Answer Set Optimization with Dependence Based Priority Propagation

Zhizheng Zhang
School of Computer Science and Engineering, Southeast University, Nanjing, China
Jiangsu Provincial Key Laboratory of Computer Information Processing Technology, Suzhou University, China
seu_zzz@seu.edu.cn

Abstract

We propose a new interpretation for a certain kind of ASO preference rules to handling prioritized symptoms. Our interpretation is based on the idea of dependence based priority propagation. Then, we present a fixpoint based answer set optimization method under such an interpretation.

Introduction

Answer set optimization (ASO) programs (Brewka et al. 2003, Brewka 2004) are powerful tools for modeling decoupled scenarios where the description of preferences are independent of a domain theory. An ASO program is a binary tuple (*P*, *Prefs*) where *P* is a domain theory encoded as an answer set program, and *Prefs* is a preferences description with a set of preference rules of the form (1).

$$pr: C_1{:}p_1 > \ldots > C_k{:}p_k \leftarrow l_1,\ldots,l_m, \text{not } l_{m+1},\ldots, \text{not } l_n \quad (1)$$

where *pr* is the rule's name. l_is are literals and C_is are Boolean combinations over literals and default negated literals. p_is are numerical penalties satisfying $p_i<p_j$ whenever $i<j$. Let $pos(pr)$ mark $\{l_1,\ldots,l_m\}$, and $neg(pr)$ mark $\{l_{m+1},\ldots,l_n\}$. Intuitively, a preference rule is interpreted as: given two outcomes S_1, S_2 such that both contain $l_1,\ldots,l_m$ and do not contain $l_{m+1},\ldots,l_n$, then S_1 is preferred to S_2 if $\exists j(S_1\models C_j)$ and $j<min\{i|S_2\models C_i\}$.

Often what is possible is determined by external factors while what is preferred is described independently by different agents (Brewka et al. 2009). So, ASO makes the overall setting well aligned with applications. However, the above interpretation of preference rules cannot handle prioritized symptoms illustrated in the following example.

Motivation Example. Medical knowledge system assists doctors in making treatment. Symptoms priorities are important in finding the best one of treatments produced from the domain theory. Given a medical domain theory and facts encoded as an answer set program *P* as follows.

r_1: $in\_bl\_co(X) \leftarrow suffer(X, emptysis)$, not $\neg in\_bl\_co(X)$
r_2: $de\_bl\_co(X) \leftarrow suffer(X, hlp)$, not $\neg de\_bl\_co(X)$
r_3: $drug(X, a) \leftarrow in\_bl\_co(X)$, not $drug(X, b)$
r_4: $drug(X, b) \leftarrow de\_bl\_co(X)$, not $drug(X, a)$
r_5: $dosage(X, a, half) \leftarrow de\_bl\_co(X), drug(X, a)$
r_6: $dosage(X, b, half) \leftarrow in\_bl\_co(X), drug(X, b)$
r_7: $suffer(tom, hlp) \leftarrow$ r_8: $suffer(tom, emptysis) \leftarrow$

where *in_bl_co*, *de_bl_co*, and *hlp* denote *increasing blood coagulation*, *decreasing blood coagulation*, *high blood pressure* respectively, *a* and *b* are drugs names. Moreover, for a patient *tom*, symptom priority is given as "*emptysis* has higher priority than *high blood pressure*.". In this case, two medical treatments are generated as answer sets of *P*:

$S_1=\{suffer(tom, hlp), suffer(tom, emptysis), in\_bl\_co(tom), de\_bl\_co(tom), drug(tom, a), dosage(tom, a, half)\}$
$S_2=\{suffer(tom, hlp), suffer(tom, emptysis), in\_bl\_co(tom), de\_bl\_co(tom), drug(tom, b), dosage(tom, b, half)\}$

According to the symptom priority, S_1 is better than S_2. The underlying logic is that "*emptysis* has higher priority than *high blood pressure*" implies "*increasing blood coagulation* precedes *decreasing blood coagulation*" implies "*a* is preferred to *b*". We call such a pattern of reasoning as dependence based priority propagation (DPP), which means the higher the priority of the (default negated) literal, the higher the priority of its dependent literals. From the example, it can be concluded that symptom priority can be naturally expressed in ASO preference rule form: $suffer(X, emptysis){:}0.1>suffer(X, hlp){:}0.2\leftarrow$. But, the DPP meaning of the rule is different from that in ASO. In ASO, it cannot even rank S_1 and S_2. Next, we present an optimization method in DPP reasoning for a certain kind of ASO programs, in which each preference rule has prioritized (default negated) literals as head.

Fixpoint based Answer set Optimization

Given an answer set program *P*, *S* is an answer set of *P*, and *pr* is a preference rule of form (1), but C_i is either a literal or a default negated literal. $\forall l\in S$, we use $pe^S_{pr}(l)$ to denote the penalty of literal *l* w.r.t. *pr* and *S*. An answer set program rule is of form $l_1\vee\ldots\vee l_k\leftarrow l_{k+1},\ldots,l_m$, not $l_{m+1},\ldots$, not l_n. where l_is are literals. Let $head(r)=\{l_1,\ldots,l_k\}$, $pos(r)=\{l_{k+1},\ldots,l_m\}$, $neg(r)=\{l_{m+1},\ldots,l_n\}$, $body(r)=\{l_{k+1},\ldots,l_m$, *not*

$l_{m+1},\ldots,$ *not* $l_n\}$. Then let P^S denote a grounded rules set: $\{r\in P|head\ (r)\cap S\neq\emptyset\wedge pos(r)\subseteq S\wedge neg(r)\cap S=\emptyset\}$. For $r\in P^S$, we mark the degree of *body*(r) w.r.t. *pr* and *S* with $rkB^S_{pr}(r)$. Before formally defining these functions, we introduce the basic DPP idea in natural language as follows: *the penalty of a literal is determined by the priority degree of the body of the applied rule with the literal as head. The priority degree of the rule's body is computed by combining penalties of (default negated) literals in it. So, priority is propagated based on a dependence relation, and the computation of literals penalties is iterative*.

Formally, we present the computation of the functions as follows ($\bigotimes$, $\bigoplus$ are generalized combining operators):

(1) if *S* doesn't satisfy *pr*, i.e., $pos(pr)\not\subseteq S$ or $neg(pr)\cap S\neq\emptyset$

$\forall l \in S,\ pe^S_{pr}(l) = U$ *%%U means undefined*

(2) if *S* satisfies *pr*, i.e., $pos(pr)\subseteq S$ or $neg(pr)\cap S=\emptyset$

-Initializing:

i $S^{irr} = \{l \in S | \nexists C_i(l = C_i)\}$ *%%* S^{irr} *is the set of %%literals cannot be directly ranked by preference rule*.

ii $\forall r \in P^S$,

$$rkB^{(0)S}_{pr}(r) = \begin{cases} \left(\bigotimes\limits_{C_i \in body(r)} p_i\right) & \exists j, C_j \in b(r) \\ U & \text{otherwise} \end{cases}$$

%%Initially ranking rule's body by combining the satisfied %%literals penalties in pr

$$S^{irr}_r = S^{irr} \cap pos(r)$$

iii $\forall l \in S, pe^{(0)S}_{pr}(l) = \bigoplus\limits_{\substack{r\in P^S \\ l\in head(r)}} rkB^{(0)S}_{pr}(r)$

%%initial penalty of l is determined by the initial ranking %%degrees of bodies of rules with l as head.

-Iterating:

$$\forall l \in S, pe^{(n+1)S}_{pr}(l) = \bigoplus_{\substack{r\in P^S \\ l\in head(r)}} rkB^{(n+1)S}_{pr}(r)$$

%%the penalty of l changes with the ranking degrees of %%bodies of rules with l as head

where for every $r \in P^S$

$$rkB^{(n+1)S}_{pr}(r) = rkB^{(n)S}_{pr}(r) \bigotimes \left(\bigotimes_{l\in S^{irr}_r} pe^{(n)S}_{pr}(l)\right)$$

%%the ranking degree of rule's body changed with penalty %%of literals in it.

Theorem 1.Let *D* be a finite chain of [0, 1], if the penalty domain is $D\cup\{U\}$, and $\bigotimes$, $\bigoplus$ are discrete (co-)t-norms on $D\cup\{U\}$ with *U* as neutral element, then there exists a natural number *n* such tha$\forall l \in S, pe^{(n+1)S}_{pr}(l) = pe^{(n)S}_{pr}(l)$.

Proof. $\forall l \in S, pe^{(k)S}_{pr}(l)$ is monotonic w.r.t *k* on the complete lattice *D*. According to *Knaster–Tarski fixpoint theory*, $pe^S_{pr}(l)$ has a fixpoint. Note that any finite chain can be one-one mapped on a finite chain of [0, 1].

Definition 1. Given an answer set program *P*, a preference rule *pr*. If both S_1 and S_2 are answer sets of *P*, then S_1 is preferred to S_2 w.r.t *pr* in DPP reasoning iff

(1)$\exists l \in S_1/S_2$,$\exists l' \in S_2/S_1$ satisfy $pe^S_{pr}(l) < pe^S_{pr}(l')$ and

(2)$\forall l' \in S_2/S_1, \nexists l \in S_1/S_2$ satisfy $pe^S_{pr}(l') < pe^S_{pr}(l)$

***Continue* Motivation Example**. let $\bigotimes$, $\bigoplus$ be *min* and *max*, symptom priority be written in *pr*: *suffer*(X, *emptysis*):0.1> *suffer*(X, *hlp*):0.2←, then penalties is computed as follows.

***-Initializing*:** *%%Assume* $r_1,\ldots r_6$ *are grounded by X=tom.*

$pe^{(0)S_1}_{pr}(in\_bl\_co(tom))=rkB^{(0)S_1}_{pr}(r_1)=0.1$,

$pe^{(0)S_1}_{pr}(de\_bl\_co(tom))=rkB^{(0)S_1}_{pr}(r_2)=0.2$,

$pe^{(0)S_1}_{pr}(drug(tom, a))=rkB^{(0)S_1}_{pr}(r_3)= U$

$pe^{(0)S_1}_{pr}(dosage(tom, a, half))=rkB^{(0)S_1}_{pr}(r_5) =U$

$pe^{(0)S_1}_{pr}(suffer(tom, hlp))= rkB^{(0)S_1}_{pr}(r_7)= U$

$pe^{(0)S_1}_{pr}(suffer(tom, emptysis)) = rkB^{(0)S_1}_{pr}(r_8)=U$

-Iterating: *%%only list functions whose value are changed*

round 1. $pe^{(1)S_1}_{pr}(drug(tom, a)) = rkB^{(1)S_1}_{pr}(r_3) = 0.1$

$pe^{(1)S_1}_{pr}(dasage(tom, a, half)) = rkB^{(1)S_1}_{pr}(r_5) = 0.2$

round 2. $pe^{(2)S_1}_{pr}(dasage(tom, a, half)) = rkB^{(2)S_1}_{pr}(r_5) = 0.1$

-Fixpoint:

$pe^{S_1}_{pr}(suffer(tom, hlp))=pe^{S_1}_{pr}(suffer(tom, emptysis))=U$

$pe^{S_1}_{pr}(in\_bl\_co(tom))=1$, $pe^{S_1}_{pr}(de\_bl\_co(tom))=2$

$pe^{S_1}_{pr}(drug(tom, a))=1$, $pe^{S_1}_{pr}(dosage(tom, a, half))=1$

From the same method, we can get the fixpoint w.r.t S_2, *pr*:

$pe^{S_2}_{pr}(suffer(tom, hlp))=pe^{S_2}_{pr}(suffer(tom, emptysis))=U$

$pe^{S_1}_{pr}(in\_bl\_co(tom))=1$, $pe^{S_2}_{pr}(de\_bl\_co(tom))=2$

$pe^{S_1}_{pr}(drug(tom, b))=2$, $pe^{S_1}_{pr}(dosage(tom, a, half))=2$

According to *definition* 1, S_1 is preferred to S_2.

Note that if multiple preference rules are necessary, for preferences combination, we can easily refer to [1,2].

Conclusion

We propose DPP reasoning for handling prioritized symptoms. Actually, a new semantics is introduced for a certain kind of ASO programs.

Acknowledge

Supported by China NSF under Grant No. 60803061, No. 60873153, NSF of Jiangsu under Grant No. BK2008293.

References

Brewka, G., Niemelä, I., and Truszczynski, M., Answer Set Optimization, In *Proc. IJCAI-03*, 867-872, 2003.

Brewka, G., Complex Preferences for Answer Set Optimization, In *Proc. KR04*, 213-223, 2004.

Brewka, G., Niemelä, I., and Truszczynski, M. 2009, Preferences and Nonmonotonic Reasoning, *AI Magazine*, 29(4): 69-78, 2009.

Mayor, G. and Torrens, J., On a Class of Operators for Expert Systems, *Internat. J. Intell. Sys*. 8:771-778, 1993.