

遺伝的アルゴリズムとそのVLSIレイアウト設計手法への 応用に関する研究

若林 真一

広島大学工学部
第二類（電気系）助教授

1 まえがき

VLSI チップのレイアウト設計においては一時に取り扱う問題のサイズを小さくして計算時間を短縮することを目的として与えられた回路を複数の部分回路に分割することが一般的である [7]. このような回路分割は与えられた論理回路のモジュールを節点でモジュール間の接続ネットをハイパー枝で置き換えることによりハイパーグラフ分割問題として定式化できる. ここでハイパーグラフを分割したときに分割されたハイパーグラフ間にまたがるハイパー枝の総数をカット数という.

ハイパーグラフ分割問題は, 分割されたハイパーグラフの節点サイズの総和がほぼ均等という制約条件を満たすカット数最小のハイパーグラフの分割を求める問題である. この問題は一般に NP 困難なことが知られており [2], 今までに多くのヒューリスティックアルゴリズムが提案されている. 代表的な分割手法として KL 法 [6], FM 法 [1] などがある. しかし, どちらの手法も反復改良法であり, 解の改良を行う過程で目的関数の減少する節点の移動のみを行なうので局所最適解への収束の回避が困難である. また, KL 法及び FM 法は最終的な解が初期解に大きく依存するという性質を持っている.

一方, 局所最適解に陥りにくい手法の 1 つとして遺伝的アルゴリズム [3] がある. 遺伝的アルゴリズムは, 生物の進化の概念に基づいた組み合わせ最適化手法であり, 一般に複数の許容解を保持し, それらに対して交配, 淘汰, 突然変異, 逆位などの遺伝的操作を適用することにより解を改良する. 著者らは既に, 遺伝的アルゴリズムを用いたハイパーグラフ分割手法を提案している [4]. また, この手法を分散実行することにより解集合の構成を多様化することで局所最適解に陥ることを回避する手法を提案している [5, 8]. 本研究ではそれらの手法に対して突然変異などを新たに遺

伝的操作として採り入れた遺伝的ハイパーグラフ分割手法を提案し, C 言語を用いて実現し実験により本手法の有効性を確認した [9].

2 ハイパーグラフ分割問題

【定義 1】 ハイパーグラフ $H = (V, E)$ は節点集合 $V = \{v_1, v_2, \dots, v_m\}$, ハイパー枝集合 $E = \{e_1, e_2, \dots, e_n\}$ よりなる. ここで各ハイパー枝 $e \in E$ は V の空でない部分集合である. もし, 全ての $e \in E$ に対し, $|e| = 2$ なら H はグラフでもある. 節点集合 V にその重みを表す関数が定義されていたならハイパーグラフ $H = (V, E)$ は重み付きハイパーグラフであると言う. 各節点 v の重みを v のサイズと呼び, $size(v)$ で表す. 節点の重みは全て正の値と仮定する. \square

【定義 2】 重み付きハイパーグラフ $H = (V, E)$ に対し, 関数 $P : V \rightarrow \{1, 2\}$ が与えられたとする. このとき $V_i = \{v \mid v \in V, P(v) = i\}$ ($i = 1, 2$) で表される 2 個の互いに素な節点部分集合を H の 2 分割と定義する. そして, 各ハイパー枝 $e \in E$ に対し, スパン関数 $q(e) = \{P(v) \mid v \in e\}$ を定義し, $\{e \mid |q(e)| = 2, e \in E\}$ なるハイパー枝の集合を 2 分割カットセットという. 2 分割のカット数を 2 分割カットセットに属しているハイパー枝の数で表す. ハイパーグラフ H の最小コスト 2 分割は

$$\left| \sum_{u \in V_1} size(u) - \sum_{v \in V_2} size(v) \right| \leq \max \left\{ \max_{v \in V} \{size(v)\}, \beta \times \sum_{v \in V} size(v) \right\}$$

を満たす最小コストの 2 分割カットである. ここで β は分割のバランスを与える正の実数である. \square

ハイパーグラフ 2 分割問題とは、与えられた重み付きハイパーグラフ $H = (V, E)$, 実数 $\beta (0.01 \leq \beta \leq 0.1)$ に対し、最小コスト 2 分割カットを求める問題である。

3 遺伝的ハイパーグラフ分割手法

遺伝的アルゴリズムは生物の進化の概念を組合せ最適化問題に応用した確率的アルゴリズムとして知られている [3]. 著者らが提案したアルゴリズム GWHB は、ハイパーグラフ分割問題に遺伝的アルゴリズムを応用したものであり [4], 初期化, 交配, 淘汰, 反復という 4 つのステップにより構成されている。アルゴリズムの各ステップは次のようになっている。

【初期化】 p 個の異なる解をアルゴリズム WHB を用いて求める。

【交配】 p 個の全ての非順序対に対して解の交配を行ない、 $\frac{p(p-1)}{2}$ 個の新しい解を作成する。

【淘汰】 $(p - m)$ 個の良い解と残りの解の中からランダムに m 個の解を選ぶ。

【反復】 以上の操作を 1 世代とし改良を繰り返す。

まず最初に初期化ステップと交配ステップで用いられる高速ハイパーグラフ分割手法 WHB について説明する。分割の対象となるハイパーグラフ $H = (V, E)$ に対応する 2 部グラフ表現 (ネットグラフと呼ぶ) $G = (V_m \cup V_n, A)$ をアルゴリズム WHB の入力とする。最初にネット節点の集合 V_n から任意に 1 つの節点を選びこれを $SEED_L$ とする。次に $SEED_L$ から最も遠いネット節点を最短経路アルゴリズムを用いて求めて $SEED_R$ とする。 $SEED_L$ と $SEED_R$ から幅優先探索を応用してネットグラフ G の 2 分割を形成する。

次に解の交配方法について説明する。 2 つの解を $S_1 = (V_{a1}, V_{a2})$, $S_2 = (V_{b1}, V_{b2})$ とする。文献 [4] の遺伝的ハイパーグラフ分割手法の交配ステップでは、解 S_1 と S_2 を重ね合わせて $V_{11} = V_{a1} \cap V_{b1}$, $V_{22} = V_{a2} \cap V_{b2}$, $V_{12} = V_{a1} \cap V_{b2}$, $V_{21} = V_{a2} \cap V_{b1}$, で表される V の 4 個の節点集合を求める。次に、こ

の中で含まれる節点数が多い順に 2 個の節点集合を選びそれぞれを 1 個の節点に縮約して $seed1$, $seed2$ とする。次に $seed1$, $seed2$ からアルゴリズム WHB を用いて新しい解を形成する。この交配プロセスから両方の解の性質を受け継いだ解を求めることができる。

4 提案手法

4.1 従来手法の改良

ここでは、3 節で説明した遺伝的ハイパーグラフ分割手法 GWHB [4] の分散アルゴリズム化について考察する。まず最初に従来からの遺伝的ハイパーグラフ分割手法に関する問題点を述べた後で従来からの遺伝的ハイパーグラフ分割手法に新たに導入する概念を述べる。

従来手法の特徴を調べた結果、解の改良は最初の 4 世代くらいで急速に行なわれその後は世代を繰り返しても解の改良はほとんど行なわれないことが分かった。これは世代が進むにつれて解の多様性が急激に失われていることも表している。人口中の解が収束するためには多様性の減少は必要であるがあまりにも急激に減少してしまうと解の改良が不十分になる可能性がある。

従来手法においてはこの点を考慮して、淘汰ステップにおいて良い解のみを選択するのではなく m 個のランダムに選んだ解を含ませることにより人口中の解に冗長性を持たせている。しかし実験によると悪い解の数を変化させてもあまり解の質は変化しないため選択を行なうときに悪い解を含めることによる効果はほとんどないと言える。

また交配ステップにおいてカット数を減少させる手続きを局所的に行なっているため高速な処理が可能であるが、一方、解の探索能力は低いと考えられる。したがって、グラフ全体を考慮した良い解を見落す可能性があり計算時間が増加してもグラフ全体を調べたほうが良い解を見つけられると予想される。そこでグラフ全体を考慮した上で FM 法のように節点をゲインにより移動することを考えたほうがよいと思われる。また、実際のチップデータにおいてはネットやセルが平均的に分布しているとは限らないため幅優先探索に基づく手法はバランスを保つためにカット数を犠牲にする可能性があると考えられる。

そこでこれらの問題点を解決するために以下に示すような手法を導入する。

- 突然変異.
- ゲインに基づく節点の移動.

以下では、これらの詳細について説明する.

【突然変異の導入】

悪い解への早期収束を改善するためには各世代の解の多様化を実現する必要がある. そこで、各世代における解の多様化を促すために突然変異を導入する. 一般に遺伝的アルゴリズムで用いられる突然変異は変異するビットの割合が小さいが、ここで用いる突然変異は発生確率が高くその適用により解の質に大きな影響を与える. また、突然変異操作を適用する個体を各世代が保持している p 個の個体全てとする. アルゴリズムでは以下のように突然変異を適用する.

人口数 p の解集合 $S = \{S_1, S_2, \dots, S_p\}$ に対し (ここで 1 つの個体は 1 つの解を表しており各個体は長さ len のビット列として表現され左側にある節点は 1、右側にある節点は 0 とする.), 突然変異率 $M[\%]$ が与えられた場合、全ての個体に対して突然変異を起させる. 1 つの個体に対する突然変異は、解のビット長を len としたときに $len \times (M/100)$ 個のビットに対してビット反転を行なうことと定義する. また、($0 < M < 50$) とする.

【ゲインに基づく節点の移動】

元の手法においては交配ステップで 2 つの解を重ね合わせるにより 2 つの解に共通の性質を受け継いだ解を生成できる. しかし、この操作においては解の改良は局所的にしか行なわれない. したがって、解の共通部分に関してはあまり改良が行なわれずに部分的な改良にとどまる. しかし、ゲインに基づいて節点を移動すればグラフ全体を考慮した解の改良が実現できる. また、バランスが崩れた場合には移動可能な節点の中でゲインの大きなものから移動を行なうことによりバランス制約を満たすために増加するカット数をできるだけ少なくすることができる.

突然変異率を設定するために行なった予備実験の結果から突然変異率が 0 のときと比べて突然変異が存在するほうが平均カット数、最小カット数のどちらに関しても良い結果が得られることが分かった. しかし、各世代に応じて適した突然変異率を設定するのは困難であると考えられるので提案分散アルゴリズ

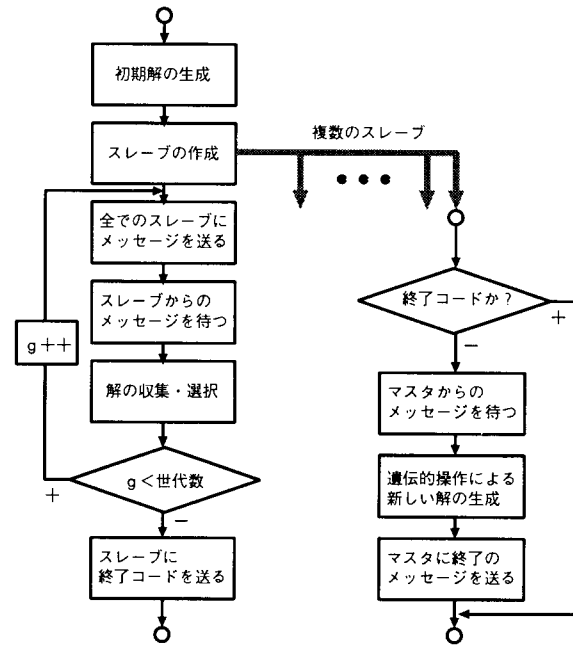


図 1: アルゴリズムのフローチャート

ムにおいてはスレーブプロセスを複数用いて、マスターから送られてきた解に大きな突然変異率を適用するプロセスと、小さな突然変異率を適用するプロセスとを同時に実行することとした.

4.2 分散処理の概要

以下では、本稿で提案する分散遺伝的ハイパーグラフ分割手法アルゴリズムの概要を示す. アルゴリズムの流れを図 1 に示す. また、アルゴリズムの各ステップは以下のようになっている.

【マスター・プロセス】

- step 1: アルゴリズム WHB により人口数 p の初期解集合を生成する.
- step 2: 各スレーブに同じ解を送る.
- step 3: 全てのスレーブから、新しく生成された解が送られて来るのを待つ.
- step 4: スレーブから送られてきた解より良い解を p 個選択する.
- step 5: step 2 から step 5 の操作を世代数繰り返す、最良解を出力し終了.

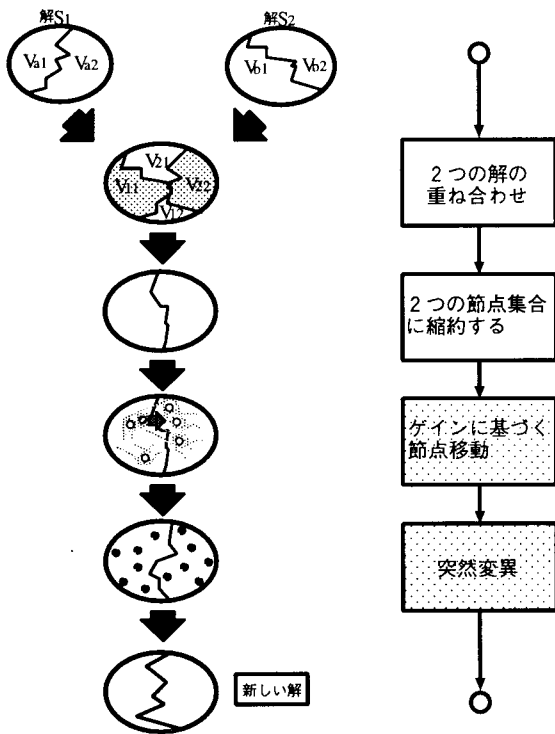


図 2: 改良した交配方法

【スレーブ・プロセス】

- step 1:** マスタから解が送られるのを待つ。
- step 2:** 受け取った解に交配，突然変異の操作を加えて新しい解を生成する。
- step 3:** 新しく生成された解の中から，良い解を p 個マスタへ送り **step 1** へ。

4.3 解の交配

提案手法の交配方法を図 2 に示す。スレーブにおいて新しい解を生成する手続きは，次の 3 つのステップから構成されている。

【重ね合わせ】 2 つの解を組合せて 4 個のブロック (節点集合) を形成し，2 つのブロックに縮約する。このとき，縮約は 2 つの大きなブロックに残りのブロック内の節点をランダムに移動することにより行なう。

【ゲインに基づく節点移動】 各節点についてゲインを計算し，ゲインの大きなものから移動を行なう。このとき，移動した

節점에隣接する節点のゲインの更新を同時に行なう。

【突然変異】 各個体中の全節点数 \times 突然変異率 [%] の節点をランダムに選び，他方のブロック内に移動を行なう。

5 実験

以下では 4 節で述べた分散遺伝的ハイパーグラフ分割手法に対して行なった実験について述べる。また，スレーブの突然変異率の決定を目的として行なった予備実験の結果についても述べる。

5.1 評価項目

アルゴリズムの有効性を以下の項目について比較し，評価する。

- (1) カット数
- (2) 分割集合間のサイズ差 (バランス)
- (3) 計算時間

項目 (2) を評価するために，評価値 *balance* を用いる。

$$balance = \frac{|wL - wR|}{wL + wR}$$

ここで， wL ， wR は 2 つのブロックに節点集合を分割したときに，それぞれのブロック内にある節点の重みの総和である。*balance* の値が小さいほど左右に均等に節点が分割されたことになる。

5.2 実験方法

分散遺伝的ハイパーグラフ分割手法をオムロンデータゼネラル社の LUNA2010 上で C 言語を用いて実現しシミュレーション実験を行なった。テストデータは，表 1 に示す MCNC (Microelectronics Center of North Carolina) ベンチマークデータを使用した。

5.3 実験結果と考察

以下では本研究で分散遺伝的ハイパーグラフ分割手法について行なった実験について説明する。まず，突然変異の効果を知るために突然変異を導入したときとそうでないときとの比較を行なった。次に従来手法との比較を行なった。また，本手法のパラメータ値の影響に関する実験を行なった。

表 1: MCNC ベンチマークデータ

データ	セル数	ネット数
primarySC1	752	1159
primarySC2	2907	3671
industry1	2271	2479
industry2	12149	13419
industry3	15059	21966

5.3.1 突然変異の効果

突然変異に関する実験を予備実験として行なった。この予備実験は、人口数 15, 世代数 6 として突然変異率を変化させてそれぞれのパラメータの組合せについて 10 回試行を行なった結果でありそれをグラフに表したものが図 3 である。図 3 から規模の小さいデータ (primarySC1, primarySC2) に関しては最小カット数, 平均カット数に関しては変化が見られないが, 最大カット数に関してはグラフにおいてかなり凹凸があり局所最適解から脱出した解とそうでない解とが存在することを表している。例えば, primarySC1 においては突然変異率が 9% における最大カット数と突然変異率が 12% における最大カット数ではおよそ 10 の違いがある。また, primarySC2 においては突然変異率が 9% のときと突然変異率が 21% のときでおよそ 40 の差がある。また, 比較的大きなデータ (industry2, industry3) における最大カット数に関してはデータの規模を考慮するとそれほど変化がないが, 最小カット数に関しては突然変異の効果が見られる。例えば, industry2 における最小カット数では突然変異率が 0% のときと 3% のときを比較するとおよそ 100 の違いがあり, industry3 において突然変異率 0 のときと突然変異率 6 のときも大きく異なる。ここで突然変異率以外のパラメータは全て同じ条件で実験を行っておりこれらの違いは全て突然変異によるものと考えられる。よって, 突然変異の導入の効果があることが示された。

5.3.2 パラメータ値の影響

提案手法のパラメータ値の影響を調べるために人口数, 世代数などのパラメータとカット数の関係を調べた。人口数を 5 から 15 まで, 世代数を 2 から 6 まで変化させた結果, まず, 世代数を固定して人口数を変化させた

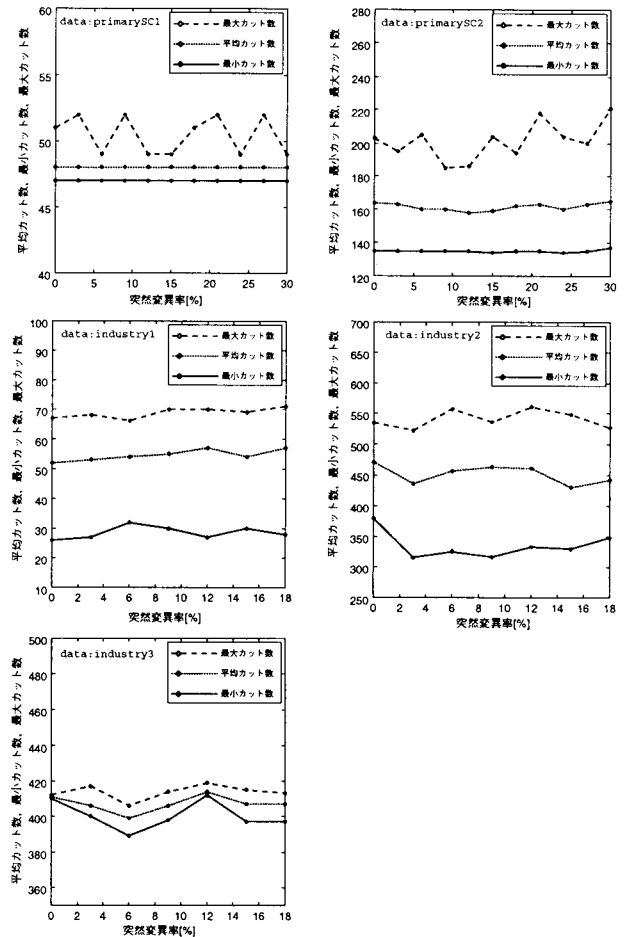


図 3: 予備実験 (突然変異率とカット数)

ときの結果より, 良い解を得るためには, 従来手法と同様にある程度の人口数が必要ながことがわかった (図 4)。

次に, 人口数を固定して世代数を変化させた場合, 4 世代くらいで解が収束している (図 5)。また, 最大カット数が世代とともに減少しており, もし局所最適解に陥ってもその状態から回復できることが分かった。

6 まとめ

本研究では VLSI レイアウト設計における基本的な問題の一つである回路分割問題に対し, 分散遺伝的ハイパーグラフ分割手法を提案した。また, 提案手法をワークステーション上に実現し, 従来の遺伝的ハイパーグラフ分割手法 GWHB と比較した結果, 提案手法の有効性が確認できた。今後の課題としては人口数や突然変異率を妥当な値に設定するための手法の検討, 交配における解の組合せの工夫による計算時間の短縮などが挙げられる。

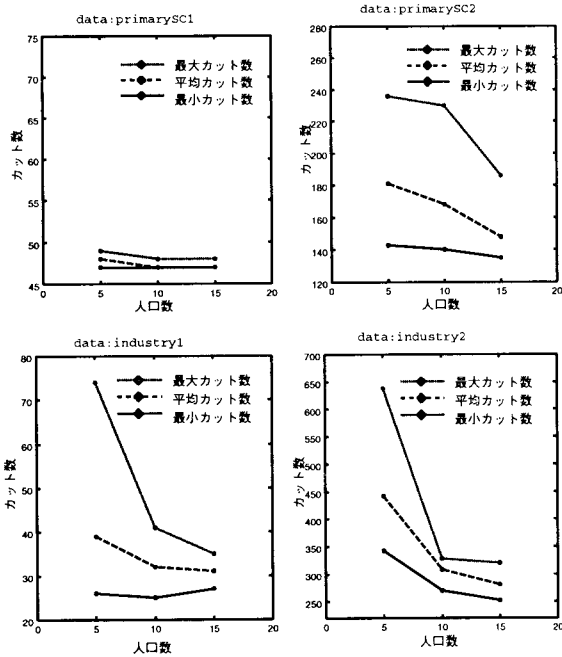


図 4: 人口数を変化させたときのカット数の
 平均値, 最大値, 最小値の関係 (世代数 6)

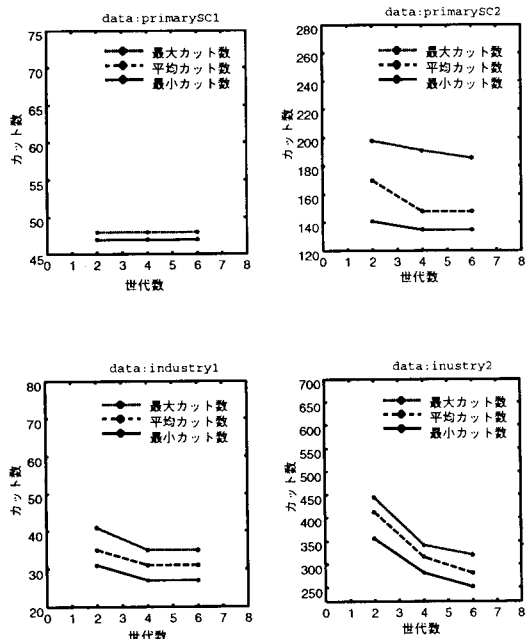


図 5: 世代数を変化させたときのカット数の
 平均値, 最大値, 最小値の関係 (人口数 15)

謝辞

本研究の遂行に多大のご支援をいただいた財団法人高柳記念電子科学技術振興財団および財団の関係者の皆様に深く感謝いたします。また、日頃ご指導いただいております広島市立大学情報科学部教授吉田典可先生、および本研究にご協力いただきました広島大学工学部小出哲士氏、広島市立大学情報科学部上土井陽子氏、ならびに広島大学大学院修了生の岸本善久氏、広島大学大学院博士課程前期2年生の松田憲治君に感謝いたします。

参考文献

- [1] C. M. Fiduccia and R. M. Mattheyses: "A linear-time heuristic for improving network partitions," Proc. of 19th Design Automation Conf., pp. 175-181 (1982).
- [2] M. R. Garey and D. S. Johnson: "Computers and Intractability: A Guide to the Theory of NP-completeness," W. H. Freeman (1979).
- [3] D. E. Goldberg: "Genetic Algorithms in Search, Optimization, and Machine Learning," Addison-Wesley (1989).
- [4] Y. Kamidoi, S. Wakabayashi and N. Yoshida: "An efficient GA hybrid for hypergraph bisection with application to VLSI placement," Proc. IEEE Asia-Pacific Conf. on Circuits and Systems, pp. 369-401 (1992).
- [5] 上土井, 若林, 吉田: "ハイパーグラフを k 分割する遺伝的分散アルゴリズムとその実験的評価," 信学技報, COMP92-97, pp. 39-48 (1993).
- [6] B. W. Kernighan and S. Lin: "An efficient heuristic procedure for partitioning graphs," Bell Sys. Tech. J., Vol. 49, pp. 291-307 (1970).
- [7] N. A. Sherwani: "Algorithms for VLSI Physical Design Automation," Kluwer Academic Publishers (1993).
- [8] 若林, 上土井, 岸本, 小出, 吉田: "遺伝的ハイパーグラフ分割手法の分散アルゴリズム化," 電気・情報関連学会中国支部第45回連合大会, pp. 431-432 (1994).
- [9] 若林, 松田, 岸本, 小出, 吉田: "大規模回路の最小コスト分割のための分散遺伝的アルゴリズム," 情報処理学会設計自動化研究会研究報告, 76-2, pp.9-16 (1995).