work we identify some natural additional conditions on deterministic balancing algorithms, resulting in a class of algorithms reaching a smaller discrepancy. This class contains well-known algorithms, e.g., the rotor-router. Specifically, we introduce the notion of cumulatively fair load-balancing algorithms where in any interval of consecutive time steps, the total number of tokens sent out over an edge by a node is the same (up to constants) for all adjacent edges. We prove that algorithms which are cumulatively fair and where every node retains a sufficient part of its load in each step, achieve a discrepancy of $O(d\sqrt{\log n/\mu}, d\sqrt{n})$ in time $O(T)$. We also show that in general neither of these assumptions may be omitted without increasing discrepancy. We then show by a combinatorial potential reduction argument that any cumulatively fair scheme satisfying some additional assumptions achieves a discrepancy of $O(d)$ almost as quickly as the continuous diffusion process. This positive result applies to some of the simplest and most natural discrete load balancing schemes.

7.2.10. Randomized local network computing

In [32], we have carried on investigating the line of research questioning the power of randomization for the design of distributed algorithms. In their seminal paper, Naor and Stockmeyer [STOC 1993] established that, in the context of network computing, in which all nodes execute the same algorithm in parallel, any construction task that can be solved locally by a randomized Monte-Carlo algorithm can also be solved locally by a deterministic algorithm. This result however holds in a specific context. In particular, it holds only for distributed tasks whose solutions that can be locally checked by a deterministic algorithm. We have extended the result of Naor and Stockmeyer to a wider class of tasks. Specifically, we proved that the same derandomization result holds for every task whose solutions can be locally checked using a 2-sided error randomized Monte-Carlo algorithm. This extension finds applications to, e.g., the design of lower bounds for construction tasks which tolerate that some nodes compute incorrect values. In a nutshell, we have showed that randomization does not help for solving such resilient tasks.

7.2.11. Proof-labeling schemes: randomization and self-stabilization

We have also carried on investigating the power of randomization for the design of proof-labeling schemes. Recall that a proof-labeling scheme, introduced by Korman, Kutten and Peleg [PODC 2005], is a mechanism enabling to certify the legality of a network configuration with respect to a boolean predicate. Such a mechanism finds applications in many frameworks, including the design of fault-tolerant distributed algorithms. In a proof-labeling scheme, the verification phase consists of exchanging labels between neighbors. The size of these labels depends on the network predicate to be checked. There are predicates requiring large labels, of poly-logarithmic size (e.g., MST), or even polynomial size (e.g., Symmetry). In [22], we introduce the notion of randomized proof-labeling schemes. By reduction from deterministic schemes, we show that randomization enables the amount of communication to be exponentially reduced. As a consequence, we show that checking any network predicate can be done with probability of correctness as close to one as desired by exchanging just a logarithmic number of bits between neighbors. Moreover, we design a novel space lower bound technique that applies to both deterministic and randomized proof-labeling schemes. Using this technique, we establish several tight bounds on the verification complexity of classical distributed computing problems, such as MST construction, and of classical predicates such as acyclicity, connectivity, and cycle length.

Next, we have established the formal connections between self-stabilization and proof-labeling scheme. Recall that self-stabilizing algorithms are distributed algorithms supporting transient failures. Starting from any configuration, they allow the system to detect whether the actual configuration is legal, and, if not, they allow the system to eventually reach a legal configuration. In the context of network computing, it is known that, for every task, there is a self-stabilizing algorithm solving that task, with optimal space-complexity, but converging in an exponential number of rounds. On the other hand, it is also known that, for every task, there is a self-stabilizing algorithm solving that task in a linear number of rounds, but with large space-complexity. It is however not known whether for every task there exists a self-stabilizing algorithm that is simultaneously space-efficient and time-efficient. In [24], we make a first attempt for answering the question of whether such an efficient algorithm exists for every task, by focussing on constrained spanning tree construction tasks. We present a general roadmap for the design of silent space-optimal self-stabilizing algorithms solving such tasks, converging in polynomially many rounds under the unfair scheduler. By applying our roadmap to the task of constructing minimum-weight spanning tree (MST), and to the task of constructing minimum-degree spanning tree (MDST), we provide algorithms that outperform previously known algorithms designed and optimized specifically for solving each of these two tasks.

7.2.12. Role of node identifiers in local decision

We have also investigated the role of IDs in network computing. This role is well understood as far as symmetry breaking is concerned. However, the unique identifiers also leak information about the computing environment — in particular, they provide some nodes with information related to the size of the network. It was recently proved that in the context of local decision, there are some decision problems such that (1) they cannot be solved without unique identifiers, and (2) unique node identifiers leak a sufficient amount of information such that the problem becomes solvable. In [33] we study what is the minimal amount of information that we need to leak from the environment to the nodes in order to solve local decision problems. Our key results are related to scalar oracles f that, for any given n, provide a multi-set $f(n)$ of n labels; then the adversary assigns the labels to the n nodes in the network. This is a direct generalization of the usual assumption of unique node identifiers. We give a complete characterization of the weakest oracle that leaks at least as much information as the unique identifiers. Our main result is the following dichotomy: we classify scalar oracles as large and small, depending on their asymptotic behavior, and show that (1) any large oracle is at least as powerful as the unique identifiers in the context of local decision problems, while (2) for any small oracle there are local decision problems that still benefit from unique identifiers.

7.2.13. Geometry on the utility space

In [31], we study the geometrical properties of the utility space (the space of expected utilities over a finite set of options), which is commonly used to model the preferences of an agent in a situation of uncertainty. We focus on the case where the model is neutral with respect to the available options, i.e. treats them, a priori, as being symmetrical from one another. Specifically, we prove that the only Riemannian metric that respects the geometrical properties and the natural symmetries of the utility space is the round metric. This canonical metric allows to define a uniform probability over the utility space and to naturally generalize the Impartial Culture to a model with expected utilities.

7.3. Network Algorithms and Analysis

7.3.1. Information dissemination on social networks

In [17], we model an online social network as a network formation game. We study convergence of selfish dynamics and show that somewhat natural metric assumption enable fast convergence towards an equilibrium with efficient collaborative filtering of content.

7.3.2. Verification of network forwarding tables

In [25], we investigate the problem of verifying forwarding network tables. We show that it is sufficient to test few representative headers when the set of rules applied by routers is complete under intersection.

7.3.3. Refreshing old datasets in a network: LiveRank

In [18], we consider the problem of refreshing a dataset. More precisely, given a collection of nodes gathered at some time (Web pages, users from an online social network) along with some structure (hyperlinks, social relationships), we want to identify a significant fraction of the nodes that still exist at present time. The liveness of an old node can be tested through an online query at present time. We call LiveRank a ranking of the old pages so that active nodes are more likely to appear first. The quality of a LiveRank is measured by the number of queries necessary to identify a given fraction of the active nodes when using the LiveRank order. We study different scenarios from a static setting where the LiveRank is computed before any query is made, to dynamic settings where the LiveRank can be updated as queries are processed. Our results show that building on the PageRank can lead to efficient LiveRanks, for Web graphs as well as for online social networks.

7.3.4. Exploiting user movement for position detection

The major issue of indoor localization system is the trade-off between implementation cost and accuracy. A low-cost system which demands only few hardware devices could save the cost but often it turns out to be less reliable. Aiming at improving classical triangulation method that requires several reference points, we propose in [34] a new method, called Two-Step Movement (2SM), which requires only one reference point (RP) by exploiting useful information given by the position change of a mobile terminal (MT), or the user movement. This method can minimize the number of reference points required in a localization system or navigation service and reduce system implementation cost. Analytical result shows that the user position can be thus derived and given in simple closed-form expression. Finally, simulation is conducted to demonstrate its effectiveness under noisy environment.

Then, in [35], we build on 2SM. We first improve the positioning performance through multi-sampling technique to combat measurement noise. Secondly, we propose the Generalized Two-Step Movement (G2SM) method for device-to-device (D2D) systems in which both the mobile terminal (MT) and RP can be mobile device. The mobile user's position can be derived analytically and given in simple closed-form expression. Its effectiveness in the presence of noise is shown in simulation results.

7.3.5. Fast diameter and radius computation in real-world graphs

In [5], we propose a new algorithm that computes the radius and the diameter of a weakly connected digraph $G = (V, E)$, by finding bounds through heuristics and improving them until they are validated. Although the worst-case running time is $O(|V||E|)$, we will experimentally show that it performs much better in the case of real-world networks, finding the radius and diameter values after 10–100 runs of Breadth First Search instead of $|V|$ BFS-s (independently of the value of $|V|$), and thus having running time $O(|E|)$ in practice. As far as we know, this is the first algorithm able to compute the diameter of weakly connected digraphs, apart from the naive algorithm, which runs in time $\Omega(|V||E|)$ performing a BFS from each node. In the particular cases of strongly connected directed or connected undirected graphs, we have compared our algorithm with known approaches by performing experiments on a dataset composed by several real-world networks of different kinds. These experiments show that, despite its generality, the new algorithm outperforms all previous methods, both in the radius and in the diameter computation, both in the directed and in the undirected case, both in average running time and in robustness. Finally, as an application example, we have used the new algorithm to determine the solvability over time of the "Six Degrees of Kevin Bacon" game, and of the "Six Degrees of Wikipedia" game. As a consequence, we have computed for the first time the exact value of the radius and the diameter of the whole Wikipedia digraph.

8. Bilateral Contracts and Grants with Industry

8.1. Collaboration with Bell Labs

Gang has a strong collaboration with Bell Labs (Alcatel-Lucent / Nokia). We notably collaborate with Fabien Mathieu and Diego Perino who are former members of GANG that have joined Alcatel-Lucent. A Cifre grant allowed to fund the PhD thesis of The-Dang Huynh to study ranking techniques and their application to social networks. An ADR (joint research action) is dedicated to content centric networks and forwarding information verification. The PhD thesis of Leonardo Linguaglossa is funded by this contract. We also collaborate with Ludovic Noirie on voting systems.

This collaboration is developed inside the Alcatel-Lucent and Inria joint research lab.

9. Partnerships and Cooperations

9.1. National Initiatives

9.1.1. ANR Displexity

Participants: Carole Gallet Delporte, Hugues Fauconnier, Pierre Fraigniaud, Amos Korman, Adrian Kosowski, Laurent Viennot.

Managed by University Paris Diderot, C. Delporte and H. Fauconnier lead this project that grants 1 Ph. D.

Distributed computation keep raising new questions concerning computability and complexity. For instance, as far as fault-tolerant distributed computing is concerned, impossibility results do not depend on the computational power of the processes, demonstrating a form of undecidability which is significantly different from the one encountered in sequential computing. In the same way, as far as network computing is concerned, the impossibility of solving certain tasks locally does not depend on the computational power of the individual processes.

The main goal of DISPLEXITY (for DIStributed computing: computability and ComPLEXITY) is to establish the scientific foundations for building up a consistent theory of computability and complexity for distributed computing.

One difficulty to be faced by DISPLEXITY is to reconcile the different sub-communities corresponding to a variety of classes of distributed computing models. The current distributed computing community may indeed be viewed as two not necessarily disjoint sub-communities, one focusing on the impact of temporal issues, while the other focusing on the impact of spatial issues. The different working frameworks tackled by these two communities induce different objectives: computability is the main concern of the former, while complexity is the main concern of the latter.

Within DISPLEXITY, the reconciliation between the two communities will be achieved by focusing on the same class of problems, those for which the distributed outputs are interpreted as a single binary output: yes or no. Those are known as the yes/no-problems. The strength of DISPLEXITY is to gather specialists of the two main streams of distributed computing. Hence, DISPLEXITY will take advantage of the experience gained over the last decade by both communities concerning the challenges to be faced when building up a complexity theory encompassing more than a fragment of the field.

In order to reach its objectives, DISPLEXITY aims at achieving the following tasks:

- Formalizing yes/no-problems (decision problems) in the context of distributed computing. Such problems are expected to play an analogous role in the field of distributed computing as that played by decision problems in the context of sequential computing.
- Formalizing decision problems (yes/no-problems) in the context of distributed computing. Such problems are expected to play an analogous role in the field of distributed computing as that played by decision problems in the context of sequential computing.

- Revisiting the various explicit (e.g., failure-detectors) or implicit (e.g., a priori information) notions of oracles used in the context of distributed computing allowing us to express them in terms of decidability/complexity classes based on oracles.
- Identifying the impact of non-determinism on complexity in distributed computing. In particular, DISPLEXITY aims at a better understanding of the apparent lack of impact of non-determinism in the context of fault-tolerant computing, to be contrasted with the apparent huge impact of non-determinism in the context of network computing. Also, it is foreseen that non-determinism will enable the comparison of complexity classes defined in the context of fault-tolerance with complexity classes defined in the context of network computing.
- Last but not least, DISPLEXITY will focus on new computational paradigms and frameworks, including, but not limited to distributed quantum computing and algorithmic game theory (e.g., network formation games).

The project will have to face and solve a number of challenging problems. Hence, we have built the DISPLEXITY consortium so as to coordinate the efforts of those worldwide leaders in Distributed Computing who are working in our country. A successful execution of the project will result in a tremendous increase in the current knowledge and understanding of decentralized computing and place us in a unique position in the field.

The project has been extended until June 2016.

9.1.2. Laboratory of Information, Networking and Communication Sciences (LINCS)

Participants: François Durand, The-Dang Huynh, Leonardo Linguaglossa, Laurent Viennot.

Gang is participating to the LINCS, a research centre co-founded by Inria, Institut Mines-Télécom, UPMC and Alcatel-Lucent Bell Labs, dedicated to research and innovation in the domains of future information and communication networks, systems and services. Gang contributes to work on online social networks, content centric networking and forwarding information verification.

9.2. European Initiatives

9.2.1. FP7 & H2020 Projects

Amos Korman has received an ERC Consolidator Grant entitled “Distributed Biological Algorithms (DBA)”, started in May 2015. This project proposes a new application for computational reasoning. More specifically, the purpose of this interdisciplinary project is to demonstrate the usefulness of an algorithmic perspective in studies of complex biological systems. We focus on the domain of collective behavior, and demonstrate the benefits of using techniques from the field of theoretical distributed computing in order to establish algorithmic insights regarding the behavior of biological ensembles. The project includes three related tasks, for which we have already obtained promising preliminary results. Each task contains a purely theoretical algorithmic component as well as one which integrates theoretical algorithmic studies with experiments. Most experiments are strategically designed by the PI based on computational insights, and are physically conducted by experimental biologists that have been carefully chosen by the PI. In turn, experimental outcomes will be theoretically analyzed via an algorithmic perspective. By this integration, we aim at deciphering how a biological individual (such as an ant) “thinks”, without having direct access to the neurological process within its brain, and how such limited individuals assemble into ensembles that appear to be far greater than the sum of their parts. The ultimate vision behind this project is to enable the formation of a new scientific field, called algorithmic biology, that bases biological studies on theoretical algorithmic insights.

9.3. International Initiatives

9.3.1. Inria International Partners

9.3.1.1. Informal International Partners

Ofer Feinerman (Physics department of complex systems, Weizmann Institute of Science, Rehovot, Israel), is a team member in Amos Korman’s ERC project DBA.

Rachid Guerraoui (School of Computer and Communication Sciences, EPFL, Switzerland) maintains an active research collaboration with Gang team members (Carole Delporte, Hugues Fauconnier).

Pierluigi Crescenzi (University of Florence, Italy) is a frequent visitor to the team and maintains an active research collaboration with Gang team members (Pierre Fraigniaud).

Sergio Rajsbaum (UNAM, Mexico) is a regular collaborator of the team, also involved formally in a joint French-Mexican research project (see next subsection).

Boaz Patt-Shamir (Tel Aviv University, Israel) is a regular collaborator of the team, also involved formally in a joint French-Israeli research project (see next subsection).

9.3.2. Participation In other International Programs

Involvement in the bilateral Franco-Israeli project MAIMONIDE (2014-2015) on "Resource Allocation in the Cloud". Pierre Fraigniaud was the project's co-coordinator for the French side. Financed by Partenariats Hubert Curien.

Involvement in the bilateral Franco-Mexican project ECOS NORD (2013-2016) on "Distributed Verification". Pierre Fraigniaud was the project's co-coordinator for the French side. Partners: LIAFA and LaBRI (France), UNAM (Mexico).

9.4. International Research Visitors

9.4.1. Visits of International Scientists

Zvi Lotker (Ben Gurion University, Israel) is a long-term visitor of the team. He has also been awarded the Junior Chair of Fondation Sciences Mathématiques de Paris (FSMP) for 2015/2016.

Andrea Pietracaprina (Univ. Padova, Italy), 1 month's visit, Fall 2015.

Geppino Pucci (Univ. Padova, Italy), 1 month's visit, Fall 2015.

Eli Gafni (UCLA, USA), 2 months' visit, June-July 2015.

Sam Toueg (Univ. Toronto, Canada), 1 month's visit, January 2015.

Flavia Bonomo (Universidad de Buenos Aires, Argentina), 1 month's visit, June 2015.

9.4.1.1. Internships

Rai Nishant
: Date: May 2015 - Jul 2015
: Institution: IITK (India)

Shah Parth
: Date: May 2015 - Jul 2015
: Institution: Indian Institute of Technology Bombay (India)

Ricardo De La Paz Guala
: Date: Feb 2015 - May 2015
: Institution: Universidad de Concepción (Chile)

Marc Heinrich
: Date: Mar 2015 - Jun 2015
: Institution: ENS Paris

Simon Collet
: Date: Mar 2015 - Jun 2015
: Institution: Paris VII

9.4.2. Visits to International Teams

9.4.2.1. Research stays abroad

Amos Korman made several monthly visits to Israel, collaborating with Weizmann Institute of Science and Tel-Aviv University.

10. Dissemination

10.1. Promoting Scientific Activities

10.1.1. Scientific events selection

10.1.1.1. Chair of conference program committees

ICDCN 2015 – The 16th ACM International Conference on Distributed Computing and Networks, India, January 2015 (Amos Korman was PC co-chair for distributed computing track).

FCT 2015 – 20th International Symposium on Fundamentals of Computation Theory, Poland, August 2015 (Adrian Kosowski was PC co-chair).

NETYS 2015 – The International Conference on Networked Systems, Marocco, May 2015 (Hugues Fauconnier was PC co-chair).

10.1.1.2. Member of the conference program committees

ICALP 2015 – 42nd Int. Colloquium on Automata, Languages and Programming, Japan, July 2015 (Pierre Fraigniaud – PC member of track C).

PODC 2015 – 34th ACM Symposium on Principles of Distributed Computing, Spain, July 2015 (Amos Korman – PC member).

SPAA 2015 – 27th ACM Symposium on Parallelism in Algorithms and Architectures, USA, June 2015 (Pierre Fraigniaud – PC member).

WWW 2015 – 24th World Wide Web Conference, Italy, May 2015 (Pierre Fraigniaud – PC Member of Social Networks and Graph Analysis track).

DISC 2015 – 29th Symposium on Distributed Computing, Japan, October 2015 (Amos Korman and Pierre Fraigniaud – PC members).

IPDPS 2015 – 29th IEEE International Parallel and Distributed Processing Symposium, India, May 2015 (Pierre Fraigniaud – PC member).

MFCS 2015 – 40th Int. Symposium on Mathematical Foundations of Computer Science, Italy, August 2015 (Pierre Fraigniaud – PC member).

CIAC 2015 – 8th Int. Conference on Algorithms and Complexity, France, May 2015, (Pierre Fraigniaud – PC member).

EUROPAR 2015 – 21st European Conference on Parallel Computing, Austria, August 2015 (Pierre Fraigniaud – PC member).

SIROCCO 2015 – 22th Colloquium on Structural Information and Communication Complexity, Spain, July 2015 (Adrian Kosowski and Pierre Fraigniaud – PC members).

SSS 2015 – International Symposium on Stabilization, Safety, and Security of Distributed Systems, Canada, August 2015 (Adrian Kosowski – PC member).

NETYS 2015 – The International Conference on Networked Systems, Marocco, May 2015 (Carole Delporte – PC member).

Algotel 2015 – 17eme Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications, France, June 2015 (Carole Delporte – PC member).

10.1.2. Journal

10.1.2.1. Members of editorial boards

Pierre Fraigniaud is member of the editorial boards of Distributed Computing (DC), Theory of Computing Systems (TOCS), Fundamenta Informaticae (FI) and Journal of Interconnection Networks (JOIN).

10.1.3. Invited talks

SIROCCO 2015 – 22th Colloquium on Structural Information and Communication Complexity, Spain, July 2015 (Amos Korman – invited speaker).

ADHOC-NOW 2015 – 14th International Conference on Ad-Hoc Networks and Wireless, special track on Distributed Computing with Mobile Agents, Greece, June/July 2015 (Pierre Fraigniaud – invited speaker).

AAAC 2015 – 8th meeting of Asian Association for Algorithms and Computation, Hiroshima, Japan, May 2015 (Pierre Fraigniaud – invited speaker).

ACBD 2015 – 2nd European Meeting on Algorithmic Challenges of Big Data, Germany, September 2015 (Pierre Fraigniaud – invited speaker).

MARAMI 2015 – Modèles et Analyses Réseau : Approches Mathématiques et Informatiques, 6ème édition, France, September 2015 (Michel Habib – invited speaker).

HaifaGraphWorkshop 2015 – 15th Haifa Workshop on Interdisciplinary Applications of Graph Theory, Combinatorics, and Algorithms, Israel, June 2015 (Michel Habib – invited speaker).

10.1.4. Leadership within the scientific community

Pierre Fraigniaud is Director of LIAFA (Laboratoire d'Informatique Algorithmique : Fondements et Applications, UMR 7089, CNRS et université Paris Diderot), since 2010.

Pierre Fraigniaud is Director of Federation IFP (Fédération d'Informatique Fondamentale de Paris-Diderot, FR 3634, CNRS et université Paris Diderot), since 2014.

Pierre Fraigniaud is Steering Committe member of the key conferences in distributed computing theory: PODC (ACM Symposium on Principles of Distributed Computing, 2010–2013 and 2015–2018) and SPAA (ACM Symposium on Parallelism in Algorithms and Architectures, since 2002).

10.1.5. Research administration

Pierre Fraigniaud has been an elected member of the EATCS (European Association for Theoretical Computer Science) since October 2013.

Pierre Fraigniaud has been a jury member of the ERC for Starting Grants (in 2013 and 2015; shadow committee member in 2014 and 2016).

Michel Habib has been a member of the NSERC evaluation committee for Computer Science (February 2015, Ottawa, Canada).

Michel Habib has been chair of the AERES evaluation committee for the LaBRI (Bordeaux, January 2015).

Pierre Fraigniaud was a member of the jury for the 2015 Edsger W. Dijkstra Prize in Distributed Computing (previously committee member in 2010 and president in 2011).

Adrian Kosowski has been acting in the Polish Academy of Sciences as elected member of Young Scientists Academy (2012–2016) and member of the Committee for Computer Science (2015–2018).

10.2. Teaching - Supervision - Juries

10.2.1. Teaching

Master: Carole Delporte and Hugues Fauconnier, Algorithmique distribuée avec mémoire partagée, 6h, M2, Université Paris Diderot

Master: Hugues Fauconnier, Cours programmation répartie, 33h, M2, Univ. Paris Diderot

Master: Carole Delporte, Cours et TP Protocoles des services internet, 44h, M2, Univ. Paris Diderot

Master: Carole Delporte, Cours Algorithme réparti, 33h, M2, Univ. Paris Diderot

Master: Carole Delporte and Hugues Fauconnier, Protocoles Réseaux, 72h, M1, Université Paris Diderot

Licence: Carole Delporte and Hugues Fauconnier, Sécurité informatique, 36h, L3, Univ. Paris Diderot

Licence: Hugues Fauconnier, Programmation objet et interfaces graphiques, 48h, L2-L3, EIDD

Licence: Boufkhad Yacine, Algorithmique et Informatique, 132h, L1, IUT de l'Université Paris Diderot

Licence: Boufkhad Yacine, Programmation Orientée Objet, 60h, L2, IUT de l'Université Paris Diderot

Master: Pierre Fraigniaud, Algorithmique avancée, 24h, Ecole Centrale

Master: Pierre Fraigniaud, Algorithmique parallèle et distribuée, 24h, Ecole Centrale

Master: Adrian Kosowski, Communication and Routing, 4h, M1, ENSEIRB-MATMECA

Master: Adrian Kosowski, Randomization in Computer Science: Games, Networks, Epidemic and Evolutionary Algorithms, 24h, M1, École Polytechnique

Master: Pierre Fraigniaud and Adrian Kosowski, Algorithmique distribuée pour les réseaux, 24h, M2, Master Parisien de Recherche en Informatique (MPRI)

Master: Fabien de Montgolfier and Michel Habib, Grand Réseaux d'Interaction, 44h, M2, Univ Paris Diderot

Licence: Fabien de Montgolfier, Introduction à la programmation, 26h, L1, Univ Paris Diderot

Licence: Fabien de Montgolfier, Programmation avancée (bio-informatique), 26h, L3, Univ. Paris Diderot

Master: Fabien de Montgolfier, Algorithmique avancée (bio-informatique), 26h, M1, Univ Paris Diderot

Licence: Fabien de Montgolfier, Systèmes et Réseaux, 52h, L3, Ecole d'Ingénieurs Denis Diderot

Licence: Michel Habib, Algorithmique, 45h, L, ENS Cachan

Master: Michel Habib, Algorithmique avancée, 24h, M1, Univ. Paris Diderot

Master: Michel Habib, Mobilité, 33h, M2, Univ. Paris Diderot

Master: Michel Habib, Méthodes et algorithmes pour l'accès à l'information numérique, 16h, M2, Univ. Paris Diderot

Master: Michel Habib, Algorithmique de graphes, 12h, M2, Univ. Paris Diderot

Licence: Pierre Charbit, Introduction a la Programmation, 30h, L1, Université Paris Diderot, France

Licence: Pierre Charbit, Automates finis, 52h, L2, Université Paris Diderot, France

Licence: Pierre Charbit, Types de Données et Objet, 52h, L1, Université Paris Diderot, France

Master: Pierre Charbit, Programmation, 60h, M2Pro PISE, Université Paris Diderot, France

Master: Pierre Charbit, Algorithmique de Graphes, 18h, M2 MPRI, Université Paris Diderot, France

E-learning

Carole Delporte and Hugues Fauconnier were involved in the project SPC "informatique en ligne" (1 year, approved in June 2015), which consisted in preparing video sequences related to Bachelor level courses at Paris VII university.

10.2.2. Supervision

HdR: Amos Korman on the topic of "Distributed Decision and Distributed Biological Algorithms", Université Paris Diderot, November 2015.

PhD: François Durand was an Inria engineer hired by Gang under a contract with Alcatel-Lucent Bell Labs France. He obtained his PhD last September [1], under the subject "Towards Less Manipulable Voting Systems".

PhD: The Dang Huynh (co-advised by Laurent Viennot and Fabien Mathieu) was a PhD Cifre with Alcatel-Lucent Bell Labs France. He obtained his PhD in June 2015 [2], under the subject “Extension of PageRank and application to social networks.” He now works in the industry on related topics.

PhD in progress: Leonardo Linguaglossa (co-advised by Laurent Viennot, Fabien Mathieu and Diego Perino, both from Bell Labs / Alcatel-Lucent / Nokia).

PhD in progress: Simon Collet (co-advised by Amos Korman and Pierre Fraigiaud). Title of thesis is: "Algorithmic Game Theory Applied to Biology". Started September 2015.

PhD in progress: Lucas Boczkowski (co-advised by Amos Korman and Iordanis Kerenidis). Title of thesis is: "Computing with Limited Resources in Uncertain Environments". Started September 2015.

PhD in progress: Laurent Feuilloley (advised by Pierre Fraigniaud). Title of thesis is: "Synchronous Distributed Computing". Started September 2015.

Foreign PhD co-supervision in progress: Alkida Balliu and Dennis Olivetti (PhD students of Gianlorenzo D’Angelo at L’Aquilla University and Gran Sasso Science Institute) have Pierre Fraigniaud as their foreign co-supervisor since September 2015. They are expected to undertake a 6-month research visit to the team in 2016.

10.2.3. Juries

Laurent Viennot was reviewer for the PhD thesis of Dimitrios Milioris on “Trend detection and information propagation in dynamic social networks” at Ecole Polytechnique (May 2015).

Laurent Viennot was co-advisor for the PhD thesis of The Dang Huynh on “Extension de PageRank et application aux réseaux sociaux” at UPMC (April 2015).

Hugues Fauconnier participated as a jury member in the PhD defense of Lena Kanellou on “Structures de données pour des architectures multi-cœur actuelles et de futures architectures many-cœur” at Univ Rennes (December 2015).

Hugues Fauconnier was a reviewer for the PhD thesis of Giuseppe Di Luna “On deterministic counting in anonymous dynamic networks” at University Sapienza in Rome (May 2015).

Pierre Fraigniaud was reviewer for the PhD thesis of Adi Shklarsh on “Modeling Bacteria-Inspired Smart Swarms ” at Tel Aviv University (2015).

Michel Habib participated as a jury member in the PhD defense of Brahim Neggazi on “Self-stabilizing algorithms for graph parameters”, Université Claude Bernard Lyon I (May 2015).

Michel Habib was a reviewer for the PhD thesis of Sid Ali Selmane on “Détection et analyse de communautés”, Université Lumière Lyon 2 (May 2015).

Michel Habib participated as a jury member in the PhD defense of Stephen Gray on “Subgraph Epimorphisms: Theory and Application to Model Reductions in Systems Biology”, Université Paris Diderot (June 2015).

Michel Habib participated as a jury member in the PhD defense of Ronan Hamon on “Analyse de réseaux temporels par des méthodes de traitement du signal”, Laboratoire de Physique, ENS Lyon (September 2015).

Michel Habib was a reviewer for the HDR of Mathieu Liedloff on “Algorithmes exponentiels pour l’étiquetage, la domination et l’ordonnancement”, Orléans (December 2015).

Michel Habib was a reviewer for the HDR of Mamadou Kante on “Studying graphs: structure via rank-width, and listing of minimal dominating sets”, Clermont-Ferrand (December 2015).

Amos Korman participated as a jury member in the PhD defense of Jara Uitto, at ETH Zurich.

Adrian Kosowski was a reviewer for the PhD thesis of Krzysztof Turowski on “The analysis of algorithmic properties of the skeleton graph coloring problem” at the Gdansk University of Technology (May 2015).

10.3. Popularization

Michel Habib has contributed a text [41] to the “Encyclopedia of Algorithms”, providing a unifying and historical overview of graph searching problems and algorithms.

11. Bibliography

Publications of the year

Doctoral Dissertations and Habilitation Theses

[1] F. DURAND. *Toward Less Manipulable Voting Systems*, Paris VII, September 2015, https://hal.inria.fr/tel-01242440

[2] T. D. HUYNH. *Extension of PageRank and application to social networks*, Université Pierre et Marie Curie - Paris VI, June 2015, https://tel.archives-ouvertes.fr/tel-01187929

Articles in International Peer-Reviewed Journals

[3] P. ABOULKER, P. CHARBIT, N. TROTIGNON, K. VUŠKOVIĆ. *Vertex elimination orderings for hereditary graph classes*, in "Contributions to Discrete Mathematics", May 2015, vol. 338, n^{o} 5 [*DOI :* 10.1016/J.DISC.2014.12.014], https://hal.archives-ouvertes.fr/hal-01230783

[4] F. BECKER, A. KOSOWSKI, M. MATAMALA, N. NISSE, I. RAPAPORT, K. SUCHAN, I. TODINCA. *Allowing each node to communicate only once in a distributed system: shared whiteboard models*, in "Distributed Computing", 2015, vol. 28, n^{o} 3, pp. 189-200 [*DOI :* 10.1145/2312005.2312008], https://hal.inria.fr/hal-01163186

[5] M. BORASSI, P. CRESCENZI, M. HABIB, W. A. KOSTERS, A. MARINO, F. W. TAKES. *Fast diameter and radius BFS-based computation in (weakly connected) real-world graphs: With an application to the six degrees of separationgames*, in "Theoretical Computer Science", May 2015, vol. 586, 21 p. , https://hal.inria.fr/hal-01255125

[6] S. BRIERLEY, A. KOSOWSKI, M. MARKIEWICZ, T. PATEREK, A. PRZYSIĘŻNA. *Nonclassicality of Temporal Correlations*, in "Physical Review Letters", September 2015, vol. 115, n^{o} 12 [*DOI :* 10.1103/PHYSREVLETT.115.120404], https://hal.inria.fr/hal-01260792

[7] H. BRUHN, P. CHARBIT, O. SCHAUDT, J. A. TELLE. *The graph formulation of the union-closed sets conjecture*, in "European Journal of Combinatorics", January 2015 [*DOI :* 10.1016/J.EJC.2014.08.030], https://hal.archives-ouvertes.fr/hal-01253152

[8] P. CHARBIT, R. AHARONI, D. HOWARD. *On a Generalization of the Ryser-Brualdi-Stein Conjecture*, in "Journal of Graph Theory", January 2015, 14 p. [*DOI :* 10.1002/JGT.21796], https://hal.inria.fr/hal-01101508

[9] D. G. CORNEIL, J. DUSART, M. HABIB, A. MAMCARZ, F. DE MONTGOLFIER. *A tie-break model for graph search*, in "Discrete Applied Mathematics", January 2016, vol. 199, 12 p. [*DOI :* 10.1016/J.DAM.2015.06.011], https://hal.inria.fr/hal-01255055

[10] C. CRESPELLE, T. H. D. PHAN, H. TRAN-THE. *Termination of the Iterated Strong-Factor Operator on Multipartite Graphs*, in "Theoretical Computer Science", 2015, vol. 571, pp. 67-77, https://hal.inria.fr/hal-01241925

[11] J. CZYZOWICZ, L. GASIENIEC, A. KOSOWSKI, E. KRANAKIS, O. MORALES-PONCE, E. PACHECO. *Position discovery for a system of bouncing robots*, in "Journal of Information and Computation", October 2015, vol. 244, pp. 122-133 [*DOI :* 10.1016/J.IC.2015.07.005], https://hal.inria.fr/hal-01251829

[12] C. DELPORTE-GALLET, H. FAUCONNIER, E. GAFNI, S. RAJSBAUM. *Linear space bootstrap communication schemes*, in "Journal of Theoretical Computer Science (TCS)", 2015, vol. 561, pp. 122-133 [*DOI :* 10.1016/J.TCS.2014.10.013], https://hal.inria.fr/hal-01251557

[13] C. DELPORTE-GALLET, H. FAUCONNIER, P. KOUZNETSOV, E. GAFNI. *Wait-freedom with advice*, in "Distributed Computing", 2015, vol. 28, n^{o} 1, pp. 3-19 [*DOI :* 10.1007/S00446-014-0231-6], https://hal.inria.fr/hal-01251549

[14] D. DERENIOWSKI, Y. DISSER, A. KOSOWSKI, D. PAJAK, P. UZNANSKI. *Fast collaborative graph exploration*, in "Journal of Information and Computation", August 2015, vol. 243, pp. 37-49 [*DOI :* 10.1016/J.IC.2014.12.005], https://hal.archives-ouvertes.fr/hal-01251826

[15] D. DERENIOWSKI, R. KLASING, A. KOSOWSKI, Ł. KUSZNER. *Rendezvous of heterogeneous mobile agents in edge-weighted networks*, in "Journal of Theoretical Computer Science (TCS)", December 2015, vol. 608, n^{o} 3, pp. 219-230 [*DOI :* 10.1016/J.TCS.2015.05.055], https://hal.inria.fr/hal-01251835

[16] D. DERENIOWSKI, A. KOSOWSKI, D. PAJAK. *Distinguishing Views in Symmetric Networks: A Tight Lower Bound*, in "Journal of Theoretical Computer Science (TCS)", May 2015, vol. 582, pp. 27-34 [*DOI :* 10.1016/J.TCS.2015.03.018], https://hal.inria.fr/hal-00875370

[17] N. HEGDE, L. MASSOULIÉ, L. VIENNOT. *Self-Organizing Flows in Social Networks*, in "Theoretical Computer Science", February 2015, 16 p. [*DOI :* 10.1016/J.TCS.2015.02.018], https://hal.inria.fr/hal-00761046

[18] T. D. HUYNH, F. MATHIEU, L. VIENNOT. *LiveRank: How to Refresh Old Datasets*, in "Internet Mathematics", 2015 [*DOI :* 10.1080/15427951.2015.1098756], https://hal.inria.fr/hal-01251552

[19] A. KORMAN, S. KUTTEN, T. MASUZAWA. *Fast and compact self-stabilizing verification, computation, and fault detection of an MST*, in "Distributed Computing", August 2015, vol. 28, n^{o} 4 [*DOI :* 10.1007/S00446-015-0242-Y], https://hal.archives-ouvertes.fr/hal-01240211

[20] A. KOSOWSKI, B. LI, N. NISSE, K. SUCHAN. *k-Chordal Graphs: from Cops and Robber to Compact Routing via Treewidth*, in "Algorithmica", 2015, vol. 72, n^{o} 3, pp. 758-777, https://hal.archives-ouvertes.fr/hal-01163494

Invited Conferences

[21] O. FEINERMAN, A. KORMAN. *Clock Synchronization and Distributed Estimation in Highly Dynamic Networks: An Information Theoretic Approach*, in "Structural Information and Communication Complexity - 22nd International Colloquium, SIROCCO 2015", Montserrat, Spain, July 2015 [*DOI :* 10.1007/978-3-319-25258-2_2], https://hal.archives-ouvertes.fr/hal-01240216

International Conferences with Proceedings

[22] M. BARUCH, P. FRAIGNIAUD, B. PATT-SHAMIR. *Randomized Proof-Labeling Schemes*, in "ACM Symposium on Principles of Distributed Computing (PODC)", Donostia-San Sebastián, Spain, ACM, July 2015, pp. 315-324 [*DOI :* 10.1145/2767386.2767421], https://hal.inria.fr/hal-01247352

[23] P. BERENBRINK, R. KLASING, A. KOSOWSKI, F. MALLMANN-TRENN, P. UZNANSKI. *Improved Analysis of Deterministic Load-Balancing Schemes*, in "PODC '15 Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing", San Sebastian, Spain, July 2015, pp. 301-310 [*DOI :* 10.1145/2767386.2767413], https://hal.inria.fr/hal-01251847

[24] L. BLIN, P. FRAIGNIAUD. *Space-Optimal Time-Efficient Silent Self-Stabilizing Constructions of Constrained Spanning Trees*, in "35th IEEE International Conference on Distributed Computing Systems (ICDCS)", Columbus, United States, IEEE, June 2015, pp. 589-598 [*DOI :* 10.1109/ICDCS.2015.66], https://hal.inria.fr/hal-01247340

[25] Y. BOUFKHAD, R. DE LA PAZ, L. LINGUAGLOSSA, F. MATHIEU, D. PERINO, L. VIENNOT. *Vérification de tables de routage par utilisation d'un ensemble représentatif d'en-têtes*, in "ALGOTEL 2015 — 17èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications", Beaune, France, June 2015, https://hal.inria.fr/hal-01148595

[26] J. CHALOPIN, S. DAS, P. GAWRYCHOWSKI, A. KOSOWSKI, A. LABOUREL, P. UZNANSKI. *Limit Behavior of the Multi-Agent Rotor-Router System*, in "DISC 2015", Tokyo, Japan, Y. MOSES, M. ROY (editors), 29th International Symposium on Distributed Computing, Springer-Verlag Berlin Heidelberg, October 2015, vol. LNCS 9363 [*DOI :* 10.1007/978-3-662-48653-5_9], https://hal.archives-ouvertes.fr/hal-01206149

[27] J. CZYZOWICZ, L. GASIENIEC, A. KOSOWSKI, E. KRANAKIS, D. KRIZANC, N. TALEB. *When Patrolmen Become Corrupted: Monitoring a Graph using Faulty Mobile Robots*, in "26th International Symposium on Algorithms and Computation (ISAAC 2015), Proceedings", Nagoya, Japan, Lecture Notes in Computer Science, December 2015, vol. 9472, pp. 343-354 [*DOI :* 10.1007/978-3-662-48971-0_30], https://hal.inria.fr/hal-01194847

[28] J. CZYZOWICZ, L. GASIENIEC, A. KOSOWSKI, E. KRANAKIS, P. G. SPIRAKIS, P. UZNANSKI. *On Convergence and Threshold Properties of Discrete Lotka-Volterra Population Protocols*, in "42nd International Colloquium on Automata, Languages, and Programming (ICALP 2015), Proceedings, Part I", Kyoto, Japan, July 2015, vol. 9134, pp. 393-405 [*DOI :* 10.1007/978-3-662-47672-7_32], https://hal.inria.fr/hal-01137486

[29] C. DELPORTE-GALLET, H. FAUCONNIER, P. KOUZNETSOV, E. RUPPERT. *On the Space Complexity of Set Agreement?* , in "ACM Symposium on Principles of Distributed Computing, PODC 2015", Donostia, France, July 2015, https://hal.inria.fr/hal-01251563

[30] C. DELPORTE-GALLET, H. FAUCONNIER, S. TOUEG. *A Separation of n-consensus and (n + 1)-consensus Based on Process Scheduling*, in "Structural Information and Communication Complexity - 22nd International Colloquium, SIROCCO 2015", Montserrat, France, Springer LNCS, July 2015, vol. 9439, pp. 385-398 [*DOI :* 10.1007/978-3-319-25258-2_27], https://hal.inria.fr/hal-01251571

[31] F. DURAND, B. KLOECKNER, F. MATHIEU, L. NOIRIE. *Geometry on the Utility Space*, in "Fourth International Conference on Algorithmic Decision Theory", Lexington, United States, September 2015, 16 p. [*DOI :* 10.1007/978-3-319-23114-3_12], https://hal.inria.fr/hal-01222871

[32] L. FEUILLOLEY, P. FRAIGNIAUD. *Randomized Local Network Computing*, in "27th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)", Portland, United States, ACM, June 2015, pp. 340-349 [*DOI :* 10.1145/2755573.2755596], https://hal.inria.fr/hal-01247357

[33] P. FRAIGNIAUD, J. HIRVONEN, J. SUOMELA. *Node Labels in Local Decision*, in "22nd International Colloquium on Structural Information and Communication Complexity (SIROCCO)", Montserrat, Spain, Springer, July 2015, vol. LNCS 9439, pp. 589-598 [*DOI :* 10.1007/978-3-319-25258-2_3], https://hal.inria.fr/hal-01247355

[34] T. D. HUYNH, C. SHUE CHEN, S.-W. HO. *Exploiting User Movement for Position Detection*, in "IEEE Consumer Communications and Networking Conference", Las Vegas, United States, IEEE, January 2015, 6 p. , https://hal.archives-ouvertes.fr/hal-01090484

Conferences without Proceedings

[35] T. D. HUYNH, C. SHUE CHEN, S.-W. HO. *Localization method for device-to-device through user movement*, in "IEEE International Conference on Communications (ICC)", London, United Kingdom, 2015 [*DOI :* 10.1109/ICCW.2015.7247279], https://hal.inria.fr/hal-01216870

Scientific Books (or Scientific Book chapters)

[36] A. KORMAN, S. K. DAS, K. DILIP, K. SANTONU, K. MOHAN, M. PORTMANN, S. SASTRY. *Proceedings of the 2015 International Conference on Distributed Computing and Networking, ICDCN 2015, Goa, India, January 4-7, 2015*, ACM Digital Library, December 2015, https://hal.inria.fr/hal-01251330

[37] A. KOSOWSKI, I. WALUKIEWICZ. *20th International Symposium on Fundamentals of Computation Theory (FCT 2015), Proceedings, Gdansk, Poland, August 17-19, 2015*, Lecture Notes in Computer Science, Springer, August 2015, vol. 9210 [*DOI :* 10.1007/978-3-319-22177-9], https://hal.inria.fr/hal-01251854

Research Reports

[38] Y. BOUFKHAD, R. DE LA PAZ, L. LINGUAGLOSSA, F. MATHIEU, D. PERINO, L. VIENNOT. *Forwarding Tables Verification through Representative Header Sets*, Inria Paris Rocquencourt, January 2016, https://hal.inria.fr/hal-01262130

[39] F. DURAND, F. MATHIEU, L. NOIRIE. *SVVAMP: Simulator of Various Voting Algorithms in Manipulating Populations*, Inria, March 2015, https://hal.archives-ouvertes.fr/hal-01135109

[40] P. FRAIGNIAUD, S. RAJSBAUM, C. TRAVERS. *Minimizing the Number of Opinions for Fault-Tolerant Distributed Decision Using Well-Quasi Orderings*, LaBRI, U, December 2015, https://hal.archives-ouvertes.fr/hal-01237873

Scientific Popularization

[41] D. G. CORNEIL, M. HABIB. *Unified View of Graph Searching and LDFS-Based Certifying Algorithms*, in "Encyclopedia of Algorithms", Springer-verlag, January 2015 [*DOI :* 10.1007/978-3-642-27848-8_685-1], https://hal.inria.fr/hal-01255083

Other Publications

[42] P. BERENBRINK, R. KLASING, A. KOSOWSKI, F. MALLMANN-TRENN, P. UZNANSKI. *Improved Analysis of Deterministic Load-Balancing Schemes*, February 2015, minor corrections; updated literature overview, https://hal.inria.fr/hal-00979691

[43] J. CHALOPIN, S. DAS, P. GAWRYCHOWSKI, A. KOSOWSKI, A. LABOUREL, P. UZNANSKI. *Lock-in Problem for Parallel Rotor-router Walks*, May 2015, working paper or preprint, https://hal.inria.fr/hal-01021930

[44] P. CHARBIT, L. HOSSEINI, P. OSSONA DE MENDEZ. *Limits of Structures and the Example of Tree-Semilattices*, September 2015, working paper or preprint, https://hal.archives-ouvertes.fr/hal-01150659

[45] P. FRAIGNIAUD, M. HEINRICH, A. KOSOWSKI. *Local Conflict Coloring*, November 2015, working paper or preprint, https://hal.inria.fr/hal-01223494