

PC クラスタにおける 2 個体分散遺伝的アルゴリズムの高速化

谷村勇輔，廣安 知之，三木光範

同志社大学 工学部

E-mail: tanisuke@mikilab.doshisha.ac.jp

遺伝的アルゴリズムは優れた最適化手法の一つである。本研究では、新しい分散遺伝的アルゴリズムとして「2 個体分散遺伝的アルゴリズム (Dual Individual DGAs, Dual DGAs)」を提案し、PC クラスタシステム上でのその有効性を検討する。提案する Dual DGAs は 1 島内に 2 個体のみが存在する簡単なモデルであり、かつ、高い探索能力を有する。Dual DGA の並列モデルでは、1 プロセッサ内に複数の島が存在し、プロセッサ内で島間の個体移住が、プロセッサ間で島移住が行われる。そのため、いくつかの通常の DGA とは異なるパラメータが存在する。本研究では、まず、それらのパラメータの解への影響について検討を行っている。また、個体数が一定の際の、プロセッサ数の解への影響についても検討を行っている。これらの検討は、典型的なテスト関数の最適化問題の数値実験を通じて行い、提案する Dual DGA の有効性および PC クラスタシステムに適したモデルであることを示している。

Performance Tuning of Dual Individual Distributed Genetic Algorithms on PC Cluster Systems

Yusuke TANIMURA, Tomoyuki HIROYASU, and Mitsunori MIKI

Knowledge Engineering Department, Doshisha University

Genetic Algorithm is one of the powerful tools for optimization problems. In this paper, on PC cluster system, we examined the characteristics of the new distributed Genetic Algorithms: Dual Individual Distributed Genetic Algorithms (Dual DGAs). In this model, there are only two individuals in each island and Dual DGA has high searching ability. In a parallel model of Dual DGAs, one processor has several number of islands. In each processor, the operation of a migration of individuals between the islands is occurred. The operation of migration of islands occurred between the processors. There are some new parameters and we studied how these parameters affect to the optimum values. We also investigated the effect of the number of the processors when the total population size is fixed. These discussions are performed through the typical numerical test functions. From these results, it is cleared that Dual DGAs is effective model of DGAs and suited for PC cluster systems.

1 緒言

遺伝的アルゴリズム (Genetic Algorithms : GA) は優れた最適化手法の 1 つである。GA は生物の進化と淘汰を模倣した確率的な多点探索を行う。つまり従来の最適化手法とは異なり、目的関数の微分値を使用することなく解探索を行うことができる。これにより、従来の最適化手法では解くことが困難であった多峰性を有する連続的問題や離散的問題にも適用できる。GA は必ずしも解が得られるという保証をもたないアルゴリズムであるが、適切に用いることで有用な手法であると考えられている。

しかしながら、GA は解を得るまでに膨大な反復計算を必要とする。高い計算コストを高速に処

理するために、GA を並列化する研究が多くなされてきた。¹⁾ これまでに提案されている並列モデルは、大きく次の 3 つのモデルに分類されると思われる。

- 1) 単純 GA を並列化したモデル
- 2) 分散 GA を並列化したモデル
- 3) 近傍 GA を並列化したモデル

これらに関して、並列化する以前の解の探索能力に関して比較した研究は多い。しかし、その並列モデルの比較はあまりなされていないといえる。加えて、各並列モデルの粒度は、ある特定のアーキテクチャを必要としていたり、解探索に大きく影響を与えかねないものであったりする。

そこで本研究では、2) の分散 GA を改良した 2

個体分散遺伝的アルゴリズム (Dual Individuals Distributed GA : Dual GA) の並列化について検討を行う。Dual GA は、他の GA に比べて解の探索能力が優れているばかりではなく、容易に並列化を行えるモデルであると考えられる。本論文では 2 章で Dual GA とその並列モデルについて説明を行い、3 章で解の探索能力と並列実行について数値実験を行う。そして 4 章で結論を述べる。

2 2 個体分散遺伝的アルゴリズム

2.1 GA と Dual GA

GA は工学的なアルゴリズムであり、次のように行われる。²⁾ GA は多点探索手法の 1 つであり、各探索点は個体 (Individual) と呼ばれる。これらの個体群に対して、個体間の交叉や突然変異といった遺伝的操作が適用される。そうして生まれた子個体群と親個体群の中から、環境への適合度の高い個体が確率的に選択される。これらの一連の操作が行われる周期を世代数と呼ぶ。これらの操作が何世代も繰り返されることにより、良い個体が増加し、やがて最適解へと近づいていく。GA の個体群は一般に母集団と呼ばれ、単純 GA (Simple GA : SGA) での母集団数は 1 つである。しかし、母集団数を複数個とした GA も存在する。それらは一般に分散 GA (Distributed GA : DGA) と呼ばれる。DGA では、新たに移住と呼ばれる遺伝的操作が適用される。移住は隔世代ごとに適用され、母集団間の個体交換を行う操作である。DGA はこの移住により、全母集団として多様性を維持し、SGA よりも良好な解を見つけることができると報告されている。また DGA の各母集団は一般にサブ母集団、あるいは島と呼ばれる。

2 個体分散遺伝的アルゴリズム (Dual Individuals Distributed Genetic Algorithms : Dual GA) は DGA における各島の個体数を 2 として、多数の島で GA を行うアルゴリズムである。そして遺伝的操作にはそれに応じた改良を加える。各遺伝的操作の改良点を以下に示す。

- 突然変異
突然変異は各個体に対して必ず 1 ビット適用されることとし、一方の個体の突然変異点と 1 ビットずれた部分を他方の突然変異点とする。これにより、島内の 2 個体が全く同一になるのを防いでいる。
- 選択
交叉を行う親個体 2 個体と交叉により生成される子個体 2 個体のうち、親個体の優れた方と子個体の優れた方を 1 個体ずつ選び、次の世代に残す。優

れた親個体を残す操作はエリート保存戦略と呼ばれ、他の GA でもしばしば用いられる。

- 移住
島内の 2 個体のうちどちらかをランダムに選択し、そのコピーを移住先へ送る。移住先では、適合度の低い方の個体を移住してきた個体で置き換える。移住のトポロジは、移住世代ごとにランダムに形成されるリング状である。移住を適用する間隔は基本的には 5 世代とする。

2.2 Dual GA の並列化

本研究では分散メモリ型の並列計算機である PC クラスタを対象として、Dual GA の並列化を考える。PC クラスタはコストパフォーマンスに優れた並列計算機であるが、専用の並列計算機に比べるとネットワーク性能は概して低い。そこで、できる限り通信のオーバーヘッドを減らすことが重要である。

DGA では、一般に 1 つの島に対して 1 つのプロセッサが割り当てられる。しかし Dual GA は前節で説明したように多数の島で GA を行うアルゴリズムである。多くの場合、PC クラスタのプロセッサ数よりも島数の方が多くなるため、1 つのプロセッサに対して複数の島を割り当てることになる。これは移住のモデルを考えた時に問題となる。Dual GA の逐次モデルでは、全島が単一のリングトポロジを構成して移住が行われる。しかし、これをそのまま PC クラスタに実装すると明らかに通信負荷が高くなってしまふ。そこで Dual GA の並列モデルでは、プロセス内の移住とプロセス間の移住を分けて考え、それぞれ独立に適用する。

Dual GA の移住モデルの概念図を図 1 に示す。プロセス内での移住は逐次モデルの移住をそのまま適用する。一方プロセス間の移住では、全プロセスが毎回ランダムにリング状を形成し、そのト

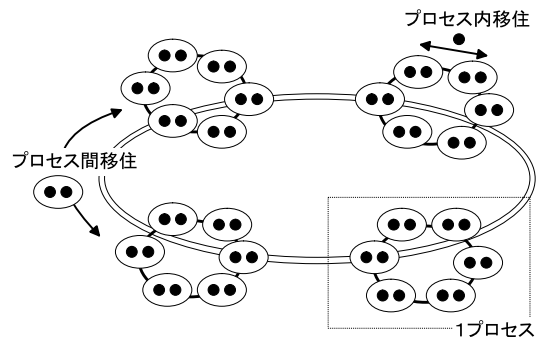


図 1: Dual GA の移住モデル

表 1: テスト関数

	関数名	式	定義域	最適値
F1	Rastrigin	$f = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i))$	$-5.12 \leq x_i \leq 5.12$	0[0,...,0]
F2	Rosenbrock	$f = \sum_{i=1}^{n-1} [100 * (x_i^2 - x_{i+1})^2 + (1 - x_i)^2]$	$-2.048 \leq x_i \leq 2.048$	0[1,...,1]
F3	Ridge	$f = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	$-64 \leq x_i \leq 64$	0[0,...,0]
F4	Griewank	$f = 1 + \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n (\cos(\frac{x_i}{\sqrt{i}}))$	$-512 \leq x_i \leq 512$	0[0,...,0]

ポロジに従って島交換の操作を行う。プロセス間の移住間隔は、基本的にプロセス内の移住間隔の5倍とする。この移住モデルを用いることにより、通信負荷を抑えたまま Dual GA を並列実行できると考えられている。

2.3 Dual GA の利点

Dual GA は以下の3つの点で、優れたアルゴリズムであると考えられる。第1に多様性を維持できるために、SGA や DGA に比べて解の探索能力が高いことである。第2に1つの島の個体数が2であるために、アルゴリズムが簡易化することである。すなわち多くの GA では、交叉する親個体の組み合わせを選び出すために乱数を発生させたり、選択やエリート保存のために個体を適合度値でソートするといった操作が必要になる。Dual GA ではこれらの操作を行う必要がない。第3に、1つの島を単位として並列化の粒度を柔軟に変更できることである。これは、Dual GA が使用可能なプロセッサ数を有効に使い、均質なクラスタだけでなく非均質なクラスタにも対応できる可能性を示している。

3 Dual GA の分散化と並列化の影響

3.1 対象問題と実験環境

本研究では表 1 に示す4つの関数を最小化する問題を用いて、Dual GA の並列モデルを検証する。これら4つの関数は最適化手法の性能を調べるためにしばしば用いられ、テスト関数と呼ばれている。それぞれの関数は表 2 に示すように、異なる形状をしており異なる性質をもっている。Dual GA の各遺伝的操作のパラメータは、論文の中で特に断らない限り表 3 に示すパラメータを用いた。

表 2: テスト関数の性質

	形状	設計変数間の依存関係
F1	多峰性	なし
F2	単峰性	あり
F3	単峰性	あり
F4	多峰性	多少あり

表 3: Dual GA の基本パラメータ

島数	192
プロセス数	任意
コーディング	10bit グレイコード
交叉手法	1点交叉
突然変異率	1 / 遺伝子長
移住トポロジ(内, 外)	リング状
プロセス内の移住間隔	5
プロセス間の移住間隔	内の5倍
プロセス間の移住率	0.1(最低1個体)

計算機環境は16ノードからなるPCクラスタ型の並列計算機である。各ノードは PentiumII 400MHz、メモリ 128MB を搭載し、OS として Linux2.2.12 が動作している。ノード間は 100BASE-TX のネットワークをイーサネット・スイッチを用いて接続している。並列プログラムは gcc (egcs-2.91.61) と MPICH1.1.2 を用いて開発を行った。

3.2 分散化による効果

Dual GA の並列モデルは、逐次モデルと異なり移住の部分がプロセス間移住とプロセス内移住の2段階に分かれている。このように複数のプロセ

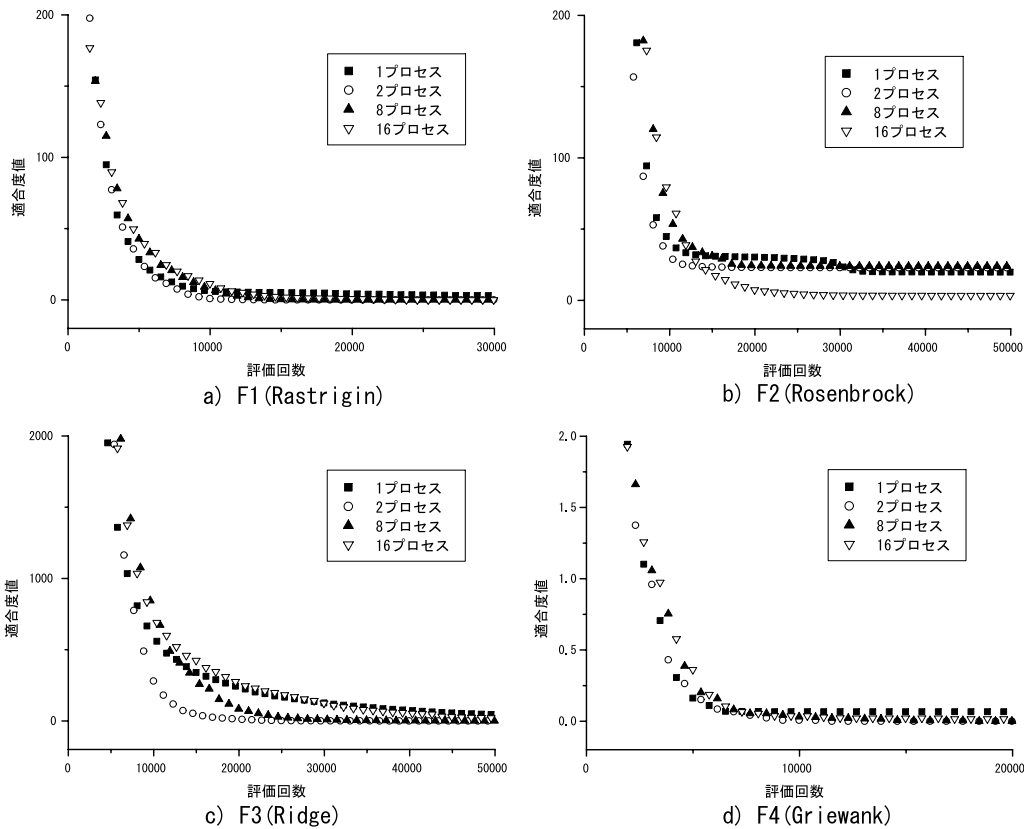


図 2: 並列化による効果

スに分散化することによりモデルが変化し、また得られる解も影響を受けるものと考えられる。そこでモデルの違いにより、Dual GA の解の探索能力がどのような影響を受けるかを調べた。4つの対象問題について数値実験を行った結果が図 2 の通りである。グラフの横軸は各プロセスに散らばる全個体の評価回数である。GA の計算では主に評価計算の部分がホットスポットとなるため、評価回数が計算負荷の指標の 1 つとなる。グラフ中の並列モデルでは、192 島を均等に各プロセスに割り当てている。

図 2 よりいずれの関数においても、Dual GA の並列モデルが逐次モデルよりも良好な結果を示しているのが分かる。そして F2 を除くその他の 3 つの関数では、2 プロセスの並列モデルが最も早く最適解を得ている。これは 2 プロセスの移住トポロジが、適切な多様性の維持を作り出すことができたからであるといえる。一方 F2 の関数は、最適解自体が求まっていない。つまり、いずれの結果も途中で局所解に収束してしまっている。結果として、この場合に最も多様性を維持できるモデル

である 16 プロセスの Dual GA が良い解を得ることができたと考えられる。

3.3 並列実行

3.3.1 移住パラメータによる影響

Dual GA では、移住のパラメータが多様性の維持に大きく関わる。そこで移住のトポロジがリング状であった場合に、移住間隔が与える解の探索能力について調査を行った。実験ではプロセス数を 16 と固定した。プロセス内の移住間隔は 5, 10 世代を設定し、プロセス間の移住間隔は 50, 100 世代をそれぞれ設定した。これらの組み合わせとして計 4 パターンを計測し、結果をまとめたグラフが図 3 である。

図 3 より F1 や F3 の問題に対しては、プロセス間の移住が小さい時に良い解を早く見つけているといえる。一方、F2 や F4 の問題では、プロセス間の移住間隔を多くとった方が良い解を見つけている。プロセス内の移住間隔は、いずれの問題においても短くした方が良い結果を得ている。これらのパラメータ設定は対象問題に依存すると考え

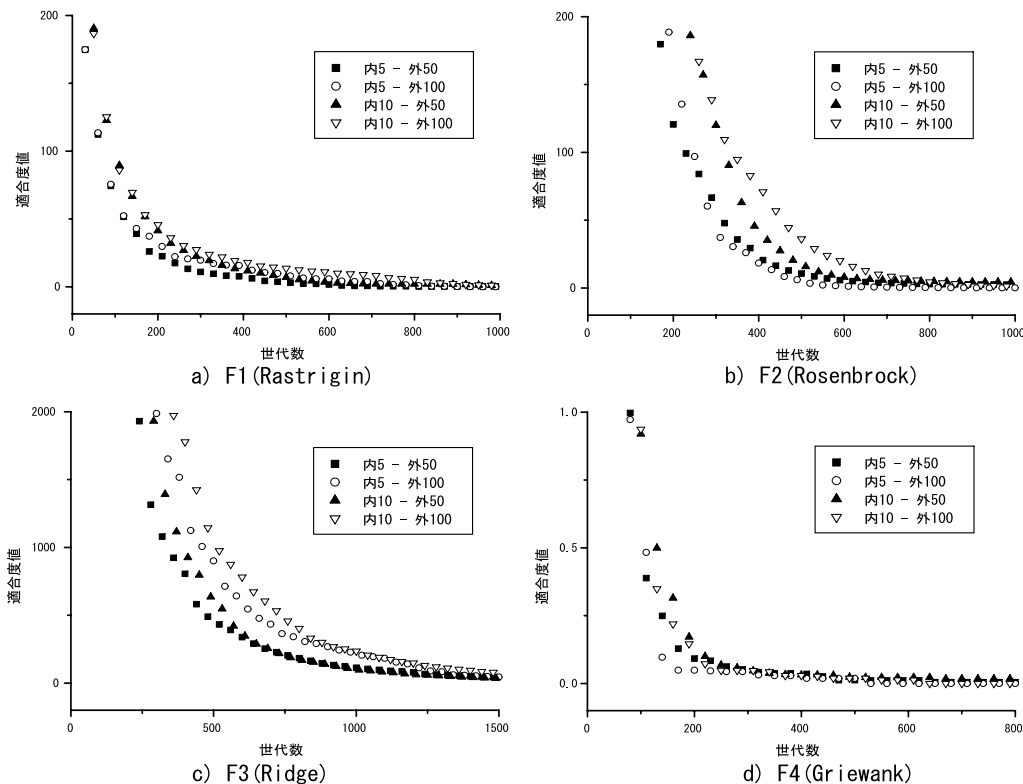


図 3: 移住間隔の影響

られるが、今回着目した関数の性質にあまり関係のない結果となった。

3.3.2 プロセッサ数と島数の割り当て

3章の数値実験より、Dual GA の並列モデルはプロセッサ数によって解の探索能力が少なからず影響を受けることが分かった。一方、プロセッサ数を増やせば、並列度が増してその分の実行時間の短縮が期待できる。そこで Dual GA を実際に並列実行した時の解の探索時間を調べた。ただし表 1 のテスト関数は実問題に比べると、計算負荷が非常に小さい。そこで計算負荷を高めるために、個体の評価計算の部分を 30 回反復計算させて実験を行った。この場合、通信のオーバーヘッドは無視できるほど非常に小さくなる。結果を図 4 に示す。解の探索性能だけに着目した場合には、図 2 の結果であったが、実際に並列実行した結果、どの問題においても 16 プロセスが高速に解を探索することができることが分かった。

しかし、これらの結果は PC クラスタの通信性能に大きく依存する。計算負荷が小さい場合には通信のコストが高つくために、16 プロセスを利

用するとかえって計算が遅くなってしまいう可能性もある。この問題は、事前にシミュレーションを行うことで解決できる。例えば、以下に示す手順でシミュレーションを行うことができる。

- i. Dual GA の逐次モデルを 100 世代行うのに必要な計算時間を S とする。
- ii. Dual GA の並列モデルの移住の部分をシミュレートする。100 世代で 4 回移住を行うとして、それに必要な時間 M を計測する。
- iii. プロセッサ数が N である時、並列モデルを 100 世代行うのに必要な計算時間は、 $P = \frac{S}{N} + M$ となる。
- iv. よって並列化による時間短縮は $\frac{P}{S}$ で求まる。

このシミュレーション結果を図 5 に示す。グラフより、100 世代の逐次モデルの実行時間が 0.01[sec] であった時、4 プロセスの並列モデルが最も実行時間が短いと分かる。これに解の探索性能を重ね合わせることができれば、適切なプロセス数を算出可能となるのである。

4 結言

本研究では Dual GA の並列モデルを PC クラスタに実装し、その評価を行った。Dual GA の並列

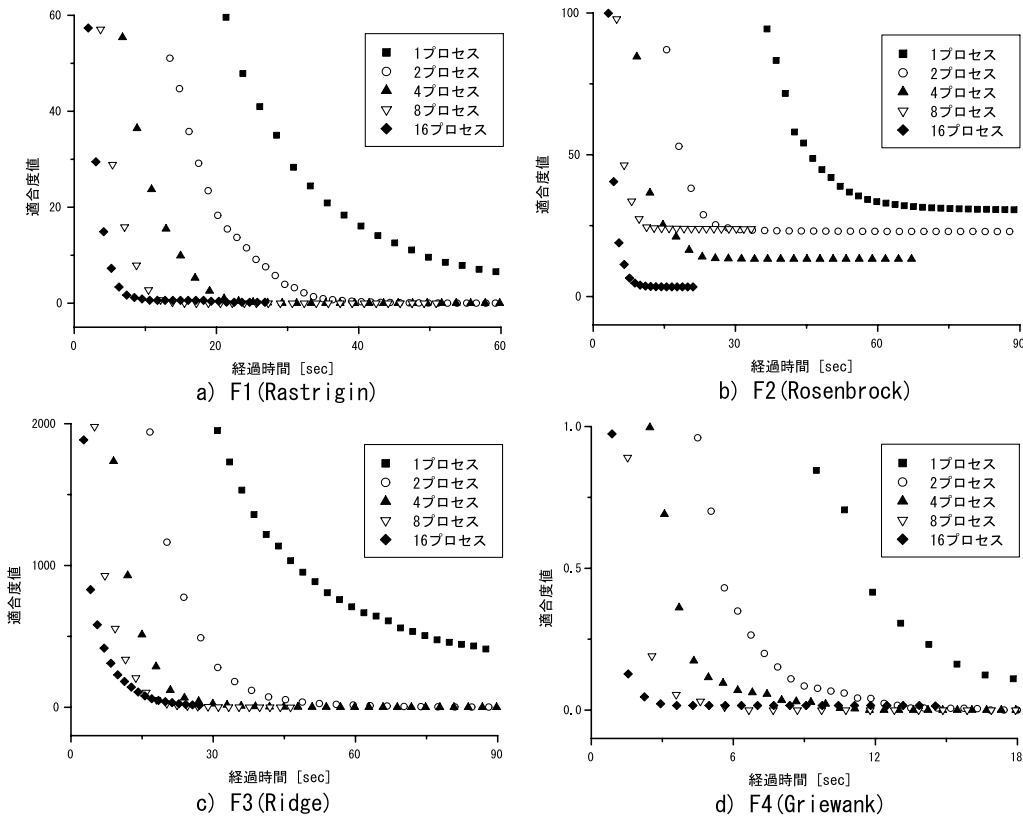


図 4: 実行時間の比較

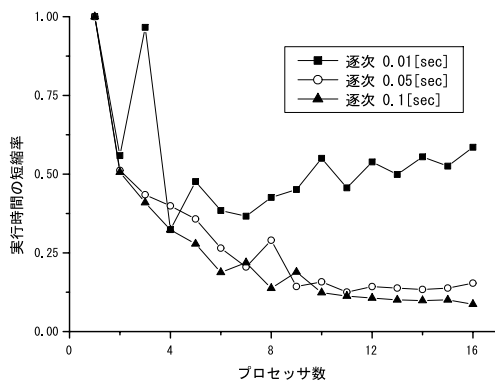


図 5: シミュレーションの例

モデルは、通信負荷を抑えるために逐次モデルとは異なる移住を行っている。しかし、これにより解の探索能力が低下することはなく、今回適用した対象問題においては全て良い結果を示した。ただし、移住に関するパラメータは重要であり、プロセス間およびプロセス内の移住間隔には、問題に依存した適切な値が存在すると思われる。さらに並列モデルの実行においては、島数をプロセッサにどう割り当てるかが重要となる。これは、対象

問題と PC クラスタのネットワーク性能に依存する。つまり最も良い割り当て方は、並列度を上げることにより解の探索能力がどう影響されるかと通信負荷の割合がどうなるかに依存しているのである。ネットワーク性能はベンチマークなどを用いることで計測できるが、求まる解についても考慮することは非常に困難である。この部分は、本研究の今後の課題である。

参考文献

- 1) Enrique Alba, Jose M. Troya, "A Survey of Parallel Distributed Genetic Algorithms", Complexity Vol.4 No.4, John Wiley & Sons, Inc., 1999
- 2) 坂和正敏, 田中雅博, "遺伝的アルゴリズム", 朝倉書店, 1997
- 3) Tomoyuki Hiroyasu, Mitsunori Miki, Masahiro Hamasaki and Yusuke Tanimura, "A New Model of Distributed Genetic Algorithm for Cluster Systems: Dual Individual DGA", Proceedings of CC-TEA, 2000
- 4) D.E. Goldberg, "Genetic algorithms in search", optimization and machine learning, Addison-Wesley, 1989

(追記)

本論文は SWoPP 松山 2000 で発表した論文で
す . 情報処理学会研究報告 (2000-HPC-82)p.161
- p.166 に掲載されました .