Relax, compensate and then integrate

Zhe Zeng[1], Paolo Morettin[2], Fanqi Yan[3], Antonio Vergari[1], and
Guy Van den Broeck[1]

[1] University of California, Los Angeles {zhezeng,aver,guyvdb}@cs.ucla.edu
[2] University of Trento, Italy paolo.morettin@unitn.it
[3] AMSS Chinese Academy of Sciences fanqi_yan@lsec.cc.ac.cn

1 WMI for advanced, real-world probabilistic inference

Consider an autonomous agent operating in the real-world and under uncertainty, e.g., a self-driving car. It would have to model both *continuous* variables like the speed and position of other cars and *discrete* ones like the color of traffic lights and the number of pedestrians. Moreover, to take decisions, it would also need to perform probabilistic reasoning over complex *algebraic constraints*, such as the geometry of vehicles and roads ahead and the laws of physics.

Performing probabilistic inference in these constrained and hybrid (mixed continuous-discrete) scenarios goes beyond the limited inference capabilities of probabilistic models such as variational autoencoders [10] and generative adversarial networks [8]. This is also the case for classical probabilistic graphical models for hybrid domains [9, 13] and more recent tractable ones [14, 16, 17] which struggle to either perform inference over complex algebraic constraints or make too simplistic representational or distributional assumptions.

On the other hand, Weighted Model Integration (WMI) [2, 15] supports general hybrid probabilistic reasoning over algebraic constraints, *by design*: mixed complex continuous-discrete interactions can be easily expressed in the language of Satisfiability Modulo Theories (SMT) [1] and answering probabilistic queries involving algebraic constraints can be naturally cast as integration of certain weight functions over the regions that satisfy those constraints. Consequently, much attention has been posed to devising more and more sophisticated general-purpose WMI solvers [15, 20, 11, 12]. However, they are generally oblivious to the structure of the problem at hand and as such do not scale. Here we advance the WMI framework on two fronts: we i) deepen the theoretical understanding of the complexity of WMI inference on real-world problems by proving some hardness results; and we ii) deliver an efficient and accurate approximate WMI solver, ReCoIn, as a practical algorithmic solution to deploy WMI in the real-world.

2 Tracing the boundaries of tractable WMI classes

Recent works have started looking for *classes of tractable WMI problems*, i.e., problems for which a solution can be computed exactly in polytime [18, 19]. These classes of problems can be characterized by two parameters: the *treewidth*

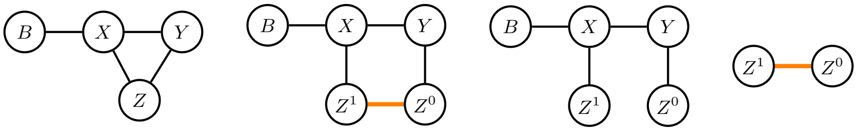

Fig. 1. Relaxing a WMI problem with an SMT formula Δ with loopy primal graph (left): first a copy of Z is added and connected in the augmented formula Δ^{aug} (center left), then the added edge (orange) is removed yielding a relaxed formula Δ^{rel} (center right) amenable to tractable inference and a remaining part Δ^{rem} (right).

and the *diameter* of the primal graph [7] of the SMT formulas considered, where the latter is generally expressed as a function of the number of variables in the problem. We provide the necessary conditions for the largest class of tractable WMI known so far[4], introduced in [19] by defining some sufficient conditions over the treewidth and diameter length of the primal graph of a WMI problem.

Theorem 1. *Let $\mathcal{WMI}(\boldsymbol{\Omega}, \log(n), t)$ be the class of WMI problems whose primal graph has treewidth t and diameter that is $\Theta(\log(n))$, where n is the number of variables in the problem and whose parametric weight function family $\boldsymbol{\Omega}$ satisfies the conditions stated in [19]. Then $\mathcal{WMI}(\boldsymbol{\Omega}, \log(n), t)$ is a tractable WMI class for inference if-and-only-if treewidth $t = 1$.*

Our complexity result sets the standard in the landscape of WMI solvers: every exact solver that aims to be efficient, needs to operate in the regime of Theorem 1. However, real-world problems do not always conform to the structural desiderata for primal graphs stated in it. This implies that efficient approximations might not only be useful in these scenarios, but *needed*. We fill this gap, by introducing ReCoIn which performs approximate inference on intractable WMI problems, by relaxing them into a tractable version in $\mathcal{WMI}(\boldsymbol{\Omega}, \log(n), t)$.

3 ReCoIn: efficient approximate WMI inference

ReCoIn comprises three phases: i) *RElaxing* an intractable WMI problem into a simpler one amenable to exact inference by removing dependencies from it; then ii) introduce certain literals and weights to *COmpensate* for the dependency structure lost in relaxation and iii) optimize them by solving a series of exact *INtegration* problems. As such, it can be cast within the *relax-compensate-recovery* (RCR) framework [4–6] for approximate inference.

Algorithm 1 sketches the main phases involved. First, given a WMI problem with an SMT formula Δ and weight functions $\mathcal{W}$, we mark some set of edges $\mathcal{E}_d$ in the primal graph of Δ to be removed to obtain a tree-shaped primal graph

[4] W.l.o.g., we consider WMI problems over continuous variables only by leveraging the polytime reduction of [18] from WMI problems with continuous-discrete variables.

Algorithm 1 ReCoIn $(\mathit{\Delta}, \mathcal{W}, K)$

```
Input: a WMI model $(\mathit{\Delta}, \mathcal{W})$, $K$ number of compensating literals
Output: $(\mathit{\Delta}^{\mathsf{rel}}, \mathcal{W}^{\mathsf{rel}})$: relaxed and compensate WMI problem.
$\mathcal{E}_d \leftarrow \mathsf{selectEdgesToRemove}(\mathit{\Delta}, \mathcal{W})$        ▷ Select edges to remove
$\mathit{\Delta}^{\mathsf{aug}}, \mathcal{W}^{\mathsf{aug}}, \mathcal{L} \leftarrow \mathsf{augmentModel}(\mathit{\Delta}, \mathcal{W}, \mathcal{E}_d)$
$(\mathit{\Delta}^{\mathsf{rel}}, \mathcal{W}^{\mathsf{rel}}), (\mathit{\Delta}^{\mathsf{rem}}, \mathcal{W}^{\mathsf{rem}}) \leftarrow \mathsf{relaxModel}(\mathit{\Delta}^{\mathsf{aug}}, \mathcal{W}^{\mathsf{aug}}, \mathcal{L})$
$\mathit{\Delta}^{\mathsf{rel}}, \mathcal{W}^{\mathsf{rel}} \leftarrow \mathsf{addingCompensations}(\mathit{\Delta}^{\mathsf{rel}}, \mathcal{W}^{\mathsf{rel}}, \mathcal{L}, K)$
while not converged do
    for $X_i \in \mathsf{copiedNodes}(\mathit{\Delta}^{\mathsf{rel}})$ do
        for $k \in [K]$ do $r^k \leftarrow \mathsf{WMI}(\mathit{\Delta}^{\mathsf{rem}}, \mathcal{W}^{\mathsf{rem}}) \ / \ \mathsf{WMI}(\mathit{\Delta}^{\mathsf{rem}} \wedge \bigwedge_{c=0}^{C_i} \ \ell_{k,i}^{c}, \mathcal{W}^{\mathsf{rem}}) - 1$
        for $c \in [C_i]$ do $\theta_{k,i}^{c,(t+1)} \leftarrow \log(r^k \alpha_{k,\sigma(c)}) - \log(1 - \alpha_{k,\sigma(c)}) - \sum_{c' \neq c} \theta_{k,i}^{c,(t)}$
Return $(\mathit{\Delta}^{\mathsf{rel}}, \mathcal{W}^{\mathsf{rel}})$
```

with bounded diameter. For each edge $X_i - X_j \in \mathcal{E}_d$, we copy one of its variables, say X_i, into X_i^c, and augment formula $\mathit{\Delta}$ into $\mathit{\Delta}^{\mathsf{aug}}$ by introducing literals representing equality constraints $\hat{\ell} : (X_i^c = X_i)$ and properly renaming occurrences of X_i in $\mathit{\Delta}^{\mathsf{aug}}$. In essence, this augmentation substitutes the dependency $X_i - X_j$ by $X_i - X_i^c - X_j$. Then, we break $\mathit{\Delta}^{\mathsf{aug}}$ into a relaxed formula $\mathit{\Delta}^{\mathsf{rel}}$ and a remaining part $\mathit{\Delta}^{\mathsf{rem}}$ by removing the previously introduced equivalence constraints between the copies. As a result, we sparsified the primal graph of $\mathit{\Delta}^{\mathsf{rel}}$ to make it amenable to exact and efficient inference. Figure 1 illustrates the process. Later, we retrieve the lost constraints by optimization. As in other RCR schemes [4, 3] we introduce K compensating literals $\{\ell_{i,k}^c\}_{k=1}^K$ between each X_i, and its copies X_i^c and equip them with compensating weight functions $w_{\ell_{i,k}^c} := \exp(\theta_{i,k}^c)$. Lastly, we iteratively optimize the compensating weights parameters $\theta_{i,k}^c$ such that we recover the probability of the compensating literals across the different copies. That is, we want to satisfy the following constraints:

$$\mathsf{Pr}_{\mathit{\Delta}^{\mathsf{rem}}}\left(\bigwedge_{c=0}^{C_i} \ \ell_{k,i}^c\right) = \mathsf{Pr}_{\mathit{\Delta}^{\mathsf{rel}}}\left(\ell_{k,i}^0\right) = \cdots = \mathsf{Pr}_{\mathit{\Delta}^{\mathsf{rel}}}\left(\ell_{k,i}^{C_i}\right), \quad \textit{for} \ \ k = 1, \cdots, K. \tag{1}$$

Therefore, at each iteration t, we need to solve $2K$ integration problems for computing $r^k = \mathsf{WMI}(\mathit{\Delta}^{\mathsf{rem}} \wedge \bigwedge_{c=0}^{C_i} \ \neg\ell_{k,i}^c, \mathcal{W}^{\mathsf{rem}}) \ / \ \mathsf{WMI}(\mathit{\Delta}^{\mathsf{rem}} \wedge \bigwedge_{c=0}^{C_i} \ \ell_{k,i}^c, \mathcal{W}^{\mathsf{rem}})$ terms and $C_i \cdot K$ integrations for $\alpha_{k,\sigma(c=} = \mathsf{Pr}_{\mathit{\Delta}^{\mathsf{rel}}}(\ell_{k,i}^{\pi(c)})$ for each pair of a variable and its copies, for an arbitrary permutation π of the copies. To this end we adopt MP-WMI [19] to solve these tractable WMI problems because it is the fastest solver yet for tree-shaped and bounded diameter problems, and even more importantly, it allows to *amortize inference across queries*. That is, we can compute all the $C_i \cdot K$ literal probabilities in a single message-passing step.

In our preliminary experiments over synthetic WMI problems with loopy primal graphs, ReCoIn delivers more accurate answers to probabilistic queries than alternatives like rejection sampling [12] and scale to larger problems than exact solvers like PA [15] or approximate ones like XSDD (Sampling) [20].

References

1. Barrett, C., de Moura, L., Ranise, S., Stump, A., Tinelli, C.: The smt-lib initiative and the rise of smt (hvc 2010 award talk). In: Proceedings of the 6th international conference on Hardware and software: verification and testing. pp. 3–3. Springer-Verlag (2010)
2. Belle, V., Passerini, A., Van den Broeck, G.: Probabilistic inference in hybrid domains by weighted model integration. In: Proceedings of 24th International Joint Conference on Artificial Intelligence (IJCAI). pp. 2770–2776 (2015)
3. Van den Broeck, G., Choi, A., Darwiche, A.: Lifted relax, compensate and then recover: From approximate to exact lifted probabilistic inference. In: Proceedings of the 28th conference on uncertainty in artificial intelligence (UAI). pp. 131–141 (2012)
4. Choi, A., Darwiche, A.: An edge deletion semantics for belief propagation and its practical impact on approximation quality. In: Proceedings of the National Conference on Artificial Intelligence. vol. 21, p. 1107. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999 (2006)
5. Choi, A., Darwiche, A.: Relax, compensate and then recover. In: JSAI International Symposium on Artificial Intelligence. pp. 167–180. Springer (2010)
6. Choi, A., Darwiche, A.: Approximating the partition function by deleting and then correcting for model edges. arXiv preprint arXiv:1206.3241 (2012)
7. Dechter, R., Mateescu, R.: And/or search spaces for graphical models. Artificial intelligence **171**(2-3), 73–106 (2007)
8. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
9. Heckerman, D., Geiger, D.: Learning bayesian networks: a unification for discrete and gaussian domains. In: Proceedings of the Eleventh conference on Uncertainty in artificial intelligence. pp. 274–284. Morgan Kaufmann Publishers Inc. (1995)
10. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
11. Kolb, S., Mladenov, M., Sanner, S., Belle, V., Kersting, K.: Efficient symbolic integration for probabilistic inference. In: IJCAI. pp. 5031–5037 (2018)
12. Kolb, S., Morettin, P., Zuidberg Dos Martires, P., Sommavilla, F., Passerini, A., Sebastiani, R., De Raedt, L.: The pywmi framework and toolbox for probabilistic inference using weighted model integration. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19. pp. 6530–6532 (7 2019). https://doi.org/10.24963/ijcai.2019/946
13. Lauritzen, S.L., Wermuth, N.: Graphical models for associations between variables, some of which are qualitative and some quantitative. The annals of Statistics pp. 31–57 (1989)
14. Molina, A., Vergari, A., Di Mauro, N., Natarajan, S., Esposito, F., Kersting, K.: Mixed sum-product networks: A deep architecture for hybrid domains. In: Thirty-second AAAI conference on artificial intelligence (2018)
15. Morettin, P., Passerini, A., Sebastiani, R.: Efficient weighted model integration via smt-based predicate abstraction. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence. pp. 720–728. AAAI Press (2017)
16. Vergari, A., Molina, A., Peharz, R., Ghahramani, Z., Kersting, K., Valera, I.: Automatic bayesian density analysis. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 5207–5215 (2019)

17. Yang, E., Baker, Y., Ravikumar, P., Allen, G., Liu, Z.: Mixed graphical models via exponential families. In: Artificial Intelligence and Statistics. pp. 1042–1050 (2014)
18. Zeng, Z., Van den Broeck, G.: Efficient search-based weighted model integration. Proceedings of UAI (2019)
19. Zeng, Z., Morettin, P., Yan, F., Vergari, A., Broeck, G.V.d.: Scaling up hybrid probabilistic inference with logical and arithmetic constraints via message passing. In: International Conference of Machine Learning (2020)
20. Zuidberg Dos Martires, P.M., Dries, A., De Raedt, L.: Exact and approximate weighted model integration with probability density functions using knowledge compilation. In: Proceedings of the 30th Conference on Artificial Intelligence. AAAI Press (2019)