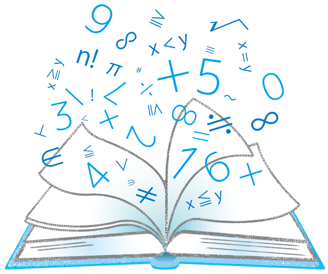

❻ MaxSAT：SATの最適化問題への拡張

―MaxSAT ソルバーの活用法―

越村三幸（九州大学）　藤田 博（九州大学）

最適化問題のための MaxSAT

MaxSAT（Maximum satisfiability）は，SAT（Boolean satisfiability）を最適化問題に拡張したもので，SAT では解に優劣はないが，MaxSAT ではあり，最適解が MaxSAT 解となる．

MaxSAT では，問題はハード節とソフト節の集合として表される．ソフト節には重みがあり，重みは正整数で表される．重みはそのソフト節の重要性の度合いを示しており，ハード節はどのソフト節よりも重要で必ず満たさなければならない制約を表す．

MaxSAT の目的は，変数の値割当の中で，すべてのハード節を満たし，かつ満たされるソフト節の重みの総和が最大となるものを見つけることである．したがって，MaxSAT を用いて問題を解く場合，良い解は，より多くのソフト節を満たすように，あるいは，より重いソフト節を満たすように問題を表現する必要がある．

なお，単に「MaxSAT」といった場合，ハード節がなくソフト節の重みがすべて等しい問題を指すことがある．この場合，ハード節がありソフト節の重みがすべて等しい問題を「部分 MaxSAT（Partial MaxSAT）」，ハード節がありソフト節の重みが複数ある問題を「重み付き部分 MaxSAT（Weighted Partial MaxSAT）」と呼んで区別する[☆1]．

SAT ソルバーの性能向上を取り込むことにより MaxSAT ソルバーの性能も向上しており，SAT の成功を背景に実用的な応用への取り組みも盛んに行われている．それらの分野は，ルーティング，バイオインフォマティクス，ハードウェアおよびソフトウェアのデバッギング，スケジューリングやプランニング，ゲーム理論等，多岐に及ぶ．

☆1 2006 年から毎年開催されている MaxSAT ソルバーの評価会 MaxSAT Evaluation では，2015 年からこの 3 つのカテゴリに分けてソルバーを評価している．

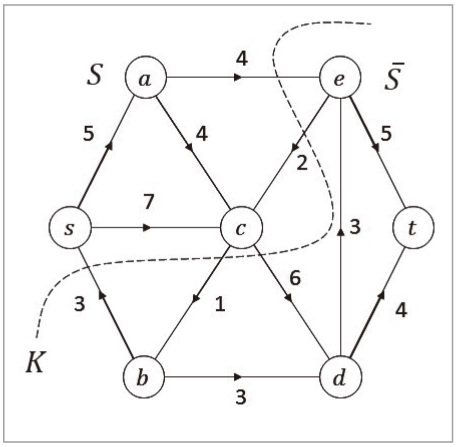

図 -1 最大流問題（恵羅ほか著「グラフ理論」より）

グラフ理論の基本的な問題のいくつかは MaxSAT に自然に翻訳（符号化）できるが，次章では，ネットワークの最大流問題の具体例を用いて，MaxSAT による解法を示す．

最大流問題の解法

図 -1 のような流れのあるネットワークを考える．アルファベットを囲む円は頂点，頂点を結ぶ矢印は弧を表し，弧の横の数字は，その弧の流れの容量を表す．たとえば，a から c への弧の容量は 4 である．このとき，頂点 s から頂点 t への流れの最大値を求める問題が最大流問題である．ネットワークを物流網，頂点を地点，弧の容量を（その向きの）地点間の 1 日に輸送可能な量と捉えると，最大流問題は s から t に 1 日で輸送可能な最大量を求める問題となる．

図 -1 のネットワークでは，s から a に 5，a から e に 4，a から c に 1，s から c に 4，c から d に

v_s	v_a	v_b	v_c	v_d	v_e	v_t	a_{sa}	a_{bs}	a_{sc}	a_{ac}	a_{ae}	a_{cb}	a_{bd}	a_{cd}	a_{ec}	a_{de}	a_{dt}	a_{et}
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

表 -1　変数の付番

5，d から e に 1，d から t に 4，e から t に 5 流せば，s から t には 9 流れる．t への流れ込みは e からと d からのみで，その容量の和は 9 なのでこれが最大流であることが分かる．

本稿では，「最大流の値は，最小カットの容量に等しい」（最大流・最小カット定理）ことを利用して問題を解く．カットとは，ネットワークの頂点集合を 2 つの集合 S と $\bar{S}$ に $s \in S \wedge t \in \bar{S}$ となるように分割することである．

図 -1 の破線 K はカットの一例である．頂点集合を s を含む S と t を含む $\bar{S}$ に分割している．$S \Rightarrow \bar{S}$ で S から $\bar{S}$ に向かう弧の集合を表すことにする．図 -1 では，a から e への弧，c から b への弧，c から d への弧の 3 つが $S \Rightarrow \bar{S}$ に属する．$S \Rightarrow \bar{S}$ に属する弧の容量の総和をカットの容量という．K の容量は 11(＝4＋1＋6) である．

物流のたとえでは，定理は次のように解釈できる．s から t へ物を運ぶ場合，それは必ず $S \Rightarrow \bar{S}$ の弧を通過することになる．したがって最小カットは s から t への物流のボトルネックに対応する．こう考えると，最大流は最小カットの容量を超えることはできないが，定理は，その容量までは物流を増やすことができることを保証している．

★最大カット問題

まず，MaxSAT で考えやすい最大カット問題を解いてみる．なお本稿では，ソフト節 c とその重み w のペアを括弧で囲って (c, w) と表す．ネットワークのカットの仕方はいろいろあるが，その中で最大容量のカットを見つけるのが最大カット問題である．頂点 x に対し変数 v_x，頂点 x から y への弧に対し変数 a_{xy} を導入する．v_x が真であるとき，$x \in S$ であることを表す．同様に，a_{xy} が真であるとき，$x \in S \wedge y \in \bar{S}$ であること，つまり，x から y への弧は $S \Rightarrow \bar{S}$ に属することを表す．図 -1 では，7 個の頂点と 12 個の弧があるので，計 19 個の変数 v_s, ..., v_t, a_{sa}, ..., a_{et} が導入される．

v_x, v_y と a_{xy} の関係は

$$v_x \wedge \neg v_y \leftrightarrow a_{xy} \qquad (1)$$

であり，これは 3 つの節

$$\neg v_x \vee v_y \vee a_{xy} \qquad (2)$$
$$v_x \vee \neg a_{xy} \qquad (3)$$
$$\neg v_y \vee \neg a_{xy} \qquad (4)$$

で表現できる．(2) は (1) の「→」部分，(3) と (4) は「←」部分を節表現したものである．このように 1 個の弧から 3 個の節が得られるので，12 個の弧からは 36 個の節が得られる．これらは必ず成り立つ必要があるのでハード節となる．

a_{xy} に対応する弧の容量を w_{xy} とすると，a_{xy} が成り立てば，カットの容量は w_{xy} 増加するので，弧の容量はソフト節 (a_{xy}, w_{xy}) として表すことができる．たとえば，$w_{ac}=4$ なので，ソフト節は $(a_{ac}, 4)$ となる．図 -1 では，12 個の弧があるので，12 個のソフト節が得られる．最後に，$s \in S \wedge t \in \bar{S}$ から，2 つのハード節 v_s と $\neg v_t$ が得られる．

以上まとめると，38 個のハード節と 12 個のソフト節，合わせて 50 個の節が得られる．これら 50 個の節を実際の MaxSAT ソルバーに与えるには，拡張 DIMACS CNF 形式で記述する必要がある．この形式では，1 以上の整数は変数を表し，節は整数の並びで表される．今扱っている問題では，19 個の変数があるので，これらを先頭から `1,...,19` と付番する（**表 -1**）．

図 -2 の左側の枠内に拡張 DIMACS CNF 形式を示す．右側は元の MaxSAT 節である．1 行目の `p wcnf` に続く数字は，変数の数，節の数，ハード節の重さを表し，2 行目以降は節を表す．ハード節の重さは，ソフト節とハード節を区別するために用

p wcnf 19 50 48	
48 1 0	v_s
48 -7 0	$\neg v_t$
48 -1 2 8 0	$\neg v_s \vee v_a \vee a_{sa}$
48 1 -8 0	$v_s \vee \neg a_{sa}$
48 -2 -8 0	$\neg v_a \vee \neg a_{sa}$
《この間 33 個のハード節を省略》	
5 8 0	$(a_{sa}, 5)$
7 10 0	$(a_{sc}, 7)$
《この下 10 個のソフト節を省略》	

図 -2　MaxSAT 記述（最大カット問題）

いられ，通常，「ソフト節の重みの総和＋1」[☆2] で与えられる．いま，12 個のソフト節の重みの総和は 47 となるので，ハード節の重さは 48 となる．

節は節の重さに続く整数の並びで表され，最後の 0 は節の終わりを表す．整数は節を構成する変数を，- は否定を表す．図 -2 では 48 で始まる各行が 1 個のハード節を表す．紙面の都合上，途中 33 個のハード節を省略した．下段の 5 で始まる行からがソフト節の記述である．12 個のソフト節のうち，10 個は省略した．

以上，51 行からなる maxcut.wcnf という名のファイルを作り，コマンドラインから MaxSAT ソルバーで解いた結果が**図 -3** である[☆3]．ほとんどの MaxSAT ソルバーはコマンドラインから

```
$maxsat <input_file_name>
```

と打ち込めば実行できる．ここで，maxsat はソルバーのコマンド名である．c で始まる行はコメントを表す．図 -3 では，最初にソルバーが起動されたことを，最後に使用メモリ量と CPU 時間を表示している．

o で始まる行は，ある解が見つかり，その解で満たされなかったソフト節の重みの和を示している．この出力は，それまでの最良解が見つかったときになされる．したがって，この値は段々と小さくなっていく．図 -3 では，35，25，24 と減少している．

```
$ maxsat maxcut.wcnf
c A MaxSAT Solver
o 35
o 25
o 24
s OPTIMUM FOUND
v 1 -2 3 -4 5 6 -7 8 -9 10
v -11 -12 -13 -14 -15 16 -17 18 19
c Memory used :  6.00 MB
c CPU time :  0 s
$
```

図 -3 MaxSAT ソルバーの実行例

「s OPTIMUM FOUND」はその直前の最良解の表示「o 24」が最適解，つまり MaxSAT 解であったことを示す．24 は，満たされなかったソフト節の重みの和なので，満たされるソフト節の重みの和はソフト節の重みの総和からの引き算で計算する．図 -1 の例では，総和は 47 なので 23 (＝47−24) がこの最大カット問題の解の容量となる．

v で始まる行は，最適解における変数の真偽値の割り当てを示す．正数は真，負数は偽を割り当てられたことを示す．v_s, ..., v_t については，1 -2 3 -4 5 6 -7 と表示されているので，$S=\{v_s, v_b, v_d, v_e\}$，$\bar{S}=\{v_a, v_c, v_t\}$ であることが分かる．したがって図 -1 より，5 個の弧（s から a, s から c, e から c, d から t, e から t）が $S \Rightarrow \bar{S}$ に属しており，この容量を計算すると 23 (＝5＋7＋2＋4＋5) となり，ソルバーの結果と一致していることが分かる．

★最小カット問題

最大ではなく最小容量のカットを見つけるのが最小カット問題である．それには，$S \Rightarrow \bar{S}$ に属さない弧の容量の和を最大にすればよい．W を弧の容量の総和とすると，$S \Rightarrow \bar{S}$ の容量は，

$$W - (S \Rightarrow \bar{S} \text{ に属さない弧の容量の和})$$

だからである．$\neg a_{xy}$ であれば x から y への弧は $S \Rightarrow \bar{S}$ に属さないので，$(\neg a_{xy}, w_{xy})$ をソフト節とすればよい．つまり，図 -2 の MaxSAT 記述の最後の 12 行のソフト節の記述を変更すれば，最小カット問題が解ける．**図 -4** に変更分を示す．

[☆2] 重要度においてソフト節は束になってもハード節にはかなわない，ことを表している．

[☆3] 筆者らが開発している QMaxSAT ソルバーの出力からの抜粋．MaxSAT Evaluation の規格に準拠しているので，この出力の意味が分かれば，どのソルバーの出力も大体理解できる，と思う．

```
5 -8 0
7 -10 0
```

$(\neg a_{sa}, 5)$

$(\neg a_{sc}, 7)$

《この下 10 個のソフト節を省略》

図 -4
MaxSAT 記述（最小カット問題：変更分のみ）

こうして得られる MaxSAT 記述のファイルを MaxSAT ソルバーに与えると $S=\{v_s, v_a, v_b, v_c, v_d, v_e\}$，$\bar{S}=\{v_t\}$ なるカットが得られる．この容量は 9 であり，こうして最大流の容量は 9 であることが示される．

節の削減と負の重み

SAT と同様に一般的に節や変数の数は，少ない方がソルバーへの負荷が少なく高速に求解できる，と期待される．実は，MaxSAT にはこれに特有の節削減法があり，前章のような手順で問題を MaxSAT に符号化した場合，これが適用できる．また，最小カット問題では，これとは別の方法で，節のみならず変数の数も減らせるが，紙面の都合上，これらについて本稿では触れない．

MaxSAT では，ソフト節の重さは正数と規定されている．しかし，解きたい問題によっては，負の重みも使いたい場合が生ずる．これも紙面の都合上，「負の重みを使いたいときは，簡単な変換で MaxSAT で解けるようになる」ということを述べるにとどめたい．

MaxSAT ソルバーの性能

MaxSAT ソルバーの評価会 MaxSAT Evaluation が毎年開催されており，2016 年で 11 回目となる．当初は，分枝限定法に基づくソルバーの性能の評価が高かったが，最近では，SAT ソルバーを推論エンジンとして利用する手法が優勢となっている．したがって，SAT ソルバーの性能向上の恩恵をそのまま享受できるソルバーが多い．このようなソルバーで扱える問題の規模は，SAT ソルバーのそれとほぼ同等である．また，整数計画ソルバーと連携するソルバーも登場しており，今後も性能向上が期待できる．

評価会の影響もありさまざまな MaxSAT ソルバーが利用可能である．どのようなソルバーがあるかは，評価会の Web ページを参考にされたい．ただソルバーの多くは Linux でのみ動作確認しているので，とりあえず使ってみたいという読者には，Java 上で動作する SAT4j Maxsat の利用を薦める．高速ではないが，とっつきやすく使いやすい．

高速性を求める場合は，いくつかのソルバーで解いてみて，どのソルバーがその問題に適しているか探ってみる必要もあろう．必ずしも評価会の上位ソルバーが解きたい問題を高速に解くとは限らない．筆者らは「AES 暗号鍵の復元」問題 76 問を昨年の評価会に提出したが，この問題に関しては，総合順位では下位のソルバーの性能が最も高かった．

整数計画ソルバーと MaxSAT ソルバーとどちらが速いか？と思われる方も多いと思う．解いてみないと分からない，というのが現状ではなかろうか．ただ，論理式で自然に表現できる問題であれば，MaxSAT を利用しない手はないと思う．

（2016 年 4 月 29 日受付）

越村三幸（正会員） koshi@inf.kyushu-u.ac.jp

九州大学大学院システム情報科学研究院助教．博士（工学）．自動推論システムに関する研究に従事．

藤田　博（正会員） fujita@inf.kyushu-u.ac.jp

九州大学大学院システム情報科学研究院准教授．博士（工学）．自動推論に関する教育研究に従事．