1R-08

制約充足問題におけるインフルエンシャル変数の特定

Identifying Influential Variables in Constraint Satisfaction Problem

浦地 勇人 [1]　沖本 天太 [2]　平山 勝敏 [2]　Nicolas Schwind[3]　井上 克巳 [4]　Pierre Marquis[5]
神戸大学海事科学部 [1]　神戸大学大学院海事科学研究科 [2]
人工知能研究センター (AIST)[3]　国立情報学研究所 [4]　CRIL-CNRS/Artois University[5]

1 序論

人工知能で扱われる問題の多くは，探索という要素を含んでおり，制約充足問題 (Constraint Satisfaction Problem; CSP) [3] として定式化可能である．CSP とは，有限で離散的な領域から値をとる複数の変数に，制約を満たすような値を割当る問題であり，典型的な例に色塗り問題や数独等がある．

CSP における既存研究にバックボーン [1, 2] と呼ばれる変数に関する研究がある．バックボーンとは，充足可能となるすべての解で同じ割当をもつ変数を指す．一般に，CSP ソルバーは充足可能となるような解をみつけた，または，そのような解が存在しないとわかった時点で終了する．これに対し，バックボーンの特定問題は，すべての充足可能な解を列挙し，そのすべてにおいて同じ割当をもつ変数を探索する必要があり，co-NP-完全な問題として知られている．バックボーンに関する既存研究では，バックボーンを特定することにより，問題を簡略化して解くことを目的とした，ヒューリスティックスの開発がほとんどである [1, 2]．一方，CSP における大域的な解の状態遷移に関する研究，具体的には，充足可能から充足不可能への状態遷移を可能とするような変数に着目した研究はほとんどない．

本論文では，CSP における大域的な解に影響を及ぼす，バックボーンを包括するような変数に着目し，このような変数をインフルエンシャル変数として定義する．さらに，インフルエンシャル変数の特定問題を定式化し，この問題の計算複雑度について議論する．実験では，異なるサイズ及びグラフ構造の色塗り問題を用いて，インフルエンシャル変数の個数を調べた．

2 制約充足問題

制約充足問題 (CSP) [3] は，$X = \{x_1, ..., x_m\}$ を変数の集合，$D = \{D_1, ..., D_m\}$ を変数値の集合，C を制約の集合とし，$\langle X, D, C\rangle$ の組により定義される．CSP を解くとは，すべての制約を満足するような変数値の組を求めることである．この問題は NP- 完全な問題として知られている．また CSP は，変数をノード，制約をノード間のエッジに対応させることにより，グラフ（制約グラフと呼ばれる）を用いて表現可能である．

例 1. 図 1 に 5 つの変数からなる色塗り問題の例を示す．各ノードは，赤，青，緑のいずれかを変数値として取る．例えば，

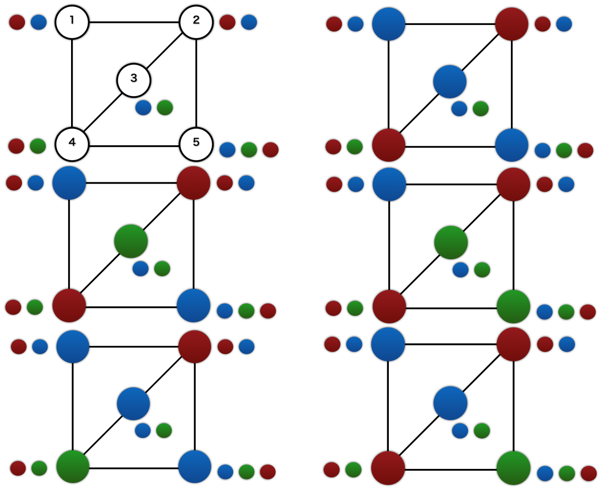

図 1　例 1 の色塗り問題及び，充足可能となるすべての解．

変数 1 は赤または青の変数値をもつ．この問題の解，すなわち，制約グラフ内のエッジでつながったノード同士が異なる色をもつ変数への割当は，例えば，変数 1 が青，変数 2 が赤，変数 3 が青，変数 4 が赤，変数 5 が青となり（上段の右），色塗りされた制約グラフが，この問題のすべての充足可能な解となる．

次に，この色塗り問題におけるバックボーンを示す．バックボーンとは，充足可能となるすべての解で同じ割当をもつ変数なので，すべての充足可能な解で同じ割当をもつ変数，すなわち，変数 1（青）及び，変数 2（赤）がバックボーンとなる．

3 インフルエンシャル変数の特定問題

CSP において，すべての制約を満たす解集合を $\mathcal{A}^{sat}$，それ以外を $\mathcal{A}^{unsat}$ と記述する．ある解 $A \in \mathcal{A}^{sat} \cup \mathcal{A}^{unsat}$ に関して，変数 x_j と x_j の変数値 d_j 含む解を $A_{(x_j,d_j)}$ と記述する．

定義 1 (インフルエンシャル変数). *CSP における変数 x_j 及び，x_j の変数値の集合 D_j に関して，以下が成立するとき，変数 x_j をインフルエンシャル変数と呼ぶ．*

- $\exists d_j \in D_j : A_{(x_j,d_j)} \in \mathcal{A}^{sat}$，かつ，
- $\exists d'_j \in D_j : A_{(x_j,d'_j)} \notin \mathcal{A}^{sat}$，$(d_j \neq d'_j)$

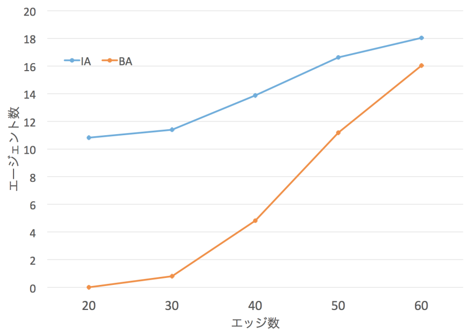

図 2　異なるエッジ数（制約密度）における IA 及び BA の数．

定義 1 は，変数 x_j が変数値 d_j を選ぶ場合，解集合 $\mathcal{A}^{sat}$ に含まれる充足可能な解が存在するが，x_j が変数値 d'_j を選んだ場合，他の変数がどのように変数値を変えても，すべての制約を満たすような解が存在しなくなることを意味している．

例 2. 図 1 の色塗り問題における大域的な解に対する，各変数のもつ影響力について考える．例えば，変数 3 の選択について考える．変数 3 が青を選択した場合，図 1（上段の右）のような充足可能な解が存在するが，もう 1 つの選択肢である緑を選択したとしても，充足可能な解が得られる（中段の左）．すなわち，変数 3 に関しては，どの変数値を選択しても常に充足可能な解が得られ，変数 4 に関しても同じである．一方，変数 5 に関しては，青を選択した場合，充足可能な解が得られるが，赤を選択した場合，この問題は他の変数がどのような変数値を選択しても充足不可能となる（充足可能から充足不可能への状態遷移）．この問題におけるインフルエンシャル変数は，定義 *1* より，変数 $1, 2, 5$ である．インフルエンシャル変数は状態遷移を可能とするが，変数 5 はバックボーンではない [*1]．

定理 1 (バックボーンとの関係)．　*CSP* において，各変数の変数値 $d \in D$ が $|d| \geq 2$ であるとき，以下の関係が成立する．

- *Backbone* $\Rightarrow$ *Influential variable*
- *Backbone* $\not\Leftarrow$ *Influential variable*

証明．"$\Rightarrow$"：定義 1 より明らか．（例えば，例 2 の変数 1 と 2.）"$\Leftarrow$"：反例として，例 2 における変数 5 を挙げる．変数 5 はインフルエンシャル変数であるが，バックボーンではない．　□

定義 2 (インフルエンシャル変数の特定問題)．

- **Input:** 制約充足問題 $CSP = \langle X, D, C \rangle$，変数 $x \in X$，
- **Question:** 変数 x はインフルエンシャル変数か？

定理 2 (計算複雑度)．　インフルエンシャル変数の特定問題は co-NP-*hard* である．

[*1] インフルエンシャル変数とバックボーンの大きな違いとして，バックボーンは，すべての充足可能となる解で同じ割当をもつ必要があるが，インフルエンシャル変数は，必ずしも同じ割当をもつ必要はない．

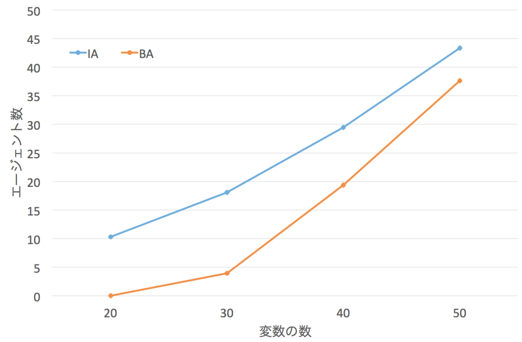

図 3　異なる変数の数における IA 及び BA の数．

4　評価実験

実験では，異なるエッジ数及び問題の規模における，インフルエンシャル変数（IA）の数及びバックボーン（BA）の数を調べた．各問題に関して，変数値は $[1, 2, 3, 4]$ とし，各変数は 2 から 4 つの変数値をもつようにランダムに選択した．実験結果は 50 インスタンスの平均値を表している．図 2 に変数の数を 20 に固定し，エッジ数を 20 から 60 とした問題における IA 及び BA 数の平均値を示す [*2]．結果より，エッジ数の少ないグラフでは，IA のみが存在することが分かった．実際，エッジ数 20 では，IA 数の平均値は 10.8，BA 数は 0 であった．また，エッジ数が増加すると IA 及び BA 数も増加し，IA における BA の割合が増加していることが分かった．このことは，制約密度を 12% に固定し，問題の規模を大きくした実験でも同じであった（図 3）．以上の結果より，充足可能な問題において，求解が容易な問題では IA における BA の割合は少なく，求解が困難な問題では，その割合は大きくなることが分かった．また，この結果は問題の規模に依存しないことが分かった．

5　結言

本論文では，CSP における大域的な解に影響を及ぼすインフルエンシャル変数を定義した．また，インフルエンシャル変数の特定問題を定式化し，この問題の計算複雑度について議論した．実験では，異なるサイズ及び構造の色塗り問題を用いて，インフルエンシャル変数を求解し，充足可能な問題において，求解が容易な問題では IA における BA の割合は少なく，求解が困難な問題では，その割合が大きくなることが分かった．今後の課題として，問題に特化したアルゴリズムの開発がある．

参考文献

[1] O. Dubois and G. Dequen. A backbone-search heuristic for efficient solving of hard 3-sat formulae. In *IJCAI*, pages 248–253, 2001.

[2] P. Kilby, J. Slaney, S. Thiébaux, and T. Walsh. Backbones and backdoors in satisfiability. In *AAAI*, pages 1368–1373, 2005.

[3] A. Mackworth. Constraint satisfaction. In *Encyclopedia of Artificial Intelligence*, pages 285–293, 1992.

[*2] エッジ数 60 以上では充足不可能となるため，IA と BA 数は 0 となる．