Hashing-Based Approximate Probabilistic Inference in Hybrid Domains: An Abridged Report

Hashing-Based Approximate Probabilistic Inference in Hybrid Domains: An Abridged Report*

Vaishak Belle
KU Leuven
vaishak@cs.kuleuven.be

Guy Van den Broeck
University of California, Los Angeles
guyvdb@cs.ucla.edu

Andrea Passerini
University of Trento
passerini@disi.unitn.it

Abstract

In recent years, there has been considerable progress on fast randomized algorithms that approximate probabilistic inference with tight tolerance and confidence guarantees. The idea here is to formulate inference as a counting task over an annotated propositional theory, called weighted model counting (WMC), which can be partitioned into smaller tasks using universal hashing. An inherent limitation of this approach, however, is that it only admits the inference of discrete probability distributions. In this work, we consider the problem of approximating inference tasks for a probability distribution defined over discrete and continuous random variables. Building on a notion called weighted model integration, which is a strict generalization of WMC and is based on annotating Boolean and arithmetic constraints, we show how probabilistic inference in hybrid domains can be put within reach of hashing-based WMC solvers. Empirical evaluations demonstrate the applicability and promise of the proposal.

Introduction

Weighted model counting (WMC) on a propositional knowledge base is an effective and general approach to probabilistic inference in a variety of formalisms, including Bayesian and Markov Networks. It extends the model counting task, or #SAT, which is to count the number of assignments (or models) that satisfy a logical sentence (Gomes, Sabharwal, and Selman 2009). In WMC, one accords a weight to every model, and computes the sum of the weights of all models. The WMC formulation has recently emerged as an assembly language for probabilistic reasoning, offering a basic formalism for encoding various inference problems. State-of-the-art reasoning algorithms for Bayesian networks (Chavira and Darwiche 2008), probabilistic programs (Fierens et al. 2013) and probabilistic databases (Suciu et al. 2011) reduce their inference problem to a WMC computation. Exact WMC solvers are based on knowledge compilation or component caching (Chavira and Darwiche 2008).

However, exact inference is #P-hard (Valiant 1979), and so, there is a growing interest in approximate model counters. Beginning with Stockmeyer (1983), who showed that approximating model counting with a tolerance factor can be achieved in deterministic polynomial time using a Σ_2^P-oracle, a number of more recent results show how random polynomial-time realizations are possible using an NP-oracle, such as a SAT solver (Jerrum, Valiant, and Vazirani 1986; Karp, Luby, and Madras 1989; Ermon et al. 2013; Chakraborty et al. 2014). The central idea here is the use of random parity constraints, in the form of *universal hash functions* (Sipser 1983), that partition the model counting solution space in an inexpensive manner. Most significantly, such methods come with tight tolerance-confidence guarantees, unlike classical variational methods that only provide asymptotic guarantees.

The popularity of WMC can be explained as follows. Its formulation elegantly decouples the logical or symbolic representation from the statistical or numeric one, which is encapsulated in the weight function. When building solvers, this allows us to reason about logical equivalence and reuse SAT solving technology (such as constraint propagation and clause learning). WMC also makes it more natural to reason about deterministic, hard constraints in a probabilistic context. Nevertheless, WMC has a fundamental *limitation*: it is purely Boolean. This means that the advantages mentioned above only apply to *discrete probability distributions*.

To counter this, in a companion paper (Belle, Passerini, and Van den Broeck 2015), we proposed the notion of *weighted model integration* (WMI). It is based on *satisfiability modulo theories* (SMT), which enable us to reason about linear arithmetic constraints. The WMI task is defined on the models of an SMT theory Δ, containing mixtures of Boolean and continuous variables. For every assignment to these variables, the WMI problem defines a weight. The total WMI is computed by integrating these weights over the domain of solutions of Δ, which is a mixed discrete-continuous space. Consider, for example, the special case when Δ has no Boolean variables, and the weight of every model is 1. Then, WMI simplifies to computing the volume of the polytope encoded in Δ. More generally, weighted SMT theories admit a natural encoding of hybrid graphical models, analogous to the encodings of discrete graphical models using weighted propositional theories.

In this work, we consider the problem of approximating inference tasks for a probability distribution defined over discrete and continuous random variables. Formulated as

*This is an abridged version of a paper that appeared in the *Proceedings of Uncertainty in Artificial Intelligence, 2015*.

a WMI task, we address the question as to whether fast hashing-based approximate WMC solvers can be leveraged for hybrid domains. What we show is that an NP-oracle can indeed effectively partition the model counting solution space of the more intricate mixed discrete-continuous case using universal hashing. (Of course, volume computation via integration is still necessary, but often over very small spaces.) In this sense, hybrid domains can now be put within reach of approximate WMC solvers.[1] In particular, the hashing approach that we consider here builds on the recent work of Chakraborty et al. (2014) on approximate WMC, and inherits their tolerance-confidence guarantees. In our empirical evaluations, the approximate technique is shown to be significantly faster than an exact WMI solver. We then demonstrate the practical efficacy of the system on a complex real-world dataset where we compute conditional queries over intricate arithmetic constraints that would be difficult (or impossible) to realize in existing formalisms.

Let us finally mention that current inference algorithms for hybrid graphical models often make strong assumptions on the form of the potentials, such as Gaussian distributions, or approximate using variational methods (Murphy 1999). There is also a recent focus on *piecewise-polynomial* potentials (Shenoy and West 2011; Sanner and Abbasnejad 2012), which are based on generalizations of techniques such as the join-tree algorithm. Such piecewise-polynomials can also be represented in the WMI context, but in a general framework allowing arbitrary Boolean connectives and deterministic hard constraints.

Weighted Model Integration

We briefly review the ideas behind WMI.

From a probabilistic perspective, we are imagining a set of Boolean random variables $\mathcal{B}$ and real-valued random variables $\mathcal{X}$. In particular, we let $(\mathbf{b}, \mathbf{x}) = (b_1, \ldots, b_m, x_1, \ldots, x_n)$ be an element of the probability space $\{0,1\}^m \times \mathbb{R}^n$, and we let $\Pr(\mathbf{b}, \mathbf{x})$ denote the probability of the assignments $(\mathbf{b}, \mathbf{x})$ to the variables in $\mathcal{B} \cup \mathcal{X}$. Assuming the joint probability density function Pr can be suitably factorized, *e.g.* via graphical models, we would like to perform probabilistic inference, that is, compute things like the partition function and conditional probabilities (Koller and Friedman 2009).

For the discrete case, that is, when limited to Boolean variables $\mathcal{B}$ only,[2], a prominent approach to perform inference is WMC (Chavira and Darwiche 2008). The idea is to encode the graphical model as a weighted propositional theory, and then compute the model count on this theory. Recall that SAT is the problem of finding a satisfying assignment M to a propositional formula ϕ. WMC, an extension of #SAT, computes the number of models of ϕ, and defines a weight to each of these based on the input theory. Formally, Given a formula Δ in propositional logic over literals $\mathcal{L}$, and a *weight function* $w : \mathcal{L} \to \mathbb{R}$, WMC is defined as:

$$\mathrm{WMC}(\Delta, w) = \sum_{M \models \Delta} w(M)$$

where, $w(M)$ is shorthand for $\prod_{l \in M} w(l)$. Here, given an assignment (or model) M, we write $M \models \phi$ to denote *satisfaction.* We write $l \in M$ to denote the literals (that is, propositions or their negations) that are satisfied at M. We often write $\mathcal{M}(\phi)$ to mean the set of models of ϕ.

The key insight behind WMI is that for hybrid graphical models, we would need to talk about satisfiability and model counting over logical theories with propositions (for $\mathcal{B}$) as well as real-valued variables (for $\mathcal{X}$). This is made possible by *satisfiability modulo theories* (SMT) technology (Barrett et al. 2009). More precisely, in SMT, DPLL is generalized to decide the satisfiability of a (typically quantifier-free) first-order formula with respect to some decidable background theory $\mathcal{T}$. Formally, assume a bijection between ground first-order atoms (from the language of linear arithmetic) and a propositional vocabulary; formula abstraction, denoted ϕ^-, proceeds by replacing the atoms by propositions, and refinement, denoted ϕ^+, replaces the propositions with the atoms. For example, if $\Delta = (x \leq 4) \wedge (x \leq 5)$, then $\Delta^- = p \wedge q$ where (say) p denotes $x \leq 4$ and q denotes $x \leq 5$; also, $q^+ = x \leq 5$. Then, suppose Δ is an linear arithmetic theory over Boolean and rational variables $\mathcal{B}$ and $\mathcal{X}$, and literals $\mathcal{L}$. Suppose $w : \mathcal{L} \to \mathrm{EXPR}(\mathcal{X})$, where $\mathrm{EXPR}(\mathcal{X})$ are expressions over $\mathcal{X}$. WMI is defined as:

$$\mathrm{WMI}(\Delta, w) = \sum_{M \models \Delta^-} \mathrm{VOL}(M, w)$$

$$\text{where, } \mathrm{VOL}(M, w) = \int_{\{l^+ : l \in M\}} w(M)\, d\mathcal{X}.$$

The intuition is as follows. The WMI of a linear arithmetic theory Δ is defined in terms of the models of its propositional abstraction Δ^-. For each such model, we compute its volume, that is, we integrate the weight values of the literals that are true at the model. The interval of the integral is obtained from the refinement of each literal. Finally, $\mathrm{EXPR}(\mathcal{X})$ is the weight function mapping an expression e to its *density function*, which is usually another expression mentioning the variables in e.

To see a very simple example, let $\Delta = (0 \leq x) \wedge (x \leq 10)$, and suppose w maps $(0 \leq x)$ to 2 and $(x \leq 10)$ to x. Suppose $p \wedge q$ is the abstraction. Then Δ has only one model, namely the one where both p and q are true, and we find:

$$\mathrm{VOL}(\{p, q\}, w) = \int_{0 \leq x \leq 10} 2x\, dx = [x^2]_0^{10} = 100.$$

Thus, $\mathrm{WMI}(\Delta, w) = 100$.

The correctness of WMI and the fact that it is a strict generalization of WMC are argued elsewhere (Belle, Passerini, and Van den Broeck 2015).

[1] In an independent and recent effort, Chistikov, Dimitrova, and Majumdar (2015) also introduce the notion of approximate model counting for SMT theories. The most significant difference between the proposals is that they focus only on unweighted model counting. Moreover, they define model counting as a measure on first-order models. Our approach is a simpler one that, as we will see, allows us to cast statements for WMI in terms of WMC.

[2] Handling random variables that take values from finite sets, rather than $\{0,1\}$, is also possible (Sang, Beame, and Kautz 2005), but we omit a discussion on this for simplicity.

Approximating WMI

The purpose of this section is to identify how to approximate $\text{WMI}(\Delta, w)$. As mentioned before, we would like such an algorithm to come with strong theoretical guarantees. To better understand what we offer, consider the well-understood WMC version (Chakraborty et al. 2014):

Definition 1 : Given a propositional sentence Δ and a weight function w, an *exact algorithm* for WMC returns $\text{WMC}(\Delta, w)$. An *approximate algorithm* for WMC given *tolerance* $\epsilon \in (0, 1]$ and *confidence* $1-\delta \in (0, 1]$, simply called an (ϵ, δ)-algorithm, returns a value v such that

$$\Pr\left[\frac{\text{WMC}(\Delta, w)}{1+\epsilon} \leq v \leq (1+\epsilon)\text{WMC}(\Delta, w)\right] \geq 1-\delta$$

Intuitively, when the weight of every model is 1, an exact algorithm returns the size of the set $\mathcal{M}(\Delta) = \{M \mid M \models \Delta\}$ while an approximate one samples from that solution space. The main question, then, is how can we sample from an unknown solution space while offering such tight bounds? We return to this question shortly.

Problem Statement

To see how the above notion applies to our task, consider an SMT theory Δ and weight function w. We observe that

$$\text{WMI}(\Delta, w) = \text{WMC}(\Delta^{\perp}, u)$$

where, for any model M of $\Delta^{\perp}$, u is a weight function such that $u(M) = \text{VOL}(M, w)$. More precisely, u is to be seen as a weight function that does not factorize over literals and directly maps interpretations to $\mathbb{R}$. (This is without any loss of generality.) Thus, our problem statement becomes:

Definition 2: An (ϵ, δ)-algorithm for a WMI problem over Δ and w is an (ϵ, δ)-algorithm for WMC over $\Delta^{\perp}$ and weight function u, where for any model M of $\Delta^{\perp}$, $u(M) = \text{VOL}(M, w)$.

The idea is that by treating the volumes of models as weights over propositional interpretations, we can view WMI simply in terms of WMC. Theoretical results can then be inherited.

There are two caveats, however. First, the weights on the propositional abstraction need to be actually computed using integration during inference. Second, such an algorithm samples feasible satisfying assignments for $\Delta^{\perp}$, but these need not be consistent in arithmetic. For example, if p denotes $x \leq 3$ and q denotes $x \geq 5$, then the interpretation $\{p, q\}$ is not a model in linear arithmetic. We refer interested readers to the full paper (Belle, Van den Broeck, and Passerini 2015) for details on how these are addressed.

Approach

Approximate algorithms for model counting (*i.e.* when the weights are uniform) with strong guarantees have been the focus of many papers, *e.g.* (Jerrum, Valiant, and Vazirani 1986; Karp, Luby, and Madras 1989; Ermon et al. 2013; Chakraborty et al. 2014). The main technical device is the use of *uniform hash functions* (Sipser 1983), a discussion of which we omit here. Roughly speaking, given a propositional theory ϕ, rather than counting $\mathcal{M}(\phi)$ exactly, which is #P-hard, one computes the models of $\phi \wedge \chi$, where χ is a random parity constraint corresponding to the hash function. The parity constraint has the effect of partitioning $\mathcal{M}(\phi)$ into a set of well-balanced *cells*: such a cell is a relatively small subset of the solution space. We count solutions for such cells, which is relatively easy owing to their size, and leverage that count as an estimate for the solution space as a whole. It can be shown that for an efficient family of hash functions, such an approach provides the desired bounds.

At this point, our formulation for approximating WMI is basically agnostic about the counting algorithm used, giving us a direct way to adapt its bounds. In this paper, we demonstrate that by leveraging the work of Chakraborty et al. (2014) (CFMSV henceforth). As argued by Ermon et al. (2013), the one major limitation when applying approximate model counters for probabilistic inference is that weights play an important role in deeming which samples are interesting. Therefore, uniformly sampling from $\mathcal{M}(\phi)$ is not appealing, and would lead to poor estimates of conditional probabilities. The approach taken by CFMSV is to bias the sampling by means of a parameter called *tilt*.

Definition 3: Suppose (Δ, w) is a weighted propositional theory. Let $w_{\max} = \max_M w(M)$ and let $w_{\min} = \min_M w(M)$. We define the *tilt* θ to be the ratio $w_{\max}/w_{\min}$.

They introduce an algorithm $\text{WeightMC}(\Delta, w, \epsilon, \delta, \theta)$ for which they show (our rewording):

Theorem 4 : *[CFMSV] Suppose* (Δ, w) *is a weighted propositional theory, and* ϵ, δ, θ *are as above. Then* $\text{WeightMC}(\Delta, w, \epsilon, \delta, \theta)$ *is an* (ϵ, δ)*-algorithm for* $\text{WMC}(\Delta, w)$*. Given a SAT-oracle, it runs in time polynomial in* $\log_2(1/\delta), \theta, |\Delta|$ *and* $1/\epsilon$ *relative to the oracle.*

For our purposes, we adapt the notion as follows:

Definition 5: Suppose (Δ, w) is a weighted SMT theory. Let $w_{\max} = \max_M \text{VOL}(M, w)$ and let $w_{\min} = \min_M \text{VOL}(M, w)$. We define the *tilt* θ to be the ratio $w_{\max}/w_{\min}$.

We then compute:

$$\text{WMI}(\Delta, w, \epsilon, \delta, \theta) = \text{WeightMC}(\Delta^{\perp}, u, \epsilon, \delta, \theta)$$

where u is calculated using $u(M) = \text{VOL}(M, w)$.

We are (almost) done. The algorithm WeightMC has to be adapted to address the caveats mentioned earlier (theory consistency, computing integrals optimally). We argued in (Belle, Van den Broeck, and Passerini 2015) that this adaption does not affect the algorithm's theoretical properties, which allows us to show:

Corollary 6: *Suppose* Δ *is an SMT theory,* w *is a weight function, and* ϵ, δ, θ *are as above. Suppose* u *is the derived weight function for* $\Delta^{\perp}$*. Then,* $\text{WeightMC}(\Delta^{\perp}, u, \epsilon, \delta, \theta)$ *is an* (ϵ, δ)*-algorithm for* $\text{WMI}(\Delta, w)$*. Suppose we are given an oracle to the weight function* u *and a SAT-oracle. Then,* $\text{WeightMC}(\Delta^{\perp}, u, \epsilon, \delta, \theta)$ *runs in time polynomial in* $\log_2(1/\delta), \theta, |\Delta^{\perp}|$ *and* $1/\epsilon$ *relative to the oracles.*

The oracle to u computes the volumes of $\mathcal{T}$-consistent models, which is shown to be efficient by Baldoni et al. (2011).

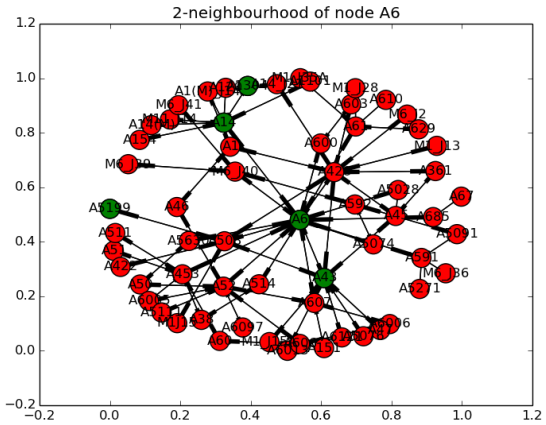

Figure 1: At most two junctions from A6.

Empirical Evaluations

We now study the scaling behavior and expressivity of an approximate inference system on a complex real-world scenario; see (Belle, Van den Broeck, and Passerini 2015) for a comprehensive report and implementation details. The scenario involves computing conditional queries over arithmetic constraints, and is based on a data series released by the UK government that provides the average journey time, speed and traffic flow information on all motorways in England, known as the Strategic Road Network.[3] Motorways are split into junctions, and each information record refers to a specific junction, day and time period. Figure 1 shows the portion of the network around the A6 motorway, limited to at most two junctions from A6.

Imagine a planning problem for a supply system for motorway service stations. The operations center (located, say, somewhere along A6) receives supply requests from a number of stations, and needs to predict whether the delivery vehicle will be able to reach all stations and return within a certain amount of time. Travel time between every pair of stations, and between stations and the operations center, is computed in terms of shortest paths across the network. We compute shortest paths for both minimum and maximum travel times, so as to get a distribution for the shortest path duration wrt every pair of relevant points (stations and operations center). Then, given a certain route between stations, the probability of completing it within the desired time can be computed by integrating over travel time distributions between consecutive stops.

For example, based on our statistical model, the probability of beginning from the operations center at 8 a.m. and completing the route touching A14, A1304, A43, and A5199 by 9 a.m. is: $\Pr(T < 3600) = 0.765$. Here, T is the time taken for the route in seconds. Suppose, however, we request that station A14 should be reached only after visiting A1304 (owing to a delivery request between these two stations) but A1304 should not be visited before 8:30 a.m. (say, because the package to deliver will not be available until then). Then the system would compute: $\Pr(T < 3600 \mid t_{\text{A14}} > t_{\text{A1304}} \wedge t_{\text{A1304}} \geq 1800) = 0.557$. Finally, suppose a last constraint were to require the station A5199 to be also visited after 8:30 a.m. (say, when a package to be delivered to the operations center will be made available). This additional constraint makes it infeasible to complete the route in the required time: $\Pr(T < 3600 \mid t_{\text{A14}} > t_{\text{A1304}} \wedge t_{\text{A1304}} \geq 1800 \wedge t_{\text{A5199}} \geq 1800) = 0$.

We use this construction as a template for considering cycle paths of increasing lengths to study the implementation extensively (see the full paper for details). To the best of our knowledge, a probabilistic inference system for hybrid specifications against intricate Boolean combinations of propositional and arithmetic constraints has not been deployed on such a scale previously.

[3] http://data.gov.uk/dataset/dft-eng-srn-routes-journey-times

Conclusions

We introduced a novel way to leverage a fast hashing-based approximate WMC methodology for inference with discrete and continuous random variables. On the one hand, SAT technology can now be exploited in challenging inference and learning tasks in hybrid domains. On the other, strong tolerance-confidence guarantees can be inherited in this more complex setting. Weighted SMT theories allow a natural encoding of hybrid graphical networks while also admitting the specification of arithmetic constraints in conditional queries, all of which are difficult to realize in standard formalisms. We demonstrated its practical efficacy in a complex novel application, deployed on a scale not considered previously.

References

Baldoni, V.; Berline, N.; De Loera, J.; Köppe, M.; and Vergne, M. 2011. How to integrate a polynomial over a simplex. *Mathematics of Computation* 80(273):297–325.

Barrett, C.; Sebastiani, R.; Seshia, S. A.; and Tinelli, C. 2009. Satisfiability modulo theories. In Biere, A.; Heule, M. J. H.; van Maaren, H.; and Walsh, T., eds., *Handbook of Satisfiability*. IOS Press. chapter 26, 825–885.

Belle, V.; Passerini, A.; and Van den Broeck, G. 2015. Probabilistic inference in hybrid domains by weighted model integration. In *IJCAI*.

Belle, V.; Van den Broeck, G.; and Passerini, A. 2015. Hashing-based approximate probabilistic inference in hybrid domains. In *UAI*.

Chakraborty, S.; Fremont, D. J.; Meel, K. S.; Seshia, S. A.; and Vardi, M. Y. 2014. Distribution-aware sampling and weighted model counting for SAT. *AAAI*.

Chavira, M., and Darwiche, A. 2008. On probabilistic inference by weighted model counting. *Artif. Intell.* 172(6-7):772–799.

Chistikov, D.; Dimitrova, R.; and Majumdar, R. 2015. Approximate counting in SMT and value estimation for probabilistic programs. In *TACAS*.

Ermon, S.; Gomes, C. P.; Sabharwal, A.; and Selman, B. 2013. Embed and project: Discrete sampling with universal hashing. In *NIPS*, 2085–2093.

Fierens, D.; Van den Broeck, G.; Renkens, J.; Shterionov, D.; Gutmann, B.; Thon, I.; Janssens, G.; and De Raedt, L. 2013. Inference and learning in probabilistic logic programs using weighted Boolean formulas. *Theory and Practice of Logic Programming*.

Gomes, C. P.; Sabharwal, A.; and Selman, B. 2009. Model counting. In *Handbook of Satisfiability*. chapter 20.

Jerrum, M. R.; Valiant, L. G.; and Vazirani, V. V. 1986. Random generation of combinatorial structures from a uniform distribution. *Theor. Comput. Sci.* 43(2-3):169–188.

Karp, R. M.; Luby, M.; and Madras, N. 1989. Monte-carlo approximation algorithms for enumeration problems. *J. Algorithms* 10(3):429–448.

Koller, D., and Friedman, N. 2009. *Probabilistic graphical models: principles and techniques*. MIT press.

Murphy, K. P. 1999. A variational approximation for Bayesian networks with discrete and continuous latent variables. In *UAI*, 457–466.

Sang, T.; Beame, P.; and Kautz, H. A. 2005. Performing Bayesian inference by weighted model counting. In *AAAI*, volume 5, 475–481.

Sanner, S., and Abbasnejad, E. 2012. Symbolic variable elimination for discrete and continuous graphical models. In *AAAI*.

Shenoy, P. P., and West, J. C. 2011. Inference in hybrid Bayesian networks using mixtures of polynomials. *International Journal of Approximate Reasoning* 52(5):641–657.

Sipser, M. 1983. A complexity theoretic approach to randomness. In *STOC*, 330–335. ACM.

Stockmeyer, L. 1983. The complexity of approximate counting. In *STOC*, 118–126. New York, NY, USA: ACM.

Suciu, D.; Olteanu, D.; Ré, C.; and Koch, C. 2011. Probabilistic databases. *Synthesis Lectures on Data Management* 3(2):1–180.

Valiant, L. G. 1979. The complexity of enumeration and reliability problems. *SIAM Journal on Computing* 8(3):410–421.