

# GA と確率学習オートマトンを用いた動的負荷分散システム

4H-8

山下 貴幸 棟朝 雅晴 高井 昌彰 佐藤 義治  
(北海道大学 工学部)

## 1 はじめに

複数の計算機を LAN で接続して資源の共有を図る分散システムにおいて、計算機間で負荷の分散を行うことにより、応答時間の短縮や資源利用率の改善など、システム性能の向上を図ることができる。この目的のため、種々の負荷分散方式が提案されてきた [1, 2]。

分散制御型の動的負荷分散方式に遺伝的操作を導入した手法として、遺伝的アルゴリズムと確率学習オートマトンによる動的負荷分散 (GeSLA) に関する研究が行われている [1]。この手法においては、タスク転送をどの計算機に対して要求するかを記述した文字列を遺伝的アルゴリズム (Genetic Algorithms, GA) における個体とし、その適合度値の更新に確率学習オートマトン (Stochastic Learning Automata, SLA) による確率的山登り法を適用している。本研究では、UNIX ワークステーションを LAN で接続した分散システム上にこの方式を実装し、実際に生成したタスクを用いた実験により性能評価を行う。

## 2 GeSLA 動的負荷分散

GeSLA 動的負荷分散アルゴリズムは、動的負荷分散方式の一つである Sender-initiated algorithm [2] に、確率学習オートマトンと遺伝的操作を導入することで、効率的なタスク転送要求を送出することを目的としたものである。

$n$  台の計算機からなる分散システムにおいて、 $\{0, 1\}$  からなる長さ  $n$  の文字列を設定し、文字列の  $k$  番目の文字が 1 のときに計算機  $k$  にタスク転送の要求を送出するものとする。この文字列が遺伝的操作の対象となる個体となる。各計算機ごとに複数の個体からなる集団を用意し、それに対して遺伝的操作を適用する。

各計算機では、ユーザからタスクが投入されたときに負荷の観測が行われ、負荷が重いと判断された場合にタスク転送手続きが起動される。負荷状態について

は、タスク待ち行列の長さが 0 のときに負荷が軽く、3 以上の場合に負荷が重いものとした。

まず始めに一つの個体はその適合度値に比例した確率で選択され、その内容にしたがって複数の計算機に対しタスク転送要求が送出される。負荷の軽い計算機が一つでも見い出されたなら要求は成功し、タスクが実際に転送される。全ての計算機から要求が拒絶された場合、そのタスクは投入された計算機で実行される。タスク転送が成功したかどうかに基づいて SLA の確率的山登り法により適合度値が更新され、その後、一定確率で遺伝的操作が集団に対して適用される。

## 3 負荷分散システムの実装

本方式で想定しているタスクは、互いに依存関係がなく独立なものであり、ネットワーク上の各計算機で実行可能なコマンドとする。

本システムは、タスクの管理を行うために各ホストで起動されるサーバと、ユーザがそのサーバに対してタスクを投入するためのクライアントから構成される。サーバはさらに、メッセージ処理を行う「メッセージ処理プロセス」と、タスクの実行を制御する「タスク処理プロセス」からなる (図 1)。

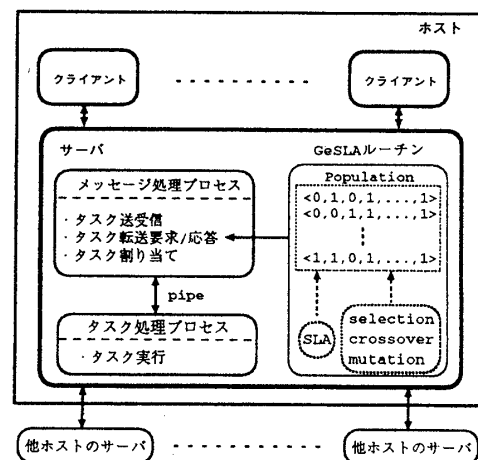


図 1: システムの構成

A Dynamic Load Balancing System using Genetic Algorithms and Stochastic Learning Automata  
Takayuki Yamashita, Masaharu Munetomo, Yoshiaki Takai, and Yoshiharu Sato  
Hokkaido University, Sapporo 060 JAPAN.

サーバ・サーバ間とサーバ・クライアント間の通信はソケットを介したメッセージ通信によって、サーバの各プロセス間の通信はパイプを介したメッセージ通信によって行われる。

ユーザはクライアントを起動することによって、複数のタスクをまとめたバッチタスクという形式でタスクをサーバに対して投入する。サーバはクライアントよりバッチタスクに関するメッセージを受け取ると、それを分解し個々のタスクを生成して負荷分散アルゴリズムを適用する。各ホストに割り当てられたタスクはサーバ内のタスク待ち行列に入り、先頭から順に一つずつ実行される。タスクの生成から終了までの流れを図2に示す。

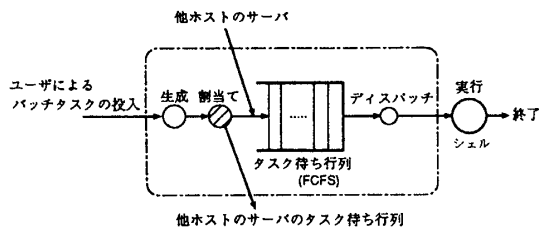


図2: タスクの生成から終了まで

サーバのメッセージ処理プロセスにおいて、タスクが投入されたホストの負荷が重いと判断された時にタスク転送アルゴリズムが起動される。そのタスク転送の成否に基づいて、タスク転送要求の送出先を決定するための文字列からなる集団に対してSLAによる適合度値の更新を行う。また、一定確率で selection, crossover, mutation といった遺伝的操作が集団に対して適用する。

実際のタスク転送は、コマンド名、コマンドの起動オプション、ワークディレクトリなどから構成されるメッセージの送信により行われる。

## 4 性能評価実験

GeSLA 動的負荷分散システムを用いた場合と、負荷分散を行わない場合のタスクの平均応答時間の比較を行うことにより、GeSLA 動的負荷分散システムの有効性を検証する。Sender-initiated algorithm, GAのみを用いた動的負荷分散システムとの比較も行う予定である。

使用したUNIX ネットワークは、SUN SPARCstation 10, SPARCstation classic, SPARCstation 2GS, SPARCstation IPX の計4台をイーサネットで接続した分散システムである。各ホストに、単純な浮動小

数点演算を繰り返すタスクを生成させ、その平均応答時間を測定した。タスクのループ回数を平均  $20 \times 10^6$  回の指数分布、到着間隔を平均  $\lambda$ 秒のポアソン到着とし、各ホスト毎に80個のタスクを生成させた。 $\lambda$ を変化させて実験を行った結果を図3に示す。

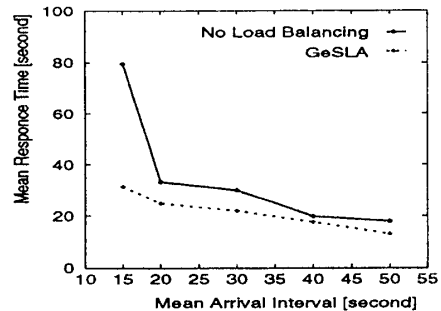


図3: 平均応答時間の比較

負荷分散を行わない場合、同時に2つ以上のタスクが並行に実行されることがある。そのため、実験結果にもあらわれているが、平均到着間隔が短くなるに従い平均応答時間が著しく増加する。

GeSLA 動的負荷分散システムを用いた場合、平均到着間隔の短縮に伴う平均応答時間の増加は、負荷分散を行わない場合と比較して少なくかつ緩やかになっている。これは、GeSLA 動的負荷分散では逐次的なタスクのディスパッチにより、一つのホストにおいて同時に高々一つのタスクしか実行されず、加えてタスク転送により負荷の軽いホストが有効に利用されるためである。

## 5 まとめ

本研究では、遺伝的アルゴリズムと確率学習オートマトンを用いた動的負荷分散アルゴリズムによる分散制御型の動的負荷分散システムを構築し、実験によりその有効性を確認した。

## 参考文献

- [1] 棟朝雅晴, 高井昌彰, 佐藤義治: “確率学習オートマトンと遺伝的アルゴリズムによる動的負荷分散”, 情報処理学会研究報告 (94-AI-93), Vol.94, No.20, pp.95-102, Mar. 1994.
- [2] N.G.Shivaratri, P.Krueger and M.Singhal: “Load distributing for locally distributed systems”, *IEEE COMPUTER*, Vol.25, No.12, pp.33-44, Dec. 1992.