数理解析研究所講究録 1426 巻 2005 年 78-83

モバイルエージェント実行計画問題について
On the Mobile Agent Allocation Problem

佐々木 淳 † 宮田 敬三 ‡ 櫟 粛之 † 増山 繁 ‡
† 日本電信電話株式会社 NTT コミュニケーション科学基礎研究所
‡ 豊橋技術科学大学 知識情報工学系

Atsushi SASAKI† Keizo MIYATA‡ Tadashi ARARGI† Sigeru MASUYAMA‡
†NTT Communication Science Laboratories, NTT Corporation
‡Department of Knowledge-Based Information Engineering, Toyohashi University of Technology

概要 モバイルエージェント実行計画問題を定義し，各エージェントの通信するエージェント数を 2 に制限した場合ですら強 NP 完全であることを証明した．また，NP 完全問題の範囲や，近似アルゴリズムについての検討を行った．

キーワード 強 NP 完全性，モバイルエージェント，通信コスト，部分グラフ同型問題，負荷分散

1 はじめに

マルチエージェントシステム [11] は，特に人工知能の分野で精力的に研究されてきた．エージェントとはプロセスなどのような動作主体 (computational entity) のことである．モバイルエージェントとは任意のホストに移動可能なエージェントのことで，分散環境におけるプロセスマイグレーションの概念とほぼ同じである．マルチエージェントシステムにおけるモバイルエージェントの移動性に関する研究 [2] は，通信コストの削減を目的としている．しかし，あるホストに膨大な数のエージェントが集中すればそのホストは過負荷となることより，CPU コストと通信コストの両方のバランスをとるようにホストにエージェントを配置することによって，効率的な実行を目指すことが必要となる．通常，高い CPU コストを最小にするように設計されたこの種の配置最適化は，マルチプロセッサ上でのタスクスケジューリングの形式で表される．よって，モバイルエージェントをタスクに対応させ，ホストをプロセッサに対応させることにより，モバイルエージェントの負荷分散を達成することを，マルチプロセッサにおけるタスクスケジューリングの拡張と見なすことができる．しかし，従来のタスクスケジューリング問題では，モバイルエージェントの相互通信と移動について考慮することは出来ない．そこで本稿にて，新たにこのような問題をモバイルエージェント実行計画問題として定義して，その計算複雑度を解析する．

マルチプロセッサ上のタスクスケジューリングは 30 年以上の間研究されている．従来の研究では，CPU コストのバランスをとること [9] や，タスク間の関係 (半順序関係など) [10] に主に着目していた．また従来の研究により，タスクが preemptive でなければ問題はマルチプロセッサスケジューリング [4] となり，強 NP 完全となることが知られている．なおタスクが preemptive の場合は多項式時間で最適なスケジュールを得ることができる [7].

その一方で，モバイルエージェントシステム上での，システム全体の観点からの負荷分散の研究は全くされてこなかった．この理由はエージェントが自律的存在であるためだと推察される．エージェントの移動の最適化に関して従来からいくつかの研究がなされているが，それらはすべて個々のエージェント移動に焦点をあてて研究したもので，システム全体の性能に関する研究ではない．例えば，データのリモートアクセス，メッセージ交換，プロセスマイグレーションの性能比較 [6], 一連のタスクのためのエージェント移動スケジュール [5], タスクとエージェントの旅程 (itinerary) を考慮したモバイルエージェントの選択 [8] などである．これらはモバイルエージェントの相互通信を考えずに，エージェントとホストとの相互通信のみを考え，エージェントの移動スケジュールを最適化しようとしたものである．システム管理エージェントに関してはそのような定式化は適切であり有用であるが，実際の多く

のエージェントシステム（例えば，情報交換や契約など）ではエージェントの相互通信を必要とする．もちろん，エージェント間での通信に関する研究（例えば，契約ネット [11] など）は多く行われているが，これらの研究は負荷分散に焦点を合わせていない．

システム全体の性能向上を目指すのは，個々のエージェント利益を平均的に良く保つと予想できることより妥当である．そこで，システム全体の仕事の完了時間を最小にすることを目的とする，モバイルエージェントの相互通信に関する問題を定義する．

モバイルエージェントは preemptive タスクと見なすことができる．よって，エージェント間に通信が全くないなら，多項式時間で最適な配置を得ることができる．しかし，エージェントが互いに通信するとき，これを得ることができるかどうかは分かっていない．本稿では，エージェントが互いに通信する場合には，時間最適な配置を求めることは強 NP 完全であることを証明する．具体的には，CPU コストと，実行途中のエージェントの移動の考慮を不要にし，さらに，各エージェントの通信相手がちょうど二つのエージェントになるように制限した問題が強 NP 完全であることを証明する．通常，エージェントは限られた数（$O(1)$）のエージェントとのみ通信するため，この制限は現実のモバイルエージェントシステムの環境を反映している．本稿の成果は，エージェントが互いに通信する必要があるほぼすべての実際の状況に対し，最適なエージェントの配置を与えるどのような効率的なアルゴリズムも無いであろうことを示したことになる．よって，そのような状況では，適切なエージェント実行計画を得るには近似解法が必要となる．そこで，近似アルゴリズムの開発可能性と NP 完全性の範囲について検討した．

次の章でモバイルエージェント実行計画問題を定義する．第 3 章では，各エージェントが通信するエージェント数が 2 に制限されても強 NP 完全であることを示す．第 4 章では，各エージェントが通信するエージェントの数が 1 に制限された場合の計算複雑性や，近似アルゴリズムの開発可能性についての検討結果を述べる．

2 問題

第 1 章で述べたように，モバイルエージェントは相互通信がある preemptive タスクと考えられる．モバイルエージェントは，任意のホストで実行することができる．また，いつでも任意のホストに移動することができ，各自の仕事を実行するために他のエージェントとの通信を行う．すなわち，CPU コスト，通信コスト，および移動コストの三つのコストを考える必要がある．これらの各コストは時間によって表され，各エージェントの実行完了時間はこれらの三つのコストの合計とする．モバイルエージェント実行計画問題の目的は，一番最後に終了するエージェントの実行完了時間を最小にすることである．モバイルエージェント実行計画問題の入力を，以下のように定義する．

問題の入力：ホストグラフとして連結グラフ $G = (H, L, C, S)$，エージェントグラフとして完全グラフ $G_A = (A, E, C_A, W_A, B)$，エージェントの初期位置として $\boldsymbol{x}^0$，納期として $D \in Z^+$ とする．ここで，H はホストの集合を表し，$|H| = n$ とする．L をホスト間のリンクの集合とし，$C = \{c_{ij} | c_{ij}$ はリンク l_{ij} の通信速度の逆数$\}$ とする（$|C| = |L|$）．$S = \{s_i | s_i$ はホスト $h_i \in H$ の CPU 処理能力の逆数$\}$ とする（$|H| = |S| = n$）．A を m 個のエージェントの集合とし，E を互いに通信するエージェント対の集合とする．$C_A = \{c^a_{ij} | c^a_{ij}$ はエージェント a_i と a_j の通信量$\}$ とする（$|E| = |C_A|$）．$W_A = \{w_i | w_i$ は a_i の負荷$\}$ とし，$B = \{b_i | b_i$ は a_i のサイズ$\}$ とする（$|A| = |W_A| = |B| = m$）．$\boldsymbol{x}^0 = \{x^0_i | a_i$ の初期位置$\}$ とする．なお，$s_i \in S$ は h_i が 1 単位の仕事を終了させるのに必要な時間，$w_i \in W_A$ は a_i が演算しなければならない仕事量を表している．

従って，エージェント a_i が他のエージェントと通信しない場合，a_i がホスト h_j によって実行され，他のエージェントが全く h_j に存在しないなら，h_j は所要時間 $w_i s_j$ でエージェント a_i を実行する．以下で，CPU コスト，通信コスト，移動コストについての詳細を述べる．

CPU コスト：エージェントが CPU 資源を使用する時間のことで，S，W_A，およびエージェントが位置するホストによって計算される．従来から議論されている preemptive タスクのスケジューリングなどでは，各プロセッサは同時にひとつのタスクのみ実行できるという仮定を持つ．しかし本問題では，同一ホスト上に k 個のエージェントが存在するならば，各エージェントはそのホストの CPU 資源を $1/k$ ずつ使用して同時に実行される．

通信コスト：エージェントがリンクを通して他のエージェントと通信するのに必要な時間のことで，C_A，C によって計算される．なお，エージェントの通信相手の存在するホストは特定可能とする．多くの現実のモバイルエージェントシステムでは，通信は与えられた時間内に実行される

ことが要求される．しかしながら，ここでは簡単化のため，エージェントはその実行時間内はお互いに通信できると仮定する．

移動コスト：エージェントがホストから別のホストまで移動するのに必要な時間のことで，B，Cおよび移動元と移動先のホストによって計算される．

エージェント a_i の実行時間 f_i の計算に関する例を以下に示す．

例 1：以下のようなシナリオについて考える：a_i の初期位置 x_i^0 をホスト h_{i_0} とし，a_i は h_{i_0} にて何も演算することなしに，エージェント a_j がすでに配置されているホスト $h_{i'}$ に移動する．なお $w_i > w_j (w_i, w_j \in W_A)$ とする．a_i はエージェント a_q および a_r と通信を行う必要があり，a_q, a_r はそれぞれ，ホスト $h_{q'}, h_{r'}$ に配置されているとする．

このシナリオでは，a_i の実行完了時間 f_i は次式のようになる．

$$f_i = (w_i + w_j)s_{i'} + \max\{c_{i'q'}c^a_{iq}, c_{i'r'}c^a_{ir}\} + c_{i_0 i'} b_i \quad (1)$$

なお，$c_{i'q'}, c_{i'r'}, c_{i_0 i'} \in C$，$c^a_{iq}, c^a_{ir} \in C_A$，$s_{i'} \in S$，$b_i \in B$ である．もし，$h_{i'}$ と $h_{q'}$（または $h_{r'}$）との間にリンクがなければ，$c_{i'q'}$ は次のように定義される．

$$c_{i'q'} = \sum_{l_{hk} \in P_{i'q'}} C_{hk} \quad (2)$$

ここで $P_{i'q'}$ とは $h_{i'}$ から $h_{q'}$ への路のことであり，$l_{hk} \in L$ とする． □

a_i が時刻 t_j^i に j 回目の移動としてホスト $h_j^i \in H$ へ移動すると表記したとき，すべての j に対して (h_j^i, t_j^i) のリストが与えられているなら，f_i を計算できる．このとき，システム中のすべてのエージェントの実行完了時刻は $\max_i f_i$ で与えられる．

この環境で，決定問題としてのモバイルエージェント実行計画問題の質問を以下のように定義する．**質問**：D 以内にすべてのエージェントの実行が完了する配置（位置および移動）が存在するか？すなわち，全エージェントの移動リストの中に $\forall i, f_i \leq D$ を満たすものがあるか？

図 1 に，モバイルエージェント実行計画，互いに独立な preemptive タスクのスケジューリング，マルチプロセッサスケジューリング，タスク間の通信を必要とするマルチプロセッサスケジューリング，の問題間の関係を示す．

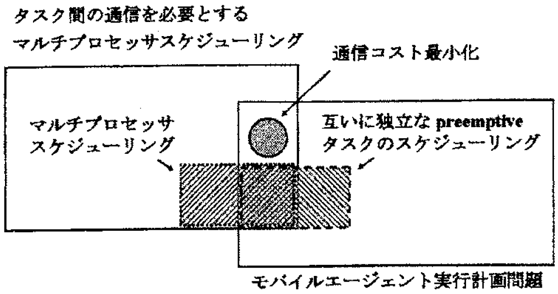

図 1: 問題間の関係

3 強 NP 完全性の証明

まず，モバイルエージェント実行計画問題を通信コスト最小化問題に制限する．通信コスト最小化問題は，図 1 に示されるように，各エージェントの負荷と実行途中での移動を考慮する必要がない問題である．次に，通信コスト最小化問題は，ハミルトン閉路問題 [4] を含む部分グラフ同型問題 [4] の部分問題と等しいことを示す．エージェントの負荷と移動コストを考慮する必要をなくすために次の五つの仮定をおく．

1. ホストの処理能力はすべて等しい（$\forall s, s' \in S, s = s'$）．
2. エージェントの負荷はすべて等しい（$\forall w, w' \in W_A, w = w'$）．
3. $n = m$
4. 仮定 1,2 を共に満たした上で，$w \in W_A$，$s \in S$ のときに，$ws > \max_{ijqr}\{c_{ij}c^a_{qr}\}$
5. 移動コストは無視できる（$\forall b \in B, b = 0$）．

仮定 1〜5 の下では，各ホストにちょうどひとつずつエージェントが配置されるのが最適となる．次に，各ホストにちょうどひとつずつエージェントが配置されているなら，それ以上エージェントの移動はないことを保障するため，さらに五つの仮定を追加する．

6. $\forall c^a \in C_A, c^a \in \{0, 1\}$.
7. $\forall i, c^a_{ij} \in C_A, \sum_j c^a_{ij} > 0$.
8. G は完全グラフ．
9. $\forall c \in C, c \in \{1, 2\}$.
10. 仮定 1,2 を共に満たした上で，$w \in W_A$，$s \in S$ のときに，$D = ws + 1$

仮定 1〜10 の下では，問題はエージェント間の最大通信時間が 1 となる配置の有無をを見つける問題に制限される．よって，preemption を考慮する必要は無い．実行途中の移動を考える必要はないので，最小通信コストがホストグラフとエージェントグラフの節点間の完全マッチングとなることより，問題は単純な通信コスト最小化問題になる．通信コスト最小化問題の正式な定義は以下の通り：二つの完全グラフ，ホストグラフ $G_H=(H,L,C)$ とエージェントグラフ $G_A=(A,E,C_A)$ がある．これらのグラフは仮定 1〜10 を満足しているとする．ここで解のグラフ $G_s=(H,L,C_s)$ について考える．もしエージェント a_i,a_j がそれぞれ，ホスト h_q,h_r に配置しているとすると，$c^s_{qr}\in C_s$ は $c^s_{qr}=c_{qr}c^a_{ij}$ となることより，この問題は，$\max_{ij}c^s_{ij}\leq 1$ となるよう，G_H と G_A の節点間の完全マッチングを見つける問題である．このことは，通信コスト最小化問題が $|V_1|=|V_2|$ となる部分グラフ同型問題と等しいことを示している．ただし，部分グラフ同型問題（強 NP 完全）は下記のように定義される [4]：二つのグラフ $G=(V_1,E_1)$ と $H=(V_2,E_2)$ がある．G の部分グラフに H と同型のものが存在するかどうかを考える．例えば，部分集合 $V\subseteq V_1$ と $E\subseteq E_1$ がそれぞれ $|V|=|V_2|,|E|=|E_2|$ であり，次のような一対一関数 $f:V_2\rightarrow V$ が存在:もし $\{f(u),f(v)\}\in E$ であれば，f は $\{u,v\}\in E_2$，を満足する．

さらに問題に対して，$k\in Z^+$，$\forall i,\sum_j c^a_{ij}=k$ という制限を追加した場合を考える．この制限は，各エージェントが通信するエージェントの個数がちょうど k であることを意味する．膨大な数のエージェントが存在する大規模なネットワークでは，通常，各エージェントが通信する必要があるエージェントの個数は定数に制限される．したがって，この制限は単なる NP 完全性の証明の道具というだけでなく，実用上も十分な意味を持つ．$|V_1|=|V_2|$ であるような部分グラフ同型問題においては，$v_i\in V_2$ の次数を d_i とすると，この制限は $\forall i,d_i=k$ と表される．よってハミルトン閉路問題は，$|V_1|=|V_2|$ かつ $k=2$ であるような部分グラフ同型問題において，H がひとつのリングの場合と考えることができる．すなわち，ハミルトン閉路問題が強 NP 完全であることより，通信コスト最小化問題は $k=2$ であっても強 NP 完全である．したがって，モバイルエージェント実行計画問題は，各エージェントが通信するエージェント数を 2 に制限した場合ですら強 NP 完全である．

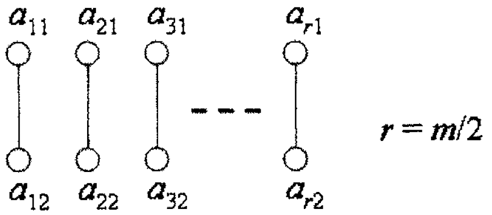

図 2: 各エージェントの通信するエージェント数を 1 に制限した場合のエージェントグラフ（リンクが描かれていないエージェント間の通信量は 0）

4 残された課題の検討

4.1 NP 完全性の範囲の模索

第 3 章で述べた 10 個の仮定の下であれば（注：仮定 2 より，$n=m$），$k=1$，つまり各エージェントが通信するエージェントの数が 1 に制限された場合（図 2 参照）は，ホストグラフ G の $c=2$ となるリンクを除いたグラフの最大マッチング M を求め [3]，$|M|=\frac{n}{2}$ であるかどうかを調べることにより $O(\sqrt{n}m)$ 時間で解が求まる．

また，第 3 章で述べた 10 個の仮定の下で，エージェントグラフを一本のパス（$k=2$ のエージェントが $m-2$，$k=1$ のエージェントが二つ存在）とすれば，ハミルトン路問題からモバイルエージェント実行計画問題に多項式時間帰着可能であることがわかるので（証明略），各エージェントが通信するエージェント数を 1 または 2 に制限した場合ですら強 NP 完全である．

モバイルエージェント実行計画問題について，問題の分類を行った．問題の分類を図 3 に示す．なお，網掛け部分は強 NP 完全問題である．

各エージェントの通信相手数が 1 で，仮定 1〜8 をおいた場合

解法

Step 1 ホストグラフ G の最小コストマッチングを求め（例えば，[1] 参照），求まったマッチングを $g_1,\cdots,g_{n/2}$ とする．

Step 2 求まったマッチングの重みの平均値 ave_g を求める．$ave_g=\frac{2}{n}\sum_{i=1}^{n/2}c_{g_i}$

Step 3 各エージェントの組は，$\frac{2ave_g}{n}$ 時間ずつ，$g_1,\cdots,g_{n/2}$ にて通信を行う． □

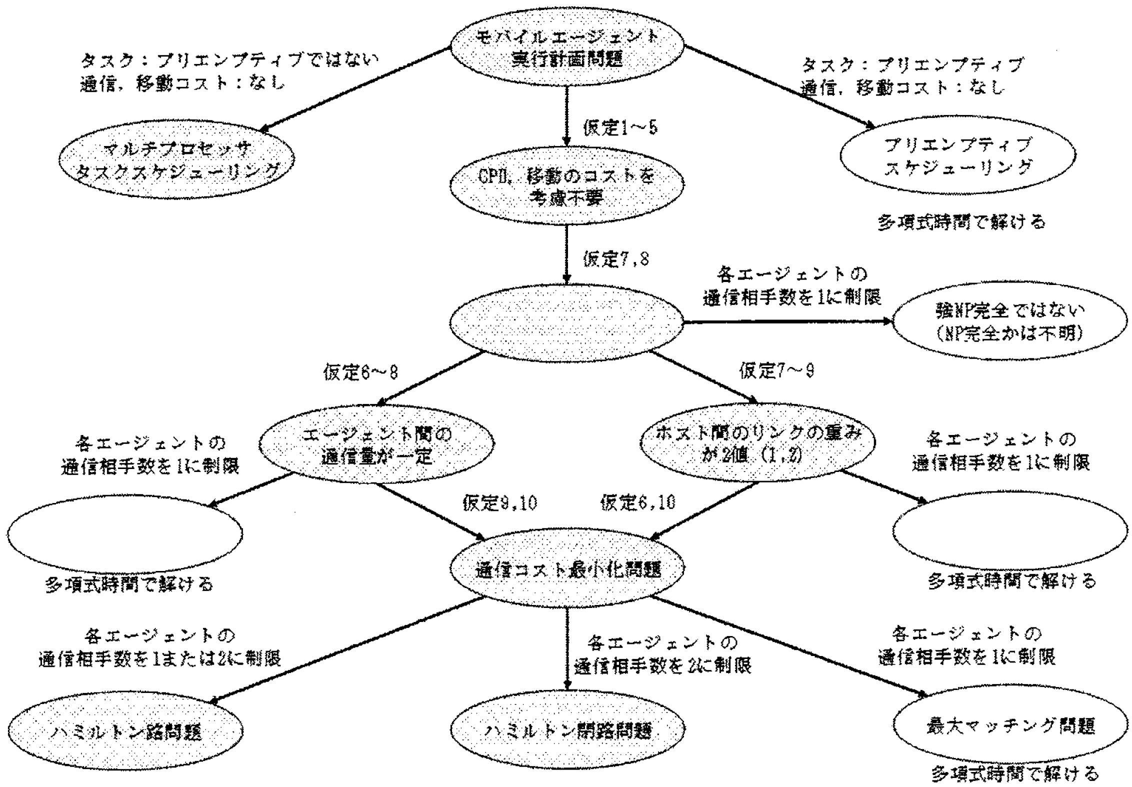

図 3: 問題の分類

最適性の証明　まず，同一ホスト上に複数エージェントを配置することは仮定 4 により最適でないことがわかるので，各ホスト上にひとつずつエージェントを配置するという条件の下で通信コストを最小化させることになる．そして，通信量はどのエージェントもすべて 1 なので通信コストはホストグラフで使用するリンクの重みのみで決まる．最小コストマッチングにより，リンクの重みの和が最小になるホストを共有しないリンクを $m/2$ 本選択できるので，これがエージェントの通信コストの総和（総通信コスト）c_s を最小化するリンクとなることがわかる．このとき，各エージェントの通信コストの最大値として実現可能な最小値が $ave_g = 2c_s/m$ であることは明らかである．従って，このアルゴリズムで求まるモバイルエージェント実行計画が最適であることが分かる． □

各エージェントの通信相手数が 1 で，仮定 1〜5,7〜9 をおいた場合

ホストグラフには重みは 1 と 2 の二種しかないことより，最適化の際にできることは通信量が少ないエージェント対が重み 2 の通信リンクを使い，多いエージェントが 1 のリンクを使う場合に，前者の通信時間が後者を上回る場合にそれを平均化することだけである．そして，これはエージェント対が三つの間でも同様にできることより，このような関係にあるエージェント対だけを対象にして平均化処理を行なえば，それで最適となる．このアルゴリズムは多項式時間で実行できる．

各エージェントの通信相手数が 1 で，仮定 1〜5,7,8 をおいた場合

各エージェントの通信相手数が 1 で仮定 1〜5,7〜9 をおいた場合の解法から，ホストグラフのリンクの重みの最大値を $c_{\max}$ とすると，擬多項式時間，すなわち，各エージェントの通信相手数が 1 で仮定 1〜5,7〜9 をおいた場合の解法の $O(c_{\max})$ 倍の時間で解ける．よって，少なくとも強 NP 困難ではない．

4.2　近似アルゴリズムの検討

CPU コスト，移動コストそれぞれの単独での最小化は多項式時間で求まるが，通信コストを考慮すると各エージェントが通信するエージェント数が 2 であって

も強NP困難である（移動コスト最小化は初期位置から移動しなければ良い）．そのため，どのような近似手法も“初期位置から移動しない戦略”よりも必ず良くなるという保障は得られない．よって，現在の配置よりもすべてのエージェントの実行完了時間が改善するなら移動，そうでないなら移動しない，という戦略が近似手法の主となるだろう．あるエージェントが移動すれば，移動元と移動先のホストに配置されたエージェントの数も変化する．よってエージェントが移動する際には，移動するエージェントが相互通信を行っているエージェント以外に，移動元と移動先に配置されているエージェントも影響を受けることになる．つまり，エージェントが移動するかどうかを決定する際には，その移動によって影響を受ける各エージェントの実行完了時間の変化が，システム全体の実行完了時間を改善させるか悪化させるかを見極める必要がある．

5 まとめ

モバイルエージェント実行計画問題を定義し，各エージェントが通信するエージェントの数を2に制限した問題が強NP完全であることを証明した．この結果は，エージェントの相互通信を考える必要がある場合，適切なエージェント実行計画を得るには近似解法が必要であることを示している．また，各エージェントが通信するエージェントの数を1に制限した場合の計算複雑性や，近似アルゴリズムの開発可能性について検討を行った．

参考文献

[1] R.K. Ahuja, T.L. Magnanti and J.B. Orlin, Network Flows, Prentice Hall, Englewood Cliffs, NJ, 1993.

[2] W.R. Cockayne and M. Zyda, Mobile Agents, Manning, Greenwich, CT, 1998.

[3] S. Even and R.E. Tarjan, "Network flow and testing graph connectivity", SIAM Journal on Computing, Vol.4, pp.507–518, 1975.

[4] M.R. Garey and D.S. Johnson, Computers and Intractability, A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, San Francisco, CA, 1979.

[5] M.D. Hamilton and I. Mitrani, "Optimal allocation policies for mobile agents," in: Lecture Notes in Computer Science, Vol.1786 (TOOLS 2000), Springer, Berlin, pp.145–155, 2000.

[6] T. Kawamura, S. Joseph, A. Ohsuga, and S. Honiden, "Designing multi-agent systems based on pairwise agent interactions," IEICE Transaction on Infomation and Systems, Vol.E84-D, No.8, pp.968–980, 2001.

[7] E.L. Lawler and J. Jabetoulle, "On preemptive scheduling of unrelated parallel processors by linear programming," Journal of the ACM, Vol.25, No.4, pp.612–619, 1978.

[8] I. Satoh, "Selection of mobile agents," Proc. the 24th IEEE Internat. Conference on Distributed Comput. Systems (ICDCS2004), pp.484–493, 2004.

[9] B.A. Shirazi, A.R. Hurson, and K.M. Kavi eds., Scheduling and Load Balancing in Parallel and Distributed Systems, IEEE Computer Society, Los Alamitos, CA, 1995.

[10] J.D. Ullman, "NP-complete scheduling problems," Journal of Computing System Science, Vol.10, 1975.

[11] G. Weiss eds., Multiagent Systems, A Modern Approach to Distributed Artificial Intelligence, MIT Press, Cambridge, MA, 1999.