Fachgruppe Komplexität

— Newsletter 13 —

In dieser Ausgabe …

Konferenzen 2022

Konferenz	Frist'22	Event'22/'23
ITCS	2.9.	11.–13.1.23
CSL	9.7.	13.–16.2.23
ICDT	???	28.03.–31.03.23
FoSSaCS	14.10.	20.–27.04.23
PODS	28.11.	18.–23.06.23
STOC	—	20.–24.06.22
LICS	—	02.–05.08.22
SAT	—	02.–05.08.22
ICALP	—	04.–08.08.22
CCC	—	21.–23.07.22
MFCS	—	22.–26.08.22
IPEC	24.06.	07.–09.09.22
Highlights	—	28.06–01.07.22
CIE	—	11.–15.07.22
AiML	—	22.–25.08.22
FOCS	—	31.10.–3.11.22

Die letzten Theorietage

TT	Wo	Wann
83	Bei Ihnen?	—
82	Frankfurt	08.–09.06.22
81	INFORMATIK'21 🌍	28.09.21
80	Berlin 🌍	13.04.21
79	Hannover 🌍	17.11.20
78	Berlin	10./11.10.19
77	Marburg	28.03.19
76	Halle	24.–25.09.18
75	Ulm	10.–11.04.18
74	Lübeck	23.–25.11.17
73	Hamburg	18./19.05.17

🌍: online

Fachgruppenleitung (2020–2023)

E-Mail an die Fachgruppenleitung

- Arne Meier (Sprecher)
- Till Tantau (stv. Sprecher)

Mitgliederzahl (GI): 299

Kostenlos Mitglied in FG-KP werden

Mai 2022

Liebe Mitglieder der Fachgruppe Komplexität,

Sie halten den dreizehnten Newsletter der GI-Fachgruppe „Komplexität“ in den Händen. Beim letzten Newsletter gab es einen Rückblick auf den zweiten virtuellen Theorietag, organisiert von der Gruppe um Rolf Niedermeier aus Berlin. Nun berichte ich mit großer Trauer von der erschütternden Nachricht, dass Rolf leider überraschend verstorben ist. Rolf war ein unglaublich sympathischer Kollege und ich habe seit seinem Vorsitz der FG Algorithmen immer gerne mit ihm zusammen gearbeitet. In der Forschung waren wir bisher leider nicht zusammengekommen, obwohl wir beide mit parameterisierten Problemen gearbeitet haben bzw. arbeiten. In diesem Newsletter gibt es nun einen Nachruf, der von André Nichterlein und Matthias Bentert geschrieben wurde.

Nun ist es schwierig zum üblichen Newsletter-Programm überzuleiten: Wie üblich gilt, wenn Sie eine spezielle Konferenz in der linken Spalte vermissen, dann melden Sie sich bei mir, damit wir die Konferenz für die Zukunft aufnehmen können.

Außerdem möchte ich, wie üblich, auf die Möglichkeit zu kurzen inhaltlichen Beiträgen hinweisen. Bei Interesse Ihrerseits melden Sie sich bitte direkt bei mir. Wir planen mit Textvorschlägen von 1–2 Seiten Länge. In dieser Ausgabe gibt es einen Gastbeitrag von Olaf Beyersdorff zu aktuellen Entwicklungen bezüglich QBF Proof Complexity.

Wenn Sie in die Fachgruppe eintreten möchten, dann ist dies kostenlos als assoziiertes Mitglied möglich — auch ohne eine GI-Mitgliedschaft.

Der Newsletter ist auch online von unserer Webseite zu beziehen (Publikationen → Newsletter).

Nun wünsche ich Ihnen viel Spass beim Lesen und bleiben Sie gesund!

Arne Meier, Sprecher der Fachgruppe KP

Die Fachgruppe Komplexität

Die Fachgruppe Komplexität ist ein Teil der Gesellschaft für Informatik. Diese Fachgruppe beschäftigt sich mit komplexitätstheoretischen Fragestellungen. Manche der Themen sind eng gekoppelt an bzw. werden gemeinsam bearbeitet mit anderen Fachgruppen, insbesondere sind dies die FG Algorithmen (Thema: Obere Schranken), FG Automaten und formale Sprachen (Thema: spezielle Berechnungsmodelle, Abschlusseigenschaften von Klassen) FG Logik in der Informatik (Thema: Komplexität logischer Entscheidungsprobleme, Komplexität des logischen Programmierens, subrekursive Hierarchien).

Ein Workshop über Algorithmen und Komplexität, gemeinsam mit der Fachgruppe Algorithmen, findet zweimal jährlich statt.

Gastbeitrag: Proof complexity of quantified Boolean formulas — a very short survey

Olaf Beyersdorff (Friedrich-Schiller-Universität Jena, olaf.beyersdorff@uni-jena.de)

The last two decades have witnessed tremendous progress and success in solving quantified Boolean formulas (QBF). In parallel, the proof complexity of QBFs has been intensively studied. In this short survey we try to explain some recent highlights from QBF proof complexity.

Quantified Boolean Formulas (QBF) extend propositional logic by adding Boolean quantifiers that range over 0/1. Typically we consider fully quantified QBFs. As these formulas have no free variables, they are either true or false. Mostly we use QCNFs, which are QBFs in prenex normal form where the matrix is in CNF. As an easy example consider the false QBFs

$$\begin{aligned} EQ_n \quad = \quad & \exists x_1 \cdots x_n \forall u_1 \cdots u_n \exists t_1 \cdots t_n \\ & \left(\bigwedge_{i=1}^{n} (x_i \vee u_i \vee \bar{t}_i) \wedge (\bar{x}_i \vee \bar{u}_i \vee \bar{t}_i) \right) \wedge \left(\bigvee_{i=1}^{n} t_i \right) \end{aligned}$$

known as the *equality formulas* in the literature [2].

The semantics of QBFs can be described by a 2-player game between an $\exists$ and a $\forall$ player who assign 0/1 values to the variables in the prefix from left to right according to their type. If a clause gets falsified the universal player wins. If, on the other hand, the matrix is satisfied, then the existential player wins. In the QBF above, the existential player would start by assigning $x_1, \ldots, x_n$, to which the universal responds by choosing values for $u_1, \ldots, u_n$. The universal player has a unique winning strategy by playing $u_i = x_i$ for all $i \in [n]$.

QBF proof systems

Propositional resolution is a refutational system operating with clauses, i.e., it demonstrates the unsatisfiability of a given CNF. It has only one rule, the *resolution rule*

$$\frac{C \vee x \qquad D \vee \bar{x}}{C \vee D} \tag{1}$$

for clauses C, D and a propositional pivot variable x.

QBF resolution systems work with fully quantified QCNFs $Q\,\phi$. As in propositional resolution, these QBF systems are refutational calculi, i.e., they refute false QBFs.

The simplest QBF resolution system *QU-Resolution* (QU-Res) can be obtained by augmenting the propositional resolution system using rule (1) by just one new rule, the *universal reduction rule*

$$\frac{C}{\sigma(C)} \tag{2}$$

where σ is a partial substitution that allows to substitute a universal variable from a clause C by either 0 or 1, provided u appears right of all variables in C in the prefix Q. Intuitively, this means that a universal variable u can be deleted from a clause C if u is rightmost in C with respect to Q, i.e., no variable in C depends on u.

As a simple example consider the first instance EQ_1 of the equality formulas $\exists x \forall u \exists t\, (x \vee u \vee \bar{t}) \wedge (\bar{x} \vee \bar{u} \vee \bar{t}) \wedge t$. The formula can be refuted in QU-Res as follows

$$\dfrac{\dfrac{\dfrac{x \vee u \vee \bar{t} \qquad t}{x \vee u}}{x}\,\text{(red)} \qquad \dfrac{\dfrac{\bar{x} \vee \bar{u} \vee \bar{t} \qquad t}{\bar{x} \vee \bar{u}}}{\bar{x}}\,\text{(red)}}{\bot}$$

where the universal reduction steps are labeled (red). The reduction steps are allowed because the only other variable x in the clause does not depend on u.

Propositional proof complexity considers many more and stronger calculi than resolution. As it turns out, almost all of these systems can be straightforwardly lifted to quantified Boolean logic by adding a slightly generalised version of the universal reduction rule (2) above. This means that starting with a line-based propositional proof system P, we obtain a QBF proof system $P + \forall\mathrm{-red}$, which operates on fully quantified QBFs and uses the rules of the system P together with a similar universal reduction rule as (2). The only difference is that the systems P and $P + \forall\mathrm{-red}$ might not operate with clauses, but use other lines, e.g. formulas in Frege systems, polynomial equations in polynomial calculus (PC), or inequalities in cutting planes (CP) [3]. The universal reduction rule is then applied to a line F as

$$\frac{F}{\sigma(F)}$$

with the same side condition as in QU-Res, i.e., σ is a partial substitution that can only substitute those universal variables with 0/1 that appear right of all other variables in F with respect to the quantifier prefix.

With this general definition of P+$\forall\mathrm{-red}$ (introduced in [3]), Resolution+$\forall\mathrm{-red}$ corresponds exactly to QU-Res as defined above.

We note that there are further QBF proof systems using other ideas than just lifting propositional calculi by adding universal reduction. In particular, there are many more variants of QBF resolution that correspond to different ideas in QBF solving [5]. While the systems $P + \forall\mathrm{-red}$ do not modify the prefix and only derive new lines from the matrix, there are also sequent systems that manipulate the quantifier prefix and are far more powerful than QBF resolution or Frege systems [3].

Lower bound techniques and characterisations

The main objective in proof complexity is to demonstrate lower bounds on the size of proofs for specific formulas in different proof systems. We will now illustrate two general techniques for proof size lower bounds in QBF proof systems.

Strategy extraction is a distinctive feature of refutational QBF proof systems for which no propositional analogue exists. Interestingly, most QBF lower-bound techniques employ strategy extraction. Strategy extraction is also used in practice to verify the answers of solvers.

From a proof π of a false QBF F, strategy extraction allows to efficiently compute a winning strategy for the universal player on F. A strategy is a function from existential variables to universal variables, specifying the action of the universal player in response to the existential player's moves.

As it turns out, from QU-Resolution proofs, winning strategies for the universal player can be efficiently constructed in AC^0 (and in fact in a much weaker computational model of decision lists) [1, 3]. This allows to show lower bounds on proof size as follows. If Φ is a false QBF where winning strategies are hard to compute in a complexity class $\mathcal{C}$, then proofs of Φ must be large in every QBF proof system with strategy extraction in $\mathcal{C}$.

To illustrate this method, consider the QBFs

$$\mathit{QParity}_n = \exists x_1 \cdots x_n \, \forall u \, \exists t_1 \cdots t_n$$
$$(x_1 \leftrightarrow t_1) \wedge \bigwedge_{i=2}^{n} ((t_{i-1} \oplus x_i) \leftrightarrow t_i) \wedge (u \oplus t_n).$$

It is easily checked that the only winning strategy for the universal player is to play $u = x_1 \oplus \cdots \oplus x_n$. As parity is hard for AC^0, the QBFs $\mathit{QParity}_n$ (transformed into QCNFs) require exponential-size proofs in QU-Resolution [4].

Similarly, lower bounds for non-uniform NC^1 would yield proof size lower bounds for QBF Frege, and lower bounds for P/poly imply lower bounds for QBF extended Frege systems [3]. While such explicit circuit lower bounds are open, this provides a direct transfer of hardness from circuit complexity to QBF proof complexity. Though much speculated upon, such a direct connection is not known for propositional proof systems.

While unconditional lower bounds for strong QBF proof systems such as QBF Frege are a major open problem, the correspondence above can be tightened to a characterisation of hardness as follows (shown in [3]): there exist hard QBFs for Frege$+\forall\mathrm{-red}$ if and only if

1. there exist hard formulas for propositional Frege or
2. $\mathrm{PSPACE} \not\subseteq$ (non-uniform) NC^1.

The equivalent result also holds for extended Frege and P/poly. This characterisation of QBF Frege lower bounds therefore combines the major problems from proof complexity (Frege lower bounds) and circuit complexity (lower bounds for unrestricted circuits). Both statements (1) and (2) are wide open.

The lower bound method above works for formulas where winning strategies are computationally hard. However, we can also show lower bounds for QBFs where winning strategies are easy to compute, but large. We illustrate this on the equality formulas from above. The winning strategies $u_i = x_i$ are trivial to compute, but when combined have a large range of size 2^n, i.e., there are exponentially many responses needed by the universal player.

Measuring the size of winning strategies can be done formally via a notion of *cost* [2] (that intuitively counts the number of responses needed on one block). The *size-cost-capacity theorem* from [2] states that for each $P + \forall\mathrm{-red}$ proof π of Φ

$$|\pi| \geq \frac{\mathrm{cost}(\Phi)}{\mathrm{capacity}(\pi)} \quad . \tag{3}$$

Informally, the *capacity* of a line in a proof counts how many responses can be at most extracted from it; and the capacity of a proof is defined as the maximum of the capacities of its lines. As an example, clauses have capacity one as the universal player has a unique strategy to falsify the clause. Hence, resolution proofs always have capacity one.

By (3) the QBFs EQ_n require QU-Resolution proofs of size 2^n.

Note that the two lower bound methods we sketched above are incomparable. While winning strategies for EQ_n are easy to compute, yet have large cost, strategies for $\mathit{QParity}_n$ are hard to compute and have cost 2.

Proof complexity and solving

We conclude by mentioning one important application and motivation of QBF proof complexity – its tight connection to QBF solving. Though deciding QBFs is PSPACE complete, the past 20 years have seen huge progress on solving QBFs in an automated fashion with particular success on instances of industrial relevance [6]. Different from SAT solving where conflict-driven clause learning (CDCL) dominates the scene, there are various conceptual approaches to QBF solving, among them QCDCL (the QBF variant of CDCL) and expansion solving. Both can be modelled by QBF resolution systems in the sense that a run of the solver on a false QBF Φ can be efficiently translated into a refutation of Φ in the corresponding proof system (while we have concentrated here on false QBFs, dual proof systems exist for true QBFs, e.g. term resolution). Therefore, lower bounds on proof size as shown above directly translate into lower bounds for the running time of algorithms.

While the relations between QBF proofs and solvers would easily lead to another such survey, we refer to the recent overview in [5] for a detailed account.

References

[1] V. Balabanov and J.-H. R. Jiang. Unified QBF certification and its applications. *Form. Methods Syst. Des.*, 41(1):45–65, 2012.

[2] O. Beyersdorff, J. Blinkhorn, and L. Hinde. Size, cost, and capacity: A semantic technique for hard random QBFs. *Logical Methods in Computer Science*, 15(1), 2019.

[3] O. Beyersdorff, I. Bonacina, L. Chew, and J. Pich. Frege systems for quantified Boolean logic. *J. ACM*, 67(2):9:1–9:36, 2020.

[4] O. Beyersdorff, L. Chew, and M. Janota. New resolution-based QBF calculi and their proof complexity. *ACM Transactions on Computation Theory*, 11(4):26:1–26:42, 2019.

[5] O. Beyersdorff, M. Janota, F. Lonsing, and M. Seidl. Quantified Boolean formulas. In A. Biere, M. Heule, H. van Maaren, and T. Walsh, editors, *Handbook of Satisfiability*, Frontiers in Artificial Intelligence and Applications, pages 1177–1221. IOS Press, 2021.

[6] A. Shukla, A. Biere, L. Pulina, and M. Seidl. A survey on applications of quantified Boolean formulas. In *Proc. IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 78–84, 2019.

Rückblick: Theorietag 81

Zum dritten (und hoffentlich letzten) Mal fand der Theorietag virtuell im Rahmen der GI Jahrestagung statt.. Für die Organisation waren Heribert Vollmer und Arne Meier zuständig. Es gab 5 Vorträge mit Themen aus den Bereichen der Energieinformatik, Parametrisierten Komplexität, Komplexität von Linear Logic sowie zur Mengenüberdeckung durch disjunkte konvexe Mengen. Der eingeladene Vortrag von Astrid Nieße (U Oldenburg) war „On distributed algorithms in Digitalized Energy Systems" betitelt. Mehr Details zum Programm sind online zu auf der Webseite des TT zu finden. Es gab angeregte Diskussionen sowohl bei den Vorträgen als auch in den Breakout-Räumen in den Pausen und insgesamt 30 Teilnehmer. *A. Meier*

Impressum

GI Fachgruppe Komplexität

Fachgruppenleitung:
Arne Meier (Sprecher, ViSdPR),
Till Tantau (stv. Sprecher).

Sekretariat +49 511 762 19692
Web https://fg-kp.gi.de
Postalisch
GI-FG Komplexität
PD Dr. Arne Meier
Institut für Theoretische Informatik
Appelstrasse 9A
D-30167 Hannover
Mail fg-kp-leitung@gi.de

Vorausschau: Theorietag 82

Der 82. Workshop über Algorithmen und Komplexität steht schon von der Tür und findet endlich wieder in Präsenz am *8. und 9. Juni 2022* an der Goethe-Universität Frankfurt statt. Organisatorisch ist Anselm Haak für ihn zuständig.
Es wird einen eingeladenen Vortrag von Pascal Schweitzer (TU Darmstadt) mit dem Titel „The Theory of Practical Graph Isomorphism Solving" geben.

Weitere Informationen entnehmen Sie bitte der Webseite: `https://tcs.uni-frankfurt.de/tt-82/`.

Vortragsanmeldungen zum Workshop per E-Mail (Titel/Abstract) an Anselm Haak. **Fristende hierfür ist der 25.5.2022!**

Nachruf auf Rolf Niedermeier (André Nichterlein, Matthias Bentert)

Mit großer Trauer geben wir den plötzlichen und überraschenden Tod von Rolf Niedermeier im Alter von 55 Jahren bekannt.

Rolfs Reise in die Welt der Informatik begann 1986 an der TU München, wo er 1991 sein Diplom erwarb. Er setzte sein Studium an der Universität Tübingen fort, wo er 1996 promovierte und nach einem einjährigen Aufenthalt als Postdoc an der Karls-Universität in Prag eine eigene Forschungsgruppe für parametrisierte Algorithmen gründete. Anschließend wechselte er 2004 an die Universität Jena, wo er seine erste Professur erhielt, und schließlich 2010 an die TU Berlin, wo er die Forschungsgruppe „Algorithmik und Komplexitätstheorie" gründete und seither leitete.

Rolf hat viele bahnbrechende Beiträge zur theoretischen Informatik geleistet, insbesondere zur parametrisierten Algorithmik unter anderem durch sein Buch „Invitation to Fixed-Parameter Algorithms". Er war als eine der führenden Persönlichkeiten auf diesem Gebiet bekannt und arbeitete mit unermüdlichem Enthusiasmus an Problemen aus verschiedenen Bereichen wie Computational Social Choice, Computational Biology und Temporal Graph Theory.

Rolf betreute dreißig Doktoranden und zahllose Bachelor- und Masterstudenten, unterrichtete eine Vielzahl spannender Kurse und leitete mehr als fünfzehn verschiedene DFG-finanzierte Forschungsprojekte. Er war Mitorganisator zahlreicher Dagstuhl-Seminare, darunter die ersten beiden über parametrisierte Algorithmen in den Jahren 2001 und 2003, „Adaptive, Output Sensitive, Online and Parameterized Algorithms" im Jahr 2009, „Application-Oriented Computational Social Choice" und „Algorithms and Complexity in Phylogenetics" im Jahr 2019, sowie „Temporal Graphs: Structure, Algorithms, Applications" im Jahr 2021. Darüber hinaus war er maßgeblich an der Organisation und Durchführung der Konferenzen „Workshop on Graph-Theoretic Concepts in Computer Science" (WG 2007), „Workshop on Challenges in Algorithmic Social Choice" (CASC 2014), und „Symposium on Theoretical Aspects of Computer Science" (STACS 2019) beteiligt. Außerdem engagierte er sich tatkräftig für die Forschungsgemeinschaft in verschiedenen Gremien und Ausschüssen als Gutachter (unter anderem für die DFG und den ERC), als Dekan der Fakultät Elektrotechnik und Informatik der TU Berlin und als Sprecher der Fachgruppe Algorithmen der GI.

Nicht nur für die Wissenschaft, sondern auch für seine Mitmenschen setzte sich Rolf in seiner warmherzigen und fürsorglichen Art unermüdlich ein. Egal wie groß sein Pensum gerade war, seine Tür stand für seine Studenten, Kollegen und Freunde zu allen Tages- und Nachtzeiten offen. Man konnte sich immer darauf verlassen, dass er einem stets seine volle Aufmerksamkeit schenken und wahrscheinlich auch eine Tasse Tee anbieten würde.

Wir trauern um einen inspirierenden Mentor, einen hervorragenden Wissenschaftler, einen hilfsbereiten Kollegen und einen guten Freund. Sein Tod hinterlässt eine große Lücke in unseren Herzen, aber sein Einfluss wird in all jenen weiterleben, die je das Vergnügen hatten, mit ihm zusammen zu arbeiten.