

# 進化計算法を用いた詰将棋の自動生成

宗藤大貴<sup>1,a)</sup> 長尾智晴<sup>2</sup>

**概要:** 人工知能の発展により、様々な論理パズル問題を人間の事前知識なしで解く能力は年々向上している。しかし、パズル問題の創作に関しては、解が一つでないことや生成された問題の評価が難しいことから解答に比べ困難なタスクである。本稿では、パズル問題の中でも、持ち駒があるなどの性質から創作が難しいと考えられる詰将棋を題材とする。エキスパートによる詰将棋創作では、短手数の詰将棋や長手数の余詰めのある詰む局面の情報を利用する。この考え方に基づき、進化計算法の1つである遺伝的アルゴリズムによって、局面の変換を最適化することで長手数の詰将棋を生成する手法を提案する。実験の結果、最長で31手詰めの詰将棋が生成できることが確認できた。

## Tsume-Shogi problem composition using evolutionary computation

DAIKI MUNETO<sup>1,a)</sup> TOMOHARU NAGAO<sup>2</sup>

**Abstract:** With the development of artificial intelligence, the ability to solve various logic puzzle problems without human knowledge is improving year by year. However, creating puzzle problems is a more difficult task than answering because there is no single solution and it is difficult to evaluate the created problem. In this paper, we focus on the chess shogi, which is thought to be difficult to create due to the nature of the puzzles problem. The expert uses the information on shorter-move mates or incomplete mates in composition. Based on this idea, we propose a method to compose longer-move mates by optimizing the board conversion by using a genetic algorithm, which is one of the evolutionary computation methods. As a result of the experiment, it was confirmed that 31-move mate can be generated.

### 1. はじめに

人工知能の急速な発展により、コンピュータにおいて数独やルービックキューブなどの論理パズル問題を人間の事前知識なしで解く能力は年々向上している[1,2]。しかし、パズル問題の創作に関しては、解が一つでないことや、生成された問題の評価が難しいことから解答に比べ困難なタスクとされている[3]。本稿では、パズル問題の中でも、持ち駒などの関係から、現在でも難しい問題の創作が困難である詰将棋を題材とし生成を試みた。

詰将棋においては、解答や解説文の生成などの研究例はあるものの[4,5]、創作に関する研究は例が少なく、人間の創作するレベルにはいまだ至っていない。詰将棋創作の研究として逆算法という手法[6]が挙げられるが、この手法では詰

み手数が11手以上の詰将棋を作成することが困難である。これは、深く逆算していくと探索空間が爆発的に増加し、計算資源が足りなくなるためである。

本稿では、進化計算法の1つである遺伝的アルゴリズム[7]を用いて、局面の変換を最適化することで長手数の詰将棋を生成する手法を提案する。本手法は、長手数の詰将棋を生成する際には、駒の配置が似ている短手数の詰将棋や余詰めのある詰みが生じる局面の情報が重要であるという考えに基づく。また、実験によって詰将棋の生成を行い手法の有効性を検証する。

### 2. 研究に関する理論

#### 2.1 詰将棋について

詰将棋とは本将棋の終盤局面を模した、1人用のパズルである。問題として与えられたある局面から相手を詰ませることを目的とする。詰将棋には本将棋で適応されるルール加えて、いくつかの専用のルールが存在する。本稿で用いる詰将棋に関する用語を以下に定義する。各定義は[6]を参考

1 横浜国立大学大学院環境情報学府  
Graduate School of Environment and Information Sciences,  
Yokohama National University

2 横浜国立大学大学院環境情報研究院  
Faculty of Environment and Information Sciences, Yokohama  
National University

a) muneto-daiki-mv@ynu.jp

にした。

**詰み** 玉方（詰まされる側）の手番であり、玉方が王手をかけられていて、合法手がない状態のこと。

**正解手順** 双方が最善を尽くし、詰む場合の手順。

**不詰め** 双方が最善を尽くした場合に詰まないこと。

**駒余り** 正解手順に従った場合、詰みの局面で攻め方（詰ます側）が持ち駒を残していること。

**余詰め** 攻め方が正解手順から手を変えた場合に詰む手順があること。ただし、最終手は除く。

（完全作の）詰将棋 駒余り、不詰め、余詰めのない問題

ここでいう「双方が最善を尽くす」とは、攻め方は王手かつ最短手順で詰むような手を選択すること、玉方は詰みまでの手順が最長になり、攻め方に持ち駒をできる限り使わせる、かつ、手数稼ぎになる無駄な合駒（無駄合）はしない手を選択するということを指す。

## 2.2 遺伝的アルゴリズム

遺伝的アルゴリズム（Genetic Algorithm; GA）は、生物の一般的な進化論をもとに提案された進化計算アルゴリズムである[7]。遺伝的アルゴリズムは解の候補を遺伝子で表現した個体を複数用意し、適応度の高い個体を優先的に選択して交叉・突然変異などの操作を繰り返しながら解を探索する手法である。基本的な遺伝的アルゴリズムの処理手順を図 1 に示す。

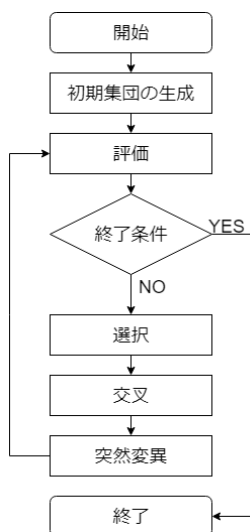


図 1 遺伝的アルゴリズムの処理手順

次にそれぞれの操作について詳細を述べる。

**選択** 選択は個体の適応度に応じて個体集団から個体を選択することによって個体の淘汰を行う操作である。

**交叉** 交叉は個体間で遺伝子の一部を入れ替える操作であり、遺伝的アルゴリズムの最も重要な操作である。

**突然変異** 個体の遺伝子の一部を変化させる操作である。

局所解に陥ることを防ぐ効果がある。突然変異の確率が高すぎるとランダム探索に近づいてしまうため、一般的に低い値が用いられる。

## 3. 関連研究

### 3.1 詰将棋の解答

パズル問題としての詰将棋は、人工知能分野の良き題材として研究されてきた。詰将棋の解が厳密に 1 つに決まることや、問題によって難易度が大きく異なることは、探索手法を試すのには最適であった。コンピュータによる解答は研究当初ミニマックス法[8]や  $\alpha$   $\beta$  探索[9]を用いて行われていた、その後、共謀数探索[10]や Proof-Number Search[11]、Depth-First Proof-Number Search[12]といった探索手法が考案され、非常に効率的に問題を解くことが可能となっている。現在では、世界最長の 1525 手詰めの詰将棋「マイクロコスモス」が解かれ、コンピュータによって解けない詰将棋はほぼ存在しないといえる。

### 3.2 詰将棋の創作について

詰将棋の生成および創作については、解答と比較すると研究例は少ない。現在までに、逆算法[6]や正算法[13]といったアルゴリズムが提案されている。以下に各手法の概要を示す。

#### 3.2.1 逆算法

逆算法は予め用意した詰み局面を利用し詰将棋を生成する手法である。詰将棋は正しい詰め手順を行えば最終的に玉方を詰ますことができる。詰みの状態から逆順に戻していけば元の詰将棋を得ることができる。これを利用し、詰み局面から一手戻した局面について検査を行い、詰将棋のルールを満たしているものを採用するという手法である。逆算法では 11 手以上の詰将棋を作成することが困難である。これは、深く逆算していくと探索空間が爆発的に増加し、計算資源が足りなくなるためである。

#### 3.2.2 正算法

正算法はランダムに配置した初期局面を利用し、駒の削除や変更という操作を繰り返すことで詰将棋を生成する手法である。相手玉を含む複数の駒を盤上および持ち駒としてランダムに配置し、詰み手順が存在するかを判断し、詰み手順があった場合は局面の駒に対し、除去、減少、変換という操作を繰り返し生成された詰将棋を保存する。この手法では主に 13 手から 19 手の詰将棋の生成が確認できている。正算法の操作では駒の追加や移動は含まれておらず、元の局面から大幅に変化した局面は生成できないといった特徴がある。

## 4. 提案手法

エキスパートが詰将棋を創作する際、短手数

から手数数を延ばすこと[14]や長手数の余詰めのある詰む局面から余詰めを消すように局面を変化させることで詰将棋を創作する。本稿では、長手数の詰将棋を生成するには駒の配置が似ている短手数の詰将棋や余詰めのある詰む局面の情報が重要であるという考え方にに基づき、遺伝的アルゴリズムを用いて、長手数の詰将棋を生成する手法を提案する。任意の詰将棋を初期局面として、局面を変化させる箇所および駒を遺伝的アルゴリズムによって最適化する。

長手数の詰む局面を生成するように評価関数を定義し、駒の配置が類似している短手数の完全作の詰将棋の情報を利用するためにアーカイブ母集団の利用を提案し、長手数の完全作の詰将棋の生成を目指す。

局面の逆算を行わないため、逆算法と異なり、長手数の詰将棋も探索可能である。また、局面全体の変化の最適化を行うため、複数の駒が相互に影響して詰手数が増加することを考慮できる。

変化させる駒の種類や場所について、詰将棋のルールによる制約以外の条件を設定していないため、初期局面に単純な詰将棋を使用して生成する場合、つまり、初期局面の選択においてエキスパートの知識を使用しない場合、エキスパートの事前知識なしでの詰将棋生成といえる。

次に提案手法の詳細を説明する。

#### 4.1 遺伝子型

本手法では、各個体を「将棋盤のマス+1 (先手の持ち駒)」×「駒の種類 (成駒を含む)」の各組み合わせを0と1のビット列で表現する。1の場合は初期局面に対して、該当するマスの該当する駒を局面に生成する。なお、生成の際に該当マスに複数の駒が生成される (遺伝子に複数の1が存在する) 場合はランダムに生成し、初期局面において該当マスに玉以外の駒が存在していた場合は駒を上書きし、玉が存在する場合は生成しない。

#### 4.2 評価関数

評価関数  $f$  は長手数の詰む局面を生成することを目的として、次のように定義する。

$$f = \begin{cases} 0 & \text{if 詰まない or 初期局面と同一手順で詰む} \\ \text{詰み手数} & \text{if 詰む and 完全作である} \\ \text{最短の詰み手数} & \text{if 詰む and 完全作でない} \end{cases}$$

#### 4.3 選択

本手法では選択の際に、短手数の完全作の詰将棋の情報を活用すること、完全作でない解への収束を防ぐことを目的として、アーカイブ母集団を用いることを提案する。評価時に完全作の詰将棋が生成された場合、その個体をアーカイブ母集団に保存する。選択の際に、設定した確率 (アーカイブ使用率) でアーカイブ母集団からランダムで個体を選

択する。アーカイブ母集団の個体を使用しない場合は、トーナメント選択によって選択を行う。

#### 4.4 交叉

本手法では、ランダム生成との比較を行うこと、変化する駒数に上限を設定することを目的とし、遺伝子中の0と1の数を一定にするように交叉を行う。交叉はそれぞれの個体間で交換する0と1の数が同数になるように行う。

#### 4.5 突然変異

交叉同様に、遺伝子中の0と1の数を一定にするように突然変異を行う。具体的には0から1への変異と1から0への変異の数が同数になるよう行う。本手法での突然変異率は0から1へ変化する遺伝子の確率を示す。

### 5. 詰将棋生成の実験

#### 5.1 パラメータ設定

遺伝的アルゴリズムにおける各種パラメータを表1に示す。なお、第1世代では各遺伝子の初期化の際に、ランダムに1を10個発生させ残りは0とした。局面の余詰めチェックには脊尾詰[15]を利用した。

表 1 遺伝的アルゴリズムのパラメータ

	パラメータ
個体数	200
世代数	50
選択方法	トーナメント選択
トーナメントサイズ	8
突然変異率	0.001
アーカイブ使用率	0.05

#### 5.2 初期局面

今回、初期局面として1~5手詰めの詰将棋を作成し、実験を行った。使用した局面を次の図2~4に示す。

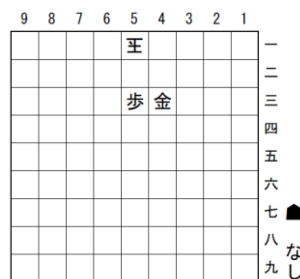


図 2 初期局面 1  
1手詰め



図 3 初期局面 2  
3手詰め

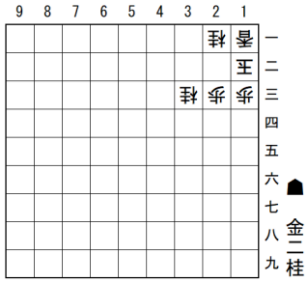


図 4 初期局面 3  
5 手詰め

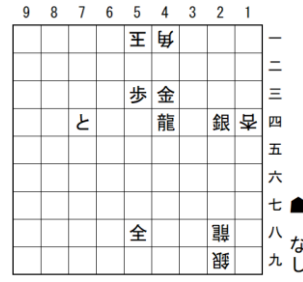


図 6 生成局面 1  
19 手詰め

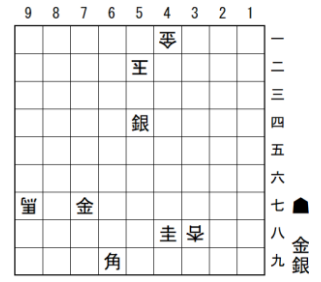


図 7 生成局面 2  
19 手詰め

### 5.3 実験結果

提案手法とランダム生成，アーカイブ母集団を使用しない場合の遺伝的アルゴリズムにおける，最終世代の平均評価値，最大評価値，および生成された詰将棋の最長手数を表 2 に示す．なお各値はシード値を変えて実験を行った 3 試行の平均値である．各世代の評価値の最大値と，その世代までに生成された詰将棋の最長手数，ランダム生成およびアーカイブ母集団を使用しない場合の遺伝的アルゴリズムのその世代までに生成された詰将棋の最長手数の推移を図 5 に示す．各推移は全ての試行の平均である．提案手法において，各初期局面から生成された最長手数の詰将棋を図 6 ～ 8 に示す．

表 2 最終世代の結果

初期局面	手法	平均評価値	最大評価値	最大詰手数
初期局面 1	Random	0.27	24.33	14.33
	GA	7.80	33.00	17.00
	GA_archive	5.44	29.00	<b>18.33</b>
初期局面 2	Random	1.37	24.33	13.67
	GA	7.66	31.00	13.00
	GA_archive	5.62	26.33	<b>15.00</b>
初期局面 3	Random	1.91	24.33	14.33
	GA	7.51	31.00	<b>19.67</b>
	GA_archive	7.86	31.67	<b>19.67</b>
平均	Random	1.18	24.33	14.11
	GA	7.66	31.67	16.56
	GA_archive	6.31	29.00	<b>17.67</b>

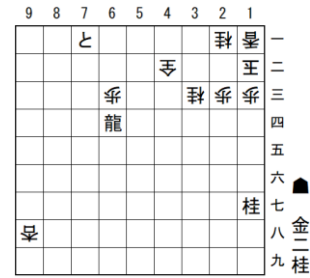


図 8 生成局面 3  
21 手詰め

### 5.4 考察

今回は 3 種類の初期局面に対して実験を行ったが，いずれの場合も 19 手以上の詰将棋の生成が確認できた．ランダム生成やアーカイブ母集団を利用しない場合よりも最長手数の詰将棋を生成することができており，提案手法の有効性が確認できた．

どの初期局面から生成した場合でも手数に大きな差はなく，複雑な詰将棋を初期局面としない場合，生成される詰将棋の手数は初期局面に依存しないといえる．初期局面 1 のような単純な詰将棋から生成でき，事前知識を使用せずに最長手数の詰将棋を生成可能な手法といえる．

次に各生成局面について考察する．生成局面 2 では全ての駒が，初期局面で存在する駒の位置から離れた，六～九段目の段に生成されていることが確認できる．これは従来の逆算法や正算法，また，範囲を絞ったランダム生成では生成が困難な詰将棋である．生成局面 3 は局面図では 1 列目にいる玉が詰め上がり図では 9 列目に移動する．詰め上がり図を図 9 に示す．正解手順の中で玉が局面の右端から左端に移動し，曲詰めのような面白さがある詰将棋といえる．

詰将棋問題では詰将棋には必要のない駒（無駄駒）の存在は，好ましくない．本手法では無駄駒が生成されてしまうため無駄駒の消去を行った．無駄駒の消去は，詰み手順が変化しないように盤上の駒を 1 枚消去することを可能な限り繰り返すという単純なアルゴリズムによって行うことができる．各生成局面から無駄駒を消去した局面を生成局面 1' ～ 3' として図 10 ～ 12 に示す．

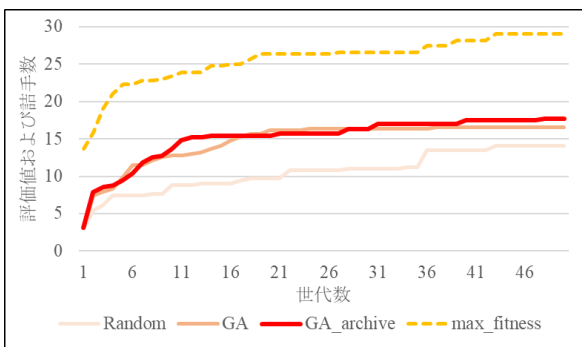


図 5 評価値と最大詰手数の推移

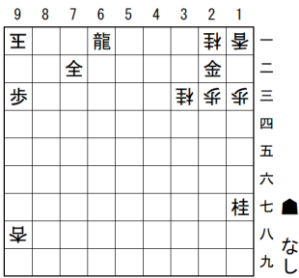


図 9 生成局面 3  
詰め上がり図

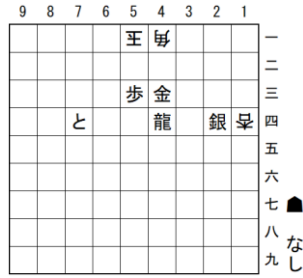


図 10 生成局面 1'

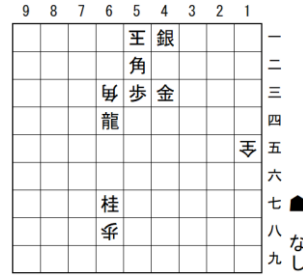


図 15 途中局面 1  
6 世代目, 19 手詰め

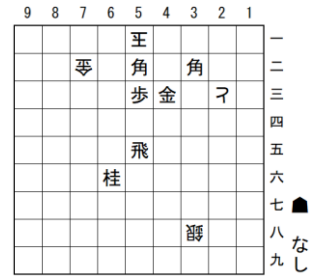


図 16 途中局面 2  
16 世代目, 25 手詰め

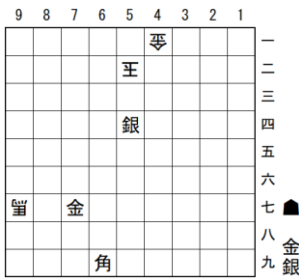


図 11 生成局面 2'

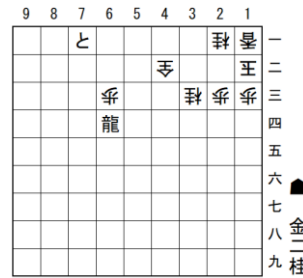


図 12 生成局面 3'

各局面とも図 14 と同様に 5 二の地点に角が存在する詰将棋であり, 図 15 においては, 3 二の地点の角の配置も図 14 と同様である. これらの情報が 31 手詰めを生成する際に使用されたと考えられる.

例えば, 図 15 のように角を中心に使用する詰将棋が生成された試行では図 13 のように飛車を中心に使用する詰将棋はほとんど生成されないことが確認できた. そのため, 本手法において, 多様な詰将棋を生成するためには複数回の試行を行う必要がある.

### 5.5 追加実験

より長手数な詰将棋の生成を目指し, 個体数と世代数を増やしての実験を行った. 計算時間の関係上, 対象とする初期局面は初期局面 1 のみとした. その結果, 前述の設定よりも長手数な詰将棋を生成することができ, 個体数 300, 世代数 50 の場合に最長で 29 手, 個体数 500, 世代数 50 の場合に最長で 31 手の詰将棋を生成することができた. それぞれの生成局面を図 13 ~ 14 に局面を示す. なお, それ以上個体数や世代数を増やしても 31 手を超える詰将棋は生成できなかった.

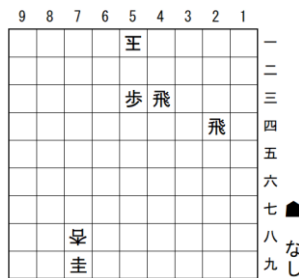


図 13 生成局面 4  
29 手詰め

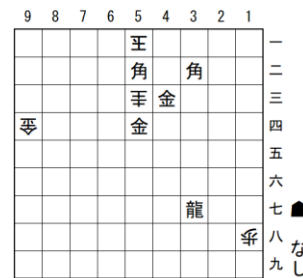


図 14 生成局面 5  
31 手詰め

図 14 の局面が生成された試行において, 進化の途中で生成された詰将棋の一部を図 15 ~ 16 に示す.

## 6. おわりに

本稿では, 長手数な詰将棋の生成を目的として, 駒の配置が似ている短手数な詰将棋や余詰めのある局面の情報が重要であるという考えのもと, 進化計算法の 1 つである遺伝的アルゴリズムを用いて, 新たな詰将棋局面を生成する手法を提案した. 実験の結果から, 逆算法では現実的に困難とされていた 11 手以上の長手数な詰将棋を複数生成し, 最長で 31 手詰めの詰将棋を生成することが確認できた.

今回は, 局面の変化を行うための遺伝子型を単純なビット列で表現した. 今後は配置と駒の種類を共有するために, 例えば, 「生成する場合の各マスの先手駒の種類」「生成する場合の各マスの後手駒の種類」「局面のマスク (先手駒を生成, 後手駒を生成, 生成しない)」「先手の各持ち駒生成」の 4 種類で表現するというような, 別の符号化方法の検討する必要がある.

### 参考文献

- [1] PALM, Rasmus; PAQUET, Ulrich; WINTHER, Ole. Recurrent relational networks. In: Advances in Neural Information Processing Systems. p.3368-3378, 2018.
- [2] MCALEER, Stephen, et al. Solving the Rubik's Cube with Approximate Policy Iteration. The International Conference on Learning Representations 2019.
- [3] Iqbal, A. Improving automatic chess problem composition by emphasizing conventions. Journal of Fundamental and Applied Sciences, Vol.10, No.6S, p.1744-1756, 2018.
- [4] 石脇滉己, 荒川達也. 「一見良い手」を含めた初心者向

- け詰将棋解説文生成の提案, 研究報告ゲーム情報学 (GI) 2015.5, p.1-7, 2015.
- [5] 千葉景太. 強化学習を用いて人間の直感を再現した詰将棋解答 AI の作成. ゲームプログラミングワークショップ 2018 論文集, p.133-136, 2018.
  - [6] 広瀬正幸, 伊藤琢巳, and 松原仁. 逆算法による詰め将棋の自動創作. 人工知能学会誌, Vol.13, No.3, p.452-460, 1998.
  - [7] Holland, John Henry. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press, 1992.
  - [8] C. E. Shannon. Programming a computer for playing chess. Philosophical Magazines, Vol.41, p.256-275, 1950.
  - [9] Knuth, D. E. and Moore, R. W. An analysis of alpha-beta pruning. Artificial Intelligence, Vol. 6, No. 4, p.293-326, 1975.
  - [10] D. A. McAllester. Conspiracy number for min-max search. Artificial Intelligence, Vol.35, p.287-310, 1988.
  - [11] M. Seo, H. Iida, and J. W. H. M. Uiterwijk. The PN\*-search algorithm: Application to tsume-shogi. Artificial Intelligence, Vol.129, No.1-2, p.253-277, 2001.
  - [12] 長井歩, 今井浩. df-pn アルゴリズムの詰将棋を解くプログラムへの応用. 情報処理学会論文誌, Vol.43, No.6, p. 1769-1777, 2002.
  - [13] 春日利文, 中家 啓文, 中山 義久, 乾 伸雄, 小谷 義行. 正算法に基づく詰将棋創作実験. 情報処理学会第 56 回全国大会講演論文集. p.307-308, 1998.
  - [14] “伊藤果先生の「詰将棋の作り方入門」”.  
<https://book.mynavi.jp/shogi/detail/id=66141>, (参照 2019-10-06)
  - [15] ” 春尾詰ダウンロード”  
<http://panashogi.web.fc2.com/seotsume.html> (参照 2019-10-06)