

# Technical Perspective

## Combining Logic and Probability

By Henry Kautz and Parag Singla

A GOAL OF research in artificial intelligence and machine learning since the early days of expert systems has been to develop automated reasoning methods that combine logic and probability. Probabilistic theorem proving (PTP) unifies three areas of research in computer science: reasoning under uncertainty, theorem-proving in first-order logic, and satisfiability testing for propositional logic.

Why is there a need to combine logic and probability? Probability theory allows one to quantify uncertainty over a set of propositions—ground facts about the world—and a probabilistic reasoning system allows one to infer the probability of unknown (hidden) propositions conditioned on the knowledge of other propositions. However, probability theory alone has nothing to say about how propositions are constructed from relationships over entities or tuples of entities, and how general knowledge at the level of relationships is to be represented and applied.

Handling relations takes us into the domain of first-order logic. An important case is collective classification, where the hidden properties of a set of entities depend in part on the relationships between the entities. For example, the probability that a woman contracts cancer is not independent of her sister contracting cancer. Many applications in medicine, biology, natural language processing, computer vision, the social sciences, and other domains require reasoning about relationships under uncertainty.

Researchers in AI have proposed a number of representation languages and algorithms for combining logic and probability over the past decade, culminating in the emergence of the research area named statistical relational learning (SRL).

The initial approaches to SRL used a logical specification of a domain annotated with probabilities (or weights) as a template for instantiating, or grounding, a propositional probabilistic representation, which is then

solved by a traditional probabilistic reasoning engine. More recently, research has centered on the problem of developing algorithms for probabilistic reasoning that can efficiently handle formulas containing quantified variables without grounding—a process called “lifted inference.”

Well before the advent of SRL, work in automated theorem-proving had split into two camps. One pursued algorithms for quantified logic based on the syntactic manipulation of formulas using unification and resolution. The other pursued algorithms for propositional logic based on model finding; that is, searching the space of truth assignments for one that makes the formula true. At the risk of oversimplifying history, the most successful approach for fully automated theorem-proving is model finding by depth-first search over partial truth assignments, the Davis-Putnam-Logemann-Loveland procedure (DPLL).


There is a simple but profound connection between model finding and propositional probabilistic reasoning. Suppose each truth assignment, or model, came with an associated positive weight, and the weights summed to (or could be normalized to sum to) one. This defines a probability distribution. DPLL can be modified in a straightforward manner to either find the most heavily weighted model or to compute the sum of the weights of the

models that satisfy a formula.

The former can be used to perform Maximum a Posteriori (MAP) inference, and the latter, called “weighted model counting,” to perform marginal inference or to compute the partition function.

PTP provides the final step in unifying probabilistic, propositional, and first-order inference. PTP lifts a weighted version of DPLL by allowing it to branch on a logical expression containing uninstantiated variables. In the best case, PTP can perform weighted model counting while only grounding a small part of a large relational probabilistic theory.

SRL is a highly active research area, where many of the ideas in PTP appear in various forms. There are lifted versions of other exact inference algorithms such as variable elimination, as well as lifted versions of approximate algorithms such as belief propagation and variational inference. Approximate inference is often the best one can hope for in large, complex domains. Gogate and Domingos suggest how PTP could be turned in a fast approximate algorithm by sampling from the set of children of a branch point.

PTP sparks many interesting directions for future research. Algorithms must be developed to quickly identify good literals for lifted branching and decomposition. Approximate versions of PTP need to be fleshed out and evaluated against traditional methods for probabilistic inference. Finally, the development of a lifted version of DPLL suggests that researchers working on logical theorem proving revisit the traditional divide between syntactic methods for quantified logics and model-finding methods for propositional logic. 

**There is a simple but profound connection between model finding and propositional probabilistic reasoning.**

**Henry Kautz** is the Robin and Tim Wentworth Director of the Goergen Institute for Data Science and a professor of computer science at the University of Rochester, New York.

**Parag Singla** is an assistant professor of computer science at the Indian Institute of Technology, New Delhi.

Copyright held by authors.