2E5-OS-09b-6

MaxSAT ソルバ用いた高分子の組成と物性との関係に関する考察

Discussion of Relationship between Compositions and Properties of Polymers using MaxSAT Solver

力 規晃[*1] 越村 三幸[*2] 西田 光生[*3] 阿部 幸浩[*3] 藤田 博[*2] 長谷川 隆三[*2]
Noriaki Chikara Miyuki Koshimura Mitsuo Nishida Yukihiro Abe Hiroshi Fujita Ryuzo Hasegawa

[*1] 徳山工業高等専門学校 情報電子工学科
Department of Computer Science and Electronic, Tokuyama College of Technology

[*2] 九州大学大学院システム情報科学研究院
Faculty of Information Science and Electrical Engineering, Kyushu University

[*3] 東洋紡株式会社
TOYOBO CO., LTD.

In order to generate chemical products from reactants by chemical reaction efficiently, we have to know appropriate chemical conditions such as temperature, flow, and density. These conditions are revealed through several chemical experiments which cost us. We developed a method extracting rules from experimental data using Inductive Logic Programming (ILP) systems. The rules show that chemical conditions which seem to be appropriate. In this work, we replace ILP for MaxSAT in order to reduce CPU time for extracting rules. MaxSAT is an optimization version of SAT which consists in finding an assignment that maximizes the number of satisfied clauses. Computational experimental results with actual chemical experimental data show that MaxSAT succeeds to extract rules faster than ILP does.

1. はじめに

化学反応を利用して，原料から所望の性能を満たす化合物を効率よく生成するには，化合物を生成する原料の濃度，流量，温度，即ち反応条件に関連したパラメータを適切に設定する必要がある．そして，所望の性能を満たす化合物を得るための反応条件パラメータの条件を得ることが重要となる．

化学反応の反応条件パラメータと性能値のデータから特定の性能条件を満たす反応条件パラメータを得るため，帰納論理プログラミングを利用することで既知の化学反応データからある性能を満たすための反応条件パラメータの規則性を得ることができる．

一方，近年 SAT ソルバや MaxSAT ソルバが高速化され，問題を SAT 符号化することで，多くの成果を上げている．

本研究では，従来，帰納論理プログラミングで行っていた化学反応データの規則性の抽出[力 12]を MaxSAT ソルバに置き換え比較検討を行う．

2. 帰納論理プログラミング

本研究が MaxSAT ソルバで置き換えを行う帰納論理プログラミングを以下に説明する

2.1 帰納論理プログラミングの枠組み

帰納論理プログラミング(Inductive Logic Programming: ILP)[古川 01]とは一階述語論理に基づいた機械学習の手法である．ILP の一般的な枠組みは，正例 と負例 ，背景知識 が節集合で与えられ，

$$\begin{cases} B \not\models E^+ \\ B \cup E^- \not\models \square \end{cases} \quad (1)$$

であるとき，つまり，背景知識では正例が説明できず，しかも，背景知識と負例が矛盾しないとき，

$$\begin{cases} H \cup B \models E^+ \\ H \cup B \cup E^- \not\models \square \end{cases} \quad (2)$$

を満たす，つまり背景知識に仮説を加えると正例を説明でき，しかも，背景知識と仮説と負例は矛盾しない仮説 を見つけることである

2.2 帰納論理プログラミングの探索手法

現実の ILP システムでは集合被覆アルゴリズムを用いて，仮説を複数抽出し，全ての正例を被覆する．

代表的なILPシステムで Progol や Aleph の集合被覆アルゴリズムは次のようなものである．

B を背景知識，H=ø を仮説，E を正例集合とする．

（1） もし E=ø ならば H を出力する．
（2） e を E 中の最初の事例とする．
（3） e と B より，最弱仮説を生成し探索空間を限定する．
（4） 探索により，最適な仮説 H'を生成する．
（5） H に H'を加える，H'を B に含める．
（6） B で説明できる正例の集合を E'とする．
（7） E から E’に含まれる正例を取り除く．
（8） (1)へ戻る．

Progol や Aleph が採用しているアルゴリズムの特徴は，一つの正例に着目し，最弱仮説と呼ばれるその正例を説明する仮説の中の最も弱い仮説を求め，その正例が説明できる仮説だけに限定することで探索空間を最初に枝刈りすることである．また，通常，最適な仮説を得るための探索時の評価関数は仮説で被覆される正例数は大きくなり，被覆される負例数や仮説の長さが小さくなるような関数が利用される．

連絡先：力 規晃，徳山工業高等専門学校 情報電子工学科，
〒745-8585 山口県周南市学園台，tikara@tokuyama.ac.jp

2.3 抽出ルールの限定

本研究では一般的な帰納論理プログラミングを対象にするのではなく，仮説として抽出されるルールの形式を限定した帰納論理プログラミングを対象とする．

一般的に帰納論理プログラミングは次のルール形式の仮説を規則性として抽出する．

$B:-A_1,\cdots,A_n$.

このルールは A_1 ～ A_n を全て満たしたとき B となることを意味する．ここで，B は事例を表わす変数を引数にもつ目標概念を表わすリテラルであり，$A_i(1\le i\le n)$ は背景知識に存在する述語を用いたリテラルであり，引数に事例を表わす変数を含む任意の変数と，背景知識で使われているアトムが使われる．

本研究では，ルールに現れるリテラルが引数に持つ変数を，第 1 引数のみとし，事例を表わす変数1つに限定し，ルールの本体部のリテラルは 2 引数のものだけを扱う．例えば次のようなルールを本研究では対象とする．

$e(X):-p(X,aa).$

$h(X,a):-q(X,ab),r(X,c),s(X,a)$.

また，3 引数のリテラルは，次のような変換が可能である．

$p(X,a,b) \Rightarrow p(X,a)\wedge p(X,b)$

この例のように 3 以上の引数の場合は 2 引数のリテラルに書き換えることができるため，次の例のような 3 引数以上の述語形式のリテラルをルールの本体部に持つルールも扱うことができる．

$e(X):-p(X,a,b),q(X,c,d).$

3. 取り扱うデータと ILP によるルール抽出

従来，化学反応データを用いて特定の性能を示す条件パラメータのルールを，ILP を用いて行っていた．以下に説明する．

3.1 取り扱うデータの概要

本研究で取り扱うデータは化学反応の条件パラメータ値(説明変数)とその性能を示す値(目的変数)とがセットになっている．条件パラメータ値と性能を示す値は1以上あり，一部の条件パラメータには合計を一定にするなどの制約条件があるものもある．

また，与えられるデータ件数についてはデータマイニングの対象のデータとしては少ない．具体的には 10 数件～50 件程度である．これは，データを得るための化学実験に長期間要することと、コストが高いためである．そのため，少ないデータからルール抽出を行うというところに挑戦課題がある．

そして，仮想実験データと実際の実験データを帰納論理プログラミングによって解析し，目標条件を満たしそうな条件パラメータを導く．導かれた条件が，実験化学者によって，妥当と判断されれば，化学実験によって本当に目標条件を満たすかどうか実証される．

3.2 ILP によるルール抽出

従来，上で説明したデータから，特定の性能を示す場合の条件パラメータの規則性をルールで抽出していた．

このとき，抽出ルールの本体部のリテラルは 2 引数で，第 1 引数は事例を表わす変数，第 2 引数は値を表わすアトムとなるように限定する．

次の例のような形式のルールが抽出される．

p_ex(A) :- p_a(A,a_50_0), p_b(A,b_1_5).

これは，パラメータ A が 50 でパラメータ B が 1.5 のとき所望の性能となることを意味する．

4. MaxSAT

命題論理の充足可能性判定は SAT と呼ばれている．SAT 問題は CNF 形式で表され，複数の節で構成される．SAT は近年高速な SAT ソルバーが開発され，様々な応用がなされている．ただし，現実の問題を SAT の問題に符号化し解を求めようとすると，全ての節を充足することは不能であるが，できるだけ多くの節を満たす解を得たい場合もある．このような場合に用いられるのが MaxSAT である[平山 10]．本研究では重み付き部分 MaxSAT(Weighted Partial MaxSAT)を用いる．重み付き部分 MaxSAT の問題を構成する節は必ず満たさなければならないハード節とソフト節に大別される．また，ソフト節については1以上の重みが設定されており，満たされる節の重みの和が最大になる解を求める．

現実の問題について，2.2 節の集合被覆アルゴリズムを実行する場合，一つの仮説で全ての正例を満たすことは少ない．多くの場合，一つの仮説で全ての正例を満たさないときは出来るだけ多くの正例を満たす仮説を得たい．そのため，できるだけ多くの節を満たす解を得る枠組みである MaxSAT を用いるべきだと考えられる．

5. ILP の MaxSAT による置き換え

3 章で説明した化学反応データから ILP を用いて，特定の性能を示す問題を Weighted-Partial-MaxSAT 問題としてエンコードし，この Weighted-Partial-MaxSAT 問題を用いて，最弱仮説を利用する集合被覆アルゴリズムを再現し，実行する．なお，本手法は，化学反応データに限らず，一つの表で表現されるような入出力関係のデータで特定の条件を満たす規則性を抽出する場合に利用可能である．

5.1 ILP 問題の MaxSAT 符号化

ここでは表 1 のような変数を MaxSAT 符号化(エンコード)して作成する MaxSAT 問題で用いる．

表 1 4.1 ILP 問題の MaxSAT 符号化で用いる変数

変数	意味
$r(i,j)$	i 番目のパラメータが j 番目の値であることがルールに現れる
$a(i,j)$	$r(i,j)$ または $f(i)$ が成り立つ
$p(x)$	x 番目の正例が成り立つ
$n(y)$	y 番目の負例が成り立つ
$f(i)$	i 番目のパラメータが無条件である

本研究の MaxSAT 符号化により生成される節を以下に示す．

(1) i 番目のパラメータはルールの上で2つの値を同時に持たない(ハード節)．

$$\{\neg r(i,j(b))\vee\neg r(i,j(c))\mid 1\le i\le l, 1\le j(b)\le m(i), 1\le j(c)\le m(i), j(b)\ne j(c)\}$$

(2) i 番目のパラメータが無条件であることと i 番目のパラメータのルールが成立することは同時に成り立たない(ハード節)．

$$\{\neg r(i,j(d))\vee\neg f(i)\mid 1\le i\le l, 1\le j(d)\le m(i)\}$$

(3) 正例が成立すれば，その正例の持つパラメータ値は成り立つことが言える(ハード節)．

$$\{\neg p(x)\vee a(i,j(i,x))\mid 1\le x\le k(p), 1\le i\le l\}$$

（4）負例が否定されるならば，負例が持つ値のどれかが否定される（ハード節）．

$$\{n(y) \vee \neg a(1, j(1,y)) \vee \ldots \vee \neg a(l, j(l,y)) \mid 1 \le y \le k(n)\}$$

（5）ルールに i 番目のパラメータが j 番目の値であることが現れることまたは i 番目のパラメータが無条件であることと，i 番目のパラメータが j 番目の値をとりうることは等しい（ハード節）．

$$\{\neg a(i,j) \vee r(i,j) \vee f(i) \mid 1 \le i \le l, 1 \le j \le m(i)\}$$
$$\{a(i,j) \vee \neg r(i,j) \mid 1 \le i \le l, 1 \le j \le m(i)\}$$
$$\{a(i,j) \vee \neg f(i) \mid 1 \le i \le l, 1 \le j \le m(i)\}$$

（6）負例が成立することはできない（ハード節）．

$$\{\neg n(y) \mid 1 \le y \le k(n)\}$$

（7）正例が成立する（ソフト節，重み：$l+1$）．

$$\{p(x) \mid 1 \le x \le k(p)\}$$

（8）i 番目のパラメータは無条件とする（ソフト節，重み：1）．

$$\{f(i) \mid 1 \le i \le l\}$$

ここで，$r(i, j(b))$, $r(i, j(c))$, $r(i, j(d))$ は i 番目のパラメータがそれぞれ $j(b)$ 番目，$j(c)$ 番目，$j(d)$ 番目の値となることを示す変数であり，また，l はパラメータの総数で，$m(i)$ は i 番目のパラメータが取りうる値の総数である．そして，$a(i, j(i,x))$ は i 番目のパラメータが x 番目の正例が持つ値になることが許されることを表し，$a(1, j(1,y))$ は i 番目のパラメータが y 番目の負例が持つ値になることが許されることを表す．$k(p)$ は正例の総数であり，$k(n)$ は負例の総数である．また，ソフト節の重みについては，(7)の節が(8)の節より常に優先されるようにするため，(7)の節の重みは(8)の節の総数 l に+1 を加えた値とした．

5.2 MaxSAT ソルバによる帰納学習アルゴリズム

次のアルゴリズムで MaxSAT ソルバを動作させ，ILPと同様の帰納学習を行う

仮説集合 H=∅とし，正例集合を E とする．
（1）もし E=∅ならば H を出力する．
（2）E 中で，最も若い番号のもの e とする．
（3）仮定により p(e)=T とする．
（4）MaxSAT ソルバで解を得る．
（5）得られた解から E の 2 以上の正例を被覆するルールが得られた場合は H に加える．
（6）得られたルールが被覆する正例と仮定の対象となった正例を合わせた集合を E'とする．
（7）E'を正例集合 E から除外する．即ち p(e')=F(e'∈E')とする．
（8）(1)へ戻る．

6. ILP と MaxSAT の比較実験

化学反応データで ILP と MaxSAT で比較実験を行った．次のような実際の化学反応データとそのデータを元に予測をして合成したデータを併せたものを使用した．次にデータの概要を示す．

- 反応条件パラメータ数：10
- 性能を示す値の数：1
- 正例：54 件
- 負例：564 件
- 背景知識：それぞれの事例（正負例）の反応条件パラメータ値を述語論理表現したもの

ILPの実行環境を次に示す．

- 使用した ILP システム：Aleph（prolog プログラム）
  - 探索評価関数：（被覆される正例数－被覆される負例数）
  - 探索法：幅優先探索
  - ルールの誤り率の上限:0.0
  - 探索ノードの上限: 50000
- 使用した Prolog：YAP-Prolog 6.2.2
- 実行前後の処理：Java プログラム
- 実行PC：（CPU:Corei7 2.7GHz，メモリ:8GB）

MaxSAT の実行環境を次に示す

- 使用した Weighted-MaxSAT ソルバ:QMaxSAT(簡易改造※)
- 実行前後の処理：Java プログラム
- 実行PC：（CPU:Corei7 2.7GHz，メモリ:8GB）

それぞれの実行結果の概要を表 2 に示し，抽出された仮説の比較を表 3 に示す．

表 2 実行結果の概要

	ILP	MaxSAT
抽出された仮説（ルール）の数	9	9
前後処理を含めた実行時間	2.630s	2.863s
ILP または MaxSAT だけの実行時間	1.482s	1.142s

表 3 抽出された仮説（ルール）の比較

比較結果	適合する仮説(ルール)の数
全く同じ仮説，被覆される正例は同じ	3
仮説の長さが等しく，仮説の条件の一部が同じ．被覆される正例は同じ	5
仮説の長さが等しい．仮説の条件は異なる．被覆される正例は同じ	1

7. 区間ルール抽出のための MaxSAT 符号化

今回扱う化学反応データは数値データであるため，所望の性能が得られるパラメータ条件を数値の区間で得たい場合もある．ILPを用いれば，次の例のようなルールを抽出することもできる．

```
p_ex(A) :- bound_a(A, a_49_0,a_50_0),
           bound_b(A,b_1_5, b_2_0).
```

このルールは，パラメータAは 49.0 以上 50.0 以下でかつ，パラメータ B は 1.5 以上 2.0 以下の時，所望の性能が得られることを示している．

このようなルール抽出を MaxSAT で実現する．その際に行う MaxSAT 符号化で用いる変数を表4に示す．

また，区間を扱うための MaxSAT 符号化により生成される節を以下に説明する．

（1） i 番目のパラメータはルールの上で2つの値を同時に持たない（ハード節）．
（2） i 番目のパラメータが無条件であることと i 番目のパラメータのルールが成立することは同時に成り立たない（ハード節）．
（3） 正例が成立すれば，その正例の持つパラメータ値は成り立つことが言える（ハード節）．
（4） 負例が否定されるならば，負例が持つ値のどれかが否定される（ハード節）．

※元の QMaxSAT は Weighted に対応してない．1より大きい重みを付けた節については，重み1の同じ節を重みの数だけ登録することで，Weighted-MaxSAT の問題を扱う．

(5) i 番目のパラメータが j 番目の値以下であることが許されるならば，i 番目のパラメータが $j-1$ 番目の値以下であることが許される(ハード節)．
(6) i 番目のパラメータが j 番目の値以上であることが許されるならば，i 番目のパラメータが $j+1$ 番目の値以上であることが許される(ハード節)．
(7) ルールに i 番目のパラメータが j 番目の値以下であることが現れることは，i 番目のパラメータが j 番目の値以下であることが許され，かつ，i 番目のパラメータが j 番目の値以下であることが許されないことと等しい(ハード節)．
(8) ルールに i 番目のパラメータが j 番目の値以上であることが現れることは，i 番目のパラメータが j 番目の値以上であることが許され，かつ，i 番目のパラメータが j 番目の値以上であることが許されないことと等しい(ハード節)．
(9) i 番目のパラメータが最大値以下でありうることは，ルールに i 番目のパラメータが最大値以下であることが現れること，または，i 番目のパラメータが無条件であることである(ハード節)．
(10) i 番目のパラメータが最小値以上でありうることは，ルールに i 番目のパラメータが最小値以上であることが現れること，または，i 番目のパラメータが無条件であることである(ハード節)．
(11) i 番目のパラメータが何らかの値以下の条件成り立つときに，i 番目のパラメータが何らかの値以上の条件が成り立つ，また逆も言える(ハード節)．
(12) 負例が成立することはできない(ハード節)．
(13) 正例が成立する(ソフト節，重み: $l+1$)．
(14) i 番目のパラメータは無条件とする(ソフト節，重み: 1)．

また，MaxSAT ソルバの実行アルゴリズムは 5.2 節で説明したものを用いる．

表 4 区間を扱うための MaxSAT 符号化で用いる変数

変数	意味
$rle(i, j)$	i 番目のパラメータが j 番目の値以下であることがルールに現れる
$rge(i, j)$	i 番目のパラメータが j 番目の値以上であることがルールに現れる
$ale(i, j)$	$rle(i, j)$ または $f(i)$ が成り立つ
$age(i, j)$	$rge(i, j)$ または $f(i)$ が成り立つ
$p(x)$	x 番目の正例が成り立つ
$n(y)$	y 番目の負例が成り立つ
$f(i)$	i 番目のパラメータが無条件である

8. 区間ルール抽出の ILP と MaxSAT の比較実験

MaxSAT については7章で説明した手法を用いる．使用したデータや実験環境は 6 章と共通である．

ILP と MaxSAT ともに反応条件パラメータ値の区間条件のみが仮説として抽出できるように設定したが，1.0 以上 1.0 以下のような区間幅がない一点だけの条件が出ることは許した．

それぞれの実行結果の概要を表 5 に示す．

9. 考察

6 章の実験により，通常の数値を扱う仮説を抽出した場合，表 2 より実行時間はほぼ互角である．また，表 3 より，抽出された全ての仮説がそれぞれ被覆する正例は全く同じであり，それぞれの長さも等しい．このことから，同等の結果が得られていると考えられる．ただし，仮説で着目したパラメータが異なるものがある．これはそれぞれのやり方で探索順が異なるため，同等の仮説だが別のものを見つけてしまったと考えられる．

8 章の実験から区間ルールを抽出する場合は，表 5 より得られたルール数とそれが被覆する正例は MaxSAT の方が多かった．また，実行時間は圧倒的に MaxSAT が速かった．Aleph(ILP)は探索に時間をかけ，探索ノード数制限により 2 つのパラメータを扱うルールまでしか抽出できなかった上に，打ち切りの際に誤った出力を行う場合がある．MaxSAT は短時間に 3 パラメータや 4 パラメータを扱うルールが抽出できた．

表 5 区間ルール抽出の比較

項目	ILP	MaxSAT
抽出された仮説(ルール)の数	9(誤抽出 2)*	12
前後処理を含めた実行時間	212.836s	2.786s
被覆される正例数	19	36
ILP または MaxSAT だけの実行時間	204.501s	1.130s

*抽出された9つのルールで正負例と矛盾するものが2つ含まれていた．

10. おわりに

ILPの抽出ルールの形式を限定し，MaxSAT で代用する手法を提案し，化学反応データを用いたILPによる規則性の抽出を MaxSAT を利用した手法で置き換え，同等以上の性能が得られた．今後の課題は，本手法をより汎用的にILPの代用ができるように拡張することである．

謝辞

本研究は科研費(25330085)の助成を受けたものである．

参考文献

[力 12] 力規晃, 越村三幸, 橋本司, 西田光生, 阿部幸浩,藤田博,長谷川隆三: 帰納論理プログラミングを用いた高分子の組成と物性との関係に関する考察, 第 11 回情報科学技術フォーラム A-012, 2012.

[古川 01] 古川康一, 尾崎 知伸, 植野 研: 帰納論理プログラミング, 共立出版, 2001.

[平山 10] 平山勝敏, 横尾真: *-SAT:SAT の拡張, 人工知能学会誌 25(1),pp.105-113, 2010.