7B-1

論理プログラミングによる優先順位付き極小限定の計算と性能評価

富田 一夫† 若木 利子†

芝浦工業大学 システム工学部†

1 はじめに

人間が日常的に行っている非単調な常識推論を形式化した極小限定の枠組みが, 1980 年代に McCarthy や Lifschitz[6, 8] によって提案された. これまで極小限定を計算する方法は数多く提案されたが, 計算量等が問題となり, システム実装に至る方法は非常に少なく, 性能評価結果はほとんど存在していない. 近年, 我々は優先順位付き極小限定のモデルを解集合プログラミング (ASP)[5] と生成検査法により, 効率的に計算する方法 [9] を提案した. 本研究では, 優先順位付き極小限定を上記方法の生成 (generator) と検査 (tester) の 2 つの論理プログラムを統合した 1 つの論理プログラムに直接変換して計算する方法を提案し実装するとともに, これらと circ2dlp[3] との性能評価を行った. 今回提案する方法を実装したシステム (circum2) は, 多くの点で優位性があるという結果が得られたので, これを報告する.

2 優先順位付き極小限定

極小限定の構文と意味論, 及び命題を以下に示す.

定義 1 [6] $A(P,Z)$ を一階述語論理式とする. A に含まれる述語に関して, P を極小化述語の集合, Z を可変述語の集合, Q を固定述語の集合とする. この時, P を k 個に分割した集合を $P^1,\cdots,P^k$ とし, P^i に属する述語が P^{i+1} に属する述語よりも優先的に極小化される優先順位付き極小限定 $Circum(A;P^1{>}\cdots{>}P^k;Z)$ は次の二階述語論理式：

$$A(P,Z)\wedge\neg\exists pz(A(p,z)\wedge p\prec P)$$

で定義される. 但し, $\prec$ の定義は文献 [6] を参照. p,z は述語変数. $k=1$ の場合は並列極小限定に対応する.

以後, A の Herbrand 基底を H, H に属する P,Q,Z の基礎原子式のそれぞれの集合を $\mathcal{P},\mathcal{Q},\mathcal{Z}$ で表す.

定義 2 [8] 任意の 2 つの構造 M_1,M_2 について, 次の (i), (ii), (iii) を満たす時, $M_1\leq^{P^1>\cdots>P^k;Z}M_2$ を表す.

(i) $|M_1|=|M_2|$
(ii) $M_1[\![Q]\!]=M_2[\![Q]\!]$
(iii) a. $M_1[\![P^1]\!]\subseteq M_2[\![P^1]\!]$
b. 任意の i $(1\leq i\leq k-1)$ について $M_1[\![P^i]\!]=M_2[\![P^i]\!]$ ならば, $M_1[\![P^{i+1}]\!]\subseteq M_2[\![P^{i+1}]\!]$

定義 3 [8] $M_1<^{P^1>\cdots>P^k;Z}M_2$ は次式で表される.

$$M_1\leq^{P^1>\cdots>P^k;Z}M_2\ \wedge\ M_2\not\leq^{P^1>\cdots>P^k;Z}M_1$$

命題 1 [8] A のモデル N,M に対して, $N<^{P^1>\cdots>P^k;Z}M$ を満たす N が存在しなければ, そしてそのときに限り, M は $Circum(A;P^1{>}\cdots{>}P^k;Z)$ のモデルである.

3 極小限定モデル生成プログラム

若木・井上の方法 [9] は, $Circum(A;P^1{>}\cdots{>}P^k;Z)$ (但し, A は節集合で与えられ関数記号は出現しない) のモデルの候補を generator: Π_{guess} で生成し, 各候補モデルは tester の論理プログラム Π_{check} の矛盾性より, 極小限定のモデルであるか否かを判定した. 本研究では, Π_{check} を僅かに変更して, 任意の侯補モデル S の $P\cup Q$-射影 $S|_{\mathcal{P}\cup\mathcal{Q}}$ を tester に埋め込んだ Π'_{check} を構成する. そして Π_{check} の矛盾性チェックを, Π'_{check} の一貫性制約 (integrity constraints) でシミュレートして不適切な侯補モデルを排除することにより, $\Pi_{guess}\cup\Pi'_{check}$ なる 1 つの GDP (general disjunctive program) の全ての解集合で所与の極小限定の全てのモデルを計算することができ, さらには 1 つの GDP で質問評価を行うことが可能になった.

3.1 GDP 論理プログラムへの翻訳

$tr[A;P^1,\ldots,P^k;Z]\stackrel{def}{=}\Pi_{guess}\cup\Pi'_{check}$ を定義する. Π_{guess} は 1〜2, Π'_{check} は 3〜9 のルールで構成される.

1. A の節 $A_1\vee\cdots\vee A_\ell\vee\neg B_1\vee\cdots\vee\neg B_m$ (但し A_i,B_j は原子式) に対して, $A_1|\cdots|A_\ell\leftarrow B_1,\ldots,B_m$.
2. $\mu\in Q,Z$ に対して, $\mu(x)\mid not\ \mu(x)\leftarrow$
3. $L\in H$ を表す項 L_t について, $m_1(L_t)\leftarrow L$
4. Π_{guess} の解集合 S の $P\cup Q$-射影 $S|_{\mathcal{P}\cup\mathcal{Q}}$ に対して, $m_2(L_t,\ j)\leftarrow\quad cardi(j)\leftarrow\quad(1\leq j\leq\xi)$
   但し, ξ は $S|_{P\cup Q}$ の総数.
5. $minp_i(L_t)\leftarrow\quad(1\leq i\leq k)$
   L_t は P^i の述語の基礎原子式を表現した項.
6. $fixed(L_t)\leftarrow$
   L_t は Q の述語の基礎原子式を表現した項.
7. $h_1(y)\leftarrow m_1(x),\ fixed(x),\ not\ m_2(x,\ y),\ cardi(y).$
   $h_2(y)\leftarrow m_2(x,\ y),\ fixed(x),\ not\ m_1(x).$
   $h(y)\leftarrow cardi(y),\ not\ h_1(y),\ not\ h_2(y).$
8. $c_i(y)\leftarrow m_1(x),\ minp_i(x),\ not\ m_2(x,\ y),\ cardi(y).$
   $d_i(y)\leftarrow m_2(x,\ y),\ minp_i(x),\ not\ m_1(x).$
   $eq_i(y)\leftarrow cardi(y),\ not\ c_i(y),\ not\ d_i(y).$
   $(1\leq i\leq k)$
9. $\leftarrow h(y),\ c_1(y),\ not\ d_1(y).$
   $\leftarrow h(y),\ eq_1(y),\ \ldots,\ eq_{i-1}(y),\ c_1(y),\ not\ d_1(y).$
   $(2\leq i\leq k)$

Computing Prioritized Circumscription and its Performance Evaluation
†Kazuo Tomita, Toshiko Wakaki
†Shibaura Institute of Technology

3.2 健全性/完全性定理と質問評価

所与の $Circum(A;P^1>\cdots>P^k;Z)$ から変換された論理プログラム (GDP) を $tr[A;P^1,\ldots,P^k;Z]$ とする.

定理 1 (健全性/完全性定理)
$M = E \cap H$ が A のエルブランモデルであるような $tr[A;P^1,\ldots,P^k;Z]$ の解集合 E が存在するならばならば, そしてその時に限り M は $Circum(A;P^1>\cdots>P^k;Z)$ のモデルである.

定理 2 (質問評価)
$Query \stackrel{def}{=} C_1 \wedge \ldots \wedge C_m \wedge \neg D_1 \wedge \ldots \wedge \neg D_n$ について,

$$Circum(A;P^1>\cdots>P^k;Z) \models Query$$

iff $tr[A;P^1,\ldots,P^k;Z] \cup \{query \leftarrow Query'.\} \cup \{violate \leftarrow not\ query.\} \cup \{\leftarrow not\ violate.\}$ は矛盾.
但し, $Query'$ は $Query$ の $\neg D_i$ を $notD_i$ で置換したもの.
又, $C_i, D_j \in H$ $(1 \leq i \leq m,\ 1 \leq j \leq n)$

4 計算機実験と性能評価

若木・井上の方法 [9] と本研究で提案した方法を ASP ソルバ DLV[2] を用いて実装し, それぞれのシステムを circum1, circum2[10] と称する. 又, Janhunen らの方法で昨年開発された prio_circ2dlp[3] と runtime(10 回の平均実行時間) の計測により性能比較を行なった.

4.1 実験環境とベンチマークケース

PemtiumⅣ 3.0GHz, L2 キャッシュ2MB, メインメモリ 2GB, Linux 環境下の PC で, A が命題論理, 述語論理の例について評価を行った. 命題論理では circ2dlp[3] の URL の Example ページにあった電子回路 (diag) の例を, 述語論理では GLbird[4], Example[1], スケジューリング問題 (meeting)[7] をベンチマークケースとした.

4.2 性能評価

4.1 のベンチマークについて極小化述語の優先順位のレベル数 k と runtime に関する表 1 と図 1 を示す.

表 1: runtime

			circum1	circum2	prio_circ2dlp
example	k	models	runtime(s)	runtime(s)	runtime(s)
GLbird	1	18	0.052	0.015	$\times^{1.1}$
GLbird1	2	6	0.062	0.014	$\times^{1.2}$
meeting0	1	3	0.194	0.026	$\times^{2}$
meeting1	3	2	0.221	0.033	$\times^{2}$
meeting2	4	1	0.246	0.039	$\times^{2}$
Example6	1	4	0.046	0.005	0.009
Example12	2	2	0.027	0.005	0.053
Example13	3	1	0.029	0.008	0.088
diag10-02-a	1	2	0.148	0.020	0.044
diag10-02-b	2	1	0.177	0.021	0.723
diag10-02-g	7	1	0.272	0.048	3.459
diag15-04-a	1	4	3.336	0.195	0.713
diag15-04-b	2	2	3.663	0.250	61.901
diag15-04-g	7	2	5.085	0.861	341.613
diag25-05-a	1	5	78.925	40.446	19.497
diag25-05-b	2	4	83.592	79.330	12023.600

(注) $^{1.1}$ モデル数が 2, $^{1.2}$ モデル数が 1, ^{2} 計算不可能

prio_circ2dlpは, 利用している lparse が原因で, GLbird のケースでは全モデルを計算することができず, meeting のケースでは計算自体が不可能 (表 1 参照) であった.

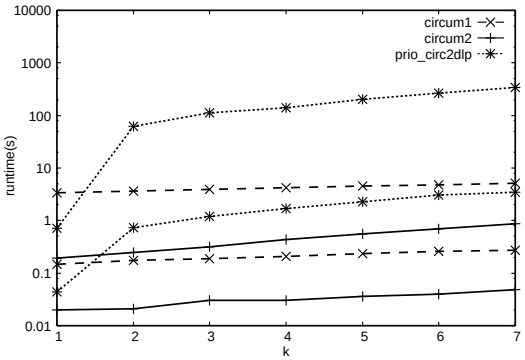

図 1: k と runtime

図 1 は diag10-02(図の下側) と diag15-04(図の上側) における極小化述語の優先順位数 k を 1〜7 まで変化させ, その runtime を示したものである. prio_circ2dlp は優先順位付き極小限定を Lifschitz schema で並列極小限定に変換して計算するため並列極小限定の性能は良いが, $k \geq 2$ になると circum1,2 より一般に性能が悪い.

5 まとめ

極小限定は, 人間が日常的に行っている非単調な常識推論を形式化したものである. そのためより柔軟な表現が可能な $k \geq 2$ の優先順位付き極小限定で Janhunen らのシステムの性能が悪いというのは大きな欠点である. また述語論理の例で計算できないものがあり, GLbird の例にいたっては命題論理で記述し直しても計算不能であった. どのようなケースが適用可能ではないのかが一切示されていないというのは重大な欠陥である.

circum2 はある程度のサイズまでの問題ならば runtime に優れ, さらに質問評価計算を行うことも可能である. だが一般に計算に多くのメモリを必要とするため巨大な問題ではメモリ不足が発生し runtime の性能が出ない (diag25 参照) 場合も存在する.

参考文献

[1] A. Nerode, T. N. Raymond, V.S. Subrahmanian: Computing Circumscriptive Databases, Part I:Theory and Algorithms, Information and Computation, Volume 116, pp.58-80, 1995.

[2] Eiter, T., Leone,N., Mateis, C., Pfeifer, G. and Scarcello, F.: A deductive system for nonmonotonic reasoning, Proc. LPNMR 1997, LNCS 1265, pp.364-375, Springer, 1997. dlv URL http://www.dbai.tuwien.ac.at/proj/dlv

[3] E.Oikarinen and T.Janhunen: CIRC2DLP - Translating Circumscription into Disjunctive Logic Programming, Proc. LPNMR 2005, LNAI 3662, pp.405-409, Springer, 2005. circ2dlp URL http://www.tcs.hut.fi/Software/circ2dlp/

[4] M.Gelfond and V.Lifschitz: Computing circumscriptive theories into logic programs. AAAI 1988, pp.455-459, 1988.

[5] M.Gelfond and V.Lifschitz: Classical Negation in Logic Programs and Disjunctive Databeses, New Generation Computing.9, pp.365-385, 1991.

[6] McCarthy, J.: Circumscription - A Form of Non-monotonic Reasoning, Artificial Intelligence, vol.13, pp.27-39, 1980.

[7] Satoh, K.; Formalizing soft constraints by interpretation ordering, Proc. ECAI 1990, pp585-590, 1990.

[8] V.Lifschitz: Computing circumscription, in: Proc. IJCAI-85, Morgan Kaufmann, LOS Altos, CA, 985, pp.121-127, 1985.

[9] Wakaki, T. and Inoue, K.: Compiling Prioritized Circumscription into Answer Set Programming, Proc. ICLP 2004, LNCS 3132, pp.356-370, Springer, 2004.

[10] circum1, circum2 http://www.ailab.se.shibaura-it.ac.jp/circumscription.html