

決定的アルゴリズムを併用した 遺伝的アルゴリズムによるピクロス解決の効率化

上水流歩望[†] 佐藤裕二[†]

概要：パズルゲームの一種であるピクロスの大規模な問題を解くための遺伝的操作の提案と、効率的に GA を実行するための決定的アルゴリズムを併用する手法の提案を行う。ランダムに生成した 20×20 のピクロス問題を 50 種類使用した評価実験から、提案するマスの確定法を行う GA が、単純にマスの確定法を行う GA より探索精度を大きく向上できる可能性を示す。

キーワード：進化計算、遺伝的アルゴリズム、組み合わせ最適化、ピクロス

Making efficient genetic algorithms applied to Picross by using deterministic algorithms

AYUMI KAMIZURU[†] YUJI SATO[†]

Keywords: Evolutionary Computation, Genetic Algorithm, Combinatorial Optimization, Picross

1. まえがき

ピクロスはパズルゲームの一種で、制約充足問題や組み合わせ最適化問題に属すると考えられる。一般に小規模の問題ならば、決定的な探索アルゴリズムによって高速に解くことができる。しかし問題の規模が巨大になると、単純な探索では解に到達できず、背理法を用いる複雑な探索では現実的な時間では解けない。

この問題に対応するためにメタヒューリスティクス手法の一つである遺伝的アルゴリズム (GA) を用いたピクロス解法の研究が始まっている[1-3]。一方、これら GA を用いた先行研究は小・中規模のピクロスの解法には成功しているが、大規模な問題に対しては局所解に陥る場合が多く、また解が求まる場合でも多大な実行時間を必要とする。本稿では、GA を用いた大規模ピクロス解法の探索精度の向上と処理時間の短縮のための遺伝的操作の提案と、GA が効率的に動作するための決定的アルゴリズムの提案を行う。ランダムに生成したテストデータを用いて、決定的アルゴリズムを併用して GA を効率化する手法を評価し、その有効性を検証する。

2. ピクロスの問題サイズと難易度

2.1 ピクロスのルール

長方形のマス目と、その各行各列に対応する 1 つ以上の数字が与えられる。初期状態でマス目は全て白マスであり、与えられた数字をヒントに、制約と矛盾ないように黒マスを配置できれば完成となる。数字は白マスを挟まずに連続して置かれる黒マスの個数を表している。複数の数字が

並んでいる場合は、数字の位置関係が対応する黒マスの位置関係を表し、対応する黒マス同士の間には 1 つ以上の白マスが存在する。

2.2 ピクロスと探索空間

ピクロス問題の難易度は一般に問題のサイズと相関関係が高いため、まずピクロスの問題のサイズと探索空間の関係を考える。ここで、ピクロスにおける探索空間の広さとは、黒マスを配置するパターン数の大きさである。一方、同じサイズの問題であってもそれぞれパターン数は異なる。ここではサイズの増加に伴うパターン数の最大値 Y' の増加を調べ、 Y' の近似関数を求めてサイズと探索空間の関係を示す。

まず、1 行中のパターン数 y を求める式を示す。図 1 は、1 行中のパターン数を計算する例である。

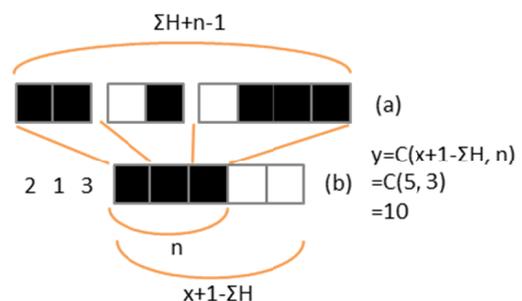


図 1 1 行中の組み合わせの計算例

行の長さ $x=10$ 、ヒントの配列 $H=[2, 1, 3]$ 、 H の長さ $n=3$ である。黒マスの配置を決定するとき、ヒントが表すマス (a) を n 個の黒マスに代表させたマス (b) にして考えることができる。(a) のマス数は制約に矛盾しないために必

[†]法政大学
Hosei University.

要な白マス数 $n-1$ とヒントの数字の和 $\Sigma H + (n-1)$ である。
 (b) のマス数は (a) を代表するマス数 n と (a) に使用されていないマス数 $x - (\Sigma H + (n-1))$ の和 $n + (x - (\Sigma H + (n-1))) = x + 1 - \Sigma H$ である。 y は (b) のマス数のうち n マスを選択する組み合わせの数となるので、次の式が成り立つ。

$$y = C(x + 1 - \Sigma H, n) \quad (1)$$

次にピクロスの 1 行の長さが x マスであるときの y の最大値 Y を求める。 $H=[0]$ のとき明らかに $y=1$ である。 $H=[0]$ 以外の全てのヒントの構成は、 $H=[1]$ より開始して、 H の末尾に新たに要素 1 を追加して $H'[n+1]=1$, $n'=n+1$ とする操作 (操作 1) と、要素の内 1 つに 1 を加えて $H'[i]=H[i]+1$ とする操作 (操作 2) を繰り返し行うことで到達することが出来る。 $w=x+1-\Sigma H$ とする。 $y=C(w, n)$ であるとき、式 (1) より、操作 1 を行うと $y=C(w-1, n+1)$ となり、操作 2 を行うと $y=C(w-1, n)$ となる。操作 1 では y と y' の大小関係は H により異なるが、操作 2 では $C(w, n) > C(w-1, n)$ より $y > y'$ が成り立つ。 y の最大値 Y を求めるためには、 y が減少する操作 2 は行わず、操作 1 のみを行えばよい。図 2 にプログラムで操作 1 を繰り返して求めた Y とその近似関数を示す。

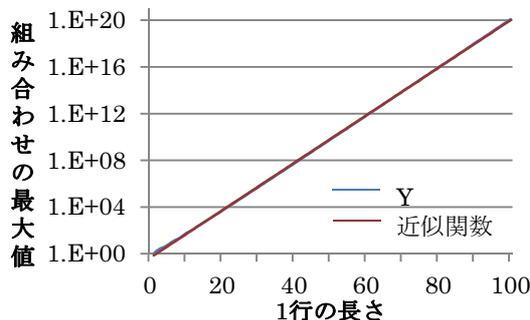


図 2 行の長さとお組み合わせの最大値

Y が片対数グラフにおいてほぼ直線状に並んでいたことから近似関数は $f(x) = \exp(ax+b)$ の形で表現できると考え、表計算ソフト Excel を用いて計算した結果 $f(x) = \exp(0.468794x - 0.802390)$ を得てそれを近似関数として示した。図 1 では Y と近似関数の 2 系列が示されているが、原点付近のわずかな差を除き 1 つの直線のように見え、高い精度で近似できている。

次に 1 行中の組み合わせの最大値 Y を元にして問題全体の組み合わせの最大値 Y' を求める。ピクロスの問題は長方形であり、行と列の長さは同程度であるものが多い。ここでは単純に行と列の長さが等しい、1 辺が x マスの正方形の問題であるとする。 x 行全てで Y 通りのパターンがある場合に組み合わせの数が最大であるので、 $Y' = Y^x$ である。図 3 に 35 マスまでの Y' とその近似関数を示す。 Y の近似関数を元にして、 Y' の近似関数は $f(x) = c(\exp(ax+b))^x$

$= \exp(ax^2 + bx + c)$ の形で表現できると考え、Excel を用いて計算した結果 $f(x) = \exp(0.465112x^2 - 0.865190x + 1.63598)$ を得てそれを近似関数として示した。

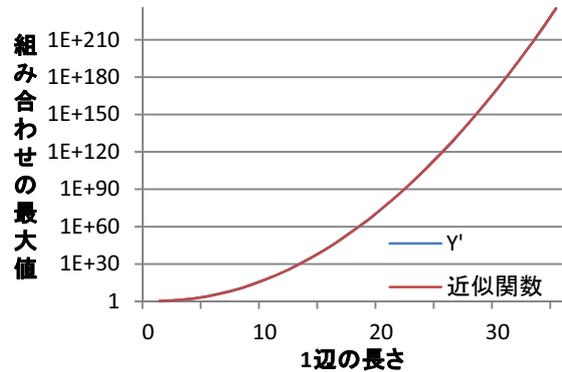


図 3 辺の長さとお組み合わせの最大値

図 2 では Y' と近似関数の 2 系列が示されているが、1 つの曲線のように見え、高い精度で近似できている。探索空間のオーダーは $\exp(O(x^2))$ と表せる。これは問題のサイズが大きくなると探索空間が非常に急激に増加することを表し、大規模なピクロスが困難な問題であることを表す。

3. 効率的な GA のための決定的アルゴリズム

3.1 確定アルゴリズムの狙い

GA を行う前にマスを確認する決定的アルゴリズム (確定アルゴリズム) を実行し、GA を効率化するための情報を得る。マスを確認するとは、あるマスで白マスと黒マスのどちらが正解であるかを知ることである。サイズの大きい問題では探索空間も巨大であり、GA を行っても局所最適解に陥る可能性が高いと考えられる。GA を適用する前に、予め効率的な確定アルゴリズムの適用を行うことにより、確定されたマスと矛盾するような黒マスの配置パターンを生成することなく探索することができ、探索効率が向上して局所最適解に陥る可能性が下がることが期待できる。

3.2 提案する確定アルゴリズムの実行手順

確定アルゴリズムは背理法を用いない単純な探索である。初期状態として全ての黒マスは制約と矛盾しない全ての位置を位置候補として持つ。あるマスが確定したとき、そのマスと矛盾するような位置候補があればその位置候補を除外する。確定するマス数は問題に依存する。全てのマスが確定した場合は問題が解かれたということであるので、GA を行わずに探索を終了する。未確定のマスが残っている状態で新たに確定できるマスが無い場合は、問題を解くために GA での探索を開始する。

マスが確定したときの位置候補の除外方法を示す。図 4 はヒント $H=[1, 3]$ で 3 番目のマスが黒、6 番目のマスが白と確定した例である。下部の数字は $H[1]$, $H[2]$ のそれぞれに 0 から 5 番目までの位置候補があることを表す。位置候補は

連続する黒マスの先頭の位置である。例えば H[2] の 0 番目の位置は 3, 4, 5 番目のマスが黒マスとなることを表す。3 番目のマスが黒マスであることで H[1] の 1, 3 番目, H[2] の 1 番目の位置は制約に反する（連続する黒マスの個数がヒントの数字を超える）ため除外される。また 6 番目のマスが白マスであることで H[1] の 5 番目, H[2] の 1, 2, 3 番目の位置は矛盾するため除外される。

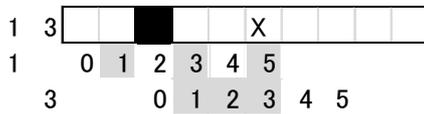


図 4 位置候補の除外の一例

マスを確定する手順を示す。マスの確定は 2 段階で行われる。第一段階はヒントのみを利用してマスを確定し、1 度だけ行われる。図 5 はヒント H=[5, 2] の例である。各行各列について制約を満たす範囲で、全ての黒マスを先頭から連続して配置した場合 (a) と、全ての黒マスを末尾から連続して配置した場合 (b) を比較する。あるマスがどちらの場合でも黒マスでありそれが同一のヒントの数字に対応しているとき、そのマスを黒マスと確定する (c)。どちらの場合でも配置が同じであるとき、すなわち配置が 1 通りのみであるときには、その行 (列) の全てのマスを得られた 1 通りの配置で白マス、黒マス共に確定する。

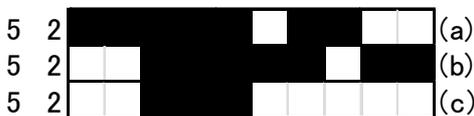


図 5 第一段階の確定の一例

第二段階はヒントと既に確定されたマスを利用してマスを確定し、新たにマスが確定されなくなるまで繰り返される。各行各列について、黒マスを真、白マスを偽として、黒マスの取り得る全ての配置について同じ位置のマス同士で論理和と論理積を計算する。

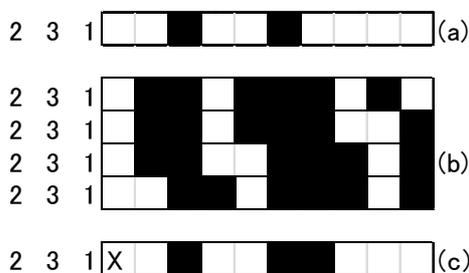


図 6 第二段階の確定の一例

図 6 はヒント H=[2, 3, 1] で 3 番目と 6 番目が黒マスと確定している例である (a)。確定しているマスに矛盾しない配置の全て (b) について、論理和を求めると 1 番目のマス

が偽、他のマスが真となり、論理積を求めると 3, 6, 7 番目のマスが真、他のマスが偽となる。論理和が偽となるのは全ての配置で白マスであったマスのみであるため、1 番目のマスが白マスと確定する (c)。同様にして論理積が真であった 3, 6, 7 番目のマスのうち未確定であった 7 番目のマスが新たに黒マスと確定する。

4. ピクロスへの GA の適用

4.1 染色体と初期個体

染色体 G は整数型の二重配列で定義する。問題の行数を x , i 行目のヒント配列 H_i の長さを n_i とすると $G[x][n_i]$ の大きさを持つ。 $G[i][j]$ の遺伝子は $H_i[j]$ の表す黒マスの位置を表す。ここで各遺伝子は制約に従う範囲で、図 4 に示されているように、対応する黒マスの最も先頭に近い位置を 0 とする。全ての初期個体は、突然変異のアルゴリズムを利用して行全体を変異させることで、制約を満たす範囲で全ての遺伝子をランダムに生成する。

4.2 評価方法

個体の評価値は、その個体の持つマス目が行列の制約に反する度合いを計算した点数とする。今回の各遺伝的操作において個体の遺伝子は常に行の制約を満たしながら変更されるので、行の制約に反する度合いは計算しない。列の制約に反する度合いは、各列の黒マスの取り得る配置のなかで、最も個体の持つ列との矛盾点が少ない配置の点数を、全列で合計したものとする。この計算による点数は、個体の持つマスを自由に変更して列の制約との矛盾を解消するための最小の変更マス数を意味する。すなわち評価値が低いほど適応度が高い優秀な個体である。点数が 0 である個体を発見したとき、解を発見したとみなし探索を終了する。制約を満たす解が複数ある場合でも、そのうちの 1 つを発見したとき終了する。この計算方式は先行研究[1]と同様である。

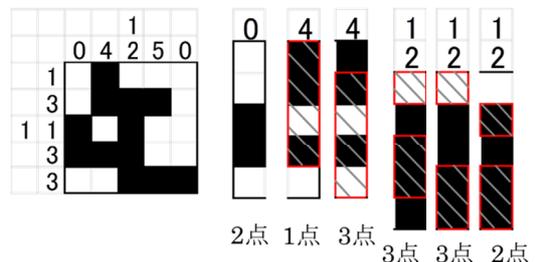


図 7 評価関数による計算のイメージ

本研究では各列の点数計算は動的計画法で行う。全ての黒マスの配置パターンを生成して計算する場合、大規模な問題では組み合わせ爆発が起こり長い処理時間がかかる。動的計画法では、組み合わせを細分化して段階的に解くことにより多項式時間で処理することができる。

図 8 は動的計画法による計算の様子を表した例である。

上部の1行はこれより評価するある個体の持つ列と、その列のヒント H である（説明のため転置してある）。列の長さ $x=10$, H の長さ $n=2$ である。

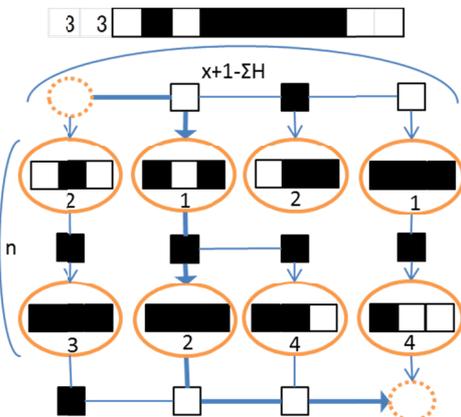


図 8 動的計画法による評価計算の例

図 8 のグラフは $n(x+1-\Sigma H) = 8$ 個のノード（楕円）とダミーのノード（点線の円）からなるグラフである。ダミーのノードは左上のものが先頭を、右下のものが末尾を表す。このグラフで先頭ノードから右か下に進み、末尾ノードに到達する経路 1 つが黒マス配置するパターン 1 つに対応し、最も点数の低い経路（最短経路）を探索することが最も点数の低い配置を探索することに対応する。最短経路の点数がその列の点数となる。 i 行 j 列のノードを通ることは $H[j]$ の表す黒マスを $j-1$ 番目の位置に置くことと想定することを表す。ノード中のマスは、想定される黒マスと同じ位置の、個体が持つマスである。そのマスに白マスが含まれている場合は制約に反するので白マスの数だけ点数が追加される。経路上のマスを通ることは白マスを置くことを表す。経路上のマスは、想定される白マスと同じ位置の、個体が持つマスである。そのマスが黒マスである場合は制約に反するので 1 点追加する。

まず $H[1]$ を配置する 4 通りについてそれぞれの点数を計算する。図中の矢印は通った経路を表す。次に $H[2]$ を配置する 4 通りについて、経路すると最短となる $H[1]$ の位置とその経路の点数を計算する。次に末尾ノードへ行くまでに経路すると最短となるノードとその経路の点数を計算する。最後に「どの配置を経路すれば最短となるか」を末尾ノードから逆算して最短経路を得る。太い矢印が最終的な最短経路を表す。

4.3 選択方法

選択にはトーナメント選択を用いる。トーナメント選択では個体群からあらかじめ決められた数（サイズ）の個体を等確率で選び出し、その中で最も適応度の高い個体を選択個体とする。先行研究[1]ではルーレット選択が用いられているが、サイズを変更することにより動作を変更できる特徴に注目して今回はトーナメント選択を採用した。ただ

し、ルーレット選択には適応度の高さが選択される確率に直接影響するという特徴があるが、トーナメント選択では個体群中での順位が同じであれば選択される確率は変わらない。今回は評価値が低いほど適応度が高いことを表す評価関数であるので、最も評価値の低い個体を選択する。

4.4 交叉

交叉では二点交叉を用いる。二点交叉では 2 個体を選択し、ランダムに決定された始点と終点の間の連続する遺伝子を交換して新たな 2 個体とする。行方向の制約と矛盾しないように、今回はランダムな数行を交換する。先行研究[1]では一点交叉が用いられているが、交叉のパターンが一点交叉より多く個体群の多様性が保たれると考え、今回は二点交叉を採用した。

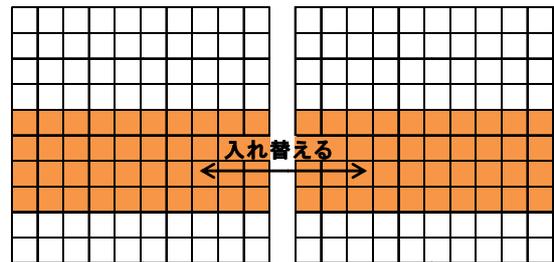


図 9 二点交叉の様子

4.5 候補情報を利用する突然変異

各個体の各行について変異判定を行い、突然変異率 r の確率で、対象の行で突然変異を行う。突然変異は二分探索法に似た動作で、ランダムに始点 s と終点 t を決定し、中間点 $m = (s+t) / 2$ の遺伝子 $g[m]$ を位置候補からランダムに決定して、他の遺伝子を再帰的に決定する。図 10 は遺伝子 $g[1]$ から $g[6]$ までの突然変異を行う様子を表す例であり、8 番目のマスが白マスと確定している。灰色マスは位置候補を表し、黒マスは位置候補のうち選択されたマスを表す。始めに $g[3]$ を決定し、制約に従って $g[1]$ と $g[5]$ の選択可能なマスを減らす。次に $g[1]$ の位置を決定し、 $g[2]$ の選択可能なマスを減らして $g[2]$ の位置を決定する。同様にして $g[5]$, $g[4]$, $g[6]$ の順で位置を決定する。単純に先頭の遺伝子より順に位置候補から等確率で決定した場合には、黒マスが末尾に偏りやすい（選択する位置により以降に決定する遺伝子の組み合わせの数が異なる）ため、偏りの少ないよう中間から決定する仕様にした。

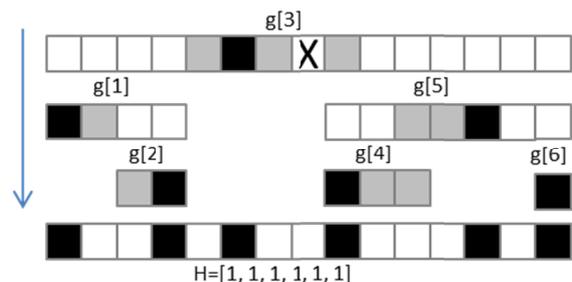


図 10 突然変異の一例

4.6 大規模突然変異

一定世代の間（世代間隔）に最良評価値の更新が無かった場合、個体群が局所最適解に陥っていると想定し、それを抜け出すことを目的として大規模突然変異を行う。大規模突然変異では各個体の各行について変異判定を行い、突然変異率より高い確率で、初期個体生成と同様にして対象の行の遺伝子を再生成する。

5. 評価実験

5.1 実験内容

先行研究[1]のマス確定法を用いる GA (GA1) と本研究で提案する GA (GA2) で、20×20 の問題 1 つにつき 20 回ずつ探索を行い正解数を計る。これを 50 種類の問題について行う。20×20 の問題は、10 種類の 10×10 の問題の中から重複を許してランダムに 4 つ選択し、それらを互いに隣接させて生成する。図 11 は用意した 10×10 の問題の一部であり、左の問題は人工衛星、右の問題はさくらんぼのイラストである。確定アルゴリズムによって全てのマスが確定される問題が生成された場合は問題を再生成する。各種パラメーターは個体数 3000、打ち切り世代 300、トーナメントサイズ 4、突然変異率 2.5%、大規模突然変異の世代間隔 10、変異率 0.5 とする。

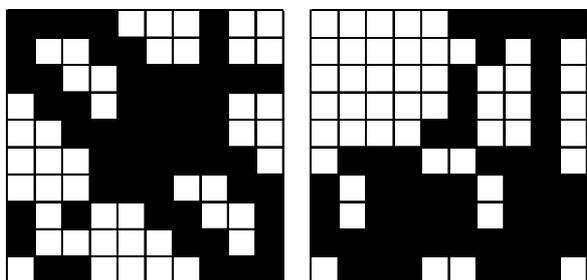


図 11 10×10 の問題例

5.2 実験結果

図 12 に確定アルゴリズムによって確定したマス数と、実行した 2 種類の GA の正解数を示す。

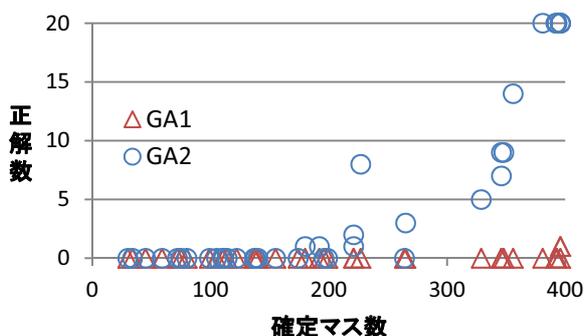


図 12 確定マス数と正解数

GA1 ではほぼ全ての問題で正解数が 0 である。GA2 では確定マス数が 200 以下の問題ではほぼ全ての問題で GA1 と同様に正解数が 0 であるが、正解している場合も見られ、確定マス数が 200 以上の問題では正解する場合が多くなる。また、確定マス数が多いほど正解数も多い傾向がある。

生成された問題の例を示す。図 13 は 47 マスが確定する問題で、正解数は GA1, GA2 共に 0 であった。図 14 は 393 マスが確定する問題で、遺伝子の初期値を変えて 20 回実行した結果正解数は GA1 では 0, GA2 では 20 であった。

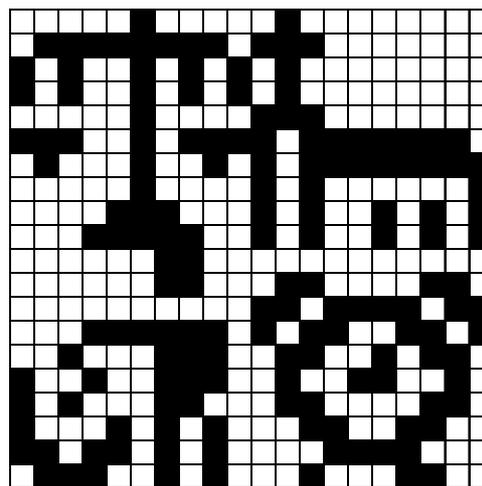


図 13 確定マス数が少ない問題の一例

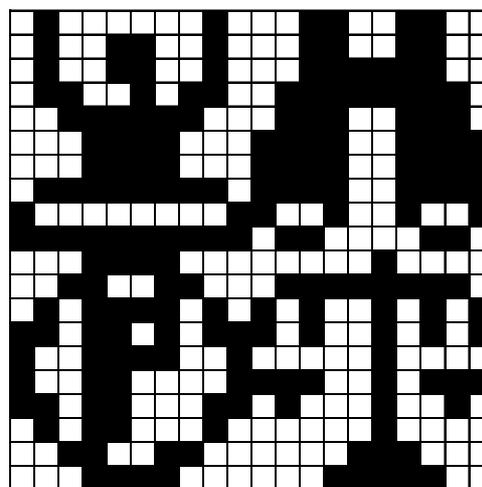


図 14 確定マス数が多い問題の一例

6. 考察

確定マス数が 200 以上の多くの問題で GA2 の正解数が GA1 の正解数を大きく超えていることから、提案したマスの確定が GA の探索精度を向上させたと考えられ、提案手法は有効であったと考えられる。一方、確定マス数が 200 以下の問題での GA2 の正解数は GA1 の正解数と並びほぼ全て 0 であり、探索精度の向上が十分でないと考えられる。この理由の一つに考えられるのは、確定したマスと矛盾しない個体のみを生成すべきであるのに、実際には矛盾する

個体も生成されることである。具体的には、あるマスが黒マスであると確定している場合でも、そのマスが白マスである個体が生成されることがある。確定したマスと矛盾する位置候補を除外するという今回の仕様では、黒マスであると確定しているマスは遺伝子の位置候補には残るが、位置候補からどの位置が選択されるかはランダムであるためである。確定したマスと矛盾しない個体のみを生成するためには、位置を決定する際に位置候補の情報と共に、確定している黒マスの情報も利用しなければならない。また、確定マス数が少ないほど除外される位置候補も少ないため、探索精度の向上率も低くなってしまう。この問題に対処する方法として、確定マス数が少ない問題ほど困難な問題であると捉え、より多くの個体を用いる探索を行うなどが考えられ、今後調査を行う必要がある。

7. むすび

本研究では、大規模ピクロス解法の探索精度の向上と処理時間の短縮のための遺伝的操作の提案と GA が効率的に動作するための決定的アルゴリズムを併用する手法を提案した。評価実験ではランダムに生成した 20×20 の問題を 50 種類用いて単純にマスの確定法を行う GA と提案するマスの確定法を行う GA による確定したマス数ごとの正解数を比較した。結果、提案する決定的アルゴリズムが有効に作用して探索空間の減少に成功した場合、GA の探索精度を大きく向上できる可能性を示した。

参考文献

- [1] 助川隆俊, 佐藤裕二 "大規模ピクロスへの部分解を利用した GA の適用方法の検討", 情報処理学会第 77 回全国大会講演論文集, pp 261-263, 2015 年
- [2] 鈴木礼智, 渡邊 俊彦 "異なる探索法と移住個体の選択を用いた DGA のイラストロジック問題への適用", BMFSA2005 論文集, pp117-120, 2005 年
- [3] Jinn-Tsong Tsai, "Solving Japanese nonograms by Taguchi-based genetic algorithm", Applied Intelligence Volume 37, Issue 3, pp 405-419, 2012 年