5.3　蒸発率変更の効果

　図5.1-5.9より，期待に反して，すべてのCasesで蒸発率変更の有無は最良解の収束性に影響を与えていないことが判明した．

5.４　アリ数の差

　図5.1-5.9の(a)(b)と(c)(d)を比べると，すべての図でアリ数20に比べて50のほうが良い解を得ている．

5.5　蒸発率の差

　図5.1-5.9の(a)(c)と(b)(d)を比べると，すべての図で蒸発率0.1に比べて0.2のほうが良い解を得ている．ただし，アリ数20，蒸発率0.2の場合（図5.1-5.9の(d)）最悪解が最良解に急激に収束している．ACOのような集団探索においては，解候補はある程度バラツキがあるほうが良いとされており，探索途中ですべて同じ候補に収束することは好ましくない．

　以上より，表5.3の範囲では，アリ数50，蒸発率0.2に設定することが妥当であると考える．

5.6　容量制約の影響

　容量制約の影響を調べるため，Cases 1A- 3A（容量不足率0.0），Cases 1B-3B（容量不足率0.2），1C-3C（容量不足率0.33）について処理完了時間を比較してみると，容量不足率0.2と0.3の収束値は不足率の差ほどの差はない．これは，処理完了時間は必ずしも不足率に比例して増大せず，容量不足を補うようにうまくマシンのスケジュールができることを意味していると考えられる．

5.7　先行制約の影響

　先行制約の影響を調べるため，Cases 1A-1C（制約層数2），Cases 2A-2C（制約層数4），Cases 3A-3C（制約層数6）について処理完了時間を比較した．先行制約の層数が増えるにしたがって処理完了時間も増えること，および，収束までの世代数も増えることが分かった．問題が複雑化するにしたがって，世代数も大きく設定する必要がある．

5.8 実験条件2

以下のように問題を設定し、数値実験を行った。

・ 表5.1を参考に、階層はCase3(階層番号6)

* マシン台数　5台 ,
* 1ジョブの含むタスク数12個

・ ジョブ数 a=3個 b=6個 c=9個

・ タスク総数　a=36個 b=72個 c=108

* 容量制約率　CasesB (不足率20%)

・ アリ数　50匹

* 世代数　2000
* 蒸発率0.2
* 初期フェロモン 100

6．まとめ

グリッドコンピューティングにおけるジョブの分割タスクに関する制約条件付スケジューリング問題について，ACOによる解法を検討し，蒸発率を変更するACOを提案した．すなわち

①ジョブ内タスクの先行制約を考慮した処理順を決める処理順ノード空間を導入する．

②ガントチャートへのタスクの配置順を決める配置順ノード空間を導入する．

③マシンへのタスクの割当てを決める割当てノード空間を導入する．

④先行制約は処理順ノード空間ならびに配置順ノード空間のノード選択確率に反映させる．マシンの容量制約は割当てノード空間の選択確率に反映させる．

⑤割当てノード空間ではすべてのノード上のフェロモン蓄積量を蒸発させる．処理順ノード空間および配置順ノード空間では制約上選択される可能性のないノードに対しては蓄積フェロモンの蒸発を行わず，選択される可能性のあるノードに対してのみ蒸発を行う．

⑥ガントチャートにより，先行制約の時間的制約を満たすセミアクティブスケジュールを作成する．

という手順である．

　数値実験の結果，以下のことが明らかとなった．

①期待に反して蒸発率変更の効果は認められなかった．

②表5.3の範囲では，アリ数50，蒸発率0.2と設定することが妥当であると考える．

③容量制約の影響に関しては，収束値に不足率の差ほどの差はない．これは，処理完了時間は必ずしも不足率に比例して増大せず，容量不足を補うようにマシンのスケジュールができることを意味していると考えられる．

④先行制約の層数が増えるにしたがって処理完了時間も増えること，および，収束までの世代数も増えることが分かった．問題が複雑化するにしたがって，世代数も大きく設定する必要がある．

　残された課題は，蒸発率の変更がなぜ効果的でないのかについて深く掘り下げて検討することである．

参考文献

[1]日本アイ・ビー・エム・システムズ・エンジニアリング株式会社**,** **“**グリッド・コンピューティングとは何か：Globus Toolkitではじめるグリッドの基礎”, ソフトバンクパブリッシング, 2004年.

[2]溝口文雄**, “グリッドコンピューティング―情報処理の新しい基盤技術**”**, 岩波書店, 2005年.**

[3]Jing Liu, Li Chen, Yuqing Dun, Lingmin Liu, and Ganggang Dong**, “**The Research of Ant Colony and Genetic Algorithm in Grid Task Scheduling”, in Proc. of 2008 Int. Conf. on Multi-Media and Information Technology, pp. 47-49, 2008.

[4]Yixiong Chen**, “**Load Balancing in Non-dedicated Grids Using Ant Colony Optimi- zation”, in Proc. of Fourth Int. Conf. on Semantics, Knowledge and Grid, pp. 279-285, 2008.

[5]大内東, 山本雅人, 川村秀憲**, “**生命複雑系からの計算パラダイム”, 森北出版, 2003年.

[6]Marco Dorigo and Thomas Stützle**, “**Ant Colony Optimization”, The MIT Press, 2004.

[7]久保幹雄，J. P. ペドロソ，“メタヒューリスティクスの数理”，共立出版，2009年.

[8]関根智明監訳**, “**スケジューリングの理論”, 日刊工業新聞社, 1971年.