Report from Dagstuhl Seminar 19211

Enumeration in Data Management

Edited by

Endre Boros[1], Benny Kimelfeld[2], Reinhard Pichler[3], and Nicole Schweikardt[4]

1 **Rutgers University – Piscataway, US,** endre.boros@rutgers.edu
2 **Technion – Haifa, IL,** bennyk@cs.technion.ac.il
3 **TU Wien, AT,** pichler@dbai.tuwien.ac.at
4 **HU Berlin, DE,** schweikn@informatik.hu-berlin.de

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 19211 "Enumeration in Data Management". The goal of the seminar was to bring together researchers from various fields of computer science, including the Databases, Computational Logic, and Algorithms communities, and establish the means of collaboration towards considerable progress on the topic. Specifically, we aimed at understanding the recent developments, identifying the important open problems, and initiating collaborative efforts towards solutions thereof. In addition, we aimed to build and disseminate a toolkit for data-centric enumeration problems, including algorithmic techniques, proof techniques, and important indicator problems. Towards the objectives, the seminar included tutorials on the topic, invited talks, presentations of open problems, working groups on the open problems, discussions on platforms to compile the community knowledge, and the construction of various skeletons of such compilations.

Seminar May 19–24, 2019 – http://www.dagstuhl.de/19211
2012 ACM Subject Classification Mathematics of computing → Combinatorial algorithms, Mathematics of computing → Enumeration, Theory of computation → Complexity classes, Theory of computation → Data structures and algorithms for data management, Theory of computation → Database query processing and optimization (theory)
Keywords and phrases constant delay, databases, dynamic complexity, enumeration, polynomial delay, query evaluation
Digital Object Identifier 10.4230/DagRep.9.5.89
Edited in cooperation with Nofar Carmeli

1 Executive Summary

Endre Boros
Benny Kimelfeld
Reinhard Pichler
Nicole Schweikardt

In recent years, various concepts of enumeration have arisen in the fields of Databases, Computational Logic, and Algorithms, motivated by applications of data analysis and query evaluation. Common to all concepts is the desire to compute a stream of items with as small as possible waiting time between consecutive items, referred to as the "delay." Alongside each concept, there evolved algorithmic techniques for developing solvers, and proof techniques for establishing complexity bounds. In addition to the traditional guarantees of "polynomial delay" and "incremental polynomial," researchers have been pursuing stronger guarantees such as "constant delay" in the context of logical query evaluation, "dynamic complexity" of incremental maintenance, and "factorized databases." The growing interest and rapid evolution of the associated research brings up opportunities of significantly accelerating the computation of big results, by devising and adopting general-purpose methodologies.

In Dagstuhl Seminar 19211 on "Enumeration in Data Management," key researchers from relevant communities have gathered to gain a better understanding the recent developments, lay out the important open problems, and join forces towards solutions thereof. These communities include researchers who explore enumeration problems in the fields of *databases*, *logic*, *algorithms* and *computational complexity*. We have had invited tutorials by

- Luc Segoufin on *Constant-delay enumeration*
- Takeaki Uno on *Enumeration algorithms*
- Yann Strozecki on *Enumeration complexity – defining tractability*
- Markus Kröll on *Enumeration complexity – a complexity theory for hard enumeration problems*
- Endre Boros on *Monotone generation problems.*

We also had presentations by most of the other participants. Moreover, the participants have prepared in advance a list of open problems in a document that we shared and jointly maintained. We have discussed the open problems during designated times of the seminar.

The organizers are highly satisfied with the seminar. We have got a very high acceptance rate for our invitations. In fact, there were further researchers whom we would have liked to invite after the first invitation round but, unfortunately, no room was left. The participants were exceptionally involved and engaged. Some considerable progress has been made on the open problems prepared in advance, as will be reported in future publications that will acknowledge the seminar. The seminar has also initiated joint efforts to disseminate toolkits for data-centric enumeration problems, including algorithmic techniques, proof techniques, and important indicator problems. To this end, we have had sessions of working groups for the different types of toolkit components. In particular, we have initiated a Wikipedia page on enumeration algorithms:

https://en.wikipedia.org/wiki/Enumeration_algorithm

This page will evolve to contain a thorough picture of the principles and techniques of enumeration problems.

2 Table of Contents

Open problems

3 Overview of Talks

3.1 Hardness for Polynomial Time Problems

Amir Abboud (IBM Almaden Center – San Jose, US)

This will be an overview of recent results in Fine-Grained Complexity. A small set of conjectures about the exact time complexity of certain core problems (such as SAT and 3SUM) are used to derive strong conditional lower bounds for many many other problems. The focus of the talk will be on the conjectures and techniques that are (or may be) of interest to the enumeration algorithms community. In particular, we will highlight the k-Clique conjectures.

3.2 A Circuit-Based Approach to Efficient Enumeration: Enumerating MSO Query Results on Trees and Words

Antoine Amarilli (Telecom ParisTech, FR)

Joint work of Antoine Amarilli, Pierre Bourhis, Louis Jahiet, Stefan Mengel, Matthias Niewerth

This talk presents our circuit-based approach to enumerate the results of monadic second-order (MSO) queries on trees. Specifically, we explain how a deterministic tree automaton encoding an MSO query can be translated to a d-DNNF set circuit representing its answers on an input tree. These answers can then be enumerated with linear-time preprocessing and constant-delay using our algorithm in [1]. The talk also explains how our methods can be extended to work with nondeterministic automata and ensure combined tractability, i.e., ensure that the preprocessing and delay are polynomial in the automaton, following our upcoming results in [4]. We last explain how these methods apply to the problem of enumerating tractably the matches of regular expressions (possibly with captures) on a text document, using our algorithm from [3], and show a demo of a preliminary implementation for this task.

References

1. Antoine Amarilli, Pierre Bourhis, Louis Jachiet, Stefan Mengel. A Circuit-Based Approach to Efficient Enumeration. ICALP 2017. https://arxiv.org/abs/1702.05589
2. Antoine Amarilli, Pierre Bourhis, Stefan Mengel. Enumeration on Trees under Relabelings. ICDT 2018. https://arxiv.org/abs/1709.06185
3. Antoine Amarilli, Pierre Bourhis, Stefan Mengel, Matthias Niewerth. Constant-Delay Enumeration for Nondeterministic Document Spanners. ICDT 2019. https://arxiv.org/abs/1807.09320
4. Antoine Amarilli, Pierre Bourhis, Stefan Mengel, Matthias Niewerth. Enumeration on Trees with Tractable Combined Complexity and Efficient Updates. PODS 2019. https://arxiv.org/abs/1812.09519

3.3 Constant delay enumeration of CQs with FPT-preprocessing

Christoph Berkholz (HU Berlin, DE)

Joint work of Christoph Berkholz, Nicole Schweikardt

In this talk I discuss for which classes of conjunctive queries (CQs) it is possible to enumerate the query result with constant delay after FPT-preprocessing (where the CQ is the parameter). While a general classification theorem is still missing, I present a dichotomy for classes of self-join-free and quantifier-free CQs.

The talk is based on an ongoing joint work with Nicole Schweikardt.

3.4 Monotone Generation Problems

Endre Boros (Rutgers University – Piscataway, US)

This tutorial talk provides an overview of monotone generation problems, the core problem of generating hypergraph transversals, and related techniques and results.

The first part introduces general techniques and results on hypergraph (monotone Boolean) dualization, the crucial task of finding a next output element, and known results about its complexity and hardness in special cases.

The second part surveys some of the standard techniques (that are actualloy useful even for non-monotone generation problems). These techniques include the Flashlight Principle (promising polynomial delay generation), the Supergraph Approach (promising incremental polynomial generation or faster), the Projection Method (that generally guarantees total polynomial time generation) and the method of Joint Generation (that provides a framework for incrementally quasi- polynomial generation). Each of these techniques can be made to work of course only if certain conditions are fulfilled.

The third part illustrates some of these techniques on problems related to/derived from the problem of generating vertices of polyhedra, leading to a special case that turns out to be NP-hard.

The talk closes with a section on open problems formulated in the very general setting of generating simplices and bodies for a given point set in a euclidian space.

3.5 Enumeration Complexity of UCQs

Nofar Carmeli (Technion – Haifa, IL) and Markus Kröll (TU Wien, AT)

Main reference Nofar Carmeli, Markus Kröll: "On the Enumeration Complexity of Unions of Conjunctive Queries", in Proc. of the 38th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2019, Amsterdam, The Netherlands, June 30 – July 5, 2019., pp. 134–148, ACM, 2019.
URL https://doi.org/10.1145/3294052.3319700

We study the enumeration complexity of Unions of Conjunctive Queries (UCQs). We aim to identify the UCQs that are tractable in the sense that the answer tuples can be enumerated with linear preprocessing time and constant delay. A union of tractable CQs is always tractable. We show that some non-redundant unions containing intractable CQs are tractable. Interestingly, some unions consisting of only intractable CQs are tractable too. The question of finding a full characterization of the tractability of UCQs remains open. We end the talk with open problems, and describe some examples of specific queries for which we have no classification.

3.6 An Optimization-based Primer on Flag Algebras

Aritanan Gruber (University of ABC – Santo André, BR)

Joint work of Aritanan Gruber, Marcel K. de Carli Silva, Fernando Mário de Oliveira Filho, Cristiane Maria Sato

Despite all the progress made in the past two decades, a unifying theory of enumeration algorithms seems, by any reasonable matter, a distance milestone – if achievable at all. In the mean time, we search for tools and inspiration in the slightly more developed settings of counting and density estimation. The theory of flag algebras offers a systematic approach to derive computer-assisted proofs of density estimation results in asymptotic extremal combinatorics. We briefly survey the theory in an optimization-based approach and then focus on a conic programming strong duality relation lying at its core, providing simpler proofs along the way. At the end, we mention possible extensions and ramifications of our approach.

3.7 Parameterized Enumeration

Heribert Vollmer (Leibniz Universität Hannover, DE)

Joint work of Nadia Creignou, Arne Meier, Julian-Steffen Müller, Johannes Schmidt, Heribert Vollmer
Main reference Nadia Creignou, Arne Meier, Julian-Steffen Müller, Johannes Schmidt, Heribert Vollmer: "Paradigms for Parameterized Enumeration", Theory Comput. Syst., Vol. 60(4), pp. 737–758, 2017.
URL https://doi.org/10.1007/s00224-016-9702-4

We introduce parameterized classes for enumeration problems. Particular interest is devoted to the class DelayFPT. We prove some structural results about this class, in particular we obtain a characterization in terms of kernelization. We also give a number of upper and lower bounds for specific problems from graph theory and propositional logic.

References

1 Nadia Creignou, Arne Meier, Julian-Steffen Müller, Johannes Schmidt, and Heribert Vollmer. Paradigms for parameterized enumeration. *Theory Comput. Syst.*, 60(4):737–758, 2017.

3.8 Some Generalizations of the Monotone Boolean Duality Testing (Hypergraph Transversal) Problem

Khaled Elbassioni (Khalifa University – Abu Dhabi, AE)

Joint work of Khaled Elbassioni, Leonid Khachiyan, Endre Boros, Vladimir Gurvich, Kazuhisa Makino

Given two sets A and B of binary vectors such that no vector in A is dominated by a vector in B, the well-known monotone Boolean duality testing problem calls for checking if A and B monotonically cover the entire binary cube, that is, if every vertex of the cube dominates some vector in A or is dominated by some vector in B. In this talk, we consider two generalizations of monotone Boolean duality testing problem. In the first generalization, we are given a hypergraph where each hyperedge intersects all but a small number of other hyperedges, and a list of colors for each vertex, and the requirement is to check if each vertex can be assigned a color from its list such that no hyperedge is monochromatic. In the second generalization, the binary cube and families A and B are replaced by a box and vectors over an integer lattice, and the requirement again is to check if A and B monotonically cover all the lattice points inside the box. We illustrate that the first generalization can be solved in quasi-polynomial time and give an application of the second generalization to finding sparse regions in multi-dimensional data.

References

1 Khaled M. Elbassioni. Quasi-polynomial algorithms for list-coloring of nearly intersecting hypergraphs. *CoRR*, abs/1904.02425, 2019.

2 Leonid Khachiyan, Endre Boros, Khaled M. Elbassioni, Vladimir Gurvich, and Kazuhisa Makino. Dual-bounded generating problems: Efficient and inefficient points for discrete probability distributions and sparse boxes for multidimensional data. *Theor. Comput. Sci.*, 379(3):361–376, 2007.

3.9 Listing Maximal Subgraphs Satisfying Strongly Accessible Properties

Andrea Marino (Univerisità degli Studi di Firenze, IT)

Joint work of Alessio Conte, Roberto Grossi, Andrea Marino, Luca Versari
Main reference Alessio Conte, Roberto Grossi, Andrea Marino, Luca Versari: "Listing Maximal Subgraphs Satisfying Strongly Accessible Properties", SIAM J. Discrete Math., Vol. 33(2), pp. 587–613, 2019.
URL https://doi.org/10.1137/17M1152206

Algorithms for listing the subgraphs satisfying a given property (e.g., being a clique, a cut, a cycle, etc.) fall within the general framework of set systems. A set system $(\mathcal{U}, \mathcal{F})$ consists of a ground set $\mathcal{U}$ (e.g., a network's nodes) and a family $\mathcal{F} \subseteq 2^{\mathcal{U}}$ of subsets of $\mathcal{U}$ that have the required property. For the problem of listing all sets in $\mathcal{F}$ maximal under inclusion, the ambitious goal is to cover a large class of set systems, preserving at the same time the efficiency of the enumeration. Among the existing algorithms, the best-known ones list the maximal subsets in time proportional to their number but may require exponential space. In this talk, we show how to improve the state of the art in two directions by introducing an algorithmic framework based on reverse search that, under standard suitable conditions, simultaneously (i) extends the class of problems that can be solved efficiently to *strongly accessible* set systems, and (ii) reduces the additional space usage from exponential in $|\mathcal{U}|$ to *stateless*, i.e., with no additional memory usage other than that proportional to the solution size, thus accounting for just polynomial space.

References

1 Alessio Conte, Roberto Grossi, Andrea Marino, Luca Versari. Listing Maximal Subgraphs Satisfying Strongly Accessible Properties. SIAM Journal Discrete Mathematics. 2019

3.10 Dichotomies for Evaluation and Enumeration Problems for Simple Regular Path Queries

Wim Martens (Universität Bayreuth, DE)

Joint work of Wim Martens, Tina Trautner
Main reference Wim Martens, Tina Trautner: "Evaluation and Enumeration Problems for Regular Path Queries", in Proc. of the 21st International Conference on Database Theory, ICDT 2018, March 26-29, 2018, Vienna, Austria, LIPIcs, Vol. 98, pp. 19:1–19:21, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2018.
URL http://dx.doi.org/10.4230/LIPIcs.ICDT.2018.19

Regular path queries (RPQs) are a central component of graph databases. We investigate decision and enumeration problems concerning the evaluation of RPQs under several semantics that have recently been considered: arbitrary paths, shortest paths, paths without node repetitions (simple paths), and paths without edge repetitions (trails). Whereas arbitrary and shortest paths can be dealt with efficiently, simple paths and trails become computationally difficult already for very small RPQs. We study RPQ evaluation for simple paths and trails from a parameterized complexity perspective and define a class of simple transitive expressions that is prominent in practice and for which we can prove dichotomies for the evaluation problem. We observe that, even though simple path and trail semantics are intractable for RPQs in general, they are feasible for the vast majority of RPQs that are used in practice. At the heart of this study is a result of independent interest: the two disjoint paths problem in directed graphs is W[1]-hard if parameterized by the length of one of the two paths.

References

1 Wim Martens, Tina Trautner. *Evaluation and Enumeration Problems for Regular Path Queries*. International Conference on Database Theory (ICDT), 2018: 19:1-19:21.

3.11 A Circuit-Based Approach to Efficient Enumeration

Stefan Mengel (CNRS, CRIL – Lens FR)

Joint work of Antoine Amarilli, Pierre Bourhis, Louis Jachiet, Stefan Mengel, Matthias Niewerth
Main reference Antoine Amarilli, Pierre Bourhis, Louis Jachiet, Stefan Mengel: "A Circuit-Based Approach to Efficient Enumeration", in Proc. of the 44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland, LIPIcs, Vol. 80, pp. 111:1–111:15, Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2017.
URL http://dx.doi.org/10.4230/LIPIcs.ICALP.2017.111

In this talk, we present a framework for efficient enumeration that is based on representations by circuits. We show that for a certain type of circuits that we call d-DNNF set circuits we can enumerate the sets computed by a circuit with constant delay and linear preprocessing. Combined with the fact that several query answering problems from database theory allow efficient representations by these set circuits, this yields a general technique for constant delay enumeration in that area.

References

1 Amarilli, A., Bourhis, P., Jachiet, L., and Mengel, S. A Circuit-Based Approach to Efficient Enumeration. ICALP 2017.
2 Amarilli, A., Bourhis, P., and Mengel, S. Enumeration on Trees under Relabelings. ICDT 2018.
3 Niewerth, M. MSO Queries on Trees: Enumerating Answers under Updates Using Forest Algebras. LICS 2018.
4 Amarilli, A., Bourhis, P., Mengel, S., and Niewerth, M. Constant-Delay Enumeration for Nondeterministic Document Spanners. ICDT 2019.
5 Amarilli, A., Bourhis, P., Mengel, S., and Niewerth, M. Enumeration on Trees with Tractable Combined Complexity and Efficient Updates. PODS 2019, to appear

3.12 Complex Event Processing and Efficient Enumeration

Cristian Riveros (Pontificia Universidad Catolica de Chile, CL)

Complex Event Processing (CEP) is a unifying field of technologies for processing and correlating distributed data sources in real-time. CEP finds applications in diverse domains, which has resulted in a large number of proposals for expressing and processing complex events. However, although these technologies have reached a good level of maturity, existing CEP systems still lack of general techniques of query evaluation with strong performance guarantees, regarding the update time per event and enumeration of complex events.

In this talk, I will present our recent proposal for processing complex events [1]. I will start by presenting our framework of a query language and automata model for defining and compiling CEP queries with local predicates, a subset of queries that are in the core of CEP. Then I will explain our evaluation strategy for processing this subset of queries with strong performance guarantees, namely, constant update time per event and constant delay enumeration of complex events.

References

1 Alejandro Grez, Cristian Riveros, and Martín Ugarte. A formal framework for complex event processing. In *ICDT*, volume 127 of *LIPIcs*, pages 5:1–5:18. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2019.

3.13 Tutorial on Enumeration Complexity: Defining Tractability

Yann Strozecki (University of Versailles, FR)

Joint work of Yann Strozecki, Florent Capelli, Arnaud Mary
Main reference Florent Capelli, Yann Strozecki: "Incremental delay enumeration: Space and time", Discrete Applied Mathematics, 2018.
URL https://doi.org/10.1016/j.dam.2018.06.038

We review the different ways tractability is defined in enumeration using as complexity measure total time, incremental time, delay and space. We present the associated complexity classes and show how they relate and can be separated modulo classical complexity hypothesis. The focus is on understanding incremental polynomial time and polynomial delay which are the classes where the majority of problems are. In particular, we present an attempt to classify which saturation problems naturally in incremental polynomial time are in fact in polynomial delay. We also briefly present low complexity classes and further desirable properties for practitioners: constant delay and strong polynomial delay and show related results and conjectures on the enumeration of models of a DNF.

References

1 Florent Capelli and Yann Strozecki. Incremental delay enumeration: Space and time. *Discrete Applied Mathematics*, 2018.

3.14 Pattern mining with MDL

Alexandre Termier (IRISA – University of Rennes, FR)

Joint work of Alexandre Termier, Jilles Vreeken, Matthijs van Leeuwen, Peggy Cellier, Esther Galbrun, Clement Gautrais, Bruno Crémilleux
Main reference Jilles Vreeken, Matthijs van Leeuwen, Arno Siebes: "Krimp: mining itemsets that compress", Data Min. Knowl. Discov., Vol. 23(1), pp. 169–214, 2011.
URL https://doi.org/10.1007/s10618-010-0202-x

Pattern mining is the field of data mining concerned with discovering (local) regularities in data. Pattern mining algorithms solve combinatorial problems. As such, they are built around an enumeration algorithm, combined with an efficient way to check that the "pattern property" holds into the data (which can be seen as an oracle for the enumeration algorithm). While this approach has been extremely popular at the beginning of the century, the exponential size of its output has increasingly turned off its potential users. Actual data scientists expect to get a synthetic representation of the main patterns occurring of the data. A promising line of research, led by Vreeken and van Leeuwen, proposed to exploit compression techniques, namely the Minimum Description Length (MDL) principle, to select a small set of patterns that are especially good representative of the data. In this talk, we introduce the main ideas of mining patterns based on the MDL principle, and briefly sketch questions that may be of interest for the enumeration community.

3.15 Enumeration of Maximum Cliques and Its Application to Coding Theory

Etsuji Tomita (The University of Electro-Communications – Tokyo, JP)

Joint work of Takayuki Nozaki, Akira Mitsutake
URL http://www.etlab.lab.uec.ac.jp/~tomita/index_e.html

We present an algorithm for enumerating all maximum cliques of a graph and its application to error correcting codes. First, we review our depth-first search algorithm CLIQUES for enumerating all maximal cliques of a graph in which pruning methods are employed as in the Bron-Kerbosch algorithm. Subsequently, we show step by step modification and improvements to the previous algorithm in order to obtain an algorithm for finding a maximum clique. We have employed techniques from our maximum-clique-finding algorithms MCQ, MCR, MCS, MCT, and some others. We note that an algorithm for enumerating all maximum cliques can be easily obtained from an algorithm for finding a maximum clique. Finally, we deal with coding theory, especially with single deletion correcting codes. The mathematically defined VT (Varshamov – Teneholtz) code is well known to be a single deletion correcting code, but many important problems remain unsolved. We show that largest single deletion correcting codes can be enumerated through enumeration of maximum cliques. Then we present new findings in non-binary single deletion correcting codes by way of enumeration of maximum cliques.

References

1 Etsuji Tomita. Efficient algorithms for finding maximum and maximal cliques and their applications. In *WALCOM*, volume 10167 of *Lecture Notes in Computer Science*, pages 3–15. Springer, 2017.
2 Akira Mitsutake, Takayuki Nozaki, and Etsuji Tomita. Construction of best non-binary single deletion correcting codes via maximum clique enumeration. In *41st Symposium on Information Theory and Its Applications*, Iwaki, Fukushima, Japan, 2018.

3.16 OBDD and Interpretability in Machine Learning

György Turan (University of Illinois – Chicago, US)

Joint work of György Turan, Karine Chubarian

The comprehensibility of models produced using machine learning methods is an important requirement in many applications. Comprehensibility, or interpretability, can refer either to a human user, or to a system using the model as a component. One approach is to construct an interpretable representation of a learned model. For naive Bayesian network classifiers, Chan and Darwiche proposed ordered binary decision diagrams (OBDD) as an exact interpretable representation. As this representation may be of exponential size, we consider approximate representations. It is shown that for tree-augmented naive Bayes classifiers, approximate OBDD representations of polynomial size can be computed efficiently. The algorithm generalizes an algorithm of Gopalan, Klivans and Meka for approximate counting of knapsack solutions to a class of quadratic polynomial threshold functions.

3.17 Constant-Delay Enumeration applied to Dynamic Query Processing: The Dynamic Yannakakis Algorithm

Stijn Vansummeren (Free University of Brussels, BE)

Joint work of Muhammad Idris, Martín Ugarte, Stijn Vansummeren, Hannes Voigt, Wolfgang Lehner
Main reference Muhammad Idris, Martín Ugarte, Stijn Vansummeren, Hannes Voigt, Wolfgang Lehner: "Conjunctive Queries with Inequalities Under Updates", PVLDB, Vol. 11(7), pp. 733–745, 2018.
URL https://doi.org/10.14778/3192965.3192966

The ability to efficiently analyze changing data is a key requirement of many real-time analytics applications like Stream Processing, Complex Event Recognition, Business Intelligence, and Machine Learning.

Traditional approaches to this problem are based either on the materialization of subresults (to avoid their recomputation) or on the recomputation of subresults (to avoid the space overhead of materialization). Both techniques have recently been shown suboptimal: instead of fully materializing results and subresults, one can maintain a data structure that supports efficient maintenance under updates and can quickly enumerate the full query output, as well as the changes produced under single updates.

In our work we are concerned with designing a practical family of algorithms for dynamic query evaluation based on this idea, for queries featuring both equi-joins and inequality joins, as well as certain forms of aggregation. Our main insight is that, for acyclic conjunctive queries, such algorithms can naturally be obtained by modifying Yannakakis' seminal algorithm for processing acyclic joins in the static setting.

This talk presents the main ideas behind this modification, offsets it against the traditional ways of doing incremental view maintenance, and discusses recent extensions such as dealing with general theta-joins.

References

1 Muhammad Idris, Martín Ugarte, Stijn Vansummeren, Hannes Voigt, and Wolfgang Lehner. Conjunctive queries with inequalities under updates. *PVLDB*, 11(7):733–745, 2018.
2 Muhammad Idris, Martín Ugarte, and Stijn Vansummeren. The dynamic Yannakakis algorithm: Compact and efficient query processing under updates. In *SIGMOD Conference*, pages 1259–1274. ACM, 2017.

3.18 Dynamic Complexity of Reachability

Thomas Zeume (TU Dortmund, DE)

Joint work of Thomas Zeume, Samir Datta, Anish Mukherjee, Raghav Kulkarni, Thomas Schwentick, Nils Vortmeier
Main reference Samir Datta, Raghav Kulkarni, Anish Mukherjee, Thomas Schwentick, Thomas Zeume: "Reachability Is in DynFO", J. ACM, Vol. 65(5), pp. 33:1–33:24, 2018.
URL https://doi.org/10.1145/3212685

Dynamic descriptive complexity theory studies how query results can be updated in a highly parallel fashion, that is, by constant-depth circuits or, equivalently, by first-order formulas, or by the relational algebra. After gently introducing dynamic complexity theory, I will discuss recent results regarding the dynamic complexity of the reachability query.

4 Open problems

4.1 Conjunctive Queries with FPT Preprocessing

Christoph Berkholz (HU Berlin, DE)

Bagan et al. [1] showed (under some algorithmic assumptions) that the result $q(D)$ of a self-join-free CQ q over a database D of size N can be enumerated with constant delay after $O(f(q) \cdot N)$ preprocessing time if and only if q is free-connex acyclic. Here, a self-join-free CQ is a query of the form $\exists \overline{x}_0 \bigwedge_{i=1}^{k} R_i(\overline{x}_i)$ for pairwise distinct relation symbols R_i. An open problem is to characterise those CQs that allow constant delay after FPT preprocessing, which means $O(f(q) \cdot N^c)$ preprocessing time for some fixed c. More precisely, let Φ be a recursively enumerable class of self-join-free CQs and consider the following problem:

Problem:	*CQ-Enum(Φ)*
Parameters:	A fixed recursively enumerable class Φ of self-join-free CQs.
Input:	A CQ $q \in \Phi$ and a database D.
Goal:	After some preprocessing, enumerate all tuples of $q(D)$ with constant delay.

The open problem is: for which (recursively enumerable classes of self-join-free CQs) Φ exist a constant c and a computable function f such that CQ-Enum($\mathcal{C}$) can be solved with $O(f(q) \cdot N^c)$ preprocessing? Let us call such classes *tractable*.

It is known that a class Φ is tractable, if it has bounded free-connex treewidth [1], bounded free-connex fractional hypertree-width (i. e. in [2]), or bounded free-connex submodular width [Berkholz, Schweikardt; unpublished]. For join queries (i. e. quantifier free CQs) it then follows from Marx' lower bound [3], that, assuming the exponential time hypothesis (ETH), a class Φ is tractable if and only if Φ has bounded submodular width. However, a full classification for classes of CQs with projections is missing. An intriguing example is the *k-star query with a quantified center*, i. e., the query $\psi_k = \exists z \bigwedge_{i=1}^{k} R_i(z, x_i)$. It is open whether the class $\Psi^{\text{star}} = \{\psi_k : k \in \mathbb{N}\}$ is tractable, that is, whether CQ-Enum(Ψ^{star}) can be solved with $O(f(k) \cdot N^c)$ preprocessing and constant delay over databases of size N.

References

1 Guillaume Bagan, Arnaud Durand, and Etienne Grandjean. On acyclic conjunctive queries and constant delay enumeration. In *Computer Science Logic, 21st International Workshop, CSL 2007, 16th Annual Conference of the EACSL, Lausanne, Switzerland, September 11-15, 2007, Proceedings*, pages 208–222, 2007.

2 Dan Olteanu and Jakub Závodný. Size bounds for factorised representations of query results. *ACM Trans. Database Syst.*, 40(1):2:1–2:44, 2015.

3 Dániel Marx. Tractable hypergraph properties for constraint satisfaction and conjunctive queries. *J. ACM*, 60(6):42:1–42:51, 2013.

4.2 Complexity of Enumeration versus Complexity of Counting

Etienne Grandjean (Caen University, FR)

It seems that in general, counting the solutions of a problem is at least as difficult as (and often much harder than) enumerating the solutions of the problem.

E.g., the solutions of problems *2-SAT* and *Perfect-Matching* are enumerable with delay $O(n)$ in the size n of each output solution. However, the corresponding counting problems are ♯P-complete.

Question

Is this "assertion" that we have the order Enumeration $\leq$ Counting (for complexity) a general phenomenon? If yes, to what extent and with what explanation (maybe informal)?

(I strongly think that this inequality holds for logical/query problems. E.g., it holds for Boolean CSP problems [1, 2] where affine queries are the only ones for which the counting problem is easy.)

Opposite challenging question

Can one exhibit a "natural" problem (in graph theory, combinatorics, etc.) which would be a counterexample to the previous assertion, that is a problem for which it is easy to count the solutions but hard (or harder) to enumerate them?

References

1 Nadia Creignou and Miki Hermann. Complexity of generalized satisfiability counting problems. *Inf. Comput.*, 125(1):1–12, 1996.
2 Nadia Creignou and Jean-Jacques Hébrard. On generating all solutions of generalized satisfiability problems. *ITA*, 31(6):499–511, 1997.

4.3 Maximal Ptime Graph Properties

Mamadou Moustapha Kanté (Université Clermont Auvergne – Aubiere, FR)

Consider the following question:

Problem:	*Maximal Ptime Graph Property*
Parameters:	A Ptime property
Input:	A graph G
Goal:	Enumerate all maximal (induced) subgraphs satisfying the property

If the property is defined with a list of finite induced subgraph obstructions, one can reduce the problem to the enumeration of minimal transversals in hypergraphs of bounded dimension, resulting in an incremental enumeration algorithm. Some natural questions:

1. For properties with finite induced obstructions, can we have a polynomial delay enumeration algorithm?

2. For which Ptime properties the enumeration is incremental? Preferred examples are minor-closed classes of graphs or generally graphs with a short description of obstructions. Short can be: a finite list of (infinite) patterns (like chordal graphs) or finite list wrt a quasi-order (like minor relation).

4.4 Enumerating Minimal Triangulations in Polynomial Delay

Batya Kenig (University of Washington – Seattle, US)

A graph is triangulated, or chordal, if every cycle of four or more vertices has a chord. Edges can be added to any given graph so that the resulting graph, called a triangulation of the input graph, is chordal. Carmeli et al. [1] have shown that the minimal triangulations of a graph (with regard to inclusion) can be enumerated in incremental polynomial delay. It remains open whether the minimal triangulations of a graph can be enumerated in polynomial delay.

Some observations that point to a negative answer to this question are the following. Deciding whether there exists a minimal triangulation that excludes some set of edges is NP-complete by reduction from the chordal graph sandwich problem of Golumbic et al. [2]. This rules out the possibility of a polynomial delay algorithm by naive backtrack search. It can also be shown that using the framework of Carmeli et al. [1] (i.e., enumerating the maximal independent sets for *succinct graph representations*) cannot lead to a polynomial delay algorithm assuming the Strong Exponential Time Hypothesis (SETH).

References

1 Nofar Carmeli, Batya Kenig, and Benny Kimelfeld. Efficiently enumerating minimal triangulations. In *Proceedings of the 36th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2017, Chicago, IL, USA, May 14-19, 2017*, pages 273–287, 2017.

2 Martin Charles Golumbic, Haim Kaplan, and Ron Shamir. Graph sandwich problems. *J. Algorithms*, 19(3):449–473, 1995.

4.5 Evaluating Automata with Capture Variables

Benny Kimelfeld (Technion – Haifa, IL)

A *variable-set automaton* (vset-automaton for short) is an automaton that can open and close variables along its run on a string, so that each variable can be opened and later closed at most once. (So, a vset-automaton is a special case of a transducer.) Therefore, a successful run defines an assignment of spans (intervals from the input string) to variables. We refer to such an assignment as a *query answer*. These automata have been studied by Fagin et al. [1] in the framework of document spanners for information extraction, and they required that a run assigns a span to *every* variable. Maturana et al. [2] have studied the variant where every variable is assigned *at most* once, hence we get *partial* query answers.

Problem:	*Maximal Answers for Vset-Automata*
Parameters:	
Input:	A string $\mathbf{s}$, a vset-automaton A.
Goal:	Enumerate the *maximal* answers of $A(\mathbf{s})$.

The query answers can be enumerated with polynomial delay under both the complete and incomplete semantics [3, 4, 2]. The above problem is about the *maximal* (rather than *all partial*) answers can also be enumerated with polynomial delay.

References

1 Ronald Fagin, Benny Kimelfeld, Frederick Reiss, and Stijn Vansummeren. Document spanners: A formal approach to information extraction. *J. ACM*, 62(2):12:1–12:51, 2015.
2 Francisco Maturana, Cristian Riveros, and Domagoj Vrgoc. Document spanners for extracting incomplete information: Expressiveness and complexity. In *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, Houston, TX, USA, June 10-15, 2018*, pages 125–136, 2018.
3 Antoine Amarilli, Pierre Bourhis, Stefan Mengel, and Matthias Niewerth. Constant-delay enumeration for nondeterministic document spanners. In *22nd International Conference on Database Theory, ICDT 2019, March 26-28, 2019, Lisbon, Portugal*, pages 22:1–22:19, 2019.
4 Dominik D. Freydenberger, Benny Kimelfeld, and Liat Peterfreund. Joining extractions of regular expressions. In *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, Houston, TX, USA, June 10-15, 2018*, pages 137–149, 2018.

4.6 Ordered Query Evaluation with Sublinear Delay

Benny Kimelfeld (Technion – Haifa, IL)

Problem:	*Ordered Query Evaluation*
Parameters:	A query $q(x_1, \ldots, x_k)$, a partial order $\succeq$ over the k-tuples.
Input:	A database D.
Goal:	Enumerate the tuples of $q(D)$ sorted by $\succeq$.

We have a good understanding of the complexity of the problem without order guaranteed, at least for Conjunctive Queries (CQs) [1] and unions of CQs [2]. Specifically, the following is known for CQs without self-joins (i.e., where each relation symbol occurs once), under conventional complexity assumptions (within polynomial time). The answers can be enumerated with a constant delay following a linear pre-processing if and only if the CQ is *acyclic and free-connex* [1]. Moreover, the algorithm of Bagan et al. [1] enumerates in an ordered fashion if the linear order $\succeq$ is the lexicographic order under an *elimination order* of the variables, meaning (intuitively) that there exists a join tree such that every prefix of the variable ordering corresponds to a rooted subtree. Nevertheless, not much is known about other orders. Moreover, the proof technique for the lower bound does not seem to be suitable to dismissing any ordering (assuming the CQ is acyclic free-connex). In particular, concrete questions include the following:

- Which variable orderings allow for a lexicographic answer ordering in sublinear delay (following a linear pre-processing)?
- Can we also support non-lexicographic orders such as decreasing *sum* of numeric values?
- Are there techniques from fine-grained complexity theory that allow to dismiss the possibility of some (natural) orders?

References

1 Guillaume Bagan, Arnaud Durand, and Etienne Grandjean. On acyclic conjunctive queries and constant delay enumeration. In *Computer Science Logic, 21st International Workshop, CSL 2007, 16th Annual Conference of the EACSL, Lausanne, Switzerland, September 11-15, 2007, Proceedings*, pages 208–222, 2007.

2 Nofar Carmeli and Markus Kröll. Enumeration complexity of unions of conjunctive queries. *CoRR*, abs/1812.03831, 2018.

4.7 Clause Sequences of a SAT Formula

Markus Kröll (TU Wien, AT)

The following problem arises when enumerating the answers to well-designed pattern trees, see [1]. In this work, Kroell et al. show that enumeration is possible in polynomial delay for several classes of well-designed pattern trees, while enumeration for other classes is intractable. One class of pattern trees however is still unclassified (see [1], Table 1 on page 16). The enumeration problem below can be reduced to this unsolved case, thus any lower bound on the enumeration complexity is especially helpful.

Let $k \geq 3$ be an integer, and let ϕ be a k-SAT formula with clauses $C_1, \ldots, C_m$ and variables $x_1, \ldots, x_n$. A truth assignment $a : \mathsf{var}(\phi) \to \{0,1\}$ of ϕ leads to a *clause sequence* $c(a) = (c_1, \ldots, c_m) \in \{0,1\}^m$ as follows: Every clause satisfied by a leads to a 1, every clause not satisfied by a to a 0, i.e., $c_i = 1$ if the assignment a restricted to clause i equals 1, otherwise $c_i = 0$. (This means that ϕ is satisfyable iff there exists some assignment a' with $c(a') = (1, \ldots, 1)$. Moreover, the problem MAX-SAT can be encoded by asking for the clause sequence with the largest sum of elements).

▶ **Example 1.** Consider the SAT instance $\phi = x_1 \wedge \neg x_2 \wedge (x_1 \vee x_2)$, thus $C_1 = \{x_1\}$, $C_2 = \{\neg x_2\}$ and $C_3 = \{x_1, x_2\}$. The assignment $a_1 = \{x_1 \mapsto 1, x_2 \mapsto 1\}$ leads to the clause sequence $c(a_1) = (1,0,1)$, the truth assignment $a_2 = \{x_1 \mapsto 0, x_2 \mapsto 1\}$ to the clause sequence $a_2 = (0,0,1)$. The set of clause sequences is given by

$$\{(0,1,0), (1,1,1), (0,1,1), (1,0,1)\}$$

The enumeration problem is now given as follows (k can be assumed to be fixed):

Problem:	*Enumerating all Clause Sequences*
Parameters:	k
Input:	A k-Sat formula ϕ.
Goal:	All clause sequences.

References

1. Markus Kröll, Reinhard Pichler, and Sebastian Skritek. On the Complexity of Enumerating the Answers to Well-designed Pattern Trees. In Wim Martens and Thomas Zeume, editors, *19th International Conference on Database Theory (ICDT 2016)*, volume 48 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 22:1–22:18, Dagstuhl, Germany, 2016. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

4.8 Unions of Conjunctive Queries

Markus Kröll (TU Wien, AT) and Nofar Carmeli (Technion – Haifa, IL)

Carmeli and Kroell [1] have initiated a systematic study on the enumeration complexity of unions of conjunctive queries (UCQs). The enumeration problem is given as follows:

Problem:	*UCQ Enumeration*
Parameters:	A UCQ $Q = \bigcup Q_i$
Input:	A database D.
Goal:	Enumerate all tuples of $Q(D)$.

As with enumerating the answers to CQs [2], the size of the query is assumed to be fixed. In this setting, the task is to fully characterize which class of UCQs allows for a constant delay enumeration with linear preprocessing. An example of a UCQ, for which the complexity is still unknown, is the following:

▶ **Example 1.** Let $Q = Q_1 \cup Q_2$ with

$$Q_1(x,z,y,v) \leftarrow R_1(x,z,v), R_2(z,y,v), R_3(y,x,v) \text{ and}$$
$$Q_2(x,z,y,v) \leftarrow R_1(x,z,v), R_2(y,t_1,v), R_3(t_2,x,v).$$

Note that Q_1 is cyclic, while Q_2 is acyclic free-connex. A result introduced in [1] shows how an easy query in a union can be used to enumerate the answers to a hard query (Theorem 12). However, for this query, it seems that this approach cannot be used. As of now, it is still an open problem to show either an upper or a lower bound for the enumeration complexity of the query Q.

References

1. Nofar Carmeli, Markus Kröll. *On the Enumeration Complexity of Unions of Conjunctive Queries*. PODS 2019: 134-148.
2. Guillaume Bagan, Arnaud Durand, Etienne Grandjean: On Acyclic Conjunctive Queries and Constant Delay Enumeration. CSL 2007: 208-222

4.9 Maximal Subgraphs with a Forbidden Pattern

Reinhard Pichler (TU Wien, AT)

The following defines a whole family of enumeration problems.

Problem:	*Maximal Subgraphs with Forbidden Pattern*
Parameters:	A graph $G' = (V', E')$, referred to as "pattern."
Input:	A (directed or undirected) graph $G = (V, E)$.
Goal:	Enumerate all maximal, induced subgraphs of G not containing pattern G'.

We say that G contains pattern G' if G contains a subgraph that is isomorphic to G'. This problem generalizes, for instance, the enumeration problem of Maximal Independent Sets, where the forbidden pattern is the graph consisting of a single edge. More generally, also a variant of the enumeration problem of Maximal Independent Sets of k-uniform hypergraphs can be cast as a special case of the Maximal Subgraphs with Forbidden Pattern problem: in case of the Maximal Independent Sets problem of k-uniform hypergraphs, we are given a hypergraph $H = (V, E)$, where each edge in E consists of precisely k vertices from V. Enumerating the Maximal Independent Sets of H corresponds to the Maximal Subgraphs with Forbidden Pattern problem where G is the primal graph of H and G' is a k-clique.

In case of Maximal Independent Sets of graphs (resp. of k-uniform hypergraphs), we know that the enumeration problem is solvable with polynomial delay (resp. in incremental polynomial time) [1, 2]. It would be interesting to come up with sufficient conditions on the graph G and or the pattern G' to make the enumeration problem solvable with polynomial delay or in incremental polynomial time. In which cases does the enumeration problem become intractable? Can one show some kind of dichotomy or trichotomy?

For instance, we have recently come across the following problem: G' is the path of length 2 and G is a tri-partite graph with vertex sets V_1, V_2, and V_3, such that $E \subseteq (V_1 \times V_2) \cup (V_2 \times V_3)$ and each vertex in V_1 and in V_3 is incident to exactly one edge. It is easy to show that this problem can be solved in incremental polynomial time (in fact, it can be cast as a Maximal Independent Sets problem of 3-uniform hypergraphs by representing each path of length 2 in G as a hyperedge in H with these 3 vertices). Can all maximal subgraphs of G not containing G' be enumerated with polynomial delay?

References

1. David S. Johnson, Christos H. Papadimitriou, and Mihalis Yannakakis. On generating all maximal independent sets. *Inf. Process. Lett.*, 27(3):119–123, 1988.
2. Thomas Eiter and Georg Gottlob. Identifying the minimal transversals of a hypergraph and related problems. *SIAM J. Comput.*, 24(6):1278–1304, 1995.

Participants

- Amir Abboud
IBM Almaden Center – San Jose, US
- Kira V. Adaricheva
Hofstra University – Hempstead, US
- Antoine Amarilli
Telecom ParisTech, FR
- Kristof Berczi
Eötvös Lorand University – Budapest, HU
- Christoph Berkholz
HU Berlin, DE
- Endre Boros
Rutgers University – Piscataway, US
- Pierre Bourhis
CNRS – CRIStAL, Lille, FR
- Nofar Carmeli
Technion – Haifa, IL
- Ondrej Cepek
Charles University – Prague, CZ
- Nadia Creignou
Aix-Marseille University, FR
- Arnaud Durand
University Paris-Diderot, FR
- Khaled M. Elbassioni
Khalifa University – Abu Dhabi, AE
- Etienne Grandjean
Caen University, FR
- Alejandro J. Grez
Pontificia Universidad Catolica de Chile, CL
- Aritanan Gruber
University of ABC – Santo André, BR
- Mamadou Moustapha Kanté
Université Clermont Auvergne – Aubiere, FR
- Batya Kenig
University of Washington – Seattle, US
- Benny Kimelfeld
Technion – Haifa, IL
- Christoph Koch
EPFL – Lausanne, CH
- Phokion G. Kolaitis
University of California – Santa Cruz, US
- Markus Kröll
TU Wien, AT
- Ester Livshits
Technion – Haifa, IL
- Kazuhisa Makino
Kyoto University, JP
- Andrea Marino
Univerisità degli Studi di Firenze, IT
- Wim Martens
Universität Bayreuth, DE
- Stefan Mengel
CNRS, CRIL – Lens FR
- Shin-Ichi Nakano
Guuma University – Kiryu, JP
- Matthias Niewerth
Universität Bayreuth, DE
- Lhouari Nourine
University Clermont Auvergne, FR
- Liat Peterfreund
Technion – Haifa, IL
- Reinhard Pichler
TU Wien, AT
- Cristian Riveros
Pontificia Universidad Catolica de Chile, CL
- Yehoshua Sagiv
The Hebrew University of Jerusalem, IL
- Nicole Schweikardt
HU Berlin, DE
- Thomas Schwentick
TU Dortmund, DE
- Luc Segoufin
INRIA & ENS Paris, FR
- Yann Strozecki
University of Versailles, FR
- Alexandre Termier
IRISA – University of Rennes, FR
- Etsuji Tomita
The University of Electro-Communications – Tokyo, JP
- György Turan
University of Illinois – Chicago, US
- Martin Ugarte
PUC Chile, CL
- Takeaki Uno
National Institute of Informatics – Tokyo, JP
- Stijn Vansummeren
Free University of Brussels, BE
- Alexandre Vigny
University of Warsaw, PL
- Heribert Vollmer
Leibniz Universität Hannover, DE
- Thomas Zeume
TU Dortmund, DE