HRC 研究報告会（2010 年 9 月 2 日）

光情報処理システムグループ

# テーマ 2 オンラインショッピング利用者の行動分析
# テーマ 3 グリッド環境におけるタスクスケジューリング

工学部 情報システム工学科　平松 綾子
工学部 電子情報通信工学科 山崎 高弘

本テーマは、近年身近になったインターネットに対し、光デバイスを利用する高速通信が与える影響を検討する。特に、データ通信容量の制約の少ない状況下でのオンラインショッピングの利用に着眼し、消費者の行動を理解することで、光デバイスの社会的影響を明らかにすることを目的とする。本稿では、研究報告として、自由記述アンケートの分類のための同意語特定手法、およびグリッド環境におけるタスクスケジューリングについて報告する。

## 限定的同意語を用いたアンケート自由回答分類

This paper proposes a method of similarity calculation to extract restrictive synonyms for classifying open answer questionnaire data. A pair of words that are not synonyms may have similar meaning on the answer of particular question. We call such a pair of words “restricted synonym”. In this paper, we use the pattern of co-occurrence of words in answered text data, without using any dictionaries. The proposed method considers whether two words are used with the same condition in the other answer. Comparison of classification by human shows the effect of proposed restricted synonyms for classifying open answer questionnaire data.

### 1. はじめに

インターネット上でのアンケートの実施により大量のテキストデータが得られる. このような大量のテキストデータの分析には多大な労力が必要となるため, テキストマイニング技術を用いたアンケート分析[1][2]が行われている. その1つに自由回答文の自動分類がある. 本研究では限定的同意語を利用してアンケート結果の自動分類手法を提案する.

### 2. 限定的同意語とは

自由記述形式のアンケートでは, 同じ意味を指す内容でも言い回しによって違う言葉で表現されることがある. それらに対応するために限定的同意語[3]を定義する. 例えば, 「沢庵を買う理由は」という質問に対して「自分では漬けられないから」と「家では漬けられないから」という回答は同じ意味と捉えるべきであるが「家」と「自分」は一般的な辞書では同義とはみなすことができない. このように一定の条件下において同義と見なした方がよい言葉を限定的同意語と呼ぶ.

### 3. 分類手法の概要

自動分類のおおまかな手順を以下に示す

**Step1** 前処理として得られたアンケート結果を形態素解析により単語に分割する
**Step2** 単語間の共起度を算出する
**Step3** 共起度から限定的同意語の候補を選ぶ
**Step4** 同意語の候補をアンケート回答にフィードバックさせ検証し, 限定的同意語を抽出する
**Step5** 限定的同意語を含めた単語の一致により回答のグループ分けを行う

以下では, Step3 の限定的同意語の候補選出のための共起度に基づく類似度計算方法について説明する.

### 4. 共起度

単語 $a$ と $b$ の共起度 $R(a,b)$ とは $a$ と $b$ が同一文書内で使用される度合を示すものである. 同意語候補の選出のための, 共起度計算式として以下の2種類を考える. 共起件数は, 単語 $a$ と $b$ が同時に使用された文書数を示す. Jaccard 係数は単語 $a$ と $b$ それぞれの使用回数をふまえた共起度数である.

共起件数: $R(a,b) = hit(a,b)$　　Jaccard 係数: $R(a,b) = \dfrac{hit(a,b)}{hit(a) + hit(b) - hit(a,b)}$

表 1 に共起件数の表を示す. これらの共起度数を各名詞と名詞以外, 各動詞と動詞以外, 各形容動詞と形容動詞以外についてそれぞれ計算して求める.

表1：共起件数の例

| 名詞以外＼名詞 | ご飯 | 自分 | 歯ごたえ | 味 | 漬物 | 沢庵 | 好き | 大根 | おかず | 家 | … |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 食べる | 63 | 15 | 24 | 17 | 17 | 36 | 7 | 16 | 18 | 3 | |
| おいしい | 45 | 16 | 34 | 16 | 16 | 22 | 6 | 24 | 8 | 3 | |
| よい | 58 | 0 | 47 | 15 | 20 | 10 | 2 | 23 | 13 | 0 | |
| 合う | 86 | 4 | 23 | 12 | 15 | 8 | 3 | 14 | 6 | 0 | |
| 漬ける{否定} | 2 | 64 | 0 | 2 | 3 | 5 | 2 | 0 | 2 | 21 | |
| 漬ける | 6 | 39 | 2 | 12 | 3 | 11 | 3 | 14 | 0 | 16 | |
| 食べる{否定} | 5 | 5 | 0 | 1 | 3 | 3 | 1 | 0 | 1 | 3 | |
| ・ | | | | | | | | | | | |
| ・ | | | | | | | | | | | |

## 5. 類似度の計算式

上記で求めた共起度により，表に示す列方向すべての語の組み合わせで類似度を式(1)のとおり計算する．閾値 $\alpha$ より小さければ限定的同意語の候補とする． ここで，$S(W_i, W_j)$ は類似度， $W_i$ は $i$ 番目の単語（同一品詞），$A_k$ は $k$ 番目の $W_i$ の品詞以外の単語 $(k = 1, \cdots, n)$，$m_i$ は $W_i$ と共起している単語の種類数，品詞は名詞，動詞，形容動詞とする．

$$S(W_i, W_j) = \sum_{k=1}^{n} \left| \mathrm{T_{ik}} - \mathrm{T_{jk}} \right| \qquad (1)$$

ただし $$T_{ik} = \left( \frac{R(W_i, A_k)}{hit(W_i)} \right) \times Log\left( \frac{n}{m_i} \right)$$

## 6. 実験

市販の沢庵漬商品の購入理由 600 件の人手による分類結果と自動分類結果による比較を行った．対象データの特徴を調べるためにアンケート分析者と被験者との人手分類の比較をした．アンケート分析者の結果を基準とした結果を表 2 に示す．

表 2：アンケート分析者に対する素人の人手分類の比較

| | A | B | C |
|---|---|---|---|
| グループ数 | 19 | 12 | 13 |
| 適合率 | 0.84 | 0.62 | 0.54 |
| 再現率 | 0.69 | 0.73 | 0.68 |
| F 値 | 0.75 | 0.67 | 0.60 |

適合率：正しく分類されているかどうか
再現率：余計な意見を含めていないかどうか
F値：適合率と再現率の調和平均

この結果人によっても表のようにばらつきがあることが分かる． 共起件数×提案類似度， Jaccard 係数×提案類似度の2種類の組み合わせとアンケート分析者との分類結果を比較した．比較した結果を図 1，図 2 に示す．

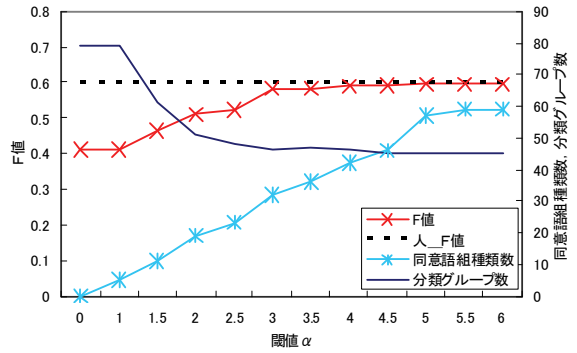



図1：共起件数×提案類似度の分類結果

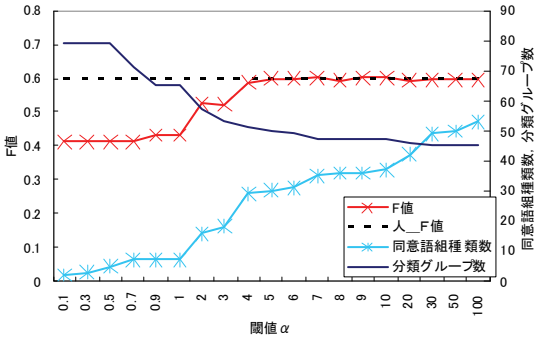



図2： Jaccard 係数×提案類似度の分類結果

比較した結果どのグラフも限定的同意語を使うことで F 値が上昇した．α の設定が容易なことから，共起件数×提案類似度の組み合わせが適切である．本手法の結果は全体的に人手分類にはおとるものの本来ばらつきがあるため，人手分類結果と同等であるといえる．

限定的同意語が分類に与える影響を見るために，共起件数×提案類似度の式で語数別再現率の変化を調べた．その結果を図3に示す．

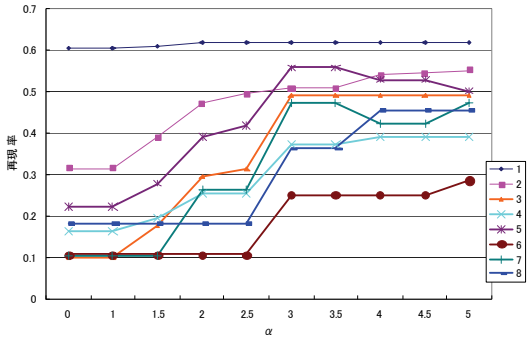



図3：共起件数×提案類似度の語数別再現率

結果から，３語のテキストに関して最も効果的であることがわかる．語数が増えるにつれ，分類が難しくなるため，再現率が低いままである場合もあるが，部分的には効果があるものと考えられる．

**7．まとめと今後の課題**

2語で成り立っているテキストを限定的同意語の対象から外しているため，同一グループ化する事ができなかったので改良の必要がある．今後，人手によって限定的同意語を設定して分類結果がどのようになるのか．また，別のアンケートでも実験する必要がある．

参考文献

[1]K.Yamanishi and H.Li: “Mining Open Answers in Questionnaire Data,” IEEE Intelligent Systems, Vol.17, No.5, pp.58-63(2002).
[2]平松綾子，田村慎吾，大礒洋明，薦田憲久：“非文法的自由回答形式アンケートデータに対応した非定型回答抽出支援方式,” 電気学会 C 部門論文誌, Vol.125, No.7, pp.1153-1159(2005).
[3]国定美佐代，平松綾子，能勢和夫: “自由記述アンケート分類のための限定的同意語特定手法,” 第 50 回自動制御連合講演会, 514(in CD-ROM)(2007)

# グリッド環境下における処理順制約を考慮したタスクスケジューリングの最適化

This paper proposes an optimization method for a task scheduling problem in grid environments. On this task scheduling problem, a job consists of many tasks which have various sizes. When these tasks are assigned to distributed calculation resources that have each different performance, various limitations or restrictions should be considered. In this paper, this problem is treated as a combinatorial optimization including restrictions and the ant colony optimization method is applied.

**1．はじめに**

近年、注目されているグリッド・コンピューティングでは、大規模な計算ジョブを構成する複数のタスクをどのコンピュータ（計算リソース）に割り当てるのかというスケジューリングが重要な問題となる．従来、グリッド環境下での様々なスケジューリング手法が考案されている[1][2]．しかしながら、これらの手法は、タスク間に連携性のない独立したタスクであることが前提とされている．そのため、タスク間に連携性のある場合には、連携性のあるタスクを大きな1つのタスクとして扱い、分散した計算リソースを有効活用することができない．　本稿では、タスク間に連携性のあるジョブに関するスケジューリング問題を対象とする．すなわち、リソース間に能力差等がある場合、および、タスク間の処理順に先行制約のある場合のタスクスケジューリング問題を制約条件付き組み合わせ最適化問題としてとらえ、アントコロニー最適化法（Ant Colony Optimization：ACO 法）による解法を提案する．

**2．想定するグリッド環境**

以下のようなグリッド環境を考える．

- スケジューリングは、マスタと呼ばれる管理マシンが行う
- マスタは、スケジューリングだけでなく、各計算リソースの管理も行う
- 各リソースは、異種環境（処理能力、メモリ容量が異なる）とする
- いくつかのタスク間には、処理順に関する先行制約があるものとする

**3. 問題の記述**

計算リソースを $m_1, m_2, \cdots, m_M$ 、ジョブを構成するタスクを $r_1, r_2, \cdots, r_N$ とする． リソース $m_j$ の処理速度を $v_j$ 、タスク $r_i$ の処理量を $q_i$ とする。このときタスク $r_i$ がリソース $m_j$ で処理されるときの処理時間は $t_{ij} = q_i / v_j$ となる. リソース $m_j$ のメモリ容量を $c_j$ 、タスク $r_i$ の処理に必要なメモリ容量を $s_i$ とするとき、もし $s_i > c_j$ なら、タスク $r_i$ をリソース $m_j$ で処理することはできない。したがって $s_i \le c_j$ でなければならない. リソース $m_j$ に割り当てられたタスクを $r_k (k = 1,2,\cdots, K_j)$ とすると、リソース $m_j$ が割り当てられた $K_j$ 個のタスクを処理するのに要する時間は $\rho_j = \sum_{k=1}^{K_j} (t_{kj})$ となる。したがって、ジョブの処理完了時刻 $F$ は、全てのリソースが処理を完了した時刻 $F = \max_{i \in M} \{\rho_i\}$ となる.

本研究でのタスクスケジューリング問題とは、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 $F$ が最も早くなるように、全てのタスクについて処理リソースとそのリソースでの処理開始時刻を決めること」となる.

**4. 最適化手法**

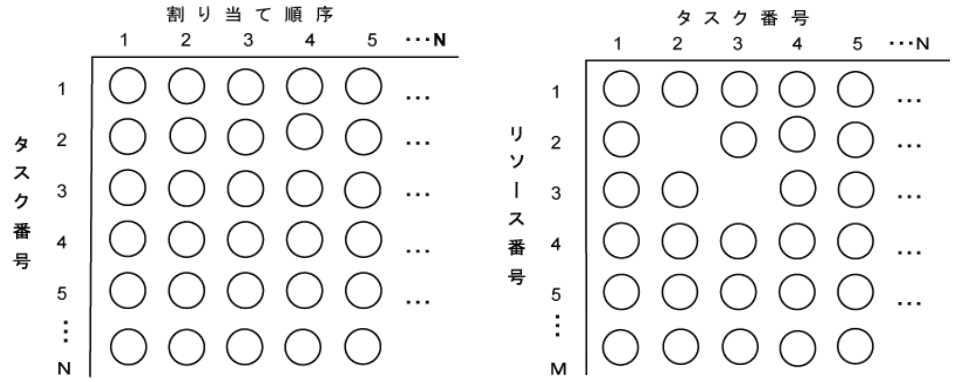
4.1 最適化手法の概要

本稿では、タスクスケジューリングの最適化手法として、組み合わせ最適化手法の一つであるACO法の適用を検討する． アリがフェロモンに基づいて餌を探索する行動を模した多点探索法である. 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.

タスクスケジューリング問題に ACO 法を適用するときの基本的な考え方は次のとおりである. まず、解候補を表現するノード空間として、タスクの計算リソースへの割り当て順を決めるノード空間（順序ノード空間）と割り当てリソースを決めるノード空間（割り当てノード空間）を準備する. ノード選択の際に、順序ノード空間ではタスク間の先行制約を考慮し、割り当てノード空間ではタスクの容量制約を考慮するものとする. そして、ガントチャート上に時間的先行制約を考慮してタスクを前詰め配置し、各タスクの処理開始時刻を得る.

4.2 ノード空間

本手法で用いる解候補のノード空間を図1に示す． タスクのリソースへの割り当て順序を決めるノード空間は、図1の(a)に示すようなノード空間で、ノード $(i, j)$ は、 $j$ 番目に割り当てられるタスクが $r_i$ であることを意味する. したがって、順序ノード空間は、一般的には、 $N \times N$ の正方形の空間となる. 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. これにより、アリがノード区間を左から右へ移動することにより、タスクの割り当て順序が決まる.




（a）割り当て順序　　(b)割り当てリソース

図1 ノード空間

タスクの割り当てリソースを決めるノード空間は、図1の(b)に示すようなノード空間で、ノード $(i, j)$ は、タスク $r_i$ をリソース $m_j$ に割り当てることを意味する. したがって、割り当てノード空間は、一般的には、$M \times N$ の長方形の空間となる. なお、アリが選択できるのは、容量制約を満たすリソース番号のみである. これにより、アリがノード空間を左から右へ移動することにより、各タスクの割り当てリソースがタスク番号順に決まる.

**5. 実験結果**

リソース 20 台、タスク 200 または 2000 個、アリ数 20 匹、世代数 3000 世代、制約率 0%～40%という条件で数値実験を行った. 制約率とは、容量制約によって割り当てることができないタスクとリソースの組み合わせの割合である. 行った数値実験の結果を図 2、図 3 に示す. 制約率が大きいほど、全タスクの処理時間は増加する. また、問題が複雑になるため解の探索に要する世代数も増加すると考えられる.

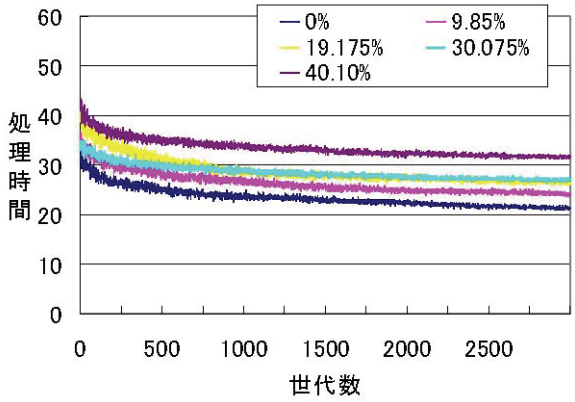



図 2 タスク数 200 のときの制約率の比較

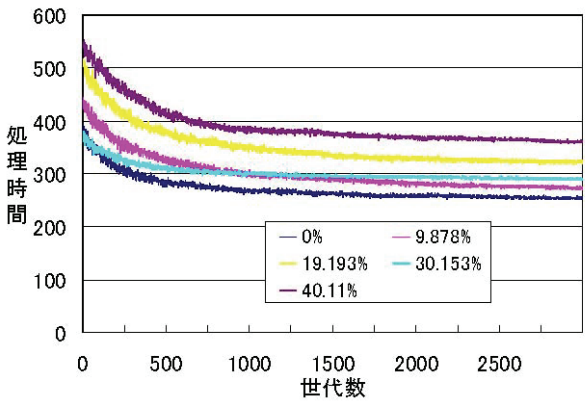



図 3 タスク数 2000 のときの制約率の比較

**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きスケジューリング問題について、ACO 法による解法を提案した. 本手法では、順序ノード空間により、先行制約を満たすタスク順を決め、割り当てノード空間により、容量制約を満たす計算リソースへのタスクの割り当てを決めることで、タスク連携性を考慮したスケジューリングを実現する. メモリの容量制約を考慮したスケジューリングが可能であることを確認した. 今後、連携性を踏まえた問題に対する数値実験を行う.

**参考文献**

[1] Jing Liu, Li Chen, Yuqing Dun, Lingmin Liu, and Ganggang Dong：“The Research of Ant Colony and Genetic Algorithm in Grid Task Scheduling,” Proc. of 2008 Int. Conf. on MultiMedia 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)