Probabilistic Finite Domains: A Brief Overview

Nicos Angelopoulos

Department of Computing, Imperial College,
London, SW7 2BZ, UK. nicos@doc.ic.ac.uk

Abstract. We propose a new way of extending Logic Programming (LP) for reasoning with uncertainty. Probabilistic finite domains (*Pfd*) capitalise on ideas introduced by Constraint LP, on how to extend the reasoning capabilities of the LP engine. Unlike other approaches to the field, *Pfd* syntax can be intuitively related to the axioms defining Probability and to the underlying concepts of Probability Theory, (PT) such as sample space, events, and probability function.
Probabilistic variables are core computational units and have two parts. Firstly, a *finite domain*, which at each stage holds the collection of possible values that can be assigned to the variable, and secondly a *probabilistic function* that can be used to assign probabilities to the elements of the domain. The two constituents are kept in isolation from each other. There are two benefits in such an approach. Firstly, that propagation techniques from finite domains research are retained, since a domain's representation is not altered. Thus, a probabilistic variable continues to behave as a finite domain variable. Secondly, that the probabilistic function captures the probabilistic behaviour of the variable in a manner which is, to a large extent, independent of the particular domain values.
The notion of *events* as used in PT can be captured by LP predicates containing probabilistic variables and the derives operator ($\vdash$) as defined in LP. *Pfd* stores hold conditional constraints which are a computationally useful restriction of conditional probability from PT. Conditional constraints are defined by $D_1 : \pi_1 \oplus \ldots \oplus D_n : \pi_n \;\mathbf{|}\; Q_1 \wedge \ldots \wedge Q_m$ where, D_i and Q_j are predicates and each π_i is a probability measure ($0 \leq \pi_i \leq 1$, $1 \leq i \leq n$, $1 \leq j \leq m$). The conjuction of Q_j's qualifies probabilistic knowledge about D_i. In particular, the constraint is evidence that the probability of D_i in the qualified cases (i.e. when $\vdash Q_1, \ldots, Q_m$) is equal to π_i. On the other hand a conditional provides no evidence for the cases where $\not\vdash Q_1, \ldots, Q_m$.
Pfd has been used to model a well known example, the Monty Hall problem, which is often used to caution about the counter-intuitive results when reasoning with probabilities. Analysis of the computations over this model, has shown that *Pfd* emulates extensional methods that are used in statistics.
The main benefits of our approach are (i) minimal changes of the core LP paradigm, and (ii) clear and intuitive way for arriving at probabilistic statements. Intuitiveness of probabilistic computations is facilitated by, (a) separation of the finite domain and the probability assigning function of a variable, and (b) using predicates to represent composite events.