5K-8

グリッドコンピューティングにおける制約条件付きタスクスケジューリング問題へのACO法の適用

Khalid Alqurashi*　　平松 綾子**　　能勢 和夫**

*大阪産業大学大学院工学研究科情報システム工学専攻　**大阪産業大学デザイン工学部情報システム学科

1. はじめに

マシンを複数繋ぐことと、それらを統合的に利用することの間には大きなギャップが存在する。複数のマシンを繋ぐことはグリッドの要件ではあるが、それだけではマシンを有効に活用することは難しい。多数のタスクから構成される大規模なジョブ群に対して、多数のマシンの統合的活用を検討するのが、グリッドコンピューティングにおけるスケジューリング問題である。本研究では、このスケジューリング問題に対し、組み合わせ最適化手法である ACO 法 (Ant Colony Optimization) の適用を試みる。

グリッドコンピューティングにおけるスケジューリング問題に関して、遺伝的アルゴリズムと ACO 法を組み合わせた手法[1]や、ACO 法を用いた手法[2]などが提案されている。しかしながら、これらの手法は､制約条件のない問題を対象としているため、処理する問題が限られる。本研究では、タスク間の先行制約などの制約条件を踏まえたスケジューリングが可能な手法を提案する。

2. 制約条件付きスケジューリング問題

次の前提で、スケジューリング問題を考える。

1. ネットワークには一つのスケジューラと M 個のマシンが接続されている。
2. ネットワークには、J 個ジョブが同時に投入される。
3. ジョブ j は $N(j)$ 個のタスクから構成されている。
4. 各マシンのメモリサイズにより、処理できるタスクに関して制限がある。(容量制約)
5. 各ジョブのタスク間には処理順に関する制約が存在する。(先行制約)

スケジューリング問題は、容量制約と先行制約を満たしながら「ジョブの処理完了時刻が最も早くなるように、全てのタスクについて処理マシンとそのマシンでの処理開始時刻を決めること」となる。

3. ACO法に基づく提案アルゴリズム

3.1 基本的な考え方

解候補の空間をノード空間とし、複数のアリエージェントが、ノード上に蓄積されたフェロモン量をたよりに最適解の探索を行う。解候補を表現するノード空間として、ジョブ内タスクの先行制約を満たす処理順序を決めるノード空間 (処理順ノード空間)、ガントチャートへの全てのタスクの配置順序を決めるノード空間 (配置順ノード空間)、容量制約を考慮して割り当てマシンを決めるノード空間 (割り当てノード空間)の3つのノード空間を定義する。

3.2 処理順ノード空間

処理順ノード空間はジョブ内先行制約を踏まえたジョブ内タスクの処理順を決める。各ジョブの処理順ノード空間は 図1に示すようなノード空間でジョブ j におけるノード (α,β) は、β 番目に割当てられるタスクが α であることを意味する。なお、一度選択されたタスク番号はそれ以降の処理において、選択候補から除外する。また、アリが選択できるのは先行制約を満たすタスク番号のみである。

3.3 配置順ノード空間

各ジョブの先行制約を満たしつつ作業のガントチャートの配置順を決めるノード空間は、図 2 に示すような空間とする。ノード (α,β) は、β 番目に配置されるタスクがジョブ α に属していることを意味する。

Ant Colony Optimization Method for Problem of Task Scheduling with Constraints in Computational Grid
*Khalid Alqurashi　**Ayako Hiramatsu　**Kazuo Nose
*Graduate School, Osaka Sangyo University
**Faculty of Design Technology, Osaka Sangyo University

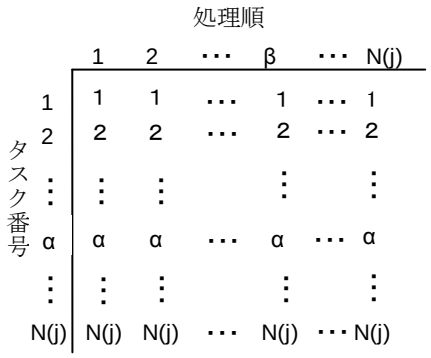

図1 処理順ノード空間

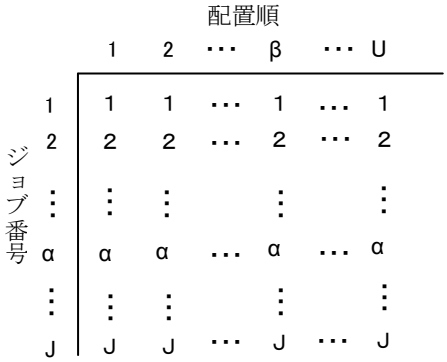

図2 配置順ノード空間

3.4 割り当てノード空間

容量制約を考慮して割り当てマシンを決めるノード空間は、図 3 に示すようなノード空間で、ジョブ j におけるノード (α,β) はタスク β をマシン α に割当てることを意味する。なお、アリが選択できるのは、容量制約を満たすマシン番号のみである。もし、タスクサイズよりマシン α のメモリが小さいなら、タスク β はマシン α に割当てることができないので、ノード (α,β) の初期フェロモン量を 0、それ以外の割当て可能なノードの初期フェロモン量は適当な正の値に設定する。これにより、ノード (α,β) の初期選択確率は 0 となり、以降のサイクルにおいても選択されることはない。

3.5 ガントチャート

上記のノード空間から解候補が得られると、縦軸をマシン番号、横軸を時刻とするガントチャート上に既配置済みのタスクに対して時間的な先行制約を考慮して前詰めでタスクを配置する。これにより解候補を評価できる。

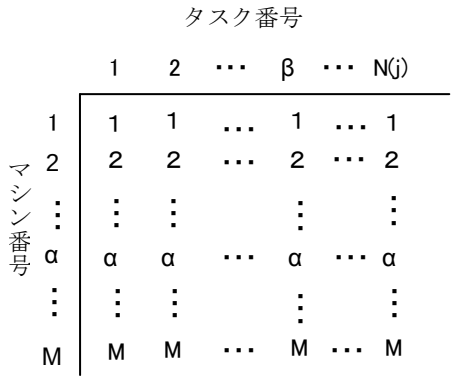

図3 割り当てノード空間

4. 数値実験

実験結果の一例を図4に示す。グラフ中の階層とは先行制約関係のあるタスクのグループ数である。例えば、2階層ではタスクのグループが2つあり、1つ目のグループの処理が終了しなければ、2つ目のグループのタスクの処理ができないことを表す。なお、詳細は当日報告する。

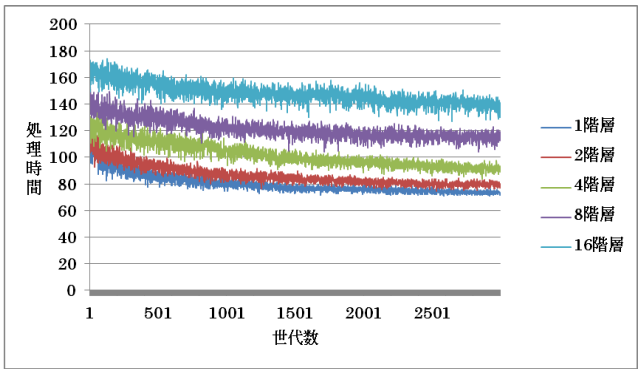

図4 先行制約の階層数と処理時間

5. あとがき

グリッドコンピューティングにおけるスケジューリング問題について、先行制約と容量制約という2つの制約条件を取り扱えるACO法を提案した。

参考文献

[1] Jing Liu, et. al. : “The Research of Ant Colony and Genetic Algorithm in Grid Task Scheduling,” Proc. of 2008 Int. Conf. on Multi-Media and Information Technology, pp. 47-49 (2008)

[2] Yixiong Chen : “Load Balancing in Non-dedicated Grids Using Ant Colony Optimization,” Proc. of Fourth Int. Conf. on Semantics, Knowledge and Grid, pp. 279-285 (2008)