RESEARCH CENTRE

Inria Centre at the University of Lille

IN PARTNERSHIP WITH:

CNRS, Université de Lille

2023 ACTIVITY REPORT

Project-Team LINKS

Linking Dynamic Data

IN COLLABORATION WITH: Centre de Recherche en Informatique, Signal et Automatique de Lille

DOMAIN

Perception, Cognition and Interaction

THEME

Data and Knowledge Representation and Processing

Inria

Contents

Project-Team LINKS

Creation of the Project-Team: 2016 June 01

Keywords

Computer sciences and digital sciences

A2.1. – Programming Languages

A2.1.1. – Semantics of programming languages

A2.1.4. – Functional programming

A2.1.6. – Concurrent programming

A2.4. – Formal method for verification, reliability, certification

A2.4.1. – Analysis

A2.4.2. – Model-checking

A2.4.3. – Proofs

A3.1. – Data

A3.1.1. – Modeling, representation

A3.1.2. – Data management, quering and storage

A3.1.3. – Distributed data

A3.1.4. – Uncertain data

A3.1.5. – Control access, privacy

A3.1.6. – Query optimization

A3.1.7. – Open data

A3.1.8. – Big data (production, storage, transfer)

A3.1.9. – Database

A3.2.1. – Knowledge bases

A3.2.2. – Knowledge extraction, cleaning

A3.2.3. – Inference

A3.2.4. – Semantic Web

A4.7. – Access control

A4.8. – Privacy-enhancing technologies

A7. – Theory of computation

A7.2. – Logic in Computer Science

A9.1. – Knowledge

A9.2. – Machine learning

A9.7. – AI algorithmics

A9.8. – Reasoning

Other research topics and application domains

B6.1. – Software industry

B6.3.1. – Web

B6.3.4. – Social Networks

B6.5. – Information systems

B9.5.1. – Computer science

B9.5.6. – Data science

B9.10. – Privacy

1 Team members, visitors, external collaborators

Research Scientists

- Mikaël Monet [INRIA, Researcher]
- Joachim Niehren [INRIA, Senior Researcher, HDR]

Faculty Members

- Sylvain Salvati [Team leader, UNIV LILLE, Professor, HDR]
- Iovka Boneva [UNIV LILLE, Associate Professor]
- Florent Capelli [UNIV LILLE, Associate Professor, until Aug 2023]
- Aurélien Lemay [UNIV LILLE, Associate Professor, HDR]
- Charles Paperman [UNIV LILLE, Associate Professor]
- Sophie Tison [UNIV LILLE, Professor, HDR]

PhD Students

- Antonio Al Serhali [UNIV LILLE, from Sep 2023]
- Antonio Al Serhali [INRIA, until Aug 2023]
- Oliver Irwin [UNIV LILLE]
- Claire Soyez-Martin [INRIA]

Technical Staff

- Nicolas Crosetti [INRIA, Engineer, until Oct 2023]

Interns and Apprentices

- Moira Fruchaud [INRIA, Intern, from Jun 2023 until Jun 2023]
- Niranjan Kumar [INRIA, Intern, from May 2023 until Jul 2023]

Administrative Assistants

- Nathalie Bonte [INRIA]
- Karine Lewandowski [INRIA]

Visiting Scientists

- Corentin Barloy [UNIV LILLE]
- Michael Cadilhac [DePaul University, Chicago (USA)]
- Olivier Idir [ENS DE LYON, from Feb 2023 until Mar 2023]
- Le Thanh Dung Nguyen [ECOLE POLY PALAISEAU, until Sep 2023]
- Sylvain Schmitz [UNIV PARIS XIII]
- Howard Straubing [Boston College (USA)]
- Sarah Winter [UNIV BRUXELLES, until Feb 2023]

2 Overall objectives

We develop algorithms for answering logical querying on heterogeneous linked data collections in hybrid formats, distributed programming languages for managing dynamic linked data collections and workflows based on queries and mappings, and symbolic machine learning algorithms that can link datasets by inferring appropriate queries and mappings.

2.1 Presentation

The following three items summarize our main research objectives.

Querying Heterogeneous Linked Data We develop new kinds of schema mappings for semi-structured datasets in hybrid formats including graph databases, RDF collections, and relational databases. These induce recursive queries on linked data collections for which we will investigate evaluation algorithms, containment problems, and concrete applications.

Managing Dynamic Linked Data In order to manage dynamic linked data collections and workflows, we will develop distributed data-centric programming languages with streams and parallelism, based on novel algorithms for incremental query answering, study the propagation of updates of dynamic data through schema mappings, and investigate static analysis methods for linked data workflows.

Linking Data Graphs Finally, we will develop symbolic machine learning algorithms, for inferring queries and mappings between linked data collections in various graphs formats from annotated examples.

3 Research program

3.1 Background

The main objective of LINKS is to develop methods for querying and managing linked data collections. Even though open linked data is the most prominent example, we will focus on hybrid linked data collections, which are collections of semi-structured datasets in hybrid formats: graph-based, RDF, relational, and NOSQL. The elements of these datasets may be linked, either by pointers or by additional relations between the elements of the different datasets, for instance the "same-as" or "member-of" relations as in RDF.

The advantage of traditional data models is that there exist powerful querying methods and technologies that one might want to preserve. In particular, they come with powerful schemas that constraint the possible manners in which knowledge is represented to a finite number of patterns. The exhaustiveness of these patterns is essential for writing of queries that cover all possible cases. Pattern violations are excluded by schema validation. In contrast, RDF schema languages such as RDFS can only enrich the relations of a dataset by new relations, which also helps for query writing, but which cannot constraint the number of possible patterns, so that they do not come with any reasonable notion of schema validation.

The main weakness of traditional formats, however, is that they do not scale to large data collections as stored on the Web, while the RDF data models scales well to very big collections such as linked open data. Therefore, our objective is to study mixed data collections, some of which may be in RDF format, in which we can lift the advantages of smaller datasets in traditional formats to much larger linked data collections. Such data collections are typically distributed over the internet, where data sources may have rigid query facilities that cannot be easily adapted or extended.

The main assumption that we impose in order to enable the logical approach, is that the given linked data collection must be correct in most dimensions. This means that all datasets are well-formed with respect to their available constraints and schemas, and clean with respect to the data values in most of the components of the relations in the datasets. One of the challenges is to integrate good quality RDF datasets into this setting, another is to clean the incorrect data in those dimensions that are less proper. It remains to be investigated in how far these assumptions can be maintained in realistic applications, and how much they can be weakened otherwise.

For querying linked data collections, the main problems are to resolve the heterogeneity of data formats and schemas, to understand the efficiency and expressiveness of recursive queries, that can follow links repeatedly, to answer queries under constraints, and to optimize query answering algorithms based on static analysis. When linked data is dynamically created, exchanged, or updated, the problems are how to process linked data incrementally, and how to manage linked data collections that change dynamically. In any case (static and dynamic) one needs to find appropriate schema mappings for linking semi-structured datasets. We will study how to automatize parts of this search process by developing symbolic machine learning techniques for linked data collections.

3.2 Research Axis: Querying Data Graphs

Linked data is often abstracted as datagraphs: nodes carry information and edges are labeled. Internet, the semantic web, open data, social networks and their connections, information streams such as twitter are examples of such datagraphs. An axis of LINKS is to propose methods and tools so as to extract information from datagraphs. We dwell in a wide spectrum of tools to construct these methods: circuits, compilation, optimization, logic, automata, machine learning. Our goal is to extend the kinds of information that can be extracted from datagraphs while improving the efficiency of existing ones.

This axis is split within two themes. The first one pertains to the use of low level representations by means of circuits to compute efficiently complex numerical aggregates that will find natural applications in AI. The second one proposes to explore path oriented query language and more particularly their efficient evaluation by means of efficient compilation and machine learning methods so as to have manageable statistics.

3.2.1 AI: Circuits for Data Analysis

Circuits are concise representations of data sets that recently found a unifying interest in various areas of artificial intelligence. A circuit may for instance represent the answer set of a database query as a dag whose operators are disjoint unions (for disjunction) and Cartesian products (for conjunction). Similarly, it may also represent the set of all matches of a pattern in a graph. The structure of the circuit may give rise to efficient algorithms to process large data sets based on representation that are often much smaller. Among others, such applications range from knowledge representation/compilation, counting the number of solutions of queries, efficient query answering, factorized databases.

In a first line of research, we want to study novel problems on circuits, in which database queries are relevant to data analysis tasks from artificial intelligence, in machine learning or data mining in particular. In particular we propose to study optimization problems on answer sets of database queries based on circuits, i.e. how to find optimal solutions in the answer set for a given set of conditions. Decompressing small circuits into large answer sets would make the optimization problem unfeasible in many cases. We believe that circuits can structure certain optimization problems in such a way that it can be phrased concisely and then solved efficiently.

Second, we propose to develop a tighter integration between circuits and databases. Indeed query-related circuits are generally produced from a database. This requires that the data is copied within the circuits. This memory cost is accompanied with the loss of the environment of the DBMS which allows many optimizations and uses many low level optimizations that are hard to implement. We propose then to encode circuits directly within the database using materialized views and index structures. We shall also develop the required computational tools for maintaining and exploiting these embedded circuits.

3.2.2 Path Query Optimization

Graph databases are easily queried using path descriptions. Most often these paths are described by means of regular expressions. This makes path queries difficult as the use of Kleene star makes them recursive. In relational DBMS, recursion is almost never used and it is not advised to use it. The natural theoretical tool that pertains to recursion in the context of relational data Datalog. There has been a wealth of optimization algorithms that have been proposed for queries written in Datalog. We propose to use Datalog as a low level language to which we will compile path queries of various kinds. The idea is that the compiler will try to obtain Datalog programs that will have low execution complexity

taking advantages of optimization techniques such as magic supplementary set rewriting, pre-computed indexes and also statistics computed from the graph. Our goal is to develop a compiler that will be able to efficiently evaluate path queries on large graphs which in particular will explore only a part of it.

3.3 Research Axis: Monitoring Data Graphs

Traditional database applications are programs that interact with database via updates and queries. We are interested in developing programming language techniques so as to interact with datagraphs rather than with traditional relational databases. Moreover, we shall take into account the dynamic aspects of datagraphs which shall evolve through updates. We will develop methods to monitor changes in datagraphs and react according to the modifications.

3.3.1 Functional Programming Languges for Data Graphs

The first question is which kind of programming language to use to enable monitoring processes for data graphs based on query answering. While languages of path queries found quite some interest on data graphs, less attention has been given to the programming language tasks, that needed to be solved to produce structured output and to compose various queries with structured output into a pipeline. We believe that transferring the generalization of ideas developed for data trees in the context of XML to data graphs will allow to solve such problems in a systematic manner.

Our approach will consist in developing a functional programming language based on first principles (the lambda calculus, graph navigation, logical connective) that generalizes full XPath 3.0 to the context of graphs. Here we can rely on own previous work for data trees, such as the language X-Fun and λ-XP. After the language for data graphs is designed we shall study its behavior empirically by means of an implementation. This implementation will help us to design optimization methods so as to evaluate the queries in that language. This will allow us to use a wealth of techniques so as to optimize the computation. Indeed, we can try to compile data structures to imperative ones when possible and also exploit possibilities of parallel executions in certain cases. Functional programming comes with nice verification techniques that we are going to use in several contexts: (i) in optimizing queries (e.g. stop the evaluation when it is possible to know that no more data can contribute to the output) and (ii) to verify that the query behaves correctly. The verification methods we shall focus on will be mainly related to automata and transducers.

Finally we shall also develop a programming language that allows to describe services that use datagraphs as a backend for storing data. Here again, functional programming seems a good candidate, we would need however to orchestrate the concurrent executions of queries so as to ensure the correct behavior of services. This means that we should have concurrent constructs that are built in the language. The high level of concurrence enabled by the notion of *futures* seems an interesting candidate to adapt to the context of service orchestration.

3.3.2 Hyperstreaming Program Evaluation

Complex-event processing requires to monitor data graphs that are produced on input streams and to write data graphs to some output stream, which can then be used as inputs again. A major problem here is to reduce the high risk of blocking, which arises when the writing of some of the output stream suspends on a data value that will become available only in the future on some input stream. In such cases, all monitoring processes reading the output stream may have to suspend as well. In order to reduce the risk of blocking, we propose to develop the hyperstreaming approach further, of which we layed the foundations in the evaluation period based on automata techniques. The idea is to generalize streams to hyperstreams, i.e. to add holes to streams that can be filled by some other stream in the future. In order to avoid suspension as possible, a monitoring process on hyperstream must then be able to jump over the holes, and to perform some speculative computation. The objective for the next period are to develop tools for hyperstreaming query answering and to lift these to hyperstreaming program evaluation. Furthermore, on the conceptual side, the notion of certain query answers on hyperstreams needs to be lifted to certain program outputs on hyperstreams.

3.4 Research Axis: Graph Data Integration

We intend to continue to develop tools for integration of linked data with RDF being their principal format. Because from its conception the main credo of RDF has been "just publish your data", the problem at hand faces two important challenges: data quality and data heterogeneity.

3.4.1 Data Quality with Schemas and Repairing with Inference

The data quality of RDF may suffer due to a number of reasons. Impurities may arise due to data value errors (misspellings, errors during data entry etc.). Such data quality problems have been thoroughly investigated in literature for relational databases and solutions include dictionary methods etc. However, it remains to be seen if the challenges of adapting the existing solutions for relational databases can be easily addressed.

One particular challenge comes from the fact that RDF allows a higher degree of structural freedom in how information is represented as opposed to relation databases, where the choice is strongly limited to flat tables. We plan to investigate suitability of existing data cleaning methods to tackle the problems of data value impurities in RDF. The structural freedom of RDF is a source of data quality issues on its own. With the recent emergence of schema formalisms for RDF, it becomes evident that significant parts of existing RDF repositories do not necessarily satisfy schemas prepared by domain experts.

In the first place, we intend to investigate defining suitable measures of quality for RDF documents. Our approaches will be based on a schema language, such as ShEx and SHACL, and we shall explore suitable variants of graph alignment and graph edit distance to capture similarity between the existing RDF document and its possible repaired versions that satisfy the schema.

The central issue here is repairing an RDF document w.r.t. schema by identifying essential fragments of the RDF that fail to satisfy the schema. Once such fragments are identified, repairing actions can be applied however there might be a significant number of alternatives. We intend to explore enumeration approaches where the space of repairing alternatives is intelligently browsed by the user and the most suitable one is chosen. Furthermore, we intend to propose a rule language for choosing the most suitable repairing action and will investigate inference methods to derive from interactions with user the optimal order in which various repairing actions are presented to the user and derive the rules for the choice of the preferred repairing action for repeating types of fragments that do not satisfy the schema.

3.4.2 Integration and Graph Mappings with Schemas and Inference

The second problem pertaining to integration of RDF data sources is their heterogeneity. We intend to continue to identify and study suitable classes of mappings between RDF documents conforming to potentially different and complementary schemas. We intend to assist the user in constructing such mappings by developing rich and expressive graphical languages for mappings. Also, we wish to investigate inference of RDF mappings with the active help of an expert user. We will need to define interactive protocols that allows the input to be sufficiently informative to guide the inference process while avoiding the pitfalls of user input being too ambiguous and causing combinatorial explosion. We intend to identify

RDF Data Quality. Approach based on a schema language (ShEx or SHACL) used to identify errors and giving a notion of a measure of quality of an RDF database. Impurities in RDF may come from data value errors (misspellings etc.) but also from the fact that RDF imposes fewer constraints on how data is structured which is a consequence of a significantly different use philosophy (just publish your data anyway you want). Repairing of RDF errors would be modeled with a localized rules (transformations that operate within a small radius of an affected node) and if several rules apply, preferences are used to identify the most desirable one. Both the repairing rules and preferences can be inferred with the help of inference algorithms in an interactive setting. Smart tools for LOD integration. Assuming that the LOD sources are of good quality, we want to build tools that assist the user in constructing mappings that integrate data in the user database. For this, we want to define inference algorithms which are guided by schemas, and which are based on comprehensible interactions with the user. For this, we need to define interactions that are rich enough to inform the algorithm, while simple enough to be understandable by a non-expert user. In particular, that means that we need to present data (nodes in a graph for instance)

in a readable way. Also, we want to investigate how the - possibly inferred - schema can be used to guide the inference.

4 Application domains

4.1 Linked data integration

There are many contexts in which integrating linked data is interesting. We advocate here one possible scenario, namely that of integrating business linked data to feed what is called Business Intelligence. The latter consists of a set of theories and methodologies that transform raw data into meaningful and useful information for business purposes (from Wikipedia). In the past decade, most of the enterprise data was proprietary, thus residing within the enterprise repository, along with the knowledge derived from that data. Today's enterprises and businessmen need to face the problem of information explosion, due to the Internet's ability to rapidly convey large amounts of information throughout the world via end-user applications and tools. Although linked data collections exist by bridging the gap between enterprise data and external resources, they are not sufficient to support the various tasks of Business Intelligence. To make a concrete example, concepts in an enterprise repository need to be matched with concepts in Wikipedia and this can be done via pointers or equalities. However, more complex logical statements (i.e. mappings) need to be conceived to map a portion of a local database to a portion of an RDF graph, such as a subgraph in Wikipedia or in a social network, e.g. LinkedIn. Such mappings would then enrich the amount of knowledge shared within the enterprise and let more complex queries be evaluated. As an example, businessmen with the aid of business intelligence tools need to make complex sentimental analysis on the potential clients and for such a reason, such tools must be able to pose complex queries, that exploit the previous logical mappings to guide their analysis. Moreover, the external resources may be rapidly evolving thus leading to revisit the current state of business intelligence within the enterprise.

4.2 Data cleaning

The second example of application of our proposal concerns scientists who want to quickly inspect relevant literature and datasets. In such a case, local knowledge that comes from a local repository of publications belonging to a research institute (e.g. HAL) need to be integrated with other Web-based repositories, such as DBLP, Google Scholar, ResearchGate and even Wikipedia. Indeed, the local repository may be incomplete or contain semantic ambiguities, such as mistaken or missing conference venues, mistaken long names for the publication venues and journals, missing explanation of research keywords, and opaque keywords. We envision a publication management system that exploits both links between database elements, namely pointers to external resources and logical links. The latter can be complex relationships between local portions of data and remote resources, encoded as schema mappings. There are different tasks that such a scenario could entail such as (i) cleaning the errors with links to correct data e.g. via mappings from HAL to DBLP for the publications errors, and via mappings from HAL to Wikipedia for opaque keywords, (ii) thoroughly enrich the list of publications of a given research institute, and (iii) support complex queries on the corrected data combined with logical mappings.

4.3 Real-time complex event processing

Complex event processing serves for monitoring nested word streams in real time. Complex event streams are gaining popularity with social networks such as with Facebook and Twitter, and thus should be supported by distributed databases on the Web. Since this is not yet the case, there remains much space for future industrial transfer related to LINKS' second axis on dynamic linked data.

5 Social and environmental responsibility

5.1 Footprint of research activities

Tison is a member of the general assembly of the European Association for Theoretical Computer Science (EATCS) (elected in 2019).

5.2 Impact of research results

Databases and methods from Artificial Intelligence are used in virtually all aspects of the modern digitalized world, from companies' web services to governments' institutions.

6 Highlights of the year

6.1 Awards

Tison HDF100 2023 laureate, "le classement des 100 personnes qui font bouger la Tech et l'Innovation en Hauts-de-France 2023" (HDFID/French Tech Lille).

6.2 International stay

Visited institution: Simons Institute for the Theory of Computing

Country: USA

Dates: August 15th - December 16th 2023

Context of the visit: Monet was selected to be a fellow of the Simons Institute during its "Logic and Algorithms in Database Theory and AI" program.

Mobility program/type of mobility: Research stay

6.3 Scientific results

Paper [25] by Capelli and Irwin.

Description The paper describes a general technique for direct access in query results that elegantly generalizes known techniques and extend them to conjunctive queries with negation.

Award The paper received the *best newcomer award* at the conference ICDT2024.

Papers [12], [13], [26] by Monet and co-authors.

Description In this series of papers, Monet and co-authors study the computation of the widely used Shapley values in many settings (machine learning, data management, probabilistic databases).

7 New software, platforms, open data

7.1 New software

7.1.1 NetworkDisk

Name: NetworkDisk

Keywords: Large graphs, Python, Databases

Functional Description: NetworkDisk provides a way to manipulate graphs on disk. The goal is to be as much as possible compatible with (Di)Graph objects of the NetworkX Python package but lifting memory requirement and providing persistence of the Graph.

URL: https://networkdisk.inria.fr/

Contact: Charles Paperman

7.1.2 Bibendum

Name: Bibendum

Keyword: Bibliography

Functional Description: Small app to fetch bibtex from a short label with the format: LastName.Year.PublicationTerm where the . denote the concatenation. For instance Codd1970Relational. LastName is the last name of one of the authors. Year is the publication year. PublicatonTerm is one meaningful word in the title of the publication.

In case of ambiguity, an extra integer is used. Ambiguous entries are resolved by sorting dois under lexicographical order. The api is idempotent, every decision taken is recorded and replayed.

URL: https://gitlab.inria.fr/cpaperma/bibendum

Contact: Charles Paperman

7.1.3 XPath AutoBench

Name: A Benchmark Collection of Deterministic Automata for XPath Queries

Keywords: XML, Querying, Tree Automata

Functional Description: We provide a benchmark collection of deterministic automata for regular XPath queries. For this, we select the subcollection of forward navigational XPath queries from a corpus that Lick and Schmitz extracted from real-world XSLT and XQuery programs, compile them to stepwise hedge automata (SHAs), and determinize them. Large blowups by automata determinization are avoided by using schema-based determinization. The schema captures the XML data model and the fact that any answer of a path query must return a single node. Our collection also provides deterministic nested word automata that we obtain by compilation from deterministic SHAs.

URL: https://archive.softwareheritage.org/browse/origin/directory/?origin_url=https://gitlab.inria.fr/aalserha/xpath-benchmark

Contact: Joachim Niehren

7.1.4 rsonpath

Keywords: JSon, Streaming, SIMD, Rust

Functional Description: The rsonpath crate provides a JSONPath parser and a query execution engine, which utilizes SIMD instructions to provide massive throughput improvements over conventional engines.

URL: https://github.com/V0ldek/rsonpath

Contact: Charles Paperman

Partner: Warsaw University

7.1.5 Coussinet

Name: Coussinet

Keywords: Enumeration, Complexity

Functional Description: Coussinet is a demo illustrating a technique called geometric amortization for enumeration algorithms introduced in the paper Geometric Amortization for Enumeration Algorithms, Florent Capelli, Yann Strozecki. The result presented in this paper is about making the delay of enumeration algorithms more regular.

URL: http://florent.capelli.me/coussinet/coussinet.html

Contact: Florent Capelli

Participants: Florent Capelli, Yann Strozecki

7.1.6 ShEx validator

Name: Validation of Shape Expression schemas

Keywords: Data management, RDF

Functional Description: Shape Expression schemas is a formalism for defining constraints on RDF graphs. This software allows to check whether a graph satisfies a Shape Expressions schema.

Release Contributions: ShExJava now uses the Commons RDF API and so support RDF4J, Jena, JSON-LD-Java, OWL API and Apache Clerezza. It can parse ShEx schema in the ShEcC, ShEJ, ShExR formats and can serialize a schema in ShExJ.

To validate data against a ShExSchema using ShExJava, you have two different algorithms: - the refine algorithm: compute once and for all the typing for the whole graph - the recursive algorithm: compute only the typing required to answer a validate(node,ShapeLabel) call and forget the results.

URL: https://github.com/iovka/shex-java

Contact: Iovka Boneva

7.1.7 gMark

Name: gMark: schema-driven graph and query generation

Keywords: Semantic Web, Data base

Functional Description: gMark allow the generation of graph databases and an associated set of query from a schema of the graph.gMark is based on the following principles: - great flexibility in the schema definition - ability to generate big size graphs - ability to generate recursive queries - ability to generate queries with a desired selectivity

URL: https://github.com/graphMark/gmark

Contact: Aurélien Lemay

8 New results

Participants: Antonio Al Serhali, Corentin Barloy, Iovka Boneva, Florent Capelli, Nicolas Crosetti, Aurélien Lemay, Mikael Monet, Joachim Niehren, Charles Paperman, Sylvain Salvati, Claire Soyez-Martin, Slawomir Staworko, Sophie Tison.

8.1 Querying Data

8.1.1 Circuits for Data Analysis in Artificial Intelligence

In a STACS'2023 paper, Charles Paperman, Sylvain Salvati, and their PhD student Claire Soyez-Martin develop theoretical aspects of *vectorial programming*, the combination of SIMD instructions with usual processor instructions. This new technology is known to speed-up many standard algorithms, such as for simple regular languages. Their idea is to take advantage of the inner algebraic structure of regular languages and produce high level representations of efficient vectorial programs that recognize certain classes of regular languages. As a technical ingredient, they establish equivalences between classes of vectorial circuits and logical formalisms, namely unary temporal logic and first order logic. Their main result is the construction of compilation procedures that turns syntactic semigroups into vectorial circuits. The circuits they obtain are small in that they improve known upper-bounds on representations of automata within the logical formalisms. The gain is mostly due to a careful sharing of sub-formulas based on algebraic tools.

In an LMCS journal article [14], Capelli, Crosetti and Niehren study the problem of optimizing a linear program whose variables are the answers to a conjunctive query. For this they propose the language LP(CQ) for specifying linear programs whose constraints and objective functions depend on the answer sets of conjunctive queries. They contribute an efficient algorithm for solving programs in a fragment of LP(CQ). The natural approach constructs a linear program having as many variables as there are elements in the answer set of the queries. Their approach constructs a linear program having the same optimal value but fewer variables. This is done by exploiting the structure of the conjunctive queries using generalized hypertree decompositions of small width to factorize elements of the answer set together. They then illustrate the various applications of LP(CQ) programs on three examples: optimizing deliveries of resources, minimizing noise for differential privacy, and computing the s-measure of patterns in graphs as needed for data mining.

In an article recently accepted at ICDT'24 [24], Capelli, Monet and co-authors study the problem of enumerating the satisfying assignments for circuit classes from knowledge compilation, where assignments are ranked in a specific order. In particular, they show how this problem can be used to efficiently perform ranked enumeration of the answers to MSO queries over trees, with the order being given by a ranking function satisfying a subset-monotonicity property. Assuming that the number of variables is constant, they show how to can enumerate the satisfying assignments in ranked order for so-called multivalued circuits that are smooth, decomposable, and in negation normal form (smooth multivalued DNNF). There is no preprocessing and the enumeration delay is linear in the size of the circuit times the number of values, plus a logarithmic term in the number of assignments produced so far. If one further assumes that the circuit is deterministic (smooth multivalued d-DNNF), then they show that it is possible to achieve linear-time preprocessing in the circuit, and the delay only features the logarithmic term.

In a preprint [25], Capelli and Irwin study the following problem. Given a conjunctive query Q and a database D, a direct access to the answers of Q over D is the operation of returning, given an index j, the j-th answer for some order on its answers. While this problem is #P-hard in general with respect to combined complexity, many conjunctive queries have an underlying structure that allows for a direct access to their answers for some lexicographical ordering that takes polylogarithmic time in the size of the database after a polynomial time precomputation. Previous work has precisely characterised the tractable classes and given fine-grained lower bounds on the precomputation time needed depending on the structure of the query. In this paper, they generalise these tractability results to the case of signed conjunctive queries, that is, conjunctive queries that may contain negative atoms. Their technique is based on a class of circuits that can represent relational data. They first show that this class supports tractable direct access after a polynomial time preprocessing. They then give bounds on the size of the circuit needed to represent the answer set of signed conjunctive queries depending on their structure. Both results combined together allow them to prove the tractability of direct access for a large class of conjunctive queries. On the one hand, they recover the known tractable classes from the literature in the case of positive conjunctive queries. On the other hand, they generalise and unify known tractability results about negative conjunctive queries – that is, queries having only negated atoms. In particular,

they show that the class of β-acyclic negative conjunctive queries and the class of bounded nest set width negative conjunctive queries admit tractable direct access.

8.1.2 Uncertainty and Explanations

In a JMLR (Journal of Machine Learning Research) journal article [12], Monet and co-authors study the computation of the SHAP-score. In Machine Learning, the SHAP-score is a version of the Shapley value that is used to explain the result of a learned model on a specific entity by assigning a score to every feature. While in general computing Shapley values is an intractable problem, they prove a strong positive result stating that the SHAP-score can be computed in polynomial time over deterministic and decomposable Boolean circuits. Such circuits are studied in the field of Knowledge Compilation and generalize a wide range of Boolean circuits and binary decision diagrams classes, including binary decision trees and Ordered Binary Decision Diagrams (OBDDs). They also establish the computational limits of the SHAP-score by observing that computing it over a class of Boolean models is always polynomially as hard as the model counting problem for that class. This implies that both determinism and decomposability are essential properties for the circuits that they consider. It also implies that computing SHAP-scores is intractable as well over the class of propositional formulas in DNF. Based on this negative result, they then look for the existence of fully-polynomial randomized approximation schemes (FPRAS) for computing SHAP-scores over such a class. In contrast to the model counting problem for DNF formulas, which admits an FPRAS, they prove that no such FPRAS exists for the computation of SHAP-scores. Surprisingly, this negative result holds even for the class of monotone formulas in DNF. These techniques can be further extended to prove another strong negative result: Under widely believed complexity assumptions, there is no polynomial-time algorithm that checks, given a monotone DNF formula φ and features x, y, whether the SHAP-score of x in φ is smaller than the SHAP-score of y in φ.

In a SIGMOD record survey article [13], Monet and co-authors review the use of the Shapley Value framework in Database Management. Attribution scores can be applied in data management to quantify the contribution of individual items to conclusions from the data, as part of the explanation of what led to these conclusions. Since its invention in the 1950s, the Shapley value has been used for contribution measurement in many fields, from economics to law, with its latest researched applications in modern machine learning. Recent studies investigated the application of the Shapley value to database management. Their article gives an overview of recent results on the computational complexity of the Shapley value for measuring the contribution of tuples to query answers and to the extent of inconsistency with respect to integrity constraints. More specifically, the article highlights lower and upper bounds on the complexity of calculating the Shapley value, either exactly or approximately, as well as solutions for realizing the calculation in practice.

In a preprint [26], Monet and co-authors study Shapley value computation for probabilistic databases. Shapley values, originating in game theory and increasingly prominent in explainable AI, have been proposed to assess the contribution of facts in query answering over databases, along with other similar power indices such as Banzhaf values. In this work they adapt these Shapley-like scores to probabilistic settings, the objective being to compute their expected value. They show that the computations of expected Shapley values and of the expected values of Boolean functions are interreducible in polynomial time, thus obtaining the same tractability landscape. They investigate the specific tractable case where Boolean functions are represented as deterministic decomposable circuits, designing a polynomial-time algorithm for this setting. They present applications to probabilistic databases through database provenance, and an effective implementation of this algorithm within the ProvSQL system, which experimentally validates its feasibility over a standard benchmark.

8.2 Monitoring Data Graphs

8.2.1 Query Answering on Streams

In a FCT'2023 (Fundamentals of Computation Theory) conference article [16], Al Serhali and Niehren show how to evaluate stepwise hedge automata (Shas) with subhedge projection. This requires passing finite state information top-down, so they introduce the notion of downward stepwise hedge automata.

Then use them to define an in-memory and a streaming evaluator with subhedge projection for SHAs. They then tune the streaming evaluator so that it can decide membership at the earliest time point. They apply our algorithms to the problem of answering regular XPath queries on XML streams. Their experiments show that subhedge projection of SHAs can indeed speed up earliest query answering on XML streams.

In an article in the Conference on Implementation and Application of Automata (CIAA'23) [21], Al Serhali and Niehren study the earliest query answering problem (EQA), which is the problem to enumerate certain query answers on streams at the earliest events. They consider EQA for regular monadic queries on hedges or nested words defined by deterministic stepwise hedge automata (dShas). They present an EQA algorithm for dShas that requires time $O(cm)$ per event, where m is the size of the automata and c the concurrency of the query. They show that our EQA algorithm runs efficiently on regular XPath queries in practice.

8.3 Graph Data Integration

8.3.1 Integration and Graph Mappings with Schemas and Inference

In a PODS'2023 article [18], Boneva, Staworko and others investigate graph transformations defined using Datalog-like rules based on acyclic conjunctive two-way regular path queries (acyclic C2RPQs). They study two fundamental static analysis problems: type checking and equivalence of transformations in the presence of graph schemas. Additionally, they investigate the problem of target schema elicitation, which aims to construct a schema that closely captures all outputs of a transformation over graphs conforming to the input schema. They show that all these problems are in EXPTIME by reducing them to C2RPQ containment modulo schema; and also provide matching lower bounds.

8.4 Others

Niehren is cooperating with the BioComputing team of the Cristal Lab at the University of Lille since many years. He uses abstract interpretation of logical formulas for predicting gene knockouts based on formal models of reaction network.

In an article at the International Conference on Formal Methods in Systems BiologyLuxembourg, 2023 [20], Niehren and co-authors improve on the systems biology markup language (SBML), that permits to represent biological models mixing reaction networks, algebraic equations, differential equations, and events. The main objective of this language is to exchange biological models between various tools for simulation and analysis. The specification of SBML, however, lacks a formal semantics. This makes it often difficult to understand SMBL models and to design correct and general interfaces with SBML. In that paper, they propose Core SBML, a novel language covering a large subset of SBML with clear formal semantics. They present a compiler of the delay-free fragment of SBML to Core SBML (without any formal correctness guarantees). They then show how to compile Core SBML further to BioCham while preserving the semantics. They implemented and applied our compilers to the more than 500 SBML models from the curated part of the BioModels database.

In a BIOKET'23 article [22], Niehren and co-authors study the lipopeptides produced by B. subtilis, that have a wide range of activities. For example, mycosubtilin is a strong antifungal molecule and surfactin is one of the most powerful biosurfactants known. These two bioactive molecules can be used in many markets (cosmetics, phytosanitary, detergents, ...) and their industrial production is being developed in Europe and Asia. Both molecules are made up of a cyclic heptapeptide linked to a fatty acid chain. The isomery and the length of the fatty acid chain (FA) are responsible for the activities of the lipopeptides. In this work, the activities of the different lipopeptide isoforms were first investigated for their surfactant or antifungal properties against several pathogens, revealing the strong surfactant activity of the surfactin isoform nC14 and the strong antifungal activity of the mycosubtilin isoform anteiso-C17. Then, the modification of lipopeptides production was studied by amino acid feeding experiments. In parallel, bioinformatic and metabolic engineering strategies were performed. The knockout (KO) or overexpression of genes, leading to the specific overproduction of the most active surfactin or mycosubtilin isoforms, were predicted using computational tools that provide logical reasoning with formal models of reaction networks. In this way, different genes impacting the production of amino acids

or branched-chain fatty acids or the regulation of these metabolic pathways were predicted and targeted for KO or overexpression. The influence of these deletions or these overexpression on the quantitative and qualitative lipopeptides production was analysed using HPLC, LC-MS-MS and GC-MS methods. To better understand the impact of certain deletions on the metabolism of strains and the production of their lipopeptides, two different approaches were undertaken. On the one hand, for the production of surfactin, a complete analysis of extracellular metabolites by quantitative 1H NMR was used. On the other hand, for the production of mycosubtilin, a comprehensive analysis of cellular fatty acid production was undertaken using GC-MS. The results obtained show important metabolic changes in the different mutants allowing an increase in the specific production of the nC14 isoform from 1.4 to 5.8 times and of the mycosubtilin anteiso-C17 from 1.4 to 2.8 times. These results show that increased selective synthesis of particular lipopeptide isoforms by metabolic engineering and bioinformatic-assisted synthetic biology is possible, revealing the interest of these approaches for the future development of lipopeptide-producing strains and for more targeted applications.

Niehren edited [23].

In an Information Processing Letters article [15], Staworko et al. study computational models that perform a folding operations on words of a given language, following directions coded on words of another given language. They consider the case in which both given languages are regular, and show that the class of languages generated by such F-systems is a proper subset of the class of linear context-free languages.

In a STACS'2023 article [19], Florent Capelli and Yann Strozecki introduce the technique of *geometric amortization* for enumeration algorithms. This technique can be used to make the delay of enumeration algorithms more regular without much overhead on the space it uses. More precisely, they are interested in enumeration algorithms having incremental linear delay, that is, algorithms enumerating a set A of size K such that for every $t \leq K$, it outputs at least t solutions in time $O(tp)$, where p is the incremental delay of the algorithm. While it is folklore that one can transform such an algorithm into an algorithm with delay $O(p)$, the naive transformation may blow the space exponentially. They show that, using geometric amortization, such an algorithm can be transformed into an algorithm with delay $O(p \log K)$ and $O(s \log K)$ space, where s is the space used by the original algorithm. They apply geometric amortization to show that one can trade the delay of flashlight search algorithms for their average delay modulo a factor of $O(\log K)$. They illustrate how this tradeoff may be advantageous for the enumeration of solutions of DNF formulas.

In a STACS'2023 article [17], Antoine Amarilli and Mikaël Monet study the task, for a given language L, of enumerating the (generally infinite) sequence of its words, without repetitions, while bounding the delay between two consecutive words. To allow for delay bounds that do not depend on the current word length, they assume a model where one produces each word by editing the preceding word with a small edit script, rather than writing out the word from scratch. In particular, this witnesses that the language is orderable, i.e., one can write its words as an infinite sequence such that the Levenshtein edit distance between any two consecutive words is bounded by a value that depends only on the language. For instance, the language $(a+b)^*$ is orderable (with a variant of the Gray code), but $a^* + b^*$ is not. They characterize which regular languages are enumerable in this sense, and show that this can be decided in PTIME in an input deterministic finite automaton (DFA) for the language. In fact, they show that, given a DFA A, one can compute in PTIME automata $A_1, \ldots, A_t$ such that $L(A)$ is partitioned as $L(A_1) \sqcup \ldots \sqcup L(A_t)$ and every $L(A_i)$ is orderable in this sense. Further, they show that the value of t obtained is optimal, i.e., it is not possible to partition $L(A)$ into less than t orderable languages. In the case where $L(A)$ is orderable (i.e., $t = 1$), they show that the ordering can be produced by a bounded-delay algorithm: specifically, the algorithm runs in a suitable pointer machine model, and produces a sequence of bounded-length edit scripts to visit the words of $L(A)$ without repetitions, with bounded delay – exponential in $|A|$ – between each script. In fact, they show that we can achieve this while only allowing the edit operations push and pop at the beginning and end of the word, which implies that the word can in fact be maintained in a double-ended queue.

9 Partnerships and cooperations

9.1 International initiatives

9.1.1 Participation in other International Programs

Informal International Partners

Northeastern University, USA Monet works with Wolfgang Gatterbauer and Neha Makhija from Northeastern University and with Antoine Amarilli from Télécom Paris on the resilience problem for regular path queries.

Chile and USA Monet works with researchers from Chile and USA to write a survey on Knowledge Compilation.

Warsaw, Poland Paperman cooperates with Filip Murlak on query evaluation on streams.

9.2 International research visitors

9.2.1 Visits of international scientists

Antoine Amarilli Télécom Paris. Regularly visits the team to collaborate with Paperman and with Monet. Antoine will in fact join the team in September 2024.

Michaël Cadilhac Visited the team (December), from Depaul University.

Niranjan Kumar Visited the team, in the contex of the Relax French-Indian "unité mixte" of CNRS.

Furthermore, the team organizes seminars that occur frequently on Fridays, and have invited numerous researchers over the year.

9.2.2 Visits to international teams

Research stays abroad

Monet

Visited institution: Simons Institute for the Theory of Computing

Country: USA

Dates: August 15th - December 16th 2023

Context of the visit: Monet was selected to be a fellow of the Simons Institute during its “Logic and Algorithms in Database Theory and AI” program.

Mobility program/type of mobility: Research stay

9.3 National initiatives

ANR JCJC KCODA

Participants: Florent Capelli *(correspondent)*, Charles Paperman, Sylvain Salvati.

- **Duration:** 2021–2025

- **Objectives:** The aim of KCODA is to study how succinct representations can be used to efficiently solve modern optimization and AI problems that use a lot of data. We suggest using data structures from the field of compilation of knowledge that can represent large datasets succinctly by factoring certain parts while allowing efficient analysis of the represented data. The first goal of KCODA is to understand how one can efficiently solve optimization and training problems for data represented by these structures. The second goal of KCODA is to offer better integration of these techniques into the systems of database management by proposing new algorithms allowing to build factorized representations of the data responses to DB requests and by proposing encodings of these representations inside the DB.

ANR Bravas

Participants: Sylvain Salvati *(correspondent)*.

- **Duration:** 2017–2023
- **Coordinator:** Jérôme Leroux, LaBRI, Université de Bordeaux
- **Scientific Partner:** LSV, ENS Cachan
- **Objective:** The goal of the BraVAS project is to develop a new and powerful approach to decide the reachability problems for Vector Addition Systems (VAS) extensions and to analyze their complexity. The ambition here is to crack with a single hammer (ideals over well-orders) several long-lasting open problems that have all been identified as a barrier in different areas, but that are in fact closely related when seen as reachability.

Participants: Antonio Al Serhali, Corentin Barloy, Iovka Boneva, Florent Capelli, Nicolas Crosetti, Aurélien Lemay, Mikael Monet, Joachim Niehren, Charles Paperman, Sylvain Salvati, Claire Soyez-Martin, Slawomir Staworko, Sophie Tison.

10 Dissemination

10.1 Promoting scientific activities

10.1.1 Scientific events: organisation

General chair, scientific chair

Niehren Co-chair of the 21st International Conference on Formal Methods in Systems Biology (CMSB), 2023.

10.1.2 Scientific events: selection

Member of conference program committees

Boneva Member of the ESWC 2023 Resource Track program committee.

Capelli Member of the IJCAI 2023 program committee.

Monet Member of the PODS'25 program committee.

Reviewer

Monet Reviewed in 2023 for MFCS.

10.1.3 Journal

Member of the editorial boards

Monet Managing editor for the TheoretiCS journal.

Reviewer - reviewing activities

Monet Reviewed in 2023 for JACM, LMCS and TODS.

10.1.4 Invited talks

Monet Gave invited talks in the context of the Simons program he participated in.

10.1.5 Scientific expertise

Capelli (Until September 2024) Member of Inria Lille CER (Commission des Emplois de Recherche).

Tison Member of the scientific committee of the GDR IM.

10.1.6 Research administration

Boneva Member of the steering committee of BDA (French association for research in databases).

Boneva Leader of the Sustainable Development Commission at CRIStAL.

Boneva Member of the Zero Carbon Network at University of Lille.

Paperman Elected member of the faculty counsel of Faculté des sciences et technologies de l'université de Lille.

Tison Member of the Steering Committee of Highlights of Logic, Automata, and Games.

Tison Member of the Inria Lille center committee.

10.2 Teaching - Supervision - Juries

10.2.1 Supervision

Al Serhali PhD project started 2020. On hyperstream programming. Supervised by Niehren.

Barloy PhD project started 2021. On circuits and lower complexity bounds. Supervised by Paperman and Salvati.

Soyez-Martin PhD project started 2020. On streaming with vectors and circuits. Supervised by Salvati and Paperman. Defended in December 2023.

Oliver Irwin PhD project started 2022. On compilation and aggregation in databases. Supervised by Capelli.

Crosetti PhD project started 2018. On enriching and solving linear programs with conjunctive queries. Supervised by Capelli, Niehren, and Tison. Defended in February 2023.

10.2.2 Teaching Responsibilities

Capelli (Until September 2024) Responsible for Parcoursup for LEA department, Université de Lille.

10.2.3 Teaching Activities

Boneva Teaches computer science in DUT Informatique of Université de Lille.

Capelli (Until September 2024) Teaches computer science in the LEA department of Université de Lille for around 200h per year (Licence and Master).

Lemay Teaches computer science in the LEA department of Université de Lille for around 200h per year (Licence and Master). He is also responsible for computer science and numeric correspondent for its department.

Monet Monet teaches computer science as a temporary lecturer at Université de Lille and at Centrale Lille. He teaches databases courses, as well as a course on "algorithms and complexity".

Niehren M2 Machine Learning, Université de Lille 2022-2023, Fondaments Theorique des Bases de Données.

Paperman Teaches CS in master degrees of the computer science department, Miashs at bachelor level in the maths department and in the data science master degree.

Salvati Teaches computer science for a total of around 230h per year in computer science department of Université de Lille.

10.2.4 Juries

Capelli Member (Supervisor) of the PhD committee of Nicolas Crosetti, University of Lille, February 2023.

Tison Member (Supervisor) of the PhD committee of Nicolas Crosetti, University of Lille, February 2023.

Tison President of the HdR committee of Antoine Amrilli, Institut Polytechnique of Paris, April 2023.

Tison Member of the HdR committee of Michaël Thomazo, ENS PAris, October 2023.

Tison President of the PhD committee of Claire Soyez-Martin, University of Lille, December 2023.

Tison President of the PhD committee of Guillaume Perution-Kihli, University of Montpellier, December 2023.

10.3 Popularization

10.3.1 Interventions

Niehren "École thématique CNRS", *Modélisation Formelle de Réseaux de Régulation Biologique*, 4-9th June, 2023. Course on "Interprétation abstraite de réseaux de réactions chimiques pour la prédiction de knock-out de genes".

10.3.2 Miscellaneous

Monet Monet conceived and corrected one of the computer science entrance exam for X/ENS (2023).

Paperman Oral examinator for X/ENS entrance exams.

Tison Member of the steering committee of the mentoring circle "Femmes et sciences", Université de Lille (since July 22).

11 Scientific production

11.1 Major publications

[1] A. Amarilli, L. Jachiet and C. Paperman. 'Dynamic Membership for Regular Languages'. In: ICALP. Vol. 48. International Colloquium on Automata, Languages, and Programming (ICALP 2021). Glasgow, Scotland, France, 2nd July 2021, 116:1–116:17. DOI: 10.4230/LIPIcs.ICALP.2021.116. URL: https://hal.archives-ouvertes.fr/hal-03466453.

[2] M. Arenas, P. Barceló, L. Bertossi and M. Monet. 'The Tractability of SHAP-Score-Based Explanations over Deterministic and Decomposable Boolean Circuits'. In: *Thirty-Fifth AAAI Conference on Artificial Intelligence*. Held online, France, Feb. 2021. URL: https://hal.inria.fr/hal-03147623.

[3] C. Barloy, M. Cadilhac, C. Paperman and T. Zeume. 'The Regular Languages of First-Order Logic with One Alternation'. In: LICS 2022 - 37th Annual ACM/IEEE Symposium on Logic in Computer Science. Haïfa, Israel, 2nd Aug. 2022, pp. 1–11. DOI: 10.1145/3531130.3533371. URL: https://hal.science/hal-03934389.

[4] C. Barloy, F. Murlak and C. Paperman. 'Stackless Processing of Streamed Trees'. In: *2021 PODS*. Xi'an, Shaanx, China, June 2021. DOI: 10.4230/LIPIcs. URL: https://hal.archives-ouvertes.fr/hal-03021960.

[5] I. Boneva, J. G. Labra Gayo and E. G. Prud 'hommeaux. 'Semantics and Validation of Shapes Schemas for RDF'. In: *ISWC2017 - 16th International semantic web conference*. Vienna, Austria, Oct. 2017. URL: https://hal.archives-ouvertes.fr/hal-01590350.

[6] P. Bourhis, M. Leclère, M.-L. Mugnier, S. Tison, F. Ulliana and L. Gallois. 'Oblivious and Semi-Oblivious Boundedness for Existential Rules'. In: *IJCAI 2019 - International Joint Conference on Artificial Intelligence*. Macao, China, Aug. 2019. URL: https://hal-lirmm.ccsd.cnrs.fr/lirmm-02148142.

[7] F. Capelli, N. Crosetti, J. Niehren and J. Ramon. 'Linear Programs with Conjunctive Queries'. In: 25th Internationcal Conference on Database Theory (ICDT 2022). Edinburgh, United Kingdom, 29th Mar. 2022. URL: https://hal.archives-ouvertes.fr/hal-01981553.

[8] P. D. Gallot, A. Lemay and S. Salvati. 'Linear high-order deterministic tree transducers with regular look-ahead'. In: *MFCS 2020 : The 45th International Symposium on Mathematical Foundations of Computer Science*. Andreas Feldmann, Michal Koucky and Anna Kotesovcova. Prague, Czech Republic, Aug. 2020. DOI: 10.4230/LIPIcs.MFCS.2020.34. URL: https://hal.archives-ouvertes.fr/hal-02902853.

[9] J. Niehren and M. Sakho. 'Determinization and Minimization of Automata for Nested Words Revisited'. In: *Algorithms* (Feb. 2021). URL: https://hal.inria.fr/hal-03134596.

[10] C. Paperman, S. Salvati and C. Soyez-Martin. *An algebraic approach to vectorial programs*. 27th Oct. 2022. DOI: 10.4230/LIPIcs.STACS.2023.14. URL: https://hal.archives-ouvertes.fr/hal-03831752.

[11] S. Staworko and P. Wieczorek. 'Containment of Shape Expression Schemas for RDF'. In: *SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (PODS)*. Amsterdam, Netherlands, June 2019. URL: https://hal.inria.fr/hal-01959143.

11.2 Publications of the year

International journals

[12] M. Arenas, P. Barceló, L. Bertossi and M. Monet. 'On the Complexity of SHAP-Score-Based Explanations: Tractability via Knowledge Compilation and Non-Approximability Results'. In: *Journal of Machine Learning Research* 24.63 (2023), pp. 1–58. URL: https://inria.hal.science/hal-04377324.

[13] L. Bertossi, B. Kimelfeld, E. Livshits and M. Monet. 'The Shapley Value in Database Management'. In: *SIGMOD record* 52.2 (11th Aug. 2023), pp. 6–17. DOI: 10.1145/3615952.3615954. URL: https://hal.science/hal-04377363.

[14] F. Capelli, N. Crosetti, J. Niehren and J. Ramon. 'Linear Programs with Conjunctive Database Queries'. In: *Logical Methods in Computer Science* (3rd Jan. 2024). URL: https://hal.science/hal-04317553.

[15] J. Lucero and S. Staworko. 'A note on the class of languages generated by F-systems over regular languages'. In: *Information Processing Letters* 179 (Jan. 2023), p. 106283. DOI: 10.1016/j.ipl.2022.106283. URL: https://hal.science/hal-03696314.

International peer-reviewed conferences

[16] A. Al Serhali and J. Niehren. 'Subhedge Projection for Stepwise Hedge Automata'. In: 24th International Symposium on Fundamentals of Computation Theory, FCT 2023. Trier, Germany, 18th Sept. 2023. URL: https://inria.hal.science/hal-04165835.

[17] A. Amarilli and M. Monet. 'Enumerating Regular Languages with Bounded Delay'. In: STACS. Hamburg, Germany, 2023. DOI: 10.4230/LIPIcs.STACS.2023.8. URL: https://hal.science/hal-03940590.

[18] I. Boneva, B. Groz, J. Hidders, F. Murlak and S. Staworko. 'Static Analysis of Graph Database Transformations'. In: Symposium on Principles of Database Systems. Seattle, United States, 18th June 2023. URL: https://hal.science/hal-03937274.

[19] F. Capelli and Y. Strozecki. 'Geometric Amortization of Enumeration Algorithms'. In: 40th International Symposium on Theoretical Aspects of Computer Science (STACS 2023). Hamburg, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. DOI: 10.4230/LIPIcs.STACS.2023.18. URL: https://hal.science/hal-03955911.

[20] J. Niehren, C. Lhoussaine and A. Vaginay. 'Core SBML and its Formal Semantics'. In: CMSB 2023 - 21th International Conference on Formal Methods in Systems Biology. Luxembourg, Luxembourg, 13th Sept. 2023. URL: https://inria.hal.science/hal-04125922.

Conferences without proceedings

[21] A. Al Serhali and J. Niehren. 'Earliest Query Answering for Deterministic Stepwise Hedge Automata'. In: 27th International Conference on Implementation and Application of Automata (CIAA). famagusta, Cyprus, 18th Sept. 2023. URL: https://inria.hal.science/hal-04106420.

[22] J.-S. Guez, F. Coucheney, J. Guy, M. Béchet, P. Fontanille, N.-E. Chihib, J. Niehren, F. Coutte and P. Jacques. 'Bioinformatics Modelling and Metabolic Engineering of the Branched Chain Amino Acid Pathway for Specific Production of Microbial Biosurfactants and Biopesticides'. In: BIOKET. Vol. 12. 2. Montreal, Canada, 16th Aug. 2023, p. 107. DOI: 10.3390/metabo12020107. URL: https://hal.science/hal-04278771.

Edition (books, proceedings, special issue of a journal)

[23] J. Pang and J. Niehren, eds. *Computational Methods in Systems Biology - 21st International Conference,CMSB 2023*. Vol. 14137. Lecture Notes in Computer Science. Springer Nature Switzerland; Springer, 11th Sept. 2023. DOI: 10.1007/978-3-031-42697-1. URL: https://inria.hal.science/hal-04325364.

Reports & preprints

[24] A. Amarilli, P. Bourhis, F. Capelli and M. Monet. *Ranked Enumeration for MSO on Trees via Knowledge Compilation*. 1st Oct. 2023. DOI: 10.4230/LIPIcs. URL: https://inria.hal.science/hal-04377344.

[25] F. Capelli and O. Irwin. *Direct Access for Conjunctive Queries with Negation*. 24th Oct. 2023. URL: https://hal.science/hal-04260203.

[26] P. Karmakar, M. Monet, P. Senellart and S. Bressan. *Expected Shapley-Like Scores of Boolean Functions: Complexity and Applications to Probabilistic Databases*. 12th Jan. 2024. URL: https://inria.hal.science/hal-04393781.