

制約違反最少化戦略に基づくハイブリッド GA による 制約充足問題の解法

松本美幸[†] 狩野均[†] 西原清一[†]

近年、山登り法やニューラルネットワークによって大規模な制約充足問題を実用的な時間内に解くための研究がさかに行われている。しかし、これらの方法には大域的探索戦略が含まれていないため探索途中で局所解に陥ってしまう恐れがある。このため、局所最適解からの脱出方法も検討されているが、積極的に大域的探索を行う方法はあまり見当たらない。そこで本論文では、大域的探索戦略を有する遺伝的アルゴリズム (GA) と制約違反最少化戦略による山登り法 (MCHC) をハイブリッド化した探索法を提案する。本手法はまず、GA で大域的に探索し、次に集団中で制約違反数が最少の個体 (エリート個体) を初期値として MCHC により局所的に探索する。その際、GA では多様性の維持を優先させるためにエリート個体とのハミング距離が近い個体は、その適応度を減少させる。また、局所最適解に陥ったエリート個体は死滅させ、GA による別の初期値から再スタートする。本手法を Haralick の定義による部分解列挙型制約充足問題に対して適用し、GA、ランダムな初期値から MCHC を繰り返す方法よりも探索成功率が優れていることを実験で示す。

Solving Constraint Satisfaction Problems by Hybrid Genetic Algorithms Based on Min-conflicts Heuristic

MIYUKI MATSUMOTO,[†] HITOSHI KANO[†] and SEIICHI NISHIHARA[†]

Several approximation algorithms using hill-climbing techniques and neural networks have recently been proposed to solve large CSPs in a practical time. In these proposals, many methods of escaping from local optima are discussed; however, there are very few methods that actively perform global search. In this paper, we propose a hybrid search method that combines the genetic algorithm with the min-conflicts hill-climbing (MCHC). Global search by using GAs is performed, and the elite, or the individual with the fewest conflicts in the population, is used as the initial value of MCHC to search locally. For the purpose of giving the first priority in the GA process to maintaining diversity of the population, fitnesses of each individual is decreased when the Hamming distances between the elite and the individual is small. The elite falling into local optima is removed from the population, and the MCHC restarts from another initial value that is created by GA. In our method, the individual having the fewest conflicts in the population is used as the initial value of MCHC to make a local search. Some detailed experiments are also performed using constraint satisfaction problems defined by Haralick to prove that the proposed method generally gives better efficiency than both of GA and the naive randomly restarting MCHC.

1. はじめに

制約充足問題 (CSP: Constraint Satisfaction Problem) は、離散値をとるいくつかの変数の値の組合せのうち、与えられた制約すべてを満たすような組合せを発見する探索問題である^{1),2)}。CSP が解を持つかどうかという決定問題は NP 完全であり、効率的な一般解法は存在しない。このため、近似解法によって、なるべく高い確率で実用的な時間内に解を求めようと

する研究がさかに行われている。これは、バックトラックによる厳密解法がアルゴリズムの完全性 (停止性) を保証しているのに対して、これを犠牲にして高速化や高度の並列化を実現しようというものである。

近年、制約違反数を最少化する方向に山登りする方法 (MCHC³⁾) や相互結合型のニューラルネットに補助ネットを付加した方法 (GDS⁴⁾) が比較的良好な結果が得られたと報告されている。しかし、これらの方法は探索途中で局所最適解に陥ってしまう危険性があり、適用範囲が限られていた⁵⁾。このため、GSAT^{6),7)}、breakout アルゴリズム⁸⁾、GENET⁹⁾、EFLOP¹⁰⁾等の局所最適解からの脱出方法が提案されているが、多

[†] 筑波大学電子・情報工学系
Institute of Information Sciences and Electronics, University of Tsukuba

点探索を行う方法はあまり見当たらない。

一方、遺伝的アルゴリズム (GA: Genetic Algorithms) は生物の進化過程をモデル化した確率的探索アルゴリズムであり、大域的探索能力が高いため、多くの分野において最適化問題の解法として利用されている^{11),12)}。しかし、GA を CSP に適用した例はあまり見当たらない。この理由は、最適化問題が、準最適解を求めればよいのに対し、CSP ではすべての制約を満たすような完全な解を求める必要があるため、局所的探索能力が低い GA はそのままでは適用できないためと考えられる。

そこで本論文では、GA と MCHC をハイブリッド化した探索方法を提案する^{13),14)}。GA のハイブリッド化に関しては多数検討されている。したがって、本論文の主題は、CSP の解法として有効なハイブリッド化の方法の提案と、その有効性の検証である。本手法はまず、GA で大域的に探索し、次に集団中で制約違反数が最少の個体 (エリート個体) を初期値として MCHC により局所的に探索する。その際、GA では多様性の維持を最優先させるためにエリート個体とのハミング距離が近い個体は、その適応度を減少させる。また、局所最適解に陥ったエリート個体は死滅させ、GA による別の初期値から再スタートするものである。

以下では、まず CSP を変数の数と制約の数の関係から 4 種類に分類する。次に、GA による CSP のコード化の方法と提案するハイブリッド化の方法について述べる。最後に、Haralick の定義による部分解列挙型 CSP を用いたシミュレーションにより、GA、ランダムな初期値から MCHC を繰り返す方法ならびに本手法の探索速度と探索成功率を比較検討する。

2. CSP の分類と本手法の基本戦略

2.1 CSP とその分類方法

CSP は、論理式の充足可能性問題やグラフ色塗り問題を例に理論的に検討されてきた。ここでは、より一般的な Haralick の定義を用いる¹⁾。この CSP は、4 つ組 (U, L, T, R) で定義される。CSP の例を図 1 に示す。 $U = \{v_1, \dots, v_n\}$ は離散的な値をとる変数の集合、 L は、各変数がとる値の集合を表す。 $T = \{t_1, \dots, t_m\}$ は制約関係 t_j の集合で、 $t_j = (v_p, v_q)$ は、 p 番目の変数と q 番目の変数の間に制約が存在することを示す。 $R = \{R_1, \dots, R_m\}$ において、 R_j は制約関係 t_j にある変数 (v_p, v_q) の部分解を列挙したもので、 $R_j = \{(r_1, r_2), (r_3, r_4), \dots\}$ は $(v_p, v_q) = (r_1, r_2)$ または $(r_3, r_4) \dots$ であることを示す。ここでは、制約はすべて 2 項制約とする。CSP は NP 完全な探索問題で

$$\begin{aligned}
 U &= \{v_1, v_2, v_3, v_4\}, \\
 L &= \{a, b, c, d, e\}, \\
 T &= \{t_1, t_2, t_3, t_4\}, \\
 &\quad t_1 = (v_1, v_2), t_2 = (v_1, v_3), \\
 &\quad t_3 = (v_3, v_4), t_4 = (v_1, v_4), \\
 R &= \{R_1, R_2, R_3, R_4\}, \\
 &\quad R_1 = \{(a, b), (e, c)\}, \\
 &\quad R_2 = \{(a, c), (a, e)\}, \\
 &\quad R_3 = \{(c, e), (a, c)\}, \\
 &\quad R_4 = \{(a, d), (a, e), (b, c)\}.
 \end{aligned}$$

解 : $(v_1, v_2, v_3, v_4) = (a, b, c, e)$

図 1 CSP の例
Fig. 1 An example of CSP.

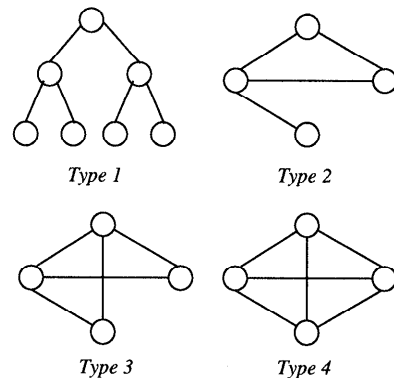


図 2 CSP の分類
Fig. 2 Classification of CSP.

あり、その意味で効率の良い汎用の探索アルゴリズムは存在しない。そこで本論文では、変数の個数 n と制約の個数 m の関係に着目して CSP を次の 4 種類に分類し、種類別に有効な解法を考察するという立場をとる。ここでは、変数の数に対する制約の数 $(d = m/n)$ を制約密度と定義する。

$$\begin{aligned}
 \text{Type1: } & m = n - 1 \\
 \text{Type2: } & n - 1 < m \ll n(n - 1)/2 \\
 \text{Type3: } & n - 1 \ll m < n(n - 1)/2 \\
 \text{Type4: } & m = n(n - 1)/2
 \end{aligned}$$

図 2 は、これらの代表例を制約グラフで表したものである。グラフの頂点は変数、辺は制約関係に対応している。なお、本論文では、グラフが連結している場合のみを考える。Type1 は、制約グラフが木構造をしている場合で、このとき、制約密度は最小となる。これに対して、Type4 は、完全グラフの場合で d は

最大となる。Type2, 3は、これらの中間の場合である。一般に、制約密度の小さい問題は、局所最適解に陥りやすく難問とされているが、これはType2に相当する。探索アルゴリズムのベンチマークとしてよく用いられる n クイーン問題は、Type4に属する。また、CSPの近似解法としてしばしば引用される文献3)のグラフ色塗り問題で、粗結合グラフはType2、密結合グラフはType3に相当する。この分類に基づいて、CSPの解法を比較検討することにより、各手法の適用範囲を明確にすることができるとと思われる。

2.2 CSPの近似解法

制約違反を最少化する方向に山登りする方法(以下、MCHC: Min Conflict Hill-Climbingと称す)³⁾を用いた大規模CSPの解法が多数提案されている^{6),8),10)}。これは値の選択規則として、次のヒューリスティクスを用いている。

制約違反最少化戦略

制約違反している変数に対して制約違反数が最少になる値を代入する(違反数が同じときはランダム)。

MCHCは、GDS⁴⁾とほぼ同等の性能が得られたと報告されている³⁾が、GDSと同様に、制約密度が低い問題に対しては、探索途中で局所最適解に陥ってしまう危険性が高いとされている。この欠点を補うため、制約の重みをダイナミックに変化させて局所最適解から抜け出す方法(breakoutアルゴリズム)⁸⁾、他の初期値から探索をやり直す方法(GSAT)^{6),7)}、また、GDSの改良であるGENET⁹⁾などが提案されている。

本論文では、GSATと同じ方法をGAと比較検討する。GSATはランダムに生成された充足可能性問題において、厳密解法(Davis-Putnam procedure)よりも優れていることが示されている⁶⁾。この方法は、1つの初期値についてMCHCにより一定回数探索を進めても制約違反数が減少しない場合は、局所最適解に陥ったと見なし探索を中止し、別の初期値から探索をやり直すものである(以下、IHC: Iterated Hill-Climbingと称す)。これは、状態空間におけるコスト関数の形状に関する次の仮説に基づいている。

再スタート仮説

制約密度の小さいCSPでは、最適解は局所最適解と離れたところにある。

MCHCとIHCのアルゴリズムを図3に示す。本論文では、MCHCの変数の選択規則として、制約違反が最も多い変数を選択するものとする⁵⁾。

2.3 遺伝的アルゴリズム

GAは、探索空間上の状態を個体と見なし、その個体の集団に対して、選択、交叉、突然変異という遺伝

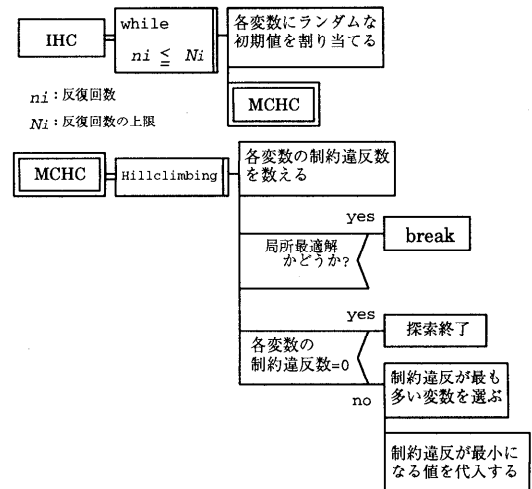


図3 IHCのアルゴリズム

Fig. 3 IHC algorithms.

的操作を行うことにより、制約違反のない状態を発見しようとするものである。すなわち、制約違反の少ない個体は、有効な部分解を多く含んでいる。これらの個体は適応度の高い個体として評価され、次世代の子個体をつくる親個体として優先的に選択される。そして交叉では親個体どうしによる遺伝子の組み替えによって子個体を生成し、突然変異ではそれら子個体の遺伝子の一部をある確率でランダムに変化させる。

ところで、GAの探索において注意しなければならない点に集団の多様性の維持がある。GAは集団という限られた部分空間内で互いの遺伝子を組み換えながら適応度を高める方向に探索を進めていくので、ある時間が過ぎると集団が一様になってしまう恐れがある。GAの最大の特徴である大域的探索機能を生かすためにはこの部分空間内に良質かつ構造が異なった候補を保持することが重要である。

3. 提案する手法

3.1 基本戦略

本論文ではGAとMCHCをハイブリッド化する方法を提案する。以下では本手法をGAHCと略記する。その基本戦略は、次の3項目である。

- I. まず、GAで大域的に探索し、次に集団中で制約違反数が最少の個体(エリート個体)を初期値としてMCHCにより局所的に探索する(エリート個体が複数のときはランダムに選択する)。
- II. GAでは、集団における多様性の維持を最優先させることにより、大域的探索能力を強調する。このため、エリート個体とのハミング距離が近い個体

表 1 CSP と GA の対応
Table 1 Correspondence between CSP and GA.

CSP	GA	例
変数	遺伝子座 i	1, 2, 3, 4
値	遺伝子 $G_k(i)$	a, b, c
解候補	染色体 G_k	(a, b, a, c)

は、その適応度を減少させる。

III. MCHC で局所最適解に陥ったエリート個体は死滅させ、GA による別の初期値から再スタートする。すなわち、再スタート仮説を適用する。

3.2 CSP のコード化と適応度の計算

本モデルでは、CSP を GA 上で表 1 のように表現する。GA の k 番目の個体の染色体を G_k 、 G_k の i 番目の遺伝子座にある遺伝子を $G_k(i)$ とかく。また、表 1 の第 3 列は、図 1 の CSP の例である。このようなコード化に対して、個体 k の適応度 F_k を以下のように計算する。

$$F_k = 1 - \sum_{j=1}^m \text{conf}_k(j)/m. \quad (1)$$

ただし、 $t_j = (v_p, v_q), R_j$ に対して、

$$\text{conf}_k(j) = \begin{cases} 1 & ((G_k(p), G_k(q)) \notin R_j). \\ 0 & (\text{otherwise}). \end{cases}$$

m : 制約数.

3.3 選択確率の計算

2.3 節で述べたとおり GA においては多様性の維持を最優先させるため、エリート個体との類似性が高い個体は選択確率を通常より減少させる。エリート個体の染色体を G_e とすると、 G_e と G_k のハミング距離 H_k を次のように定義する。

$$H_k = n - \sum_{i=1}^n \delta(G_k(i), G_e(i)). \quad (2)$$

ただし、

$$\delta(x, y) = \begin{cases} 1 & (x = y). \\ 0 & (\text{otherwise}). \end{cases}$$

n : 遺伝子座の数=変数の数.

このような H_k を用いて、選択確率 PS_k は次の式 (3) で計算する。

$$PS_k = F_k \times H_k / n. \quad (3)$$

3.4 本手法のアルゴリズム

本手法のアルゴリズムを図 4 に示す。図 4 で、 N_p は集団サイズ (個体数)、 N_g は世代数の上限とする。まず、 N_p 個の個体 (染色体 $G_k, k = 1, \dots, N_p$) を

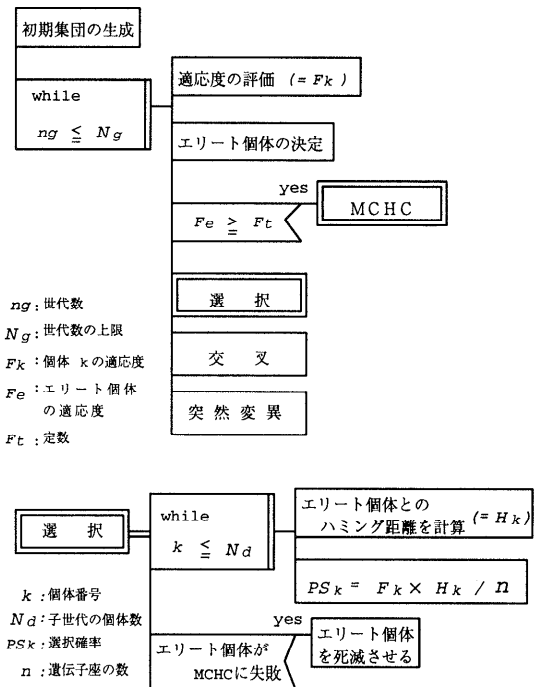


図 4 本手法のアルゴリズム

Fig. 4 GAHC algorithms.

ランダムに発生させる。次に以下の (i), (ii) を解が発見されるまで (または N_g 回) 繰り返す。

- (i) 局所的探索：全個体の適応度 F_k を式 (1) で計算し、 F_k が最大の個体をエリート個体とする。エリート個体の適応度 F_e がしきい値 F_t を超えた場合、エリート個体の染色体を初期値として MCHC で局所的探索を実施する。
- (ii) 遺伝的操作：まず、式 (3) の選択確率によるルーレット戦略とエリート保存戦略を併用し、集団から N_d 個の個体を選択する。この個体から、一様交叉により N_p 個の個体を次世代として生成する。次に、ランダムに選んだ個体の遺伝子座に対して、ランダムな遺伝子をあらかじめ設定された確率 (突然変異率) で割り当てる。

4. 性能評価

4.1 実験方法

IHC, GA, GAHC の性能を比較検討するため、部分解の集合 R_j のサイズの平均値 $|\bar{R}_j| = 1, 3, 6, 9$ の CSP を各 100 個ずつ、Type1~4 ごとにランダムに発生させ (計 1600 個)、解探索を試みた。なお、本論文では制約の強さ ($=s$) を $s = 1 - |\bar{R}_j|/|L|^2$ として評価する。以下では、値の数 (対立遺伝子数) は 4、変数の数は特に断らない限り 50 とした。したがって、

探索空間の大きさは約 10 の 30 乗である。表 2 は、制約数と制約密度をタイプ別に示したものである。また手法の評価は、次の 2 項目について行った。

((評価項目))

(a) 探索成功率 (%) :

探索した CSP のうち解を発見できたものの割合。

(b) 平均探索時間 (min) :

解を発見できた CSP についての平均探索時間。なお、計算機は HP715/33 を使用し、プログラムはすべて C 言語で記述した。

4.2 パラメータの設定

IHC と GA では探索方法がまったく異なるので、性能を比較するときには注意が必要である。特に、解が得られない場合に探索をどこで打ち切るかによって、成功率と探索時間の統計値が変化することがある。本論文では、IHC における反復回数の上限 N_i と GA における世代数の上限 N_g を、探索に要する計算コストの理論的な最大値 (最悪ケース) が各手法でほぼ等しくなるように設定した。これらのコストはそれぞれ以下のように求めることができる。

$$\text{cost}(IHC) = C_{HC} \times N_h \times N_i. \quad (4)$$

$$\text{cost}(GA) = C_{GA} \times N_p \times N_g. \quad (5)$$

$$\text{cost}(GAHC) =$$

$$C_{GA} \times N_p \times N_g + C_{HC} \times N_h \times N_g. \quad (6)$$

ただし、

$$C_{HC} = C1 \times m \times n, \quad (7)$$

$$C_{GA} = C2 \times m + C3 \times n. \quad (8)$$

$C1, C2, C3$: 定数

C_{HC} は、山登りの 1 ステップに要するコストで、変数ごとの制約違反数を計算するコストにほぼ比例する。 C_{GA} は、GA の 1 世代 1 個体あたりの計算コストである。これは次の 2 つの項の和となる。第 1 項は、個体の制約違反数を計算するコストで式 (1) のように m に比例する。第 2 項は、選択確率の計算コストと遺伝的操作の計算コストで n に比例する。

表 3 に C_{HC}/C_{GA} の実測値を示す。また、 N_h は実際に山登りの行われた回数の平均値である。以下の実験では、表 3 の $Type1$ の値を用いて、計算コストの理論的な最大値を推定し、次の式 (9) の条件を満たす範囲で各パラメータを設定するようにした。

$$\text{cost}(GAHC) \leq \text{cost}(IHC), \text{cost}(GA). \quad (9)$$

式 (4)~(8) は解が得られない場合に探索をどこで打ち切るかを判断するとき重要となる。なお、GA および GAHC では全実験を通して、生存率 $N_d/N_p = 0.5$ 、1 遺伝子座あたりの突然変異率=1%とした。

表 2 制約数と制約密度

Table 2 Number of constraints and their density for $Type1$ to CSPs.

Type	1	2	3	4
m	$n-1$	$9(n-1)$	$17(n-1)$	$n(n-1)/2$
d	0.98	8.82	16.7	24.5

表 3 C_{HC} と C_{GA} の実測値

Table 3 Experimental values of C_{HC}/C_{GA} .

Type	1	2	3	4
C_{HC}/C_{GA}	2.98	9.37	14.5	15.8

表 4 GA の実験結果

Table 4 Experimental results for GA.

N_p	500	1000	2000	3000
Succ	72	73	77	82
Time	4.0	8.2	7.5	12

Succ: 探索成功率 (%)

Time: 平均探索時間 (min)

表 5 GAE の実験結果

Table 5 Experimental results for GAE.

N_p	500	1000	2000	3000
Succ	72	74	74	74
Time	3.2	6.0	10	18

4.3 通常の GA による CSP の解法

GA では、式 (1) の適応度 F_k の 2 乗を選択確率としたルーレット戦略、交叉方法は一様交叉を用いた。また、世代数の上限は $N_g=200$ とした。比較のためにエリート個体を必ず次の世代に残す選択戦略 (エリート保存戦略) を併用した GA でも実験を行った (以下、GAE と称す)。表 4、表 5 にこの結果を示す。探索成功率は $Type1 \sim 4$ の CSP の合計に対する値を示す。表 4 では、集団サイズが大きくなるとともに成功率が増加している。一方、表 5 では、成功率は 72~74% の範囲と、ほぼ一定であるにもかかわらず、探索時間は集団サイズの変化量に比例して増加している。一般的に最適化問題では、エリート保存戦略を併用することにより、より最適解に近い準最適解が得られるといわれている。しかし、本実験のような CSP では解が発見できた否かを問題にしているためこの効果が現れていないと思われる。表 4、表 5 を通じて、エリート保存戦略を併用しない GA で $N_p = 2000$ のときが比較的良好な結果が得られた。

4.4 GAHC による CSP の解法

表 6 は表 4、表 5 と同様の実験を GAHC で行った結果である。世代数の上限は $N_g=100$ 、MCHC を適用するエリート個体のしきい値は $F_i = 0.5$ とした。

表 6 では、 $N_p=1000$ のとき最も良い結果が得られた。

図 5 は、 $N_p = 1000$, $N_g = 100$, $F_t = 0.4, 0.5, 0.6$ としたときの実験結果である。横軸は制約密度 $d = n/m$ で、表 2 に示した *Type1*~4 の値である。 F_t は、大域的探索から局所的探索へ移行するときの適応度のしきい値で、GAHC においては最も重要なパラメータである。GAHC の動作は、 $F_t = 0$ では IHC と一致し、 $F_t = 1$ では GA と一致する。図 5(a) では、 $F_t = 0.6$ のときは他のケースより成功率、探索時間ともに劣っている。また、 $F_t = 0.4, 0.5$ のときを比べると $d \geq 16.7$ で成功率、探索時間ともに後者の方が優れている。したがって、 $F_t = 0.5$ のときが最も良い結果が得られたといえる。

4.5 GAHC と他の手法の比較

実験結果を図 6 に示す。IHC では、 $N_i = 500$,

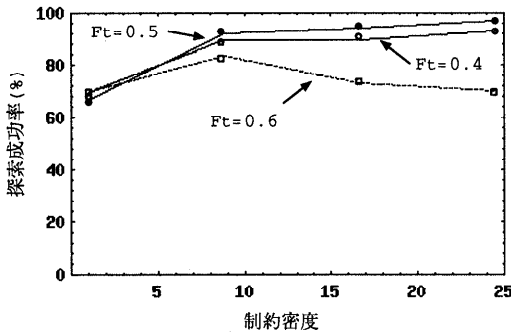
$N_h = 500$ とした。GA と GAHC のパラメータは 4.3, 4.4 節の実験結果から設定した。本実験中に測定した N_h は、平均で IHC では約 180, GAHC で約 35 であった。これらのパラメータは、式 (7) の条件を満たしている。

図 6 から以下のことが分かる。IHC と GAHC を比べると $d = 8.82 \sim 24.5$ では成功率 (図 6(a)) がほぼ一致しているが、 $d = 0.98$ では前者が約 20% であるのに対し、後者では約 65% と優れている。近似解法では、 d の小さな CSP は探索の手がかりとなる制約が少ないため、局所最適解に陥りやすく難問とされている⁵⁾。本手法は大域的探索戦略を兼ね備えているため、IHC に比べ良い結果が得られたと考えられる。

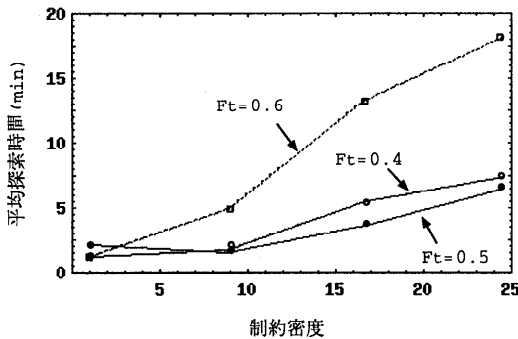
また探索時間 (図 6(b)) は、 $d \leq 8.82$ では、IHC, GAHC とともにほぼ一致しているが、 $d = 16.7 \sim 24.5$ では、GAHC の方が IHC の約 4 倍の速さで解の探索に成功している。また、 $d = 8.82 \sim 24.5$ のときに前者の平均探索時間が d に比例して急激に増加しているのに対し、後者ではゆっくり増加している。これは、4.2 節で述べたように IHC, GAHC の探索に要する計算コストの違いによるものである。すなわち、式

表 6 GAHC の実験結果
Table 6 Experimental results for GAHC.

N_p	500	1000	2000	3000
Succ	76	87	88	83
Time	2.9	3.1	3.5	9.0



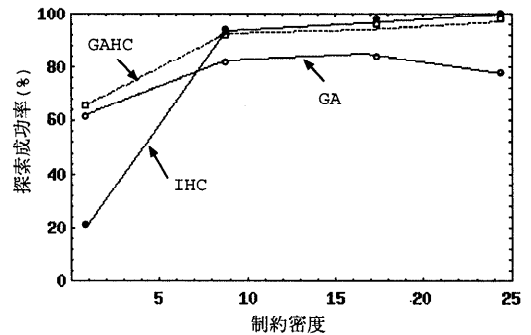
(a) 探索したCSPのうち解を発見できたものの割合



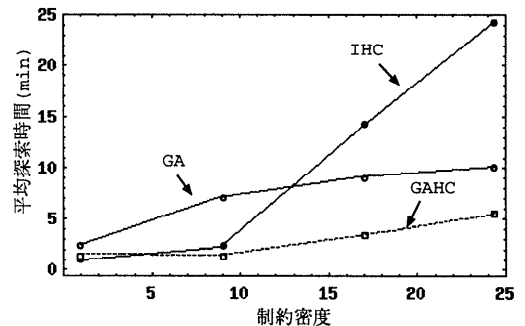
(b) 解を発見できたCSPについての平均探索時間

図 5 F_t に関する実験結果

Fig. 5 Experimental results for GAHC.



(a) 探索したCSPのうち解を発見できたものの割合



(b) 解を発見できたCSPについての平均探索時間

図 6 IHC, GA, GAHC に関する実験結果

Fig. 6 Experimental results for IHC, GA and GAHC.

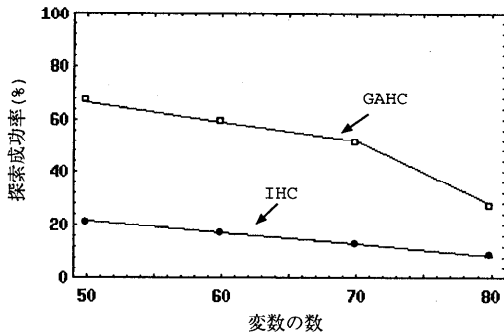


図7 探索空間のサイズに関する実行結果

Fig. 7 Experimental results with respect to search space size.

表7 制約の強さと探索成功率
Table 7 Strength of constraint and percentage of success.

$ \bar{R}_j $	1	3	6	9
s	0.94	0.81	0.63	0.44
IHC	99	100	98	76
GAHC	99	99	99	72

$|\bar{R}_j|$: 部分解の集合 R_j のサイズの平均値

(4), (6)~(8) から $cost(GAHC)/cost(IHC)$ を計算すると、これは制約の数に反比例して減少することが分かる。一方、GA と GAHC を比べると成功率、探索時間とも全領域で本手法が優れている。

4.6 探索空間のサイズに関する比較

制約密度の低い *Type1* の CSP について本手法と IHC で比較実験を行った。図7は、変数の数 $n = 50, 60, 70, 80$ としたときのこの結果を示したものである。また、IHC と GAHC のパラメータは4.5節の実験と同じ値を用いた。

図7から以下のことが分かる。変数の数の増加にともなって成功率が本手法、IHC ともに減少しているものの、全領域において本手法の方が IHC よりも優れている。CSP では変数の数の増加にともなって探索空間のサイズが指数関数的に増加する。これに対して、本手法の探索回数を決めるパラメータ N_p, N_g は、一定であるにもかかわらず、成功率の低下は緩やかなものとなっていることが分かる。

4.7 制約の強さに関する比較

表7に *Type2* の CSP の制約の強さ s を変化させたときの IHC, GAHC の探索成功率を示す。表7から GAHC と IHC は、ほぼ同じ傾向を示していることが分かる。

表8 breakout アルゴリズムとの比較

Table 8 Comparison between breakout algorithm and GA+breakout.

$ \bar{R}_j $	1	3	6	9
s	0.94	0.81	0.63	0.44
breakout	100	71	68	53
GA+breakout	100	89	73	55

5. 本手法の拡張性

本章では、本手法の拡張として GA と breakout アルゴリズム⁸⁾をハイブリッド化した手法 (GA+breakout) と breakout アルゴリズムを単独で用いた方法との性能を比較検討する。breakout アルゴリズムは、制約密度の小さいグラフ色塗り問題に対して、制約違反最少化バックトラッキングよりも効率的であることが示されている⁸⁾。GA+breakout のアルゴリズムは、図4における MCHC を breakout アルゴリズムに置き換えたものである。すなわち、エリート個体を初期値とした MCHC で局所最適解に陥った場合、違反している制約の重みを増加させるものである。また、無限ループを回避するため、MCHC のステップ数の総計が制限回数を超えた場合には、GA による別の初期値から再スタートする。

ここでは、文献15)で難問とされる *Type2* ($d = 4.3$ の場合) の CSP を $|\bar{R}_j| = 1, 3, 6, 9$ ごとに各100個ずつランダムに発生させ(計400個)、解探索を試みた。このときの値の数は $|L| = 4$ 、変数の数は $n = 150$ とした。また、集団のサイズは $N_p = 2000$ 、エリート個体のしきい値は $F_t = 0.2$ とした。

表8に制限時間(10分)以内に解を発見できた CSP の割合(探索成功率)を示す。表8から $|\bar{R}_j|$ が増加するにつれて(すなわち、制約の強さが小さくなるほど)探索成功率が減少しているものの、本手法が優れていることが分かる。

6. おわりに

本論文では、GA と MCHC をハイブリッド化した CSP の解法を提案した。また、通常の GA や反復山登り法 (IHC) を単独で用いた場合よりも探索速度、探索成功率ともに優れていることを実験で確認した。本論文では、Haralick の部分解列挙型の CSP を用いて実験を行ったが、この CSP 特有のヒューリスティクスは用いずに解候補の制約違反のみで議論した。したがって、本論文で得られた結論は、他の CSP にもあてはまると考えられる。

本論文では、CSP を4つのタイプに分類して考察

した。GA をベースとした方法は、これらのタイプにあまり依存しないロバストな方法であると考えられるので、今後は、より詳細な分類に基づいて GA の適用範囲を明確にすることが重要であると考えられる。また、解法の性能評価については、本論文では、コストを一定として得られた解の質を規範としていたが、一般にコストと解の質は、トレードオフの関係にあると考えられている。したがって、コストを変えたときに得られる解の質がどのように変化するかという視点からの評価が残された課題であると考えられる。

謝辞 本研究を進めるにあたり有益なご意見をいただいた、本学構造工学系星野力教授、本学電子・情報工学系李江洪講師、同じく本学工学研究科内野寛治氏（現（株）富士通研究所）に深く感謝いたします。また、プログラムの開発にご協力いただいた本学理工学研究科水野一徳氏に深く感謝いたします。

参 考 文 献

- 1) Haralick, R.M. and Shapiro, L.G.: The Consistent Labeling Problem, Part I, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.PAMI-1, No.2, pp.173-184 (1979).
- 2) 西原清一：整合ラベリング問題と応用，情報処理，Vol.31, No.4, pp.500-507 (1990).
- 3) Minton, S., Johnston, M.D., Philips, A.B. and Laird, P.: Solving Large-scale Constraint Satisfaction and Scheduling Problems Using a Heuristic Repair Method, *AAAI*, pp.17-24 (1990).
- 4) Adorf, H.M. and Johnston, M.D.: A Discrete Stochastic Neural Network Algorithm for Constraint Satisfaction Problems, *IJCNN III*, pp.917-924 (1990).
- 5) Minton, S. et al.: Minimizing Conflicts, A Heuristic Repair Method for Constraint Satisfaction and Scheduling Problems, *Artif. Intel.*, 58, pp.161-205 (1992).
- 6) Selman, B., Levesque, H., Mitchell, D.: A New Method for Solving Hard Satisfiability Problems, *AAAI*, pp.440-446 (1992).
- 7) Selman, B. and Kautz, H.: Domain-independent Extension to GSAT: Solving Large Structured Satisfiability Problems, *IJCAI*, pp.290-295 (1993).
- 8) Morris, P.: The Breakout Method for Escaping from Local Minima, *AAAI*, pp.40-45 (1993).
- 9) Davenport, A., Tsang, E., Wang, C.J. and Zhu, K.: GENET, A Connectionist Architecture for Solving Constraint Satisfaction Problems by Iterative Improvement, *AAAI*, pp.325-

330 (1994).

- 10) Yugami, H., Ohta, Y. and Hara, H.: Improving Repair-based Constraint Satisfaction Methods, *AAAI*, pp.344-349 (1994).
- 11) Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley (1989).
- 12) 北野宏明：遺伝的アルゴリズム，産業図書 (1993).
- 13) 松本，内野，狩野，西原：遺伝的アルゴリズムによる制約充足問題の解法，情報処理学会人工知能研究会報告，Vol.95, No.86, pp.33-40 (1995).
- 14) Kanoh, H., Matsumoto, M. and Nishihara, S.: Genetic Algorithms for Constraint Satisfaction Problems: *Proc. IEEE, International Conference on System, Man and Cybernetics*, pp.626-631 (1995).
- 15) Mitchell, D., Selman, B. and Levesque, H.: Hard and Easy Distributions of SAT Problem: *Proc. National Conference on Artif. Intel.*, pp.459-465 (1992).

(平成 8 年 3 月 5 日受付)

(平成 9 年 3 月 7 日採録)



松本 美幸 (学生会員)

1972 年生。1995 年筑波大学第三学群情報学類卒業。1997 年同大学大学院理工学研究科修士課程修了。現在、ソニー株式会社勤務。在学中は、遺伝的アルゴリズム、カーナビゲーションのための経路探索に関する研究に従事。



狩野 均 (正会員)

1954 年生。1978 年筑波大学第一学群自然科学類卒業。1980 年同大学大学院理工学研究科修士課程修了。同年、日立電線(株)入社。同社オプトロシステム研究所において人工知能・神経回路の応用に関する研究に従事。1993 年より筑波大学電子情報工学系。現在、同助教授。制約に基づく問題解決、遺伝的アルゴリズムの研究に従事。工学博士。1992 年電気学会論文賞受賞。電気学会、人工知能学会、計測自動制御学会等各会員。



西原 清一（正会員）

1968年京都大学工学部数理工学科卒業。同年、同大学大型計算機センター助手。1975年より筑波大学電子・情報工学系。現在、同教授。工学博士。1982～1983年文部省在外研究員として米国ヴァージニア工科大学。図形画像処理、グラフィックスとCAD、組合せ探索アルゴリズム、知識処理、制約充足問題、GAの研究に従事。著書に「データ構造」（オーム社）など。電子情報通信学会、人工知能学会、ACM、IEEE各会員。
