

Grid 環境における大規模ジョブ分配問題の作成
斉藤 宏樹

1 今月の課題

今月の課題を以下に示す .

- Grid 環境における大規模ジョブ分配問題の作成
- MGG に関する文献調査
- 用語集の作成

2 課題の進捗状況

2.1 Grid 環境における大規模ジョブ分配問題の作成

Grid 環境における大規模ジョブ分配問題とは , Grid のようなヘテロな環境において , 大規模なジョブをどう最適に処理するか , その最適なジョブ分配を求める問題である . 以下にその概要を示す .

- 処理性能がそれぞれ異なるプロセッサに対し , 大規模なジョブを複数に分割し割り当てる .
- 通信するプロセッサが増加するために起こる通信遅延や , ジョブを分配するために必要となる分配作業などのオーバーヘッドが発生する .
- 分配するジョブの大きさは異なっていて良い (分配されない CPU が存在しても良い) .

このような概要に基づいて , 以下に示す問題を作成した .

使用するプロセッサの処理能力 P は 20 , 40 , 60 , 80 , 100 と 5 段階に分かれ , それぞれ 100 個ずつ計 500 個利用できるとする (式 (1) 参照) .

$$P_i = \{20, 40, 60, 80, 100\} \quad (1 \leq i \leq 500) \quad (1)$$

そして大規模ジョブの計算量 X を 30,000 とし , 各プロセッサに分配されるジョブの計算量を x_i とすると , 次のような関係式が成り立つ .

$$\sum_{i=1}^{500} x_i = X = 30,000 \quad (2)$$

次に分配におけるオーバーヘッドとして , 通信遅延 (latency) と分配作業による負荷 (load) を考慮する . それぞれは分配数 d に比例する . そしてそれぞれのオーバーヘッドは , 使用するプロセッサの個々の処理時間に関わるものと , 全体の処理時間に関わるものにわかれ

る . 全体の処理時間に関わるオーバーヘッドは , 分配による通信遅延と分配作業の負荷である . 式 (3) , 式 (4) に示す .

$$OVERHEAD = (latency + X \times loadfactor) \times d \quad (3)$$

$$0.0 \leq loadfactor \leq 1.0 \quad (4)$$

個々のプロセッサに関わるオーバーヘッドは , 分配するために発生する処理であり , 式 (5) , 式 (6) のようになる .

$$overhead_i = x_i \times distributecost \times d \quad (5)$$

$$0.0 \leq distributecost \leq 1.0 \quad (6)$$

これらから , 大規模ジョブの処理時間 T を求める式は , 式 (7) のようになる .

$$T = \max \left\{ \frac{x_i + overhead_i}{P_i}, \dots, \frac{x_{500} + overhead_{500}}{P_{500}} \right\} + OVERHEAD \quad (7)$$

この大規模ジョブの処理時間 T を最小にする分配ジョブの計算量を求める . これは , 式 (2) より制約付きの最適化問題である . GA は一般的に制約条件を陽には扱わないので , 実行可能領域を外れた解候補には制約条件を扱うための何らかの処理が必要になる . そのため , 今回においてはペナルティ関数法を適用する .

式 (7) より , $f(X) = T$ とすると , ペナルティ関数法では式 (8) のような関数を定義して , 大規模ジョブにおける処理時間の最適化を行うことができる .

$$\text{minimize } F_\rho(X) = f(X) + \rho \left(\left| \sum_{i=1}^{500} x_i - X \right| \right) \quad (8)$$

2.2 MGG に関する文献調査

現在追加内容を調査中であるが , ほぼ完成している . (<http://mikilab.doshisha.ac.jp/dia/research/person/hiroki/report-mgg.html>)

3 翌月への課題

翌月への課題を以下に示す。

- Grid 環境における大規模ジョブ分配問題の実装と検討
- MGG に関する文献調査
- 用語集の作成