Model Counting: Symbolic Quantitative AI and Practical Solving (Invited Talk)

Johannes K. Fichte[1]

[1]TU Wien, Austria

Abstract

Model counting (#SAT) asks to compute the number of satisfying assignments for a propositional formula. The decision version (SAT) received widespread interest in computational complexity, formed many applications in modern combinatorial problem solving, and can be solved effectively for millions of variables on structured instances. #SAT is much harder than SAT and requires more elaborate solving techniques. We revisit the problem, its complexity, and explain applications in symbolic quantitative AI. We briefly overview solving techniques and finally list connections to abstract argumentation.

Keywords

Propositional Model Counting, Satisfiability, Abstract Argumentation, Computational Complexity, Solving

Introduction. The *satisfiability* (SAT) *problem* asks to decide whether a given propositional formula has a *model*, which is a satisfying assignment to the variables. When extending SAT to *model counting* (#SAT), we ask for computing the number of models instead [1]. The SAT problem and their solvers have manifested as a core tool for *qualitative symbolic artificial intelligence (AI)* allowing for efficient modeling and solving of a wide variety of real-world problems, in areas such as hardware or software verification, planning, combinatorial design [1, Chapters 18,19,20], dependency solving [2], and knowledge representation and reasoning [3]. Model counting lifts SAT to *quantitative tasks* and directly applies to exact probabilistic reasoning [4]. Thereby, #SAT provides the link to quantitative symbolic tasks in AI that occur in probabilistic reasoning, statistics, and combinatorics with manifold applications [5, 6, 7, 8, 9, 10, 11, 9, 6, 12, 13, 14]. In the practical community two more problems are of high practical interest, namely, (i) *weighted model counting* (WMC) and (ii) *projected model counting* (PMC). For WMC, we additionally associate to each literal a weight and we are interested in the weighed model count[1]. For PMC, we are interested in hiding some variables and counting the models after restricting them to a set P of projection variables.

SAFA'22: Fourth International Workshop on Systems and Algorithms for Formal Argumentation 2022, September 13, 2022, Cardiff, Wales, United Kingdom
johannes.fichte@tuwien.ac.at (J. K. Fichte)
https://www.dbai.tuwien.ac.at/staff/fichte/ (J. K. Fichte)
0000-0002-8681-7470 (J. K. Fichte)

[1]More formally, the weight of a model is the product of its weights. We are interested in the sum of weights over all models. The WMC problem relates to sum-of-product, partition function, or probability of evidence.

Computational Complexity. While the SAT problem is already known to be NP-complete [15, 16], its generalization #SAT is believed to be even harder. Namely, #SAT is known to be $\#\cdot\mathsf{P}$-complete [17], and by direct implications from a result by Toda [18], any problem on the polynomial hierarchy [19, 20] can be solved in polynomial-time by a machine with access to an oracle that can output the model count for a given formula. The problem WMC is also $\#\cdot\mathsf{P}$-complete. But, PMC is harder under standard assumptions and complete for the class $\#\cdot\mathsf{NP}$ [21]. Common notions on counting complexity follow works by Hemaspaandra and Vollmer [22] and Durand, Herman, and Kolaitis [21]. Complexity classes with the sharp-dot operator '$\#\cdot X$' are counting classes for which the witness can be tested in decision complexity class X. Similar to reductions in decision complexity, there are dedicated reductions between two counting problems. A *parsimonious* reduction preserves the cardinality between the corresponding witness sets and is computable in polynomial time. A *subtractive* reduction between two counting problems consists of two functions. The first function may over-count and the second function determines the precise correction for the over-count, which is then compensated by a subtraction.

Counting and its Connection to Quantitative Reasoning. To illustrate the connection, we elaborate on an introductory example about Bayesian inference, which can be found in lectures on Bayesian reasoning, c.f. [23]. Assume that we have a bag of four marbles of which each could be either red or blue. However, we cannot see inside the box and are hence unaware of the exact colors. According to the colors and the setting, we may have five potential combinations of colors. To determine the individual colors, we are only allowed to take out one marble at a time and put it back into the bag. While we could repeat this experiment for a very long time and obtain the combination that has the highest probability, we assume that we repeat the experiment only three times. Suppose that we observed the combination blue-red-blue and want to conjecture how likely blue-red-red-red is. Therefore, we can simply illustrate the possible combinations as paths. In the first step, we can have a blue, red, red, or red marble assuming that blue-red-red-red was inside the bag. In steps two and three, we can have the same situation as a marble is put back inside the bag after each step. However, only a few paths are "accepting" according to the observation that we had seen blue-red-blue in our experiment when taking three marbles. Overall, we have 3 paths out of the 4^3 possible paths that are consistent with the observation. If we consider the same experiment for each of the five possible conjectures of marbles and sum the paths that are consistent with the observation, we count 20 paths in total. By counting 3 paths and in total 20 paths, we obtain a probability of $3/20$ for our conjecture blue-red-red-red under the observation blue-red-blue. In that way, we can obtain the probabilities by counting.

Abstract Argumentation. When modeling problems in reasoning and artificial intelligence [24, 25], we oftentimes rely on more extended frameworks to enable high-level encoding and avoid spacious encodings. A popular framework for modeling problems that are related to arguments and their interaction is abstract argumentation [26, 27]. There, counting complexity has been considered in the literature [28, 29]. The most recent competition also featured counting questions [30], which will hopefully enable probabilistic questions in abstraction argumentation beyond a simple enumeration.

Practical Solving. Over the last years, tremendous progress in solving has been made and a recently established competition makes continuing improvements visible [31]. Today's solvers provide exact, high precision, or approximate solving capabilities. Some solvers extend techniques from SAT solving by concepts such as component caching [32], knowledge compilation [33], or approximate solving [34], but also techniques from parameterized algorithmics proved helpful [35, 36].

Conclusion. Model counters are key tools for symbolic quantitative reasoning, which allow for faster reasoning than simple enumeration of models. We believe that key challenges for the next years are scalability, trustworthiness, and provability correctness of practical solvers. We hope that effective solvers for model counting make probabilistic symbolic reasoning accessible to a variety of scientific fields such as computational biology and psychology as cognitive reasoning.

References

[1] A. Biere, M. Heule, H. van Maaren, T. Walsh (Eds.), Handbook of Satisfiability (2nd Edition), Frontiers in Artificial Intelligence and Applications, IOS Press, Amsterdam, Netherlands, 2021.
[2] P. Abate, R. D. Cosmo, R. Treinen, S. Zacchiroli, Dependency solving: A separate concern in component evolution management, J. Systems and Software 85 (2012) 2228–2240.
[3] M. Gebser, R. Kaminski, B. Kaufmann, T. Schaub, Answer Set Solving in Practice, Morgan & Claypool, 2012.
[4] A. Darwiche, Three modern roles for logic in AI, in: PODS'20, ACM, 2020, pp. 229–243.
[5] A. Choi, G. Van den Broeck, A. Darwiche, Tractable learning for structured probability spaces: A case study in learning preference distributions, in: IJCAI'15, The AAAI Press, 2015, pp. 2861–2868.
[6] O. Pourret, P. Naim, M. Bruce, Bayesian Networks - A Practical Guide to Applications, John Wiley & Sons, 2008.
[7] D. J. Fremont, Algorithmic Improvisation, University of California, Berkeley, 2019.
[8] L. Dueñas-Osorio, K. S. Meel, R. Paredes, M. Y. Vardi, Counting-based reliability estimation for power-transmission grids, in: AAAI'17, The AAAI Press, 2017, pp. 4488–4494.
[9] A. L. D. Latour, B. Babaki, A. Dries, A. Kimmig, G. Van den Broeck, S. Nijssen, Combining stochastic constraint optimization and probabilistic programming, in: CP'17, Springer, 2017, pp. 495–511.
[10] C. D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, Cambridge, Cambridge, England, 2008.
[11] M. Sahami, S. Dumais, D. Heckerman, E. Horvitz, A Bayesian approach to filtering junk e-mail, in: T. Joachims (Ed.), Proceedings of the AAAI-98 Workshop on Learning for Text Categorization, volume 62, The AAAI Press, Madison, Wisconsin, 1998, pp. 98–105.
[12] B. Abramson, J. Brown, W. Edwards, A. Murphy, R. L. Winkler, Hailfinder: A Bayesian system for forecasting severe weather, International Journal of Forecasting 12 (1996) 57–71.
[13] J. K. Fichte, M. Hecher, M. A. Nadeem, Plausibility reasoning via projected answer set counting – a hybrid approach, in: IJCAI'22, IJCAI Organization, 2022, pp. 2620–2626.

[14] E. Dietz, J. K. Fichte, F. Hamiti, A quantitative symbolic approach to individual human reasoning, in: CogSci'22, 2022, pp. 2838–2846.
[15] S. A. Cook, The complexity of theorem-proving procedures, in: STOC'71, ACM, 1971, pp. 151–158.
[16] L. Levin, Universal sequential search problems, Probl. Inf. Transm. 9 (1973) 265—266.
[17] D. Roth, On the hardness of approximate reasoning, AIJ 82 (1996).
[18] S. Toda, PP is as hard as the polynomial-time hierarchy, SIAM J. Comput. 20 (1991) 865–877.
[19] L. J. Stockmeyer, The polynomial-time hierarchy, Th. Computer Sci. 3 (1976) 1–22.
[20] L. J. Stockmeyer, A. R. Meyer, Word problems requiring exponential time, in: STOC'73, ACM, 1973, pp. 1–9.
[21] A. Durand, M. Hermann, P. G. Kolaitis, Subtractive reductions and complete problems for counting complexity classes, Th. Computer Sci. 340 (2005) 496–513.
[22] L. A. Hemaspaandra, H. Vollmer, The satanic notations: Counting classes beyond #P and other definitional adventures, SIGACT News 26 (1995) 2–13.
[23] R. McElreath, Bayesian inference is just counting, 2020. Talk.
[24] L. Amgoud, H. Prade, Using arguments for making and explaining decisions, AIJ 173 (2009) 413–436.
[25] A. Rago, O. Cocarascu, F. Toni, Argumentation-based recommendations: Fantastic explanations and how to find them, in: IJCAI'18, 2018, pp. 1949–1955.
[26] P. M. Dung, On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games, AIJ 77 (1995) 321–357.
[27] I. Rahwan, Argumentation in artificial intelligence, AIJ 171 (2007) 619–641.
[28] J. K. Fichte, M. Hecher, A. Meier, Counting complexity for reasoning in abstract argumentation, in: AAAI'19, The AAAI Press, 2019, pp. 2827–2834.
[29] J. Fichte, M. Hecher, Y. Mahmood, A. Meier, Decomposition-guided reductions for argumentation and treewidth, in: IJCAI'21, IJCAI Organization, 2021, pp. 1880–1886.
[30] J.-G. Mailly, E. Lonca, J.-M. Lagniez, J. Rossit, 4th intl. competition on computational models of argumentation (ICCMA'21), 2021. http://argumentationcompetition.org/2021/index.html.
[31] J. K. Fichte, M. Hecher, F. Hamiti, The model counting competition 2020, ACM Journal of Experimental Algorithmics 26 (2021).
[32] M. Thurley, sharpSAT – counting models with advanced component caching and implicit BCP, in: SAT'06, Springer, 2006, pp. 424–429.
[33] J.-M. Lagniez, P. Marquis, An improved decision-DDNF compiler, in: IJCAI'17, The AAAI Press, 2017, pp. 667–673.
[34] S. Chakraborty, D. J. Fremont, K. S. Meel, S. A. Seshia, M. Y. Vardi, Distribution-aware sampling and weighted model counting for SAT, in: AAAI'14, The AAAI Press, 2014, pp. 1722–1730.
[35] J. M. Dudek, V. H. N. Phan, M. Y. Vardi, ADDMC: Weighted model counting with algebraic decision diagrams, in: V. Conitzer, F. Sha (Eds.), AAAI'20, The AAAI Press, 2020, pp. 1468–1476.
[36] J. K. Fichte, M. Hecher, V. Roland, Parallel model counting with cuda: Algorithm engineering for efficient hardware utilization, in: CP'21, 2021, pp. 24:1–24:20.