

## 制約充足的手法を応用したクロスワードパズルの解法

2 S - 8

Nguyen Viet Ha<sup>†</sup> 石川 勉<sup>†</sup>金杉 友子<sup>††</sup><sup>†</sup> 拓殖大学工学部情報工学科<sup>††</sup> NTT アドバンステクノロジー

### 1 はじめに

クロスワードパズルを解く問題は、パズルのヒントに合う単語をどのように選ぶかという意味的な問題と、ヒントに合った単語群を用いて如何にクロスワードを埋めていくかという問題に分けられる。

ここでは、パズルのヒントに合った単語群が何らかの方法で得られると仮定し、後者の問題の解法について述べる。具体的には、パズルを制約充足問題とみなし、解候補を段階的に絞っていく高速な解法を提案する。

### 2 クロスワードパズルの解法

#### 2.1 問題の定式化

##### (1) クロスワードパズルの定義

パズルは横と縦のN個の枠からなるとする。枠と枠の交わるマス目を交点と呼ぶ。枠に対し、ヒントから選ばれた答えの候補の集合を候補群と呼ぶ。

パズルは、各候補群からクロス条件を満たす答えの単語を求めることで解かれる。クロス条件とは横と縦の二つの枠が交わるならば、その交点の文字が一致することをいう。

##### (2) 部分パズルの定義

n 個の連結する枠からなる部分を「n次部分パズル」と呼ぶ。ここで、「連結」とは、その部分パズルのどの枠も、他の枠のどれかと交点を持つことをいう。全体のパズルは、各部分パズルを解くことにより解かれる。

#### 2.2 パズルの解法

クロスワードパズルは、縦や横の解が交点で互いに束縛し合っているため制約充足問題とみなすことができる。従って、この種の問題で一般的な併合法[1]を用い、候補を絞っていくことにより解くことができる。ここでは以下のように解く。まず、全体

のパズルを複数の部分パズルに分け、各部分パズルに対しクロス条件を充足できない、つまり部分解の要素になれない候補を消去する。次に、部分パズルのサイズを徐々に拡大して、これを繰り返す。

##### (1) 部分パズルでの候補絞り

n次部分パズルとその時点の各候補群に対して、以下のアルゴリズムで候補を絞り込む。

##### [アルゴリズム 1]

- ① 元の各候補群からすべての部分解を求める。
- ② 元の各候補に対し、どの部分解にも属さない候補を消去し、新たな候補群を作成する。

ここで、元の各候補群の候補総数と新しい各候補群の候補総数の差を、消去し得た候補数と呼ぶ。

2次部分パズルの例でこのアルゴリズムについて説明する。図1のような部分パズルについて考える。

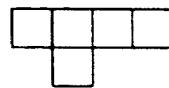


図1 2次部分パズル

横と縦の枠の候補群を、  
 (“こうばん”、“こう  
 たい”) と (“うま”、  
 “ぶた”) とする。こ  
 の場合、部分解は

(“こうばん”、“うま”) と (“こうたい”、“うま”)のみである。よって、これらに含まれない“ぶた”が消去され、新しい候補群はそれぞれ {“こうばん”、“こうたい”} と {“うま”} になる。この例では1個の候補が消去されたことになる。

##### (2) 全体のパズルの解法

上記のアルゴリズムを用いたパズル全体の解法を以下に示す。

##### [アルゴリズム 2]

- ①  $n = 2$  とする。
- ② 全ての n 次部分パズルを求め、これらを  $(n)P_1, (n)P_2, \dots, (n)P_m$  とする。
- ③ 各部分パズル  $(n)P_i$  に対し、アルゴリズム 1 を用い、 $(n)P_1$  から  $(n)P_m$  まで逐次的に候補絞りを行う。消去した候補の総数を D とする。

Solving crossword puzzles by a scheme of constraint satisfaction problems

Nguyen Viet Ha<sup>†</sup>, Tsutomu Ishikawa<sup>†</sup>, Tomoko Kanasugi<sup>††</sup>

<sup>†</sup> Department of computer science, Takushoku university

<sup>††</sup> NTT Advanced technology corporation

1 (14)	(15)	(17)				(21)		(25)
2					3 (20)			(23)
		4		(18)				
5	(16)				2 (22)			(24)
	7				8			
8				(19)				
			10					11 (24)
12							13	

“よこ”のヒント

1. 交替、ワケ別、出張所
2. 忍耐、塩忍、根比べ
3. 演芸、二人、掛合い
4. 東緯、枷、犬
5. 文字、重要、事務
6. 粥、病人、上登り
7. 面倒、お節介、手段
8. 飛行、際どい、地面
9. 師匠、医者、教師
10. 弦楽器、ばち、法師
11. 仲間、同伴、配偶者
12. 駅、切符、入り口
13. 壁、聞く、飾り

“たて”のヒント

14. 初冬、風、冷たい
15. 家畜、たてがみ、走る
16. 卒業、送別、宴
17. 勝負、大穴、予想外
18. 賭事、投げる、丁半
19. 邪、歪む、変形
20. 球、遊び、手
21. 第六感、胸騒ぎ、鋭い
22. 借り、負担、思
23. 家具、板、材料
24. 罰、深い、悪事
25. 雨、軒、簷
26. 黄昏、日没、暗い

こ	う	ば	ん		か			と
が	ま	ん			ま	ん	ざ	い
ら			く	さ	り			い
し	よ	る	い			お	も	ゆ
		せ	わ		て	い	く	う
せ	ん	せ	い			め		く
	か			び	わ		つ	れ
か	い	さ	つ			み	み	

図2 3ヒント・クロスワードパズルの例

図3 本手法による解答

④  $D > 0$  なら③に戻る。さもなければ次に進む。  
 ⑤  $n < N - 1$  なら  $n$  を1増やし②に戻る。  
 ⑥ パズル全体の解を求める。  
 ③では、逐次的に候補絞りを行うので、 $(n)P_i$  まで候補絞りした結果は、 $(n)P_{i+1}$  の候補絞りに影響する。即ち、候補群内の候補の数が部分パズルの候補絞りごとに段階的に減っていく。④では、 $D > 0$  ならば、また消去できる候補が存在する可能性があることを意味する。即ち、 $(n)P_i$  で実際に候補が消去されると、そのことが  $(n)P_1 \sim (n)P_{i-1}$  候補絞りに影響し、さらにこの段階で候補を絞れる可能性が出てくる。⑤、⑥では、 $N - 1$  次パズルまで候補絞りをを行い、最後に  $(n=N)$  全体のパズルの解を求める。この段階で、残っている候補は全てのクロス条件を満足するので、最終的にクロスワードパズルの解はこの候補群の中から適切な組み合わせを選べばよい。具体的には、縦形探索等で簡単にパズルの解を求めることができる。あるいは、候補の順位等を考慮して解を評価しながら最適な解を選んでよい。

3 実験と評価

3.1 概念ベースからの候補群抽出

実験では、図2のような3ヒント・クロスワードパズルを用いた。また、解の候補群抽出には、概念ベース[2]を用いた。ここで、概念ベースとは、国語辞典等における語義文の情報をもとに、自動的に構築した大規模な知識ベースである。各概念は複数個の属性と属性値のペアによって現される。属性は、その概念に関連する概念であり、属性値はその関連の度合いである。

この概念ベースを用い、以下のように候補の抽出を行う。ある概念がヒントの単語を属性として含む

なら、そのヒントと関連すると考える。概念ベースから、ヒントの単語を属性として含む全ての概念(単語)を選び出し、これを3ヒントで集計して枠の候補群とする[3]。なお、各候補は文字数の制約を満たすものだけに絞る。

3.2 実験結果と評価

8×8のパズルを図2の例を含め5個用意して実験を行った。実験では、全ての例題に対し正解が得られることが確認できた。

図2のパズルに対するの実行結果を図3に示す。処理時間はSun SS/5 で約0.4秒であった。なお、この例では、概念ベースから得られた各枠の候補群の候補数平均は70であった。

従来の手法[3]では、パズル全体の約80%しか求められなかったのに対し、完全にパズルを解け、かつ一桁以上の高速化が達成できた。

4 むすび

クロスワードパズルを制約充足問題とみなし、併合法の考えを用いた候補絞りアルゴリズムを提案した。また、実験により従来の手法に対して一桁以上の高速化が実現できることを確認した。

参考文献

[1] 西原：制約充足問題の基礎と展望(特集)、人工知能学会誌 Vol12 No3 (1997)  
 [2] 笠原、松沢、石川：国語辞書を利用した日常語の類似性判別、情報処理学会論文誌 Vol38 No7 pp1272-1283  
 [3] 金杉、松沢、笠原：アバウト推論の「言葉遊び」への適用、信学技法 NLC96-31(1996-10)