数理計画法を用いた制約充足問題の定式化についての検討

2Q−3

永井 保夫
(株) 東芝 情報処理・機器技術研究所

長谷川 隆三
(財) 新世代コンピュータ技術開発機構

1 はじめに

制約充足問題は組み合わせ問題を対象としたNP完全問題であることが知られている．このような制約充足問題の代表的な解法としては，探索法や局所近似を用いる方法がある [1]．しかしながら，大規模な問題を取り扱う場合には，このような解法を用いて充足解，特に全解を探索することはほとんど不可能である．そこで，われわれは探索による全解を求めるかわりにひとつの解を求めるアプローチをとることとし，ここでは最適解を求める数理計画法を用いたアプローチについて検討している．本アプローチでは制約充足問題を整数計画法を用いて定式化し，得られた制約式(条件)を解くことにより(最適)解を求める．本稿では，アプローチの概要について説明する．まず，制約充足問題ならびに数理計画法の一種である整数計画法について触れる．次に，整数計画法を用いた制約充足問題の定式化と具体的な問題を用いた定式化の具体例について説明する．最後に，整数計画法の解法ならびに制約充足問題の充足不能性の判定についても論じる．

2 制約充足問題と数理計画法

2.1 制約充足問題

制約充足問題(CSP)とは，次のような変数の有限集合および各変数に対してとりうる値を離散値の有限集合として与えた場合に，すべての制約を満足するように各変数に対して値を求める問題である．ここでは，すべての制約が満足されるようにすべての変数に対する値を求めることに相当し，単解または全解を求める．

- 変数: $V = \{v_1, \ldots, v_n\}$
- ドメイン: 各変数 v_n は離散値の有限集合 $D_i = \{d_1, \ldots, d_m\}$ $i = 1, \ldots, n$ から値 d_m を選択する．
- 制約: 集合 $C = \{c_1, \ldots, c_l\}$ where $c_n \equiv (v_1, v_2, \ldots, v_n) \subseteq D_1 \times D_2 \times \ldots \times D_n$

2.2 数理計画問題と整数計画法

数理計画問題(最適化問題)は制約条件:$x \in S$ を満足するもののなかで，目的関数: $f(x)$ を最小または最大にするように変数の値を決定する問題である．(但し，変数 x はベクトル $x \in R^n$，集合 S は実行可能領域 $S \subseteq R^n$，目的関数 $f(x)$ は S をふくむ集合上で定義された実数値関数である)．この場合，変数が連続的な値をとるか，離散的な値をとるかによって大別される [2][3]．

整数計画問題は変数が離散的な値をとる組み合わせ最適化問題の一種であり，整数値をとらねばならないという制約(整数制約)が与えられている変数を含んだ線形計画問題(目的関数が一次関数，制約条件が一次の等式ならびに不等式で与えられる問題)である．ここでは，特に，次のように整数変数の値が0または1に限られている0-1整数計画問題を取り上げる．

$$P_0 \begin{cases} minimize & z = \sum_{j=1}^{n} c_j x_j \\ subject\ to & \sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad (i = 1, \ldots, m) \\ & x_j \in \{0, 1\} \quad (j = 1, \ldots, n) \end{cases}$$

3 整数計画法による制約充足問題の定式化検討

我々は制約充足問題を0-1整数計画法を用いて定式化し，得られた制約式に対して上記の解法を適用することで充足解を求めるアプローチをとる．従来のCSPで求められる充足解は前述したように単解または全解 [1] であるのに対して，今回われわれが提案したアプローチでは全体の解集合を考慮した最適解を単解としてもとめる．制約充足問題はNP完全問題であり，取り扱うべき問題の規模が大きくなった場合には，探索法を用いた全解の収集は不可能であると考えられる．われわれの提案した手法は，存在する解をすべて数え上げる必要がなく，実際にはひとつの解を求めることで十分であるような場合や一次関数形式の目的関数が存在する場合(最適化問題)に対して有効であると考えられる．

まず，整数計画法を用いたCSPの定式化について示し，次に具体例を用いて定式化について説明をおこなう．以下では，すべての制約が2項関係からなるCSPを対象とする．

3.1 整数計画法による定式化

- 変数集合 $V = \{v_1, \ldots, v_n\}$，各変数 v_i がとりうる離散値の有限集合 $D_i = \{d_1, \ldots, d_m\}, i = 1, \ldots, n$ とし，変数 v_i に対してドメイン D_i の要素 d_j をひとつだけ割り当てることを変数 x_{ij} によりあらわす．

$$\sum_{j=1}^{m} x_{ij} = 1, \quad i = 1, \ldots, n$$

- 制約を2つの変数間において成り立つ二項関係として考える．ここでは，2つの変数間に対して値の割り当てを許してはいけない関係(禁止制約)，たとえば，変数 v_i へのドメイン D_i の要素ならびに変数 v_j へのドメイン D_j の要素の割り当てを禁止する組み合わせ関係を二項タプル集合としてあらわす．具体的には，変数 v_i と変数 v_j のそれぞれのドメイン D_i と D_j に対して値の割り当てを許さない d_k と d_l からなる組み合わせを二項のタプル集合 $\overline{T}_{ij} = \{(d_k, d_l) \mid D_i \times D_j - T_{ij}\}$ とし，以下の制約式により表現する．

$$\sum_{(d_k, d_l) \in \overline{T}_{ij}} x_{ik} + x_{jl} \le 1, \quad (i = 1, \ldots, n, j = 1, \ldots, m)$$

- 各変数の0-1整数条件は

$$x_{ij} \in \{0, 1\} \quad (i = 1, \ldots, n, j = 1, \ldots, m)$$

となる．

- 上記の制約条件における目的関数 z は

$$z = \sum_{j=1}^{n} c_j x_{ij}$$

となる．

3.2 制約充足問題の具体例による説明

制約充足問題 [1] では，制約が制約ネットワークとして静的な問題の記述に適した知識表現をとる場合が多い．ここでは，図のような3つの変数 X，Y，Z とそのドメイン $D_X = \{5, 2\}$，$D_Y = \{2, 4\}$，$D_Z = \{5, 2\}$ からなる制約ネットワークを例とした制約充足問題について説明する．制約は変数 X と Z 間の関係を示すタプル集合 $(X, Z) = \{(5, 5), (2, 2)\}$ と変数 Y と Z 間の関係を示すタプル集合 $(Y, Z) = \{(2, 2), (4, 2)\}$ からなる．

Mathematical Programming Approach to Constraint Satisfaction Problems

Yasuo NAGAI　Ryuzo HASEGAWA
Toshiba Corp.　ICOT

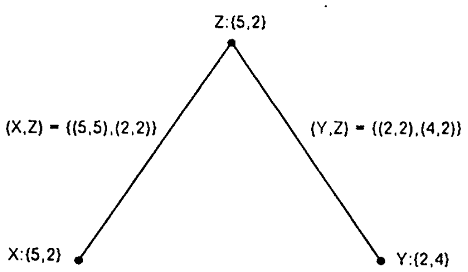

図: 制約ネットワーク

変数 $X(=v_1)$ はそのドメインが $D_X = \{5,2\}$ であり，ドメインの要素からは値 d_j をひとつだけとるとすると，$X = v_1$, $D_X = D_1$, $d_1 = 5, d_2 = 2$ であり，これを制約式により表現すると次のようになる.

$$x_{11} + x_{12} = 1$$

同様に，変数 $Y(=v_2)$ と $Z(=v_3)$ に対しても制約式を求めると次のようになる.

$$x_{21} + x_{22} = 1,$$
$$x_{31} + x_{32} = 1$$

さらに制約 $(X, Z) = \{(5,5),(2,2)\}$ と $(Y, Z) = \{(2,2),(4,2)\}$ では，次のような定式化が考えられる. 変数 X と Z 間および変数 Y と Z 間において値の割り当てが許されないタプルをそれぞれ $\overline{T}_{13}$, $\overline{T}_{23}$ とすると
$\overline{T}_{13} = \{(d_1,d_2),(d_2,d_1)\}$, $\overline{T}_{23} = \{(d_1,d_1),(d_2,d_1)\}$ となる.
これを制約式により表現すると次のようになる.

$$x_{11} + x_{32} \le 1, x_{12} + x_{31} \le 1,$$
$$x_{21} + x_{31} \le 1, x_{22} + x_{31} \le 1$$

3.3 生成される制約式と求められるべき充足解

上記の例題の整数計画法による定式化よって，次のような制約式の集合 CS が生成される.

$$\left.\begin{array}{lr} minimize & z = x_{11} + x_{12} + x_{21} + x_{22} + x_{31} + x_{32}^{\dagger} \\ subject\ to & x_{11} + x_{12} = 1, \\ & x_{21} + x_{22} = 1, \\ & x_{31} + x_{32} = 1, \\ & x_{11} + x_{31} \le 1, \\ & x_{12} + x_{31} \le 1, \\ & x_{21} + x_{32} \le 1, \\ & x_{22} + x_{31} \le 1, \\ & x_{11}, x_{12}, x_{21}, x_{22}, x_{31}, x_{32} \in \{0,1\} \end{array}\right\} = CS$$

一般，CSP を数理計画法を用いて制約式に変換すると，式の数が非常に増えるため，大規模な制約式の集合を解くことが必要となる. 例えば，CS の場合には，$m \times n$ 個の変数 x_{ij} からなる $n + |\overline{T}|$ 個の制約式が生成される. ここでは $minimize\ z = \sum_{j=1}^{n} c_j x_j$ を $c_j = 1 (j = 1, \ldots, n)$ となる変数の線形和としているが，c_j をコストとして与えられれば，目的関数を有効に利用して効率的な求解ができる †.

最終的には，上記の制約式の集合 CS からなる整数計画問題を線形計画法に基づいた分枝限定法を用いて解くことにより，次のような充足(最適)解を求める.

$$x_{11} = 0, x_{12} = 1, x_{21} = 0, x_{22} = 1, x_{31} = 0, x_{32} = 1, z = 3$$

これらの結果から，充足解は変数 X がドメイン D_X の2番目の要素2，Y が変数 Y のドメイン D_Y の2番目の要素4，Z がドメイン D_Z の2番目の要素2であり，目的関数の値 z が3であることがわかる.

3.4 効率的な解法ならびに充足不能性についての検討

ここでは，CSP に対する整数計画法の効率的な解法ならびに充足不能性について検討する.

整数計画問題の解法には，切除平面法，分枝限定法，群緩和法などがある [2][3]. 一方，命題論理の推論に対する整数計画法によるアプローチが研究されており，そこでの有効な解法として線形計画法を用いた分枝限定法と切除平面法が知られている [4].

われわれは整数計画問題として定式化された CSP の効率的な解法として，このような線形計画法を用いた分枝限定法と切除平面法の適用について検討中である. これらの手法では，まず，整数計画問題が整数条件を緩和した連続緩和問題に直され，線形計画法が解かれる. 次に，効率化を考え，線形計画法により得られた情報ならびに問題の特殊な構造 [3] を有効に利用して分枝限定法や切除平面法をおこなう.

[連続緩和問題]

$$\overline{P_0} \left\{ \begin{array}{ll} minimize & z = \sum_{j=1}^{n} c_j x_j \\ subject\ to & \sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad (i = 1, \ldots, m) \\ & 0 \le x_j \le 1 \ (j = 1, \ldots, n) \end{array} \right.$$

上記の連続緩和問題ともとの問題である整数計画問題の間には，次のような定理が成り立つ. 特に，2番めの項目はCSPにおける充足不能性の判定に利用できる.

[定理]

- $\overline{P_0}$ が最適値 x^0 をもち，$x^0 \in Z^n$ ならば，x^0 は P_0 の最適解である.
- また，$\overline{P_0}$ が実行可能解をもたないならば，P_0 も実行可能解をもたない (CSPにおける充足不能性の判定).

連続緩和問題の解法である線形計画法の計算複雑度は最悪のケースでは指数オーダーとなるが，平均的なケースでは線形オーダーとなることが示されており [2]，上記の充足不能性の判定は大規模な問題に対しても十分可能であると考えられる.

4 むすび

制約充足問題に対する従来の探索を基本とした解法とは異なる数理計画法を用いた解法ならびに充足不能性について論じた. 本解法は制約充足問題を整数計画法，特に 0-1 計画法を用いて定式化し，得られた制約式を解くアプローチをとる. 具体的には制約充足の例題を取り上げて定式化をおこない，連続緩和問題に変換し，線形計画法を用いた分枝限定法ならびに切除平面法を適用する. ここでは，線形計画法を取り扱うために，制約論理プログラミング言語 CAL[5] の線形不等式制約評価系を用いた. 従来制約充足問題の解法として探索法がよく知られているが，われわれは数理計画問題における解析的な方法が有効となる可能性を示した. 今後は処理(分枝限定法と切除平面法)の効率化を考慮して，CSP のもつ特殊な構造を利用したアルゴリズムの検討ならびに提案した手法の評価実験をおこなっていく予定である.

謝辞

本研究は第5世代コンピュータプロジェクトの一環として行なわれた. 本研究の機会を与えてくださり、常にご指導いただいたＩＣＯＴの渕一博所長、古川康一研究次長，新田克己第7研究室室長に深く感謝いたします. また，CAL を利用するにあたりいろいろ教えて戴いた相場亮第4研究室室長代理ならびに整数計画法について有益な助言を戴いた川岸太郎氏をはじめとした第4研究室の皆様に感謝いたします.

参考文献

[1] Mackworth, A. K.: Constraint Satisfaction, In Encyclopedia of Artificial Intelligence (Ed. Shapiro, S.C.), John Wiley & Sons Ltd. (1987).

[2] Schrijver, A.: Theory of Linear and Integer Programming, John Wiley & Sons Ltd. (1986).

[3] 今野浩，鈴木久敏編: 整数計画法と組合せ最適化，日科技連 (1982).

[4] Hooker, J. N.: A Quantitative Approach to Logical Inference, Decision Support System 4, North-Holland, pp.45-69 (1988).

[5] Aiba, A., Sakai, K., Sato, Y., Hawley, D.J.and Hasegawa, R.: Constraint Logic Programming Language CAL, Procs. of Int. Conf. on FGCS88 (1988).