

人工知能

第2回講義課題 課題番号 06

提出締切: 2013.11.07

提出日: 2013.11.07

工学部電子情報工学科

03-123006 岩成達哉

1 ATMS を用いた N-Queens 問題の解答プログラム

今回は、ATMS を用いる例として、N-Queens 問題を解き、解法の数を出力するプログラムを作成した。プログラムは、参考 URL[1][2] を基に Common Lisp によって記述した。その実行結果をリスト 1 に示した。これは対称な解も含めて出力している。プログラムは、atms.lisp と nqueens.lisp の 2 種類である。

リスト 1: 実行結果

```
1 [1]> (load "atms.lisp")
2 ;; Loading file atms.lisp ...
3 ;; Loaded file atms.lisp
4 T
5 [2]> (load "nqueen.lisp")
6 ;; Loading file nqueen.lisp ...
7 ;; Loaded file nqueen.lisp
8 T
9 [3]> (n-queens 1)
10 1
11 [4]> (n-queens 2)
12 0
13 [5]> (n-queens 3)
14 0
15 [6]> (n-queens 4)
16 2
17 [7]> (n-queens 5)
18 10
19 [8]> (n-queens 6)
20 4
21 [9]> (n-queens 7)
22 40
23 [10]> (n-queens 8)
24 92
```

2 ATMS による N-Queens の解法説明

2.1 ATMS の主要な動作

ATMS は、Assumption-based TMS の略であり、仮説を基にして、推論を繰り返し、矛盾のない仮説集合 (環境) を求めることが狙いである。ここで用語をいくつか取り上げると、「仮説」は成立するかどうかわからないもの、「前提」はいつも整理するもの、「ノード」は推論結果や仮説、前提のいずれか、「正当化」は推論結果が成立する理由付けを行うことである。また、ATMS では、矛盾する条件を「Nogood」と呼び、推論に用いる。JTMS と違い、ATMS は導出と仮説ノードを分ける [3]。ATMS では、ノードに対して「ラベル」を付与する。ラベルは複数の「環境」から成り、環境は複数の「仮説」から成る [4]。これは例えば、図 1 のようになる [4]。

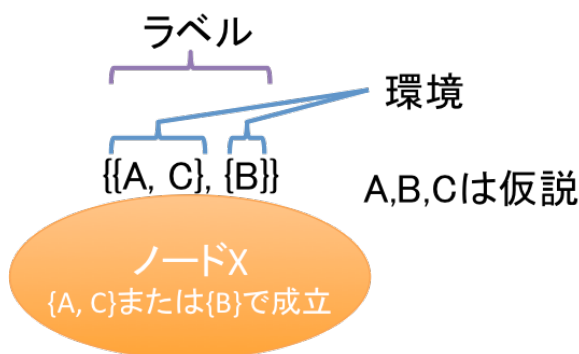


図 1: ラベル、環境、仮説、ノードの整理

このノードと、仮説に基づいて、矛盾のない仮説の集合を求めるわけである。ATMS では、バックトラックを行わず、仮説と Nogood を用意しておくことで、仮説の検証を全ての場合について行うことができる。今回の N-Queens の問題では、このラ

ベルを更新することで解の個数を求めるということを行った．ラベルの更新は例えば，図 2 のようになる．ノード X のラベルとノード Y のラベルの全ての組み合わせを列挙する．この際に，環境が他のより強い条件に含まれるならばそれを除く．このような操作で，ノード Z のラベルとしては， $\{A, B, C\}$ が除かれて，3 つの環境が残る．また，Nogood となるような環境が含まれた場合はそれも除く．例えば， $\{B, D\}$ が Nogood であるならば，ノード Z のラベルは， $\{\{A, C\}, \{C, D\}\}$ となる．

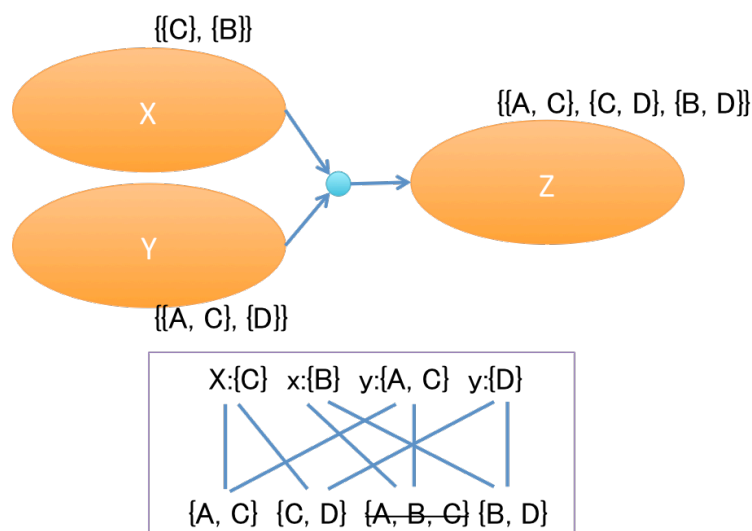


図 2: ラベルの更新

2.2 仮説と Nogood の作成

さて，以上が ATMS の主要な動作であるが，この仕組みを N-Queen に適用する．これまでの内容からわかるように，ATMS では仮説と Nogood を列挙することから始める．N-Queens では， N 個の Queen を $N * N$ の盤上で，縦・横・斜めの列に一つずつしかないように配置する．例えば，8-Queens では，図 3 が解の 1 つである．

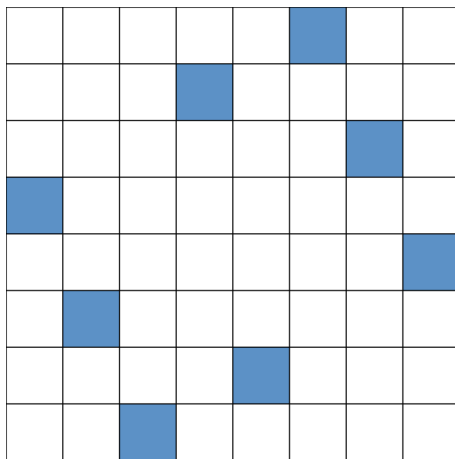


図 3: 8 クイーンの 1 つの解

仮説を作る際には， $N * N$ の位置に Queen が設置される可能性があるため，単純に $N * N$ の数だけその位置についての仮説を用意する．次に，Nogood を作成するわけだが，N-Queen の解答では「同じ列には 1 つの Queen しか配置されない」ことを利用する．つまり，同じ列に Queen が配置されることを考えずに，

1. 同じ行に Queen がいないか
2. 斜めの列に Queen がいないか

の2つを考えればよいとする．すると，検証を大きく減らすことができる．したがって，Nogoodの作成には，2つのQueenの位置を $(i_1, j_1), (i_2, j_2)$ とすると，

1. j については考えず， i が同じか
2. $|i_1 - i_2| = |j_1 - j_2|$ か

を確認し，どちらか一方でも成り立てば，その2つのQueenの位置を記憶するのである．さて，以上で仮説とNogoodを作成することができた．次は，これらを用いてノードを作り，ラベルを更新していく．

2.3 ノードの作成とラベルの更新，解の導出

まずは，1列目にQueenを置く仮説 $(k, 1)(k = 1..N)$ 全てに対してノードを作り，それらを正当化する．なぜなら，最初のQueenの位置は任意であり，この時点では矛盾を含んでいないからである．これは，図4のようになる．

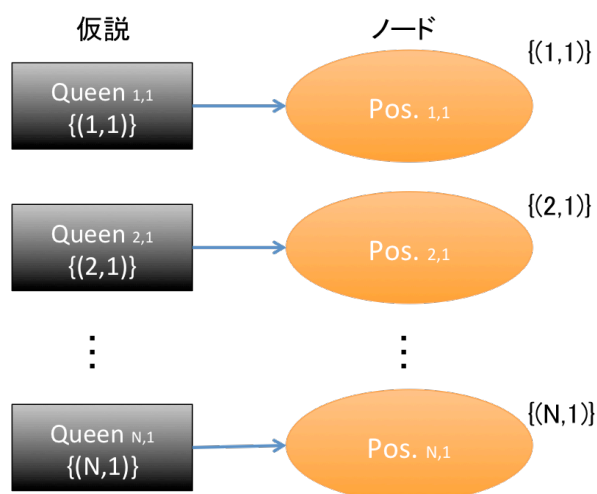


図 4: 最初の列

次は，2列目に移行し，それぞれのノードと，2列目にQueenを置く仮説を用いて，ラベルを更新する．このときに，Nogoodとなる環境を含むノードがあれば，そのノードは正当化できない．この様子を，図5に示した．上部の $\{(1, 1), (1, 2)\}$ はNogoodに含まれているので，正当化されず， $\{(1, 1), (3, 2)\}$ は正当化される．この操作によって，残ったノードのみについて， N 列目まで同様の操作を繰り返していく．

そして，最後まで残ったノードを数え上げれば，それが解法の数になるというアルゴリズムである．

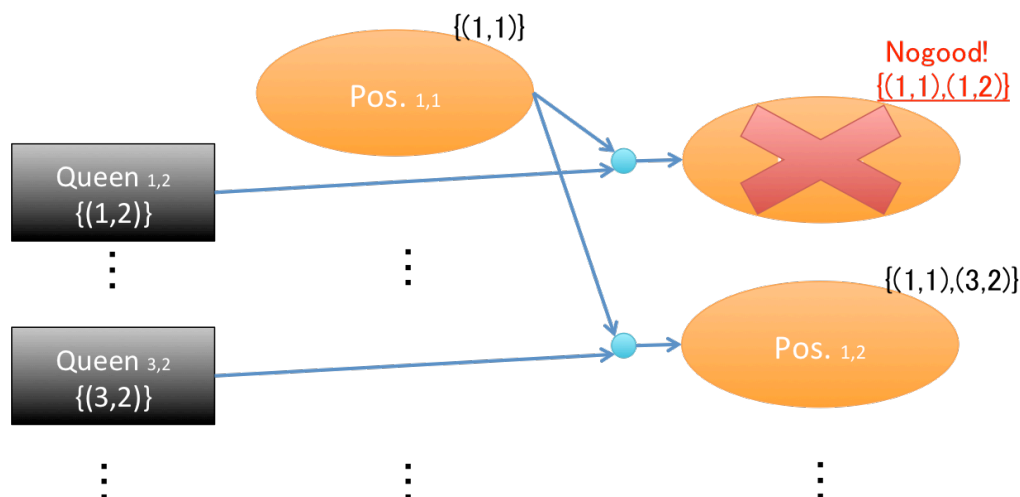


図 5: ラベルの更新

参考文献

- [1] 『人工知能特論』(last accessed at 2013/11/07), <http://winnie.kuis.kyoto-u.ac.jp/~okuno/Lecture/02/AI/ai-02-9-ohp.pdf>
- [2] 『Peter@Norvig.com』(last accessed at 2013/11/07), <http://norvig.com/ltd/test/atms.lisp>
- [3] Takeuchi Lab: 『講義資料置き場>知識工学』(last accessed at 2013/11/07), <http://cl.it.okayama-u.ac.jp/kougi/data/knowledge>
- [4] 情報工学教育類 / 産業戦略工学専攻 伊藤孝行: 『知能処理アルゴリズム論 第 6 回 推論システムと TMS』(last accessed at 2013/11/07), <http://www.itolab.nitech.ac.jp/~ito/Lecture/IntelligentAlgorithm2011/IntelligentAlgorithm6.pdf>
- [5] 伊庭斉志: 『人工知能と人工生命の基礎』, オーム社, pp.37-51, (平成 25 年 5 月 24 日)