

遺伝的プログラミングによる調停グラフの交差数減少とその並列分散処理

札埜 有香[†] 田村 慶一[†] 森 康真[†] 北上 始[†]

[†] 広島市立大学情報科学部 〒731-3194 広島県広島市安佐南区大塚東 3-4-1

E-mail: † {fudano, ktamura, mori, kitakami}@db.its.hiroshima-cu.ac.jp

あらまし 生物分類樹と分子進化系統樹から作成される調停木を比較することにより、遺伝子重複の研究や、生物分類樹の見直しがされている。しかしながら、このためには膨大な計算量が必要である。本研究では、局所解への収束をできるだけ避け、高速によりよい調停グラフを導くために、Minimal Generation Gap (MGG) を用いた遺伝的プログラミングに対する新しい並列分散処理方法を提案する。

キーワード データベース一般、並列・分散DB、最適化、遺伝的プログラミング

Reducing Crossovers in Reconciliation Graphs with a Genetic Programming and Its Parallel Distributed Processing

Yuka FUDANO[†] Keiichi TAMURA[†] Yasuma MORI[†] Hajime KITAKAMI[†]

[†] Faculty of Information Sciences, Hiroshima City University 3-4-1 Ozukahigashi, Asa-Minami-ku, Hiroshima, 731-3194, Japan

E-mail: † {fudano, ktamura, mori, kitakami}@db.its.hiroshima-cu.ac.jp

Abstract By comparing reconciliation graphs constructed from taxonomic and gene trees, gene duplications are researched and taxonomic trees are corrected. However, a huge computational complexity is necessary for this. The authors propose a new parallel distributed processing method of a Genetic Programming using Minimal Generation Gap (MGG), for leading a better reconciliation graph by high speed and avoiding settling to the limited part solution.

Keyword Database, Parallel and Distributed DB, Optimization, Genetic Programming

1. はじめに

系統樹間の調停とは、構造の異なった2つの木の節点間の対応を1対1に対応付けることである。調停は、生物の多様性を解明するために行なわれている。例えば、図1のような生物分類樹と分子進化系統樹から調停木を作成、比較することにより、遺伝子重複の研究や、生物分類樹の見直しがされている。しかしながら、このためには膨大な計算量が必要であり、さまざまな工夫がなされている^{[1][2]}。

本論文では、GPによる調停グラフの交差数減少とその並列分散処理を提案する。提案するGPによる調停グラフの交差数減少とその並列分散処理の特徴は、世代交代モデルとしてMinimal Generation Gap (MGG)^[3]を使用していることと、MGGの並列化においてはワーカプロセスの台数分個体を選択する並列化モデルを採用したことがあげられる。

MGGは初期収束を回避し、探索の後半においても進化的停滞を抑制するというすぐれた特徴を持つ。調停グラフを解くために、GPを用いて調停グラフの交差数減少を行なうアルゴリズム^[2]が存在する。文献[2]の研

究では世代交代のモデルとしてSimple GA^[3]が用いられている。Simple GAは初期収束する可能性があり、MGGを世代交代のモデルとして採用したことで、探索の後半においても進化的停滞を抑制できると期待される。

MGGの並列化は、文献[4][5]でいくつか提案されている。提案されたMGGの並列化では、選択操作において2つの個体のみを選択するが、提案手法では、選択操作においてワーカプロセスの台数分個体を選択する。選択操作において複数の個体を選ぶことにより、多様性を維持しつつランダム性が向上し、従来手法よりも提案手法の方が世代交代の早い段階で良い解を求めることができるかと期待される。

提案手法をPCクラスタ上で実装し、実際の生物分類樹と分子進化系統樹を使い性能評価を行なった。性能評価により、Simple GAでは求めることができなかった解を求めることができ、提案手法の方が従来手法と比べて世代交代の早い段階でよい解を求めることができることを確認した。

本論文の構成は以下の通りである。2章ではデータモデルを説明する。3章ではGPによる調停グラフの交

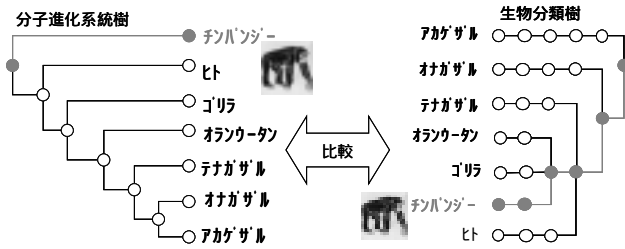


図 1. 調停

差数減少方式について述べる．4 章ではその並列分散処理について説明し，5 章で関連研究を紹介する．6 章で実験と評価を示し，最後に 7 章でまとめる．

2. 問題の定義

本節では，問題の定義を木構造とその構造に対する操作という観点から説明する．図 2 の木のように，初期の調停グラフは 2 種類の順序木から構成されている．最初に 2.1 節では，調停グラフの構成要素である順序木の構造について述べる．次の 2.2 節では，計算手法の説明に重要な構造である結合行列とそれに対する基本操作について述べる．最後の 2.3 節では，順序木から最大クラスタと最小クラスタを抽出する方法について述べる．両者はクラスタ交換と呼ばれる基本操作に利用される．

2.1. 順序木

グラフの各順序木のすべての枝を (d,e) の形で表わす．入力データはこの形式の枝情報とする．以下では枝 (d,e) において，始点 d を親節点と呼び，終点 e を子節点と呼ぶことにする．例えば，図 2 の左側順序木において，枝は (1,a), (1,2), (2,d), (2,3), (3,b), (3,c) となる．

このとき，子節点にはなるが，親節点にはならないものが，順序木の葉節点となる．葉節点の並びを表わすものを葉節点列とし，右側と左側の両葉節点列が 1 つの染色体を構成する．

2.2. 結合行列

左側順序木と右側順序木についての葉節点の対応に対して結合行列を定義するアプローチをとる．図 1 の葉節点間の結合関係を表わす結合行列は図 2 に示されるとおりである．図の左側順序木及び右側順序木に存在する葉集合 {a,b,c,d} の順序関係は，各々， $a < d < b < c$ ， $a < b < c < d$ である．これらを，各々， $OL_1 = [a,d,b,c]$ 及び $OL_2 = [a,b,c,d]$ で表現する．左側順序木の葉 b と右側順序木の葉節点列 [a,b,c,d] との間の結合関係は，結合行列 Mat の 3 行目の行ベクトル (0,1,0,0) として表現される．左側順序木の b は右側順序木の a,c,d と接続関係を持たないので，行ベクトルの第一，第三，第四の要素は 0 で表現されているが，第二要素は两部分木間の葉 b 同士で接続関係をもつので 1 で表

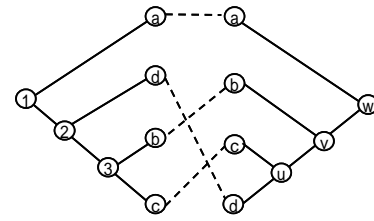


図 2. 調停グラフ

	a	b	c	d
a	1	0	0	0
d	0	0	0	1
b	0	1	0	0
c	0	0	1	0

図 3. 結合行列

現されている．結合行列の要素を m_{ij} で表現すると，葉節点列間の交差数 $C(\text{Mat})$ を計算する式は以下のとおりである．

$$C(\text{Mat}) = \sum_{[1 \leq j < k \leq n]} \sum_{[1 \leq \alpha < \beta \leq n]} m_{j\beta} m_{k\alpha} \quad (1)$$

上記の結合行列に対する操作には，2 種類の基本操作がある．それらは，相異なる列同士の交換操作及び相異なる行同士の交換操作である．その 1 つは，左側順序木において相異なる葉節点同士の順序を交換することに相当し，他の 1 つは右側順序木の葉節点に関する交換に相当する．例えば，図 3 において 1 列目のベクトル (1,0,0,0) と 4 列目のベクトル (0,0,1,0) を交換することは，図 2 において右側順序木の 2 つの葉節点 a と d の順序を交換することに相当する．以上は交換操作の単位が 2 つの列または 2 つの行同士だが，ここではこの交換操作を複数の列または複数の行同士に拡張する．2.3 節で詳述するが，ある性質を満たす複数行または複数列をクラスタと呼ぶ．例えば，右側順序木において，葉節点集合 {b,c,d} に相当する 2~4 までの列集合は，クラスタと呼ばれる．このクラスタと，1 行目に相当する葉節点集合 {a} との順序を交換しても，右側順序木の枝に新たな交差を作らない．この交換により新たな順序関係として， $b < c < d < a$ が得られる．このように，結合行列に対する我々の操作は，列または行に関するクラスタ同士の交換操作を基本にしている．この交換操作は，GP に利用される．

2.3. クラスタ

ここでは，前述のクラスタを最大クラスタと最小クラスタの 2 種類に分け，各々のクラスタを見つけ出す方法について述べる．最大クラスタは，GP における初期集団の作成や突然変異操作におけるクラスタ交換に利用する．最小クラスタは，GP の探索処理における組

替え操作で利用されるクラスタ交換の対象である。結合行列に対する最大クラスタを探すためには、まず結合行列の行側か列側かのどちらに着目するかを決めなければならない。行側（列側）に着目することは左側（右側）順序木に着目していることに相当する。

2.3.1. 最大クラスタ

相異なる2つの葉節点を指定し、両者を区別するのに有用な2つの最大部分木を探す。以下、図4を用いて、葉節点集合を有する順序木から2つの葉節点 p, q を各々含む2つのクラスタ C_1, C_2 を見つけ出す方法について述べる。2つのクラスタ C_1, C_2 は、2つの葉節点 p, q を各々含むクラスタの中で最大のものである。ただし、 $p \in C_1, q \in C_2$ であり、 $C_1 \cap C_2 = \emptyset$ である。また、葉節点 p のクラスタとは、 p を含む部分木の葉節点集合を指す。以上の性質を満たす2つの最大クラスタ C_1, C_2 は、以下の手順により見つけ出すことができる。

上方探索により、2つの葉節点 p, q の分岐点 r を見つけ、 r に連結される複数の部分木の中から葉節点 p を含む部分木 $SubT_1$ 及び葉節点 q を含む部分木 $SubT_2$ を見つける。ただし、 $SubT_1$ と $SubT_2$ の節点には共通節点がないとする。また、 $SubT_1$ と $SubT_2$ は、各々、考えられる部分木の中で最大の部分木であるとする。そして、下方探索により、2つの部分木 $SubT_1$ と $SubT_2$ の各々に対する葉節点集合 C_1, C_2 を見つけることができる。

2.3.2. 最小クラスタ

まず相異なる2つの葉節点 p, q を指定する。図5に示すように両葉節点 p, q を含む最小部分木 $SubT$ を見つけると、 $SubT$ の葉節点集合が最小クラスタに相当する。 p, q を含む最小部分木の根 r は上方探索をすることにより見つけ出すことができ、部分木 $SubT$ の葉節点集合は、下方探索により見つけることができる。

3. GP による探索

一般に GP では、予め定めた遺伝子型を用いて初期遺伝子集団を作成し、その集団に対して、選択操作、組替え操作、突然変異操作の3つの基本操作を繰り返し適用する。そして、予め定めた基準を満足したときに、これらの操作による探索処理を停止している。調停グラフの交差数減少問題を解く場合、その基準はゼロ交差になるかあるいは個体の改良が見られなくなったときに満足する。

3.1. 選択操作

3.1.1. 適応度

調停グラフの交差数を減少させるため、ここでは、交差数のより小さな調停グラフを適応度のより高い個体と見なしている。

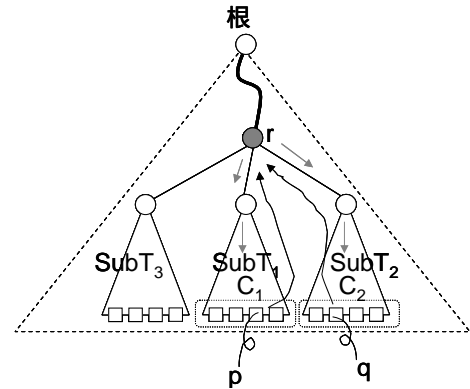


図 4. 最大クラスタの探索方法

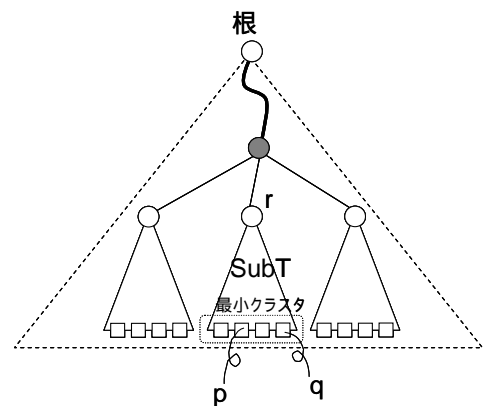


図 5. 最小クラスタの探索方法

3.1.2. 世代交代モデル MGG

MGG は、探索序盤における選択圧をできるだけ下げて初期収束を回避するとともに、探索の後半においても集団内に多種多様な個体を生存させやすくして進化的停滞を抑制することを意図したモデルであり、そのためには世代間での個体分布の差異を最小化することが望ましいとの考えに基づく。

選択操作には、2種類の選択、すなわち複製選択と生存選択が含まれる。複製選択は子を生成する親を選ぶための選択であり、生存選択は次世代に生き残る個体を選ぶための選択である。

MGG による選択は以下のように要約される。

(1) 複製選択

適応度を無視して、集団から2個体をランダムに抽出する。

(2) 生存選択

エリート戦略により選ばれた最良1個体およびルーレット選択により選ばれた1個体を次世代に残す。ルーレット選択では、適合度に比例した割合で個体を選択する。

(1)により初期収束を回避し、(2)によって適応度の分

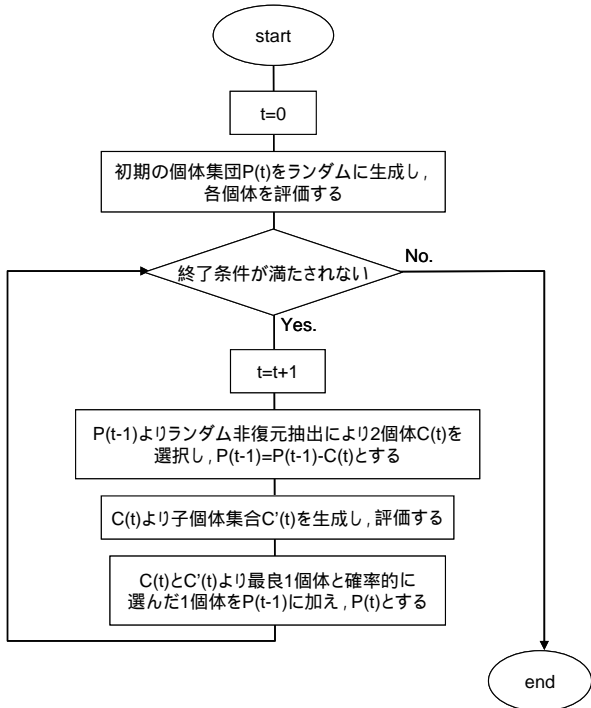


図 6. MGG のアルゴリズム

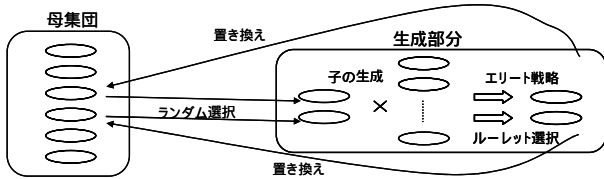


図 7. MGG

散をできるだけ維持できるように配慮することを目指す。

MGG のアルゴリズムは図 6 のとおりである。また、操作の一連の流れを図 7 の示す。母集団中の染色体は、1 世代に 2 個体を更新している。

3.2. 組替え操作

組替え操作は、2 本の親染色体の各々から選択される部分列同士を交換し、2 本の子染色体を生成する方法により達成している。この操作では、2 つの親染色体の中で、左側（または右側）の部分列同士が交換対象であり、2 つの部分列はお互いに同じ葉節点名を持たなければならない。お互いに異なる葉節点を持つことを許すと、生成される子染色体には致死遺伝子が発生する。従って、ここでは同じ部分列同士の組替え操作だけを許している。組替え操作は、葉節点列の並びの変更を目的としているので、交換対象とする部分列は 2 つ以上の要素を含んでいる必要がある。その部分列は、2 つの葉節点 p, q に対する最小クラスタを順

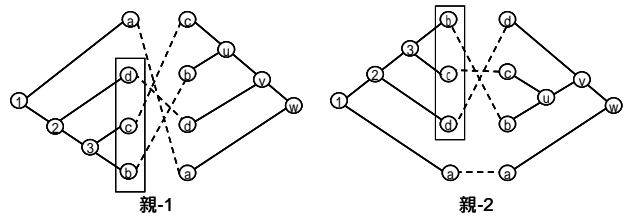


図 8. 二つの親染色体の例

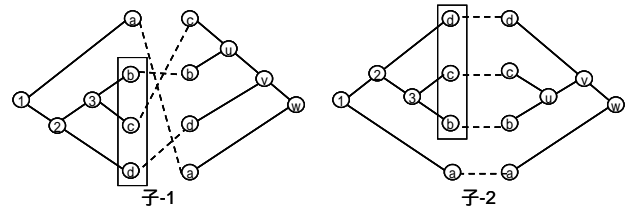


図 9. 生成された二つの子染色体の例

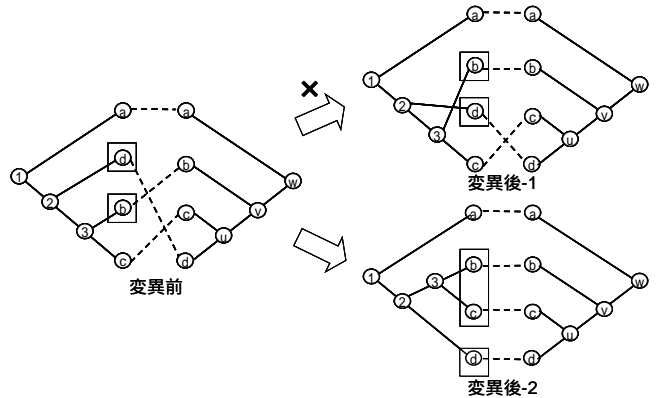


図 10. 突然変異操作

序木から探索することにより得られる。また、この操作では、最小クラスタ同士の交換を左右双方の各順序木に対して実施している。

図 8, 2 つの親染色体の例である。図中の親-1 は、染色体として $[(a, d, c, b), (c, b, d, a)]$ を持つ。この場合のグラフの交差数は 5 である。親-2 の例は、染色体として $[(b, c, d, a), (d, c, b, a)]$ を持つ。従って、このグラフの交差数は、3 である。

図 8 の例を用いて、左側の順序木に着目し、両者の組替え操作について考える。最初にランダムに、2 つの葉節点 b および d が選択されたとすると、最小クラスタは図 8 点線で囲まれた部分になる。図 8 親染色体同士を、最小クラスタで交換すると、図 9 示されるようになる。図中の子-1 は、染色体として $[(a, b, c, d), (c, b, d, a)]$ を持ち、グラフの交差数は、4 である。子-2 は、遺伝子型として、各染色体 $[(d, c, b, a), (d, c, b, a)]$ を持つ。従って、このグラフの交差数は、ゼロである。右順序木に着目した場合も、同様の組替え操作を行なっ

ている。

3.3. 突然変異操作

突然変異操作は、集団からランダムに選択された染色体上で2つの葉節点同士を交換する方法を

基本操作にしている。ただし、2つの葉節点 p, q は、右側または左側の葉節点列のどちらかをランダムに選択した葉節点列に対して、さらにランダムに選択されたものである。選択された2つの葉節点 p, q を単純に交換すると、順序木の枝の間に交差が生じてしまうことがある。ここでは、2.3節で述べた2つの最大クラスタ C_1, C_2 を見つけ出し、両クラスタを交換している。

図10は突然変異操作の例である。図中の変異前は、染色体として $[(a, d, b, c), (a, b, c, d)]$ を持つ。この場合のグラフの交差数は2である。この染色体について突然変異操作を行なってみる。最初にランダムに、左側の順序木が選択され、2つの葉節点 b および d が選択されたとする。これを単純に交換すると、変異後-1になり、左側順序木の枝に交差が生じる。そこで、変異後-2のように b, d それぞれを含む最大クラスタ同士を交換し、新しい染色体を作成する。このように、順序木の枝の間に交差が生じることを防いでいる。

4. 並列分散処理

4.1. システム構成

大規模な調停グラフを扱う場合、処理に時間がかかる。そこで、並列分散処理を行なうことで、高速に解を導く。図11示されるような移住を行なわない MGG の並列分散モデルを提案する。

マスタプロセスでは初期集団の生成、複製選択、親と子の入れ替えの処理を行なう。複製選択では、ランダムに選択された2つの親染色体をワーカプロセスへ受け渡す。このとき、それぞれのワーカプロセスへ受け渡す2個体は、それぞれ選択する。つまり、 $(2 \times \text{サイト数})$ の染色体が選択される。

ワーカプロセスでは子の生成、適合度計算、生存選択の処理を行なう。生存選択では、両親と生成されたすべての子を合わせた固体集合からエリート戦略とルーレット選択によって2つの染色体を選択し、母集団中の両親と置き換える。

通信量は、母集団からランダムに選択した2個体をワーカプロセスに転送し、2個体を受信するであるため、 $(2 \times \text{ワーカプロセス数} \times \text{個体のサイズ} \times \text{世代交代数}) \times 2$ となる。

4.2. 処理手順

MGG の並列分散モデルの処理手順を説明する。

ここでは、母集団数を n 個、母集団作成時に1個体を作成するために行なう突然変異の回数を m 回、次世代の2個体を選択するために生成する候補回数を x 個、

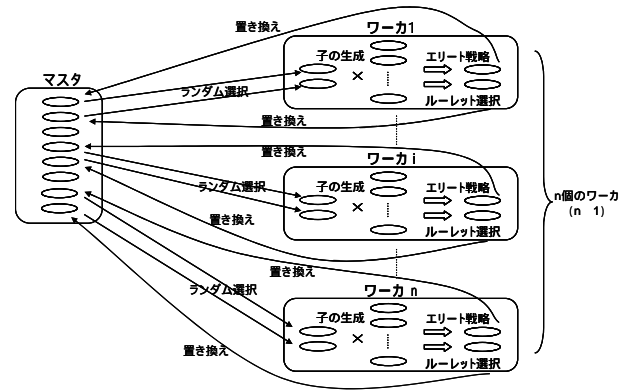


図 11. システム構成図

突然変異の割合を $y\%$ とする。

(1) マスタプロセスでの処理

① 初期集団の生成

- 枝の情報を基に、初期染色体を生成する。
- 初期染色体に対して1個体につき m 回の突然変異操作を行ない、 n 個の初期母集団を作成する。

② 複製選択

- 母集団からランダムに2個体を選択する。
- 2個体をワーカプロセスへの受け渡す。

(2) ワーカプロセスでの処理

③ 子の生成

- 組替え操作により、次世代の2個体を選択するために生成する候補解を x 個生成する。
- $y\%$ の割合で、突然変異操作を行なう。

④ 適合度計算

- ③で生成した候補解の適合度を計算する。

⑤ 生存選択

- 候補の中から、エリート戦略で最良1個体を選択する。
- ルーレット選択で確率的に1個体を選択する。
- 生存選択で選んだ2個体をマスタプロセスへの受け渡す。

(3) マスタプロセスでの処理

⑥ 親と子の置き換え

- 選択した2個体を、母集団中の親と置き換える。

⑦ 探索終了の判定

- ゼロ交差になるか、あるいは一定の世代数を超えたとき、または個体の改良が見られなくなったときに探索処理を停止する。
- それ以外の場合は、②～⑦を n 回繰り返す。

⑧ 終了

- 探索を終了した時点での最良 1 個体を解とする。

5. 関連研究

並列化についての研究は、マスタワーカモデルでグリッド向け実行環境 Jojo 上で実装されているもの^[4]やパラメータサーベイ型のマスタワーカ型の並列プログラムを簡単に作成できるツールである Ninf を用いて並列化をおこなったもの^[5]がある。研究[4]では、評価モデルと交叉モデルという 2 つのモデルを提案している。また、研究[5]は研究[4]と関連しており、Ninf 版並列化ツールを使用するため研究[4]で提案されている評価モデルで実装を行なっている。ここでは、効率のよい交叉モデルについて説明する。

交叉モデルでは、図 12 示すように、マスタプロセスで母集団より 2 個体を選び、その 2 個体をすべてのワーカプロセスに転送し、ワーカプロセスで子の生成、評価、選択を行なう。ワーカプロセスで選択された 2 個体をマスタプロセスに戻し、戻った 2 個体よりエリート戦略、ランダム選択により 2 個体を選び世代交代を完了させる。世代交代のモデルは、生存選択のエリート戦略は同じであるが、ランダム選択の意味合いが MGG と若干異なる。

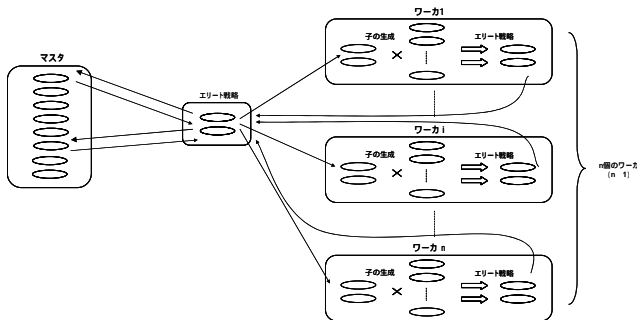


図 12. 交叉モデルのシステム構成図

6. 実験・評価

アルゴリズムの実装は C 言語によって行った。PC クラスターの構成は次の通りである。Intel Pentium4 プロセッサ 2.8GHz, 1.0GB メモリを搭載し、各計算機は 1,000 Mbit/sec イーサネットに接続されている。OS には Fedora 2.0 を使用し、通信には MPICH の version 1.2.5 を MPI ライブラリとして使用した。

6.1. 目的

実験の目的は、提案手法の有効性を確認するため、以下の項目を確認することである。比較対象として前述の交叉モデルを従来手法として用いた。

(1) 提案手法の並列性能の測定

今回提案する方法を並列化したとき、逐次の場合と比べて、交差数の少ない調停グラフを選び出す世代が短縮されているか。また、求めた解が向上している

か。

(2) 提案手法と従来手法の比較

提案手法と従来手法の並列性能を比較し、提案手法がよりよい解をより少ない世代数で求めることができるか。

6.2. 方法

ここでは、実験の詳細な内容について述べる。

(1) 提案手法の並列性能の測定

作成したプログラムは以下のように動作する。

- 入力データは左右の系統樹の枝情報と葉節点数とする。枝は (d,e) の形で表わす。
- 入力データから元になる 1 個体を作成する。この 1 個体は、左右の葉節点の情報とその交差数の情報を持っている。
- 元になる 1 個体から初期集団 (母集団) を生成する。
- 提案手法を適応する。ここでは、以下のことを繰り返す。
 - ① それぞれのワーカが母集団から 2 個体を選択し、そこから新たな 2 個の個染色体を生成する。
 - ② 母集団中の選択した 2 個体を生成した 2 個体と置き換える。
- 終了条件を満たしたら、最良 1 個体を出し、終了する。

このプログラムを、サイト数を 1 から 16 まで変化させて、それぞれ 2000 世代まで動作させる。このときの、交差数や収束性を比較し、並列性能の測定を行なう。

(2) 提案手法と従来手法の比較

従来手法について、(1) と同様の実験をし、その結果と (1) の結果を比較する。

6.3. 実験データ

使用した調停グラフは、実際に利用されている生物分類樹と分子進化系統樹の 2 種類の木 (葉節点数は各々 40 件) から構成されているもの (Hashimoto97) と、それを元に 2 つの系統樹 (葉節点数は各々 80 件) を人工的に生成したもの (Test80) を使用する。Hashimoto97 の概要を、表 1 に示す。

このとき、パラメータは以下のように設定した。

- 提案手法の母集団数: サイト数 × 200
- 従来手法の母集団数: 400
- 次世代の 2 個体を選択するために生成する候補の染色体数: 1000
- 突然変異の割合: 1%

表 1：測定に用いたデータ

データ名	葉節点数	生物分類樹		分子進化系統樹		最小交差数
		非葉節点数	総節点数	非葉節点数	総節点数	
Hashimoto97	40	201	241	39	79	9

表 2：提案手法の測定結果（2000 世代後）

サイト数	交差数	
	Hashimoto97	Test80
1	22	267
2	16	23
4	12	8
8	10	9
16	9	4

表 3：従来手法の測定結果（2000 世代後）

サイト数	交差数	
	Hashimoto97	Test80
1	17	265
2	15	127
4	15	118
8	19	167
16	15	482

提案手法と従来手法の母集団数が異なっているのは、1 世代で行なわれる複製選択で選ばれる染色体数が異なっているためである。例えば、サイト数 16 のとき、提案手法の母集団数は 3200、従来手法の母集団数は 400 となる。このとき、探索開始から終了までの間に、提案手法では 1 染色体につき平均 20 回の更新が行なわれ、従来手法では平均 10 回の更新が行なわれる。

次世代の 2 個体を選択するために生成する候補の染色体数は、ここでは多様性を維持するために大きいサイズ（1000）にしている。

突然変異については、割合を高くすると得られる解の精度が向上するが、探索の後半においても収束性を見せず、無駄な探索を行うことになる。そのため、ここでは突然変異率を 1% とした。

6.4. 測定結果

6.4.1. (1) 提案手法の並列性能の測定

(1) について、測定した結果を表 2 に、このときの交差数が減っていく過程のグラフを図 13, 14 に示す。

表 2 より、Hashimoto97, Test80 の両方において、サイト数が増えるに連れて交差数は小さくなっていることがわかる。サイト数が 16 のとき、Hashimoto97 の最

小交差数である 9 が得られた。また、図 13, 14 より、どちらの場合もサイト数が大きいほど収束性が速く、世代のはやい段階で小さい交差数が得られた。

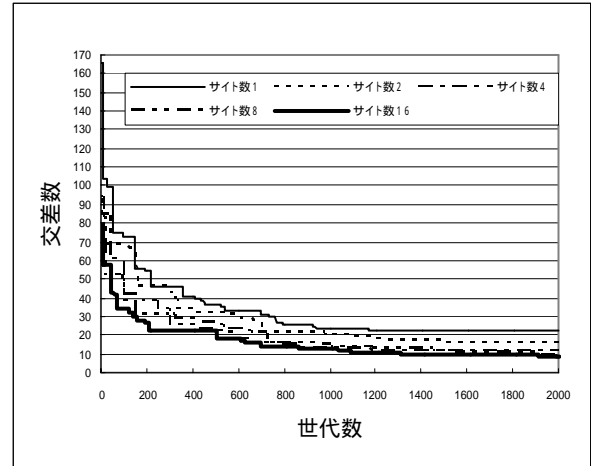


図 13. 提案手法の測定結果：Hashimoto97

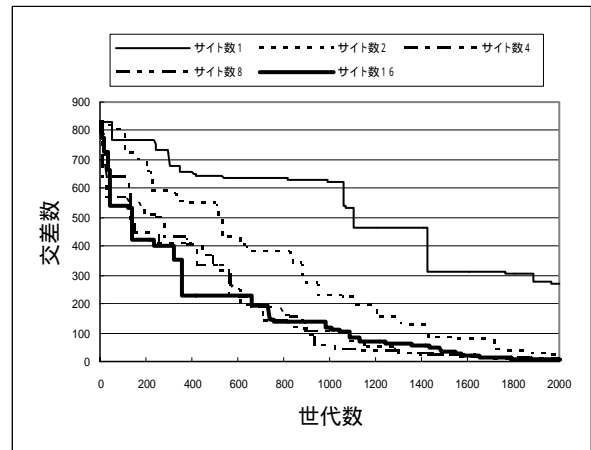


図 14. 提案手法の測定結果：Test80

6.4.2. (2) 提案手法と従来手法の比較

(2) について、上記 (1) の結果と、従来手法に対して同様の測定をした結果を比較した。従来手法の測定結果を表 3, 図 15, 16 に示す。

従来手法では、サイト数を増やしても解の向上は見られなかった。サイト数 16 のとき 2000 世代目の Hashimoto97 の交差数は 15 となり、最小交差数は得られなかった。Test80 に対しても、サイト数を 16 にしても、交差数は全く減らなかった。また、収束性についても、サイト数による変化はなかった。

サイト数が 16 のときの提案手法と従来手法の測定結果を比較する。このとき、提案手法の 2000 世代後の交差数は、従来手法のものより Hashimoto97 では約 2/3、Test80 では約 1/100 となった。また、収束性も提案手法のほうが従来手法に比べて速かった。

6.5. 考察

前節の実験結果から次のようなことがわかった。

(1)の測定結果より、提案手法では、サイト数が増えるにつれてよりよい解が得られた。したがって、提案手法の並列性能はよいといえる。

(2)の結果より、提案手法のほうが従来手法より並列性能がよく、よりよい解が得られたのは、従来手法が1世代に2個染色体のみ更新させるのに対して、提案

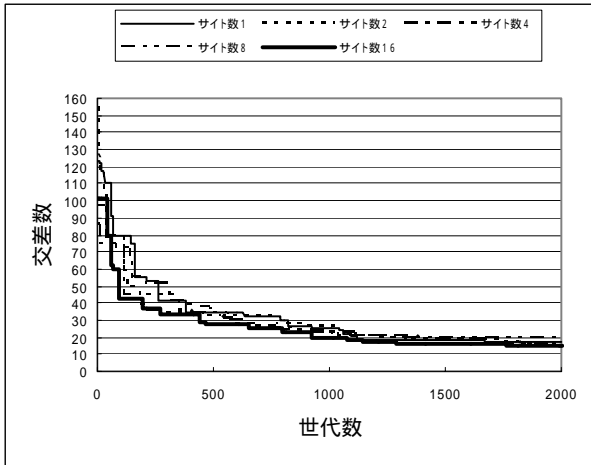


図 15. 従来手法の測定結果 : Hashimoto97

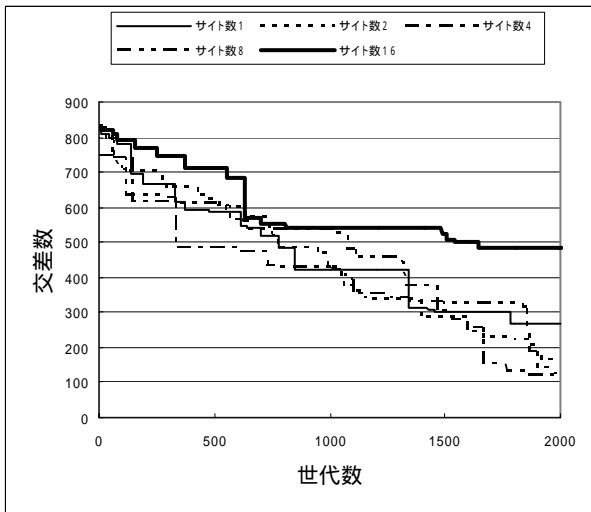


図 16. 従来手法の測定結果 : Test80

手法では1世代に(ワーク数×2)個の染色体を変更させるからであると思われる。そのため、従来手法ではサイト数が増えなくても1世代で更新される染色体の数が変わらないので、収束性や交差数にあまり変化が見られない。しかし、提案手法ではサイト数が増えると1世代で更新される染色体の数が増えるので、より速くよい解を得ることができた。また、1度に多くの染色体をルーレット選択によって更新させるので、多様性も保つことができ、よりよい解が得られた。

以上のことから、提案手法は従来手法より並列分散処理に向いており、サイト数が増えるほど解の品質が向上することがわかった。

7. おわりに

本論文では、世代交代モデルとして MGG を採用した GP による調停グラフの交差数減少の方法を述べ、それを並列分散処理する方法について提案した。

GP による調停グラフの交差数減少では、GP の基本操作である、選択操作、組替え操作、突然変異操作を繰り返し行なっている。選択操作では、世代交代モデル MGG を使用することによって、多様性を維持し、局所解への収束をできるだけ避けている。

提案した MGG の並列化の有効性を確かめるために、実際に利用されている 40 件の葉節点を持つ順序木から構成される調停グラフと、人工的に生成した 80 件の葉節点を持つ順序木から構成される調停グラフに適応した。並列分散処理を行なうことによって、従来手法である交差モデルよりも、大規模な調停グラフに対して収束性がよくなり、解の品質が向上することがわかった。調停グラフの交差数減少について、提案手法は従来手法より有効であるといえる。

今回は、提案手法と従来手法の 2000 世代までの測定結果を比較し、提案手法の有効性を示した。今後は候補解の分散についても調べ、提案手法が有効である理由を検討する。また、提案手法と MGG を用いないシンプルな GP との比較についての検討も残されている。

謝 辞

本研究の一部は広島市立大学・特定研究費(一般研究費(コード番号:3106))、文部科学省科学研究費補助金(課題番号:16700114)の支援により行なわれた。

文 献

- [1] 北上始, 森康真, 太田聡史, 斎藤成也; “異種系統樹間の調停のためのゼロ交差制約の充足”, 情報処理学会論文誌, データベース, 第 40 巻 SIG 5(TOD2)号, pp.1-14, May 1999.
- [2] Hajime Kitakami and Yasuma Mori; “Reducing Crossovers in Reconciliation Graphs using the Coupling Cluster Exchange Method with a Genetic Algorithm”, H. Motoda (Ed.), Active Mining, IOS Press, Vol.79, p163-174, 2002.
- [3] 佐藤 浩, 小野 功, 小林 重信; “遺伝的アルゴリズムにおける世代交代モデルの提案と評価”, 人工知能学会誌, Vol.12 No.5, pp.734-744, 1996.
- [4] 中田秀基, 中島直敏, 小野功, 松岡聡, 関口智嗣, 小野典彦, 楯真一; “グリッド向け実行環境 Jojo を用いた遺伝的アルゴリズムによる蛋白質構造決定”, 情報処理学会研究報告 2002-HPC-93, pp.155-160, March 2003.
- [5] 小野功, 今出広明, 中田秀基, 小野典彦, 松岡聡, 関口智嗣, 楯真一; “蛋白質立体構造の進化的解析のための Ninf 版並列 MGG とその性能評価”, 情報処理学会研究報告 2002-HPC-93, pp.149-154, March 2003.