

通信制限のある複数エージェントの協調連続巡回問題における担当領域の重複とその抑制手法の提案

吉村 祐^{1,a)} 杉山 歩未¹ 菅原 俊治^{1,b)}

受付日 2018年2月1日, 再受付日 2018年3月22日,
採録日 2018年4月1日

概要: 本研究では通信に制限がある環境で, エージェント間の交渉を通じて自分が作業すべき担当領域を自律的に決定する手法を提案する. 近年のロボット技術の発達により, ロボットの活躍の領域が広がっている. しかし環境の大きさ, 求められる作業量, バッテリー容量などの制限を考慮すると, 複数のロボットによる協調作業が必須となることがある. 我々は本目的のためにロボットをエージェントとしてモデル化し, 協調連続巡回問題としてとらえ, 作業領域を分担しながら公平で効率的な作業のための環境の分割方法を提案してきた. しかし, 通信に制限があり, たまたまエージェントが近づいたときのみ通信を可能となるように制限を加えると, 隣接エージェントと長期間離れていて通信できない間に担当領域に冗長性が発生し, 効率低下を招くことが分かった. そこで本論文では, 各エージェントが隣接エージェントの作業負荷を推定しながら, 拡大を制限する手法を提案する. これにより, 整合性があり公平な分業が実現できることに加え, この冗長な拡大を必要な領域には活用し, 不要な部分にはなるべく排除させた. この結果, 既存手法で発生した不要な冗長性を防ぎ効率を向上させるとともに, 必要な相手には助けるという形でこの冗長性を活用し, 通信がつかぬに可能な環境を想定した手法よりも効率化が実現できたことを実験を通して示す.

キーワード: 協調連続巡回問題, 領域分割, 分業, 通信範囲, マルチエージェントシステム

A Method for Preventing Redundant Responsible Areas Caused by Limited Communications in Continuous Multi-agent Patrol Problem

YU YOSHIMURA^{1,a)} AYUMI SUGIYAMA¹ TOSHIHARU SUGAWARA^{1,b)}

Received: February 1, 2018, Revised: March 22, 2018,
Accepted: April 1, 2018

Abstract: We propose a method for enabling agents to autonomously divide the given environment into the subareas for the individual responsibilities through negotiations between local agents in the environment where communication range is limited. Recent advance in robot and computer technologies expands the range of robot activities, but if we consider the requirement such as the size of environments and the required workload, and the limitation such as communication range and battery capacity, cooperation among robots becomes inevitable. Along this line, we formulated a continuous cooperative patrol problem by modeling robots as agents, and proposed the method by which agents can identify their responsible subareas of environments so that their workloads are fair and balanced. However, if the communication range is limited, their divided subareas contained so many redundant parts, and thereby, lowering the entire performance. In this paper, we propose a novel method in which agents not only reduce the redundant parts, but also use the redundant activities to help the busier neighboring agents. We experimentally show that our method can reduce the unnecessary redundancy and can improve the higher entire performance than that in the previous method that are assumed that communications are always available.

Keywords: continuous cooperative patrol problem, division of labor, partitioning, limited communication range, multi-agent systems

1. はじめに

近年のロボット技術の発達により、ロボットの活躍の領域が広がっている。このような領域には、人間の負担を減らすための作業や人間が活動するのが困難な場面などがあり、たとえば、前者として清掃や警備、後者の例としては惑星探索や災害時の人命救助があげられる。このような応用分野では、継続的な探索や作業と、これらの行動を通して要求条件、たとえば作業や探索の頻度の条件などを満たすように環境全体を訪れることが求められる。しかし、すべての作業を1台のロボットのみで処理するにはバッテリー容量や移動速度などの物理的あるいは能力上の制限があり現実的ではない。他方、コンピュータや通信技術の発展により、複数のロボットが協力し、共同作業で要求課題を解決することが現実的になってきた。

複数ロボットの協調作業の実現には、集中制御と自律分散制御の2つの方法が考えられる。前者では1台の制御プログラムあるいは1台のロボットが、全ロボットを集中管理する方法がある。しかし、大規模なタスクの場合ロボットの行動を決めるためのデータ量が大きくなるほか、制御部分が故障するとシステム全体が停止するという欠点もある。上記で述べた応用領域では作業の継続性は必須な機能であり、仮にシステムの一部あるいはロボットの一部が故障しても協調により相互に補い合い、作業を維持できるように、ロボットそれぞれが自律的に行動を決定し、全体としての目的を達成・維持する知的制御が必要となる。そのため本研究では、後者による自律分散制御による協調作業に焦点を当てる。

このような高い自律性の実現に焦点を当てた研究として、マルチエージェントシステムによる継続的巡回探索問題がある [1]。この研究では、複数のエージェントが環境内にある戦略で巡回し、ある確率分布（頻度）に従い継続的に発生するイベントを協力して観測あるいは検出する問題を扱っている。たとえば、セキュリティ巡回では与えられた重要度に応じてエージェントが訪問し、監視する頻度を増減させる。巡回清掃問題ではゴミが発生しやすい（汚れやすい）領域をより頻繁に訪れて作業するような制御が該当する。我々もこのような課題に焦点を当て、エージェントの能力や環境の差を考慮しながら担当領域を自律的に分割し、全体のパフォーマンスを向上させ、同時に個々の負荷をバランスさせる手法を提案してきた [2]。しかし、この研究ではエージェントどうしがつねに通信可能であることを仮定しており、自走式ロボットやドローンなど移動するエージェントを想定すると、この条件は現実的ではないこ

ともある。特に無線を使った通信では距離や障害物の影響を受け、確実な通信はそれぞれが近づいたときのみ可能という制限が加わる。しかし、エージェント間の距離が遠いとき、特に、頻繁に訪れるべき重点領域で隣接エージェントと通信できないと情報交換の間隔が伸び、重複領域が広く発生して効率が低下することや、公平な分業が実現できないなどの課題を確認している [3]。

そこで本論文では、この課題に対処するために、通信が不可能な期間でも、周辺のエージェントの状態を推測し、これを作業領域の分割戦略に反映させる手法を提案し、これによりつねに通信可能とした場合と遜色なく全体の効率化が実現できることを示す。なお、以下では説明を容易にするために、巡回清掃問題を例題として述べるが、本手法は、セキュリティ巡回などの領域にも適用可能である。

本論文の構成は以下のとおりである。まず、2章で関連研究について述べ、3章で本研究で用いるエージェントや環境のモデル、本研究で求める課題と既存研究の概要を説明する。4章で通信が不可能な間でも、不要な領域拡大をできる限り抑える領域分割法の提案をする。5章では実験で用いるエージェントや環境について説明したのち、評価実験と考察を加える。特に、重複領域の大きさによる影響と負担の大きいエージェントの領域拡大を抑えたことによって清掃効率の向上につながったことを確認し、提案手法の有効性を議論する。最後に6章で結論と今後の課題を述べる。

2. 関連研究

複数のエージェントが協調して、継続的に環境を巡回するアプローチは大きく2つに分けられる。第1の手法は全エージェントが作業領域全体を共有して清掃する手法である [1], [4], [5], [6], [7], [8]。

たとえば、Ahmadiら [4] は1台のエージェントが異なる確率で発生するイベントを、発生確率の高い場所を優先的に巡回する手法を提案している。しかし、複数台のエージェントによる協調は考慮されていない。Chevaleyr [6] は、巡回セールスマン問題をベースにして分業しながら巡回するマルチエージェント巡回戦略を提案し、それをあらかじめ経路分割した場合と比較している。Yonedaら [7] は、協調巡回問題において、バッテリー残量を考慮しながら探索を継続するか充電に戻るかの判断をしながら、エージェントが自分のチーム内で効率を上げられる巡回探索戦略の学習法を提案している。Sugiyamaら [1] では、文献 [7] を拡張し、さらに分業を促進するための交渉手法を提案し、効率化を実現している。しかし領域を分割しない方法では、他のエージェントとの領域重複作業が起き、これだけでは十分な効率性を引き出せない可能性がある。

第2のアプローチは領域を分割して、各エージェントが担当領域で作業する手法である [2], [9], [10], [11], [12],

¹ 早稲田大学
Waseda University, Shinjuku, Tokyo 169-8555, Japan
a) y.yoshimura@isl.cs.waseda.ac.jp
b) sugawara@waseda.jp

[13], [14]. たとえば, Ahmadi ら [10] では, 文献 [4] を複数エージェントに適用できるように拡張し, 各エージェントが互いの境界に関する情報を交換し, より頻繁にその境界領域を訪れるエージェントにその領域を渡すよう調整している. Elor ら [12] は, ガスで膨張する Balloon モデルとアリのフェロモンモデルを活用し, 各エージェントの担当領域の広さを反映した値として膨張力を導入し, これを均衡させることで領域分割をしている. しかし, どちらの研究においてもバッテリーによる制限は考慮されていない. さらに環境には障害物の存在, 汚れやすさ, 形状やスロープによる探索の困難さなどの差があり, 他方エージェントにも移動速度, 作業 (清掃) 効率, 探索アルゴリズムの差, バッテリー容量など, ハードあるいはソフトウェア上の能力差があり, 単純に各担当領域の広さを均等にしたり, 訪れる頻度だけで領域を分割したりすると作業負荷が不均衡になる可能性がある.

Kato ら [13] では, エージェントがバッテリー残量を考慮しながら作業の余力を表す指標として上記の膨張力を拡大力として定義し直し, エージェントの能力や環境の差を考慮した分割手法を提案した. また Sea ら [2] では, ゴミの発生しやすさが未知の環境下で複数エージェントがゴミの発生確率を学習しながら探索する手法と, 上記の再定義した拡大力を統合し, 環境の差, エージェントの探索アルゴリズムの差, バッテリー容量の差などを反映した分割が可能であることを示した. しかし, これらの研究では隣接エージェントとはつねに通信が可能であることを仮定しており, 状況によってはこの条件は厳しいこともある.

実際に吉村ら [3] は, ゴミの発生確率を学習しながらエージェント間の通信範囲の制限を加えた領域分割法を適用してマルチロボットの巡回清掃問題に対する手法を実装したが, 隣接エージェントとしばらく通信ができない間に分業に冗長性, 具体的には担当領域の重複が発生し, 環境によってはそれが大きな効率低下を招くことを示した. また, Mao ら [15] でも通信範囲に制限がある設定で, 複数エージェントが領域を分割して巡回する手法を提案しているが, 環境の規模が小さく現実的とはいえない.

本研究では, 既存の手法 [2] に通信可能範囲の制限を加えても, 作業効率を低下させない手法を提案する.

3. モデルの定義

本章では, 環境とエージェントのモデル, 既存研究 [2], [3], [13] で採用されている手法の概要, および問題の定式化について述べる.

本モデルは当初文献 [13] で提案したものであるが, 文献 [2] ではこれを拡張し, 3.3.1 項で述べるノードのゴミの発生確率の学習を導入した. 我々はさらに文献 [2] を拡張し, 3.4.4 項で述べる通信範囲の制限を導入したモデルを提案し, その場合に起こりうる課題を示唆した [3].

本論文では文献 [3] で発見された課題を解決すべく 3.4.2 項で述べる領域拡大行動および 3.4.4 項で述べる通信による交渉に関して拡張を加えた手法を提案するが, その前に文献 [2] のモデルを説明する. ゴミの発生確率の学習とその影響については本論文の範囲を超えるのでここでは省略する. 詳細については文献 [2] を参照されたい. また, 文献 [3] で提案した通信範囲の制限を加えたモデルは 3.4.4 項で説明する.

3.1 環境のモデル

$A = \{a_1, \dots, a_N\}$ をエージェントの集合とし, 移動の表現のために離散時間を導入する. エージェントが清掃を行う空間を 2 次元もしくは 3 次元空間に埋め込み可能な連結グラフ $G = (V, E)$ とする. ここで $V = \{v_1, \dots, v_x\}$ はノードの集合, E はエッジの集合とし, ノード $v_i, v_j \in V$ にエッジが存在するときそれを $e_{i,j}$ と表し, v_i, v_j 間に到達できる道があると考え. 必要に応じてダミーのノードを加えることで, エッジの長さをすべて 1 とし, エージェントは 1 単位時間でエッジでつながった隣接ノードへと移動し, そのゴミを吸引する. それぞれのエージェント $a_i \in A$ は充電を行うための基地 $v_{base}^i (\in V)$ を個別に持つ.

エージェント a_i には, 自分が担当すべき領域 (担当領域) がある. 時刻 t における a_i の担当領域を連結サブグラフ

$$G_t^i = (V_t^i, E_t^i) \text{ ただし } (V_t^i \subseteq V, E_t^i \subseteq E)$$

と表す. 全体の領域を均一に清潔に保つために, それぞれのエージェントの担当領域 G_t^i は, 清掃状況に応じて領域拡大行動ならびに他のエージェントとの通信を行い調整するため, 時刻ごとに変化する. また, 既存の手法 [2] と異なり, $G_t^i \cap G_t^j \neq \emptyset$ となりうる.

環境内のゴミの発生は時間ごとに確率的に起こり, 各ノード v の汚れやすさを 1 単位時間あたりのゴミの発生確率 p_v ($0 \leq p_v \leq 1$) で表現する. 時刻 t におけるノード v のゴミの量 $L_t(v)$ は,

$$L_{t+1}(v) = \begin{cases} L_t(v) + 1 & (\text{ゴミの発生時, 確率 } p_v) \\ L_t(v) & (\text{その他, 確率 } 1 - p_v) \end{cases} \quad (1)$$

のように蓄積されるが, エージェントが時刻 t に v に到達すると, v 上のゴミはすべて清掃され, $L_{t+1}(v) = 0$ となる.

3.2 エージェントのモデル

エージェントのモデルに以下の 2 つを仮定する. 第 1 にエージェントは領域のマップ (グラフ G) を既知とする. 現在多くのマップ作成やそれに基づく位置同定のアルゴリズムが提案されており (たとえば, 文献 [16], [17], [18]), 基地から徐々に地図を作成することが可能となっている.

将来的にはこれらを活用することを想定してマップを既知とし、本研究では領域分割を用いたエージェント間の協調的分業に主眼を置く。第2に複数のエージェントが同一ノードに存在でき、衝突は考えない。実際には、本手法は作業領域を分割する手法であり、複数のエージェントが接近したときには通信により分担領域を分けるため衝突は起こらないと考える。しかし衝突の可能性があっても通信可能な範囲では衝突回避のアルゴリズムが多数提案されており [19], [20], それらを仮定することで衝突回避ができると考え、ここでは前者と同様に領域分割手法に主眼を置くこととする。

エージェント a_i は有限容量のバッテリーを持ち、自分の基地から探索を開始し、自分の基地に戻り充電を行う。エージェントは探索状態と充電状態の2つの状態を交互に繰り返す。探索状態とは、エージェントが巡回清掃のために領域内を探索している状態のことである。エージェント a_i が持つバッテリーの最大容量を B_{max}^i 、1単位時間あたりのバッテリー消費量を B_{drain}^i とする。時刻 t におけるバッテリー残量を B_t^i とすると、時刻 $t+1$ におけるバッテリー残量は

$$B_{t+1}^i = B_t^i - B_{drain}^i \quad (2)$$

となる。そのため、 a_i の最大連続稼働時間は $\lfloor B_{max}^i / B_{drain}^i \rfloor$ となる。

$Len(v^m, v^l)$ をノード v^m から v^l の経路長としたとき、エージェント a_i がノード v から基地 v_{base}^i までの移動に必要なエネルギー $Pot^i(v)$ は以下のとおりである。

$$Pot^i(v) = Len(v, v_{base}^i) \times B_{drain}^i \quad (3)$$

したがって、 v^i をエージェント i の現在ノードとしたとき、条件

$$B_t^i < Pot(v) + Len(v^i, v) \times B_{drain}^i \quad (4)$$

を満たすノード v からは基地に帰ることはできないため、以下に述べる探索アルゴリズムにより次の目標ノード v が上記の条件を満たすとき、エージェントは v に移動せず充電のため基地へ移動する。本論文では通信が行えない場合でのエージェントによる領域分割が主眼であることや、通信装置に関してロボットに搭載するものと考えており、高出力の無線を想定しておらず、移動用や清掃用のモータと比べると消費電力は小さいと仮定し、1単位時間あたりのバッテリー消費量 B_{drain}^i には通信の電力も含むとした。

エージェント a_i は基地 v_{base}^i で充電する。一般的に、稼働時間と充電時間は異なるため、充電にかかる時間をバッテリーの消費量と比例すると仮定し、充電の比例定数を k_{charge}^i とすると、充電の所要時間 t_{charge}^i は以下のようになる。

$$t_{charge}^i = k_{charge}^i (B_{max}^i - B_t^i) \quad (5)$$

なお、充電容量最大まで充電しなくても移動開始は可能

だが、ここでは簡単化のために、基地では満充電になるまで待機し、その後、ただちに移動するものとする。

3.3 担当領域の同定とゴミの発生確率の学習

エージェント a_i には、初めに基地 v_{base}^i を中心に距離 d_{ini} ノード以下の範囲を初期担当領域 $G_0^i = (V_0^i, E_0^i)$ とし与える。なお本論文の実験では、距離としてマンハッタン距離を採用した。また、 $v_{base}^i \in V_t^i$ とする。以下、担当領域の同定とゴミの発生確率の学習について述べる。

3.3.1 ゴミの発生確率の学習

エージェント a_i はノード v のゴミの発生確率 p_v を知らないため、 a_i は、担当領域内の各ノード v に対し、 p_v を学習する。時刻 t における a_i による p_v の推定値を $p_v^i(t)$ とし、初期値は $p_v^i(0) = 0$ とする。エージェント a_i がノード v を時刻 t に訪れたとき、その前に v に訪れた時間を t_v^i とし（もし v が新しく担当領域になり、 a_i が v を訪れたことがなければ v が V_t^i に加えられた時刻を t_v^i とする）、ノード v を訪れた間隔 $t - t_v^i$ とノード v でのゴミの回収量 $AL_t(v)$ と学習率 α ($0 \leq \alpha \leq 1$) から v でのゴミの発生確率 $p_v^i(t)$ を以下のように計算する。

$$p_v^i(t) = (1 - \alpha) \cdot p_v^i(t-1) + \alpha \cdot \frac{AL_t(v)}{t - t_v^i} \quad (6)$$

ここで、式 (6) は、観測値の差分を埋めることを示している。訪れていないノード v については、 $p_v^i(t) = p_v^i(t-1)$ とする。なお、 $AL_t(v)$ は回収量、つまり吸い込んだゴミの量 $L_t(v)$ としているが、これを簡略化して $AL_t(v)$ をゴミを吸い込んだとき1、その他を0の2値としても、ゴミの発生頻度の差については同程度の学習が可能であることを確認している [1]。エージェントは時刻 t におけるノード v の実際のゴミの量 $L_t(v)$ を知らないが、ゴミの発生確率の推定値 $p_v^i(t)$ を用いて、その期待値として $L_t(v)$ を推定する。ノード v におけるゴミの量の推定値 $E(L_t(v))$ は次式で求められる。

$$E(L_t(v)) = p_v^i(t) \cdot (t - t_v^i) \quad (7)$$

なお、ノードの集合 V に対し、以下のように定義する。

$$L_t(V) = \sum_{v \in V} L_t(v) \quad (8)$$

$$E(L_t(V)) = \sum_{v \in V} E(L_t(v)) \quad (9)$$

したがって、たとえば、 a_i の担当領域に存在するゴミの推定値は $E(L_t(V_t^i))$ と表せる。

3.4 エージェントの行動

エージェントの行動は文献 [2], [3], [13] に基づくものであり、ここではそれらについて簡単に述べる。

3.4.1 探索アルゴリズム

エージェントは探索アルゴリズムを使って担当領域を巡

回する。本研究では探索アルゴリズム自体の評価は対象外であり、分業のための自律的担当領域分割を主題とするため、共通的に、簡易な探索アルゴリズムである有向深さ優先探索アルゴリズムを用いる。この探索アルゴリズムでは、エージェント a_i は基地から出発するとき担当領域の中からゴミの量の推定値 $E(L_t(v))$ の値が最も高いノードへ最短距離で移動する。ターゲットノードに到達したら、隣接ノードのうち訪れていないノードをランダムに選択し、そのノードに移動し、そのノードをスタックにプッシュする。このプロセスをエージェント a_i が訪れたことがないノードを選べる限り続ける。この際、もしエージェント a_i は選択するノードがない場合、スタックの一番上のノードをポップし、そのノードに戻り、別の訪れていないノードを選択する。この動作を繰り返したのち、最初に選択したターゲットノードへ戻った場合、基地に最短経路で戻る。ただし探索の途中で次のノードが条件 (4) を満たすとき、エージェントはそのノードの代わりに基地を目標ノードに変え基地に戻る。

3.4.2 領域拡大行動の開始

領域拡大行動とは、担当エージェントがいない領域、あるいはより忙しいエージェントの担当領域を一部引き受けるために、担当領域の拡大を試みることである。エージェントは後述する条件を満たしたとき担当領域をほとんど清掃したと判断し、担当領域の拡大を試みる。この判断のために、エージェント a_i は時刻 t_b に基地 v_{base}^i を出発するとき、担当領域における少し未来のゴミの残量の推定値 $E(L_{t_b+\gamma}(V_{t_b}^i))$ を計算する。ここで γ は正の整数とする。これはエージェントが巡回清掃している間もゴミは蓄積し続けるため、もし現在のゴミの残量の推定値を基準とすると、推定値よりも多くのゴミが残っているにもかかわらず担当領域を十分に清掃したと判断し不要に領域拡大をすることを防ぐためである。なお、少し未来の推定値の増分は、その担当領域の特徴により異なることに注意されたい。エージェント a_i は t_b から始まった現在の巡回行動中に、時刻 $t (> t_b)$ における自分が訪れたノード数 $N_{vis}(t)$ と回収したゴミの総和 $N_d(t)$ を記録する。そして、エージェントは、

$$N_{vis}(t) \geq R_1 \cdot |V_t^i| \tag{10}$$

$$N_d(t) \geq R_2 \cdot E(L_{t_b+\gamma}(V_{t_b}^i)) \tag{11}$$

のいずれかの条件を満たしたとき、担当領域を十分清掃できたと判断し、拡大行動を開始する。ここで、 $0 \leq R_1, R_2 \leq 1$ とする。また、環境の構造などによっては $t_b + \gamma$ よりも前に式 (11) を満たす場合もあり、逆にゴミの回収に時間がかかり $t_b + \gamma$ よりも後になる場合もある。なお、エージェントは十分なバッテリー残量を持っていても、領域拡大行動は基地を出てから 1 回のみとし、過剰頻度の拡大行動を避ける。また、 γ はエージェントのバッテリー容量 B_{max}^i

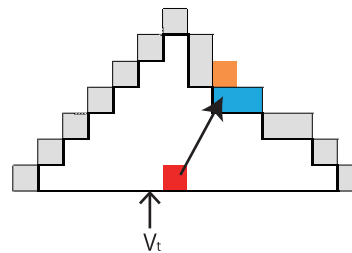


図 1 拡大戦略

Fig. 1 Expansion strategy.

を超えない値とする。これは、エージェントが基地に戻った際に自分が訪れたノード数 $N_{vis}(t)$ と回収したゴミの総和 $N_d(t)$ をリセットするため、バッテリー容量よりも高い値でゴミの残量を推定した場合、基地を出発してから基地へ戻ってくるまでに推定値以上のゴミを回収するのが困難になるからである。

3.4.3 拡大戦略

ここでは既存手法 [2] における拡大戦略を述べる。拡大戦略は、エージェントやシステム全体の性能に影響すると考えられる。基地から相対的に離れたノードの取り込みや、頻繁な領域の取り合い、拡大行動交渉の連続失敗を回避するなどの要因を考慮して、拡大対象となるノード $I^i \subset V$ を以下のように決定する。

- (1) エージェント a_i は自分の担当領域の境界ノードの集合 B を選択する (B と担当領域は排反とする)。
- (2) エージェント a_i は B から、 I_{avoid}^i (これは後に説明する) に含まれず、かつ自分の基地 v_{base}^i から最も近い $k_{inc} (> 0)$ 個のノードを選び、その集合を I_{inc}^i とする。
- (3) ノード I_{inc}^i とそれらの隣接ノードの集合のうち、 V_t^i にも I_{avoid}^i にも含まれないものを I^i とする。

たとえば、図 1 に示すようなグリッド状の環境 G で、 V_t^i を太線内のノードの集合とし、境界ノード B を灰色と水色のノードとする (オレンジ色は含まれない)。 $k_{inc} = 1$ かつ $I_{avoid}^i = \emptyset$, I_{inc}^i として矢印で指すノードを選んだとする。このとき水色とオレンジのノードが I^i となる。 $I^i = \emptyset$ となったときは a_i は領域拡大行動はしない。

重複領域の担当の決定や拡大戦略の指標として作業の余力を示す拡大力 (extension power) を定義する。エージェント a_i は時刻 t に充電基地に戻った際に、現在の担当領域に対する拡大力を計算するために、その時点でのゴミの残量の推定値 $E(L(V_t^i))$ を求め、エージェント a_i の拡大力 $\xi(i, t)$ を以下のように定義する。

$$\xi(i, t) = E(L(V_t^i))^{-1}$$

ただし $E(L(V_t^i)) = 0$ のときは $\xi(i, t)$ は十分大きな整数値とする。各エージェントはこの値を再び基地に戻り再計算するまで保持する。

3.4.4 通信と領域の更新（領域拡大のための交渉）

既存研究 [2] ではつねに通信が行うことが可能であり、エージェントは担当する領域の情報を隣接エージェントと交換し、 I^i の新領域を拡大するかどうかを交渉し決定する。そのため、隣接エージェントとの担当領域の重複は発生しない。しかし本研究では、いつも通信可能とは限らないため、しばらく遅延が発生する。そのため各エージェントは I^i の要素の選択とそれらを担当領域に加えるべきかどうかの判断を個別に行う。通信範囲に制限がある既存手法 [3] では、たとえ隣接エージェントと通信が行えないとしても I^i を次の交渉の機会まで担当領域に加えるため、隣接エージェントとの担当領域の重複が発生する。そして、隣接エージェントと通信が可能となった際にそれぞれの担当領域を比較し重複領域の有無を調べ、存在した場合はどちらが担当するかを交渉により決定する。提案手法における通信と交渉に関する部分は 4.1 節で説明する。

既存手法 [2], [3] では拡大力を相互に比較して、 I^i として拡大候補とした領域をどちらの担当領域に加えるべきかを決定した。この際に、候補にはしたが拡大できなかったノードの集合を I_{avoid}^i として記録し、しばらくの間 (k_{avoid} 回分)、領域拡大行動の際に I^i の要素として選ばないようにしていた（ただし、 k_{avoid} は正整数）。これには領域拡大行動の際の頻繁な失敗や、担当領域の取り合いのような振動現象を穏やかにする目的がある。しかし、本研究が想定する環境は通信に制限があり、他のエージェントの正確な情報が得られない環境であり、文献 [3] の結果から正確な情報をつねには得られないことによって分業に冗長性が生まれ効率低下を招くことが分かった。そのため、つねには隣接エージェントの情報が得られない場合の手法を提案手法にて説明する。

3.5 システムの評価指標

本システムの評価は、ある期間 $[t_s, t_e]$ における全ノード V のゴミの存在量の総和の平均値 $D_{t_s, t_e}(V)$ とし、以下の式で表す。

$$D_{t_s, t_e}(V) = \sum_{t=t_s}^{t_e} L_t(V) / (t_e - t_s + 1) \quad (12)$$

この値をなるべく小さく保つことが本研究の目的である。この定義から、ゴミが放置される（より一般に継続的巡回探索問題としてはイベントの発生を放置した）時間が長いと、 $D_{t_s, t_e}(V)$ は大きくなる。また、全領域を均一に清潔に保つためには、各担当領域が均一に清掃されることが望ましい。そのため、 $D_{t_s, t_e}(V)$ に加え、各担当領域におけるゴミの存在量（汚れの度合い） $L_t(V_t^i)$ の時間変化と、担当領域間の差も評価対象とする。

4. 提案手法

提案手法では、エージェント a_i は領域拡大にあたって、既存手法 [3] で定義した I_{avoid}^i ではなく、通信できない隣接エージェントの清掃状況を推定し、それに基づいた拡大戦略を用いることで、領域全体の清掃効率の向上を図っている。以下、詳細に説明する。

4.1 提案手法における通信と領域の更新

本研究では、エージェントはある一定の範囲内のエージェントとのみ通信できると想定する。エージェント a_i が送信した情報が届く通信範囲を距離 d_{comm} ($d_{comm} > 0$) 以下とする。本論文の提案手法における重複領域に関する交渉の流れは以下のとおりである。

- (I) 通信可否の確認 (Hello メッセージ)：作業中のエージェントは、時間 N_{comm} ごとにエージェント a_i は通信の確認のメッセージを送る。Hello メッセージに対する返事がエージェント a_j からあったとき、 a_i は V_t^i と現在の拡大力 $\xi_i = \xi(i, t)$ を a_j に送信する。
- (II) 送信情報の受信：時刻 t にエージェント a_j は、 a_i との距離が d_{comm} 以下のとき a_i と通信ができ、Hello メッセージへの返信後、担当領域と拡大力を受け取り、以下の動作を行う。
 - (1) 重複領域を $V_{overlap} = V_t^j \cap V_t^i$ とする。 $V_{overlap} = \emptyset$ の場合、何もせず終了する。 a_j は、 a_i の拡大力 ξ_i を記録する。
 - (2) $V_{overlap} \neq \emptyset$ なら、 a_j は、 a_i の拡大力 ξ_i と自分の拡大力 $\xi(j, t)$ を比較する。もし $\xi(j, t) \geq \xi_i$ ならば、 a_j は a_i に拡大拒否メッセージとして $V_{overlap}$ と $\xi_j = \xi(j, t)$ 、 V_t^j を送信する。もし $\xi(j, t) < \xi_i$ ならば、 a_j は重複領域のノードを担当領域から削除し、 a_i に受諾メッセージとして $V_{overlap}$ と $\xi_j = \xi(j, t)$ および修正した V_t^j を送信する。
- (III) a_i は a_j からの拡大拒否メッセージを受信した場合、そのノードを V_t^i から削除する。

上記の一連の通信は、1 単位時間で完了するものとした。なおこのとき、エージェント a_i と a_j は、情報交換した相手の拡大力 ξ_j と ξ_i を保存・更新し、さらに a_i は境界ノードの集合 B と a_j の担当領域 V_t^j の和集合 B_j^i を保存する。なお、通信範囲外（あるいは通信状況が悪く正常な通信ができない場合）では、領域の更新はない。

4.2 提案手法における拡大戦略

提案手法における拡大戦略は、隣接エージェントの清掃状況を考慮して拡大するノードの集合 I^i を以下のように決定する。なお、隣接エージェントとは通信の履歴のあるエージェントであり、履歴のないエージェントについては考慮しない。通信した a_i の隣接エージェントの集合を K^i

とおく.

- (1) すべての隣接エージェント $a_j \in K^i$ に関して, $\xi(i, t)$ と保存した ξ_j を比較し, もし $\xi_j \geq \xi(i, t)$ ならば, $B = B \setminus B_j^i$ とする. ここで B は, 3.4.3 項で求めた境界領域である.
- (2) a_i は B の中から自分の基地 v_{base}^i から最も近い順に k_{inc} 個のノードの集合である I_{inc}^i を選ぶ.
- (3) ノード I_{inc}^i とそれらの隣接ノードの集合のうち, V_t^i と $\xi_j \geq \xi(i, t)$ を満たす B_j^i の両者に含まれないものを I^i とする.

I^i の要素をさしあたり担当領域 V_t^i に加え, 清掃対象とする. なお, 拡大戦略直後に近隣のエージェントと通信できるとは限らない.

5. 評価実験

5.1 実験設定

提案手法の性能と特徴を実験を通して明らかにする. 実験で用いるエージェントと環境の設定を表 1 に示す. 4 台のエージェント $A = \{a_1, a_2, a_3, a_4\}$ はすべて同じ性能とする. 本実験の環境は図 2 に示すように, エージェントが清掃を行う空間 G を縦横ともに 51 の 2 次元グリッドとし, マンハッタン距離を導入する. エージェント a_i は各々の充電基地 v_{base}^i (図 2 では丸囲み数字で表す) から各担当領域 V_t^i の巡回清掃を行う. また本実験の環境では, エージェントが基地にいるときは距離の制約で通信はできないので, 充電中は通信を行わないとした.

環境に汚れやすさの偏りを持たせるために, ゴミの発生確率として $p_h = 2 \times 10^{-4}$, $p_m = 2 \times 10^{-5}$, $p_l = 2 \times 10^{-6}$ の 3 つがあり, 図 2 のように配置した. 環境 1 (Env. 1) はごみの発生分布が平均的であるが, 環境 3 (Env. 3) は担当領域の境界と思われる付近に発生確率の高い領域があり, 他方, 環境 2 (Env. 2) では境界とはやや離れた, 基

表 1 実験設定

Table 1 Experimental setting.

	パラメータ	値
エージェント台数	$ A $	4 台
学習率	α	5×10^{-2}
拡大戦略を行うためのパラメータ	R_1	0.7
	R_2	0.7
	γ	300
	k_{inc}	15
拡大戦略の制御パラメータ (既存手法のみ)	k_{avoid}	7
バッテリー消費量	B_{drain}	1
バッテリー最大容量	B_{max}	900
充電にかかる時間の比例定数	k_{charge}	3
通信が可能な範囲	d_{comm}	10
初期担当領域の範囲	d_{ini}	5
通信間隔	N_{com}	1

地に近い部分の発生確率が高くなっている. 実験は 1 回 100 万単位時間とし, 3600 単位時間 (つまり式 (12) において $t_e - t_s + 1 = 3600$) ごとに以下の値を記録する. なお, 3600 は, バッテリーの最大容量時に最大限行動できる時間と満充電までの時間を加えた値である.

- (1) 全ノードのゴミの存在時間の総和 $D_{t_e, t_s}(V)$ (以下 $D(V)$ と略記する)
- (2) 各担当領域の広さ $|V_t^i|$
- (3) 各担当領域におけるゴミの残量 $D(V_t^i)$

なお, 以下に示すデータは, 上記の実験を 100 回行った平均値である.

比較手法として文献 [3] で提案した既存手法を採用する. 既存手法では, 3.4.3 項で述べた拡大戦略で担当領域を拡大し, 次の通信可能なタイミングまでその領域も清掃を続ける. 次の通信のタイミングまでは, 重なる領域があるが複数のエージェントが清掃すること自体は問題はない. また, つねに通信が可能であると仮定した手法 [2] も, 比較のために一部加える. この手法を常時通信可能手法と以下記述する. なお参考として, 図 3 に環境 1 における担当領域のスナップショットを示す. ここで, 赤, 黄, 緑, 青はそれぞれエージェント単独の担当領域であり, 白はそれぞれの担当の境界にある重複領域を示す.

5.2 実験結果

初めに, 均一的な環境である環境 1 の結果を示す. 3 手法の $D(V)$ の推移を図 4 に, 既存手法と提案手法の $|V_t^i|$ の推移を図 5 に, $L(V_t^i)$ の推移を図 6 に示す. 図 4 から,

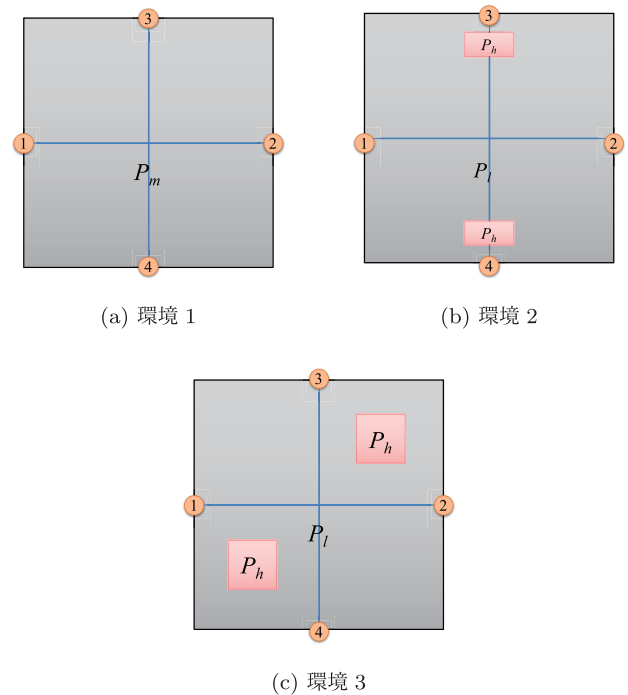


図 2 実験環境

Fig. 2 Experimental environments.

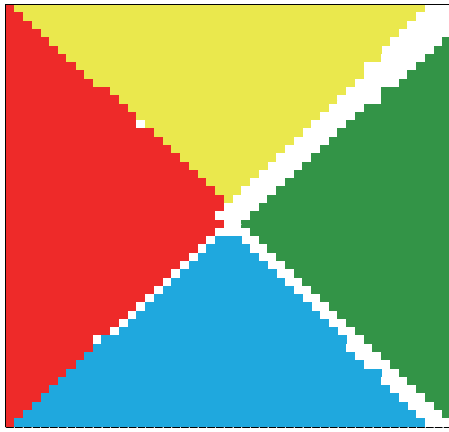


図 3 担当領域の分割の一例 (環境 1)

Fig. 3 Example of dividing area (Env. 1).

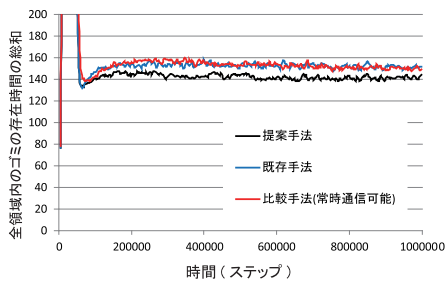
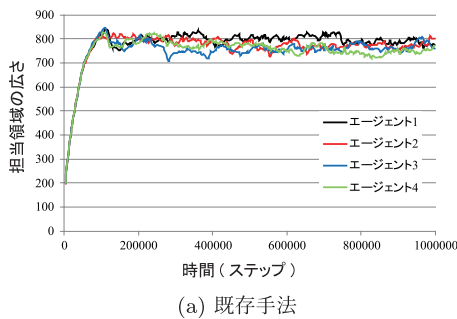
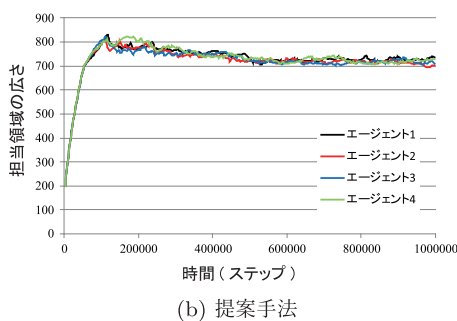


図 4 全ノードのゴミの存在時間の総和の推移 (環境 1)

Fig. 4 Accumulating existence duration of dirt (Env. 1).



(a) 既存手法

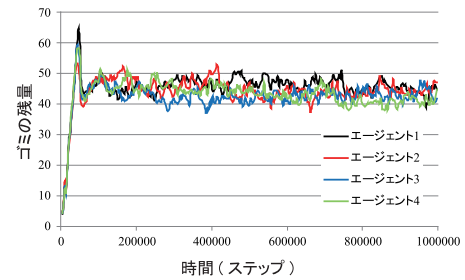


(b) 提案手法

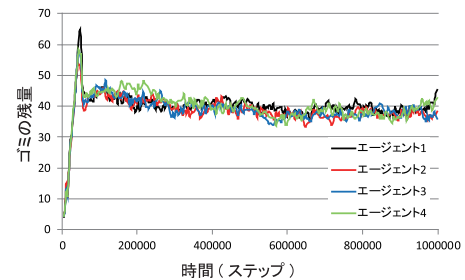
図 5 担当領域の広さの推移 (環境 1)

Fig. 5 The sizes of the responsible area (Env. 1).

提案手法によって、既存手法ならびに常時通信可能手法と比較してゴミの存在時間の総和 $D(V)$ は減少 (減少量は約 7%) したことが分かる。今回の提案手法が常時通信可能手法より効率が高い理由として、担当領域の揺らぎとそれに



(a) 既存手法



(b) 提案手法

図 6 担当領域のゴミの残量の推移 (環境 1)

Fig. 6 Remaining dirt in responsible area (Env. 1).

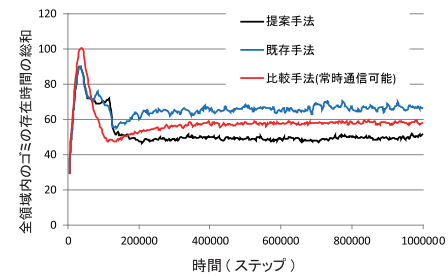


図 7 全ノードのゴミの存在時間の総和の推移 (環境 2)

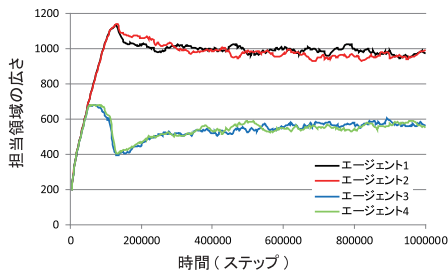
Fig. 7 Accumulating existence duration of dirt (Env. 2).

対する相互援助がある。詳細については他の環境での結果を含め、考察でまとめて述べる。

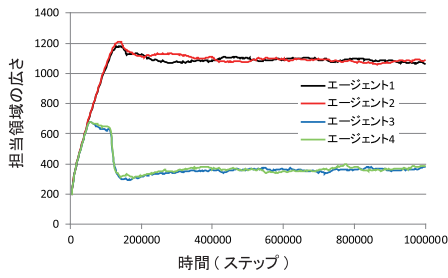
また、図 5(a) と (b) および図 6(a) と (b) との比較から提案手法では各担当領域の広さ、ゴミの残量の両方が減少している。提案手法では、記録した隣接エージェントの拡大力に基づいて余力 (あるいは負荷の度合い) を推定しているため、担当領域の拡大が抑制された結果であり、また不要な重なり排除がゴミの残量を減らしていると考えられる。本観点についても考察で述べる。

次に環境 2 と環境 3 をまとめて比較すると、その結果に大きな差があることが分かる。図 7 は、環境 2 における 3 手法の $D(V)$ の推移を、図 8 に提案手法と既存手法の $|V_t^i|$ の推移、図 9 に $L(V_t^i)$ の推移を示す。同様に環境 3 の 3 手法の $D(V)$ の推移を図 10 に、既存手法と提案手法の $|V_t^i|$ の推移を図 11 に、 $L(V_t^i)$ の推移を図 12 にそれぞれ示す。

図 7 から環境 2 においては、既存手法と比べて提案手法が $D(V)$ を減少させている (減少量は約 15%)。また常時通信可能手法と比較しても 7%ほど減少させている。他

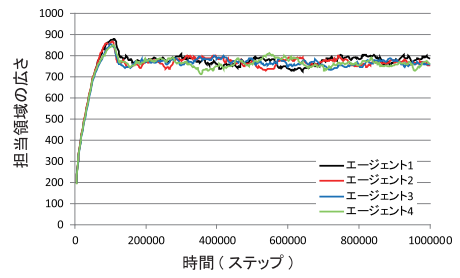


(a) 既存手法

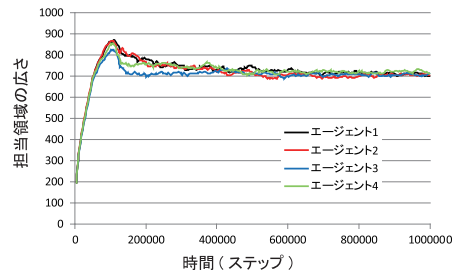


(b) 提案手法

図 8 各担当領域の広さの推移 (環境 2)
Fig. 8 The sizes of the responsible area (Env. 2).

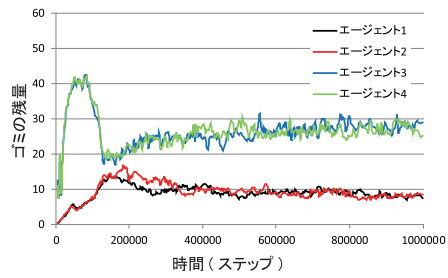


(a) 既存手法

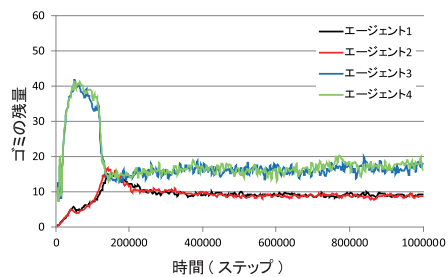


(b) 提案手法

図 11 各担当領域の広さの推移 (環境 3)
Fig. 11 The sizes of the responsible area (Env. 3).

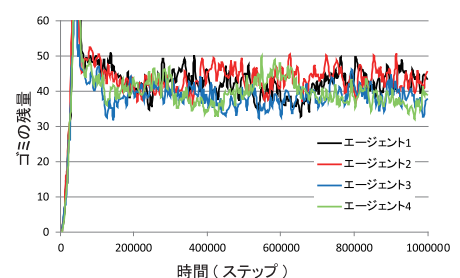


(a) 既存手法

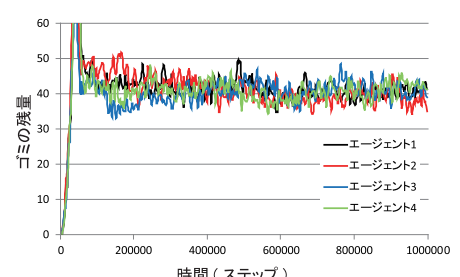


(b) 提案手法

図 9 各担当領域のゴミの残量の推移 (環境 2)
Fig. 9 Remaining dirt in responsible area (Env. 2).



(a) 既存手法



(b) 提案手法

図 12 各担当領域のゴミの残量の推移 (環境 3)
Fig. 12 Remaining dirt in responsible area (Env. 3).

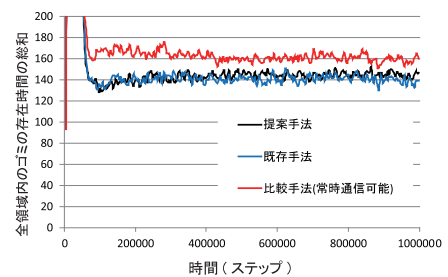


図 10 全ノードのゴミの存在時間の総和の推移 (環境 3)
Fig. 10 Accumulating existence duration of dirt (Env. 3).

方, 図 10 から環境 3 においては, 既存手法と提案手法では $D(V)$ に差はほとんどないが, 通信に制限があり重複領域が発生する既存手法・提案手法がともに常時通信可能手法より効率を向上させている. 図 8 (a), (b) を比較すると, 環境 2 の各担当領域の広さに関しては, 提案手法のエージェントは既存手法と比較して, エージェント 3 と 4 が担当領域を大きく減少させ, エージェント 1 と 2 は増加させている. これに合わせて, 図 9 (a), (b) の各担当領域のゴミの残量を比較すると, 提案手法では, 既存手法に比べて

エージェント 1, 2 の領域と 3, 4 の領域間の差が少なく、より均一な作業が実現できていることが分かる。

これは環境の差、特にエージェント 3, 4 の基地の目の前に汚れやすい領域があるため、その特徴をより強く反映できたことによる。環境 2 では、各担当領域の境界付近 (図 3 参照) には汚れやすい領域はない。しかし、通信の遅延によりその境界付近に重複領域が発生し、重複領域で複数のエージェントが作業する。既存手法では、他者を考慮した拡大行動を行う提案手法に比べ、その境界付近で複数のエージェントがともに清掃する重複領域が大きくなる。結果として、エージェント 3, 4 の負荷も大きくなり、基地前の汚れやすい領域に十分に着手できないため、作業効率が低下すると考えられる。常時通信可能手法との比較も含め詳しくは考察で述べる。

他方環境 3 では、境界領域に汚れやすい領域がある。その領域は重点的に訪れるべきであり、複数のエージェントが清掃すること自体は効率の向上につながる。学習が進むにつれてエージェントはゴミの発生頻度を学習しこの領域に訪問しやすくなり、その結果、通信できる可能性も高くなる。通信成功率への影響とその効果については次節で確認する。

5.3 重複領域の広さと通信成功率の比較

これまでの実験結果は、提案手法が既存手法に比べて作業負荷の観点から、担当領域をより公平にし、全体としての効率も向上させることができたことを示している。特に提案手法が効率を上げた要因として、担当領域の冗長性の減少、通信可能なタイミングの頻度の向上、などが推定される。以下では、これらの項目を調査する。

まず、各手法における担当領域の広さの和 $\sum_{a_i \in A} |V_t^i|$ を図 13 (a), (b), (c) に示す。なお、全領域の広さは $51 \times 51 = 2601$ であり、この差が重複領域となる。図 13 からどちらの手法でも冗長な重複領域が発生しているが、すべての環境において既存手法よりも提案手法の方が隣接エージェントとの重複領域の発生を抑えられたことが分かる。また、詳しく図を比較すると、提案手法では環境 1, 3 と比べ環境 2 の方が若干ではあるが重複領域が大きい。これは、汚れた領域が基地の目の前にあるのでそこに滞在する時間が長くなり、他のエージェントとの通信の機会が減ったためと考えられる。そこで通信の成功率を調査した。

通信の成功率については既存手法と提案手法で差はないと考えられるが、環境ごとには変化する。ここで提案手法における通信の成功率を、エージェントが Hello メッセージを出し、応答するエージェントが存在したときの確率とする。これは、基地に戻ったときは通信はせず通信間隔 $N_{com} = 1$ としているため、移動時間中の通信成功率ととらえられる。図 14 に各環境での通信成功率の推移を示す。また各環境ごとの後半の時刻 600,000 以降の通信成功

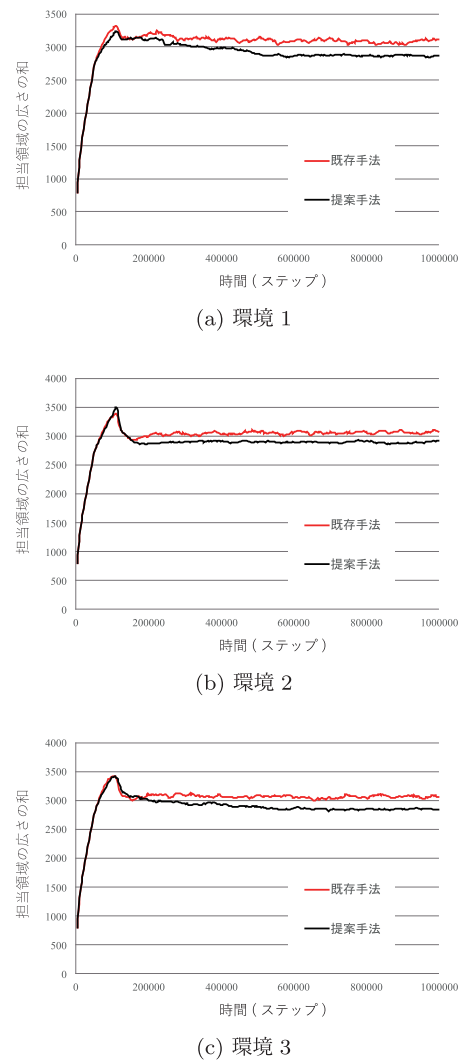


図 13 各環境における各手法での担当領域の広さの和
Fig. 13 Sum of the size of each responsible area in Envs. 1, 2, 3.

率平均は、環境 1 は 0.00954、環境 2 は 0.00654、環境 3 は 0.00799 であった。これらの図から通信可能な確率はそれほど高くはなく、長い間、重複領域を解消せずに行動することになる。環境 2 では、基地の前にある汚れやすい領域に高頻度で滞在するため、通信の成功率がやや低い。環境 3 では、境界付近の汚れやすい領域に頻繁に訪問するため、特定の相手とは通信成功の確率が上がる。しかし、反対の境界側の相手との通信頻度は減少し、総合的には通信成功確率は環境 1 ほど上がらない。環境 1 ではこのようなバイアスはなく、結果として成功率はほかより高くなったと考えられる。通信成功確率は重複領域の減少にとって重要な観点であるが、本実験では各環境間のごみ存在時間や重複領域の差に比べると環境ごとの通信成功確率の差はきわめて小さい。よって、提案手法による効率化や重複領域の差は、他の要因も大きく影響していると考えられる。

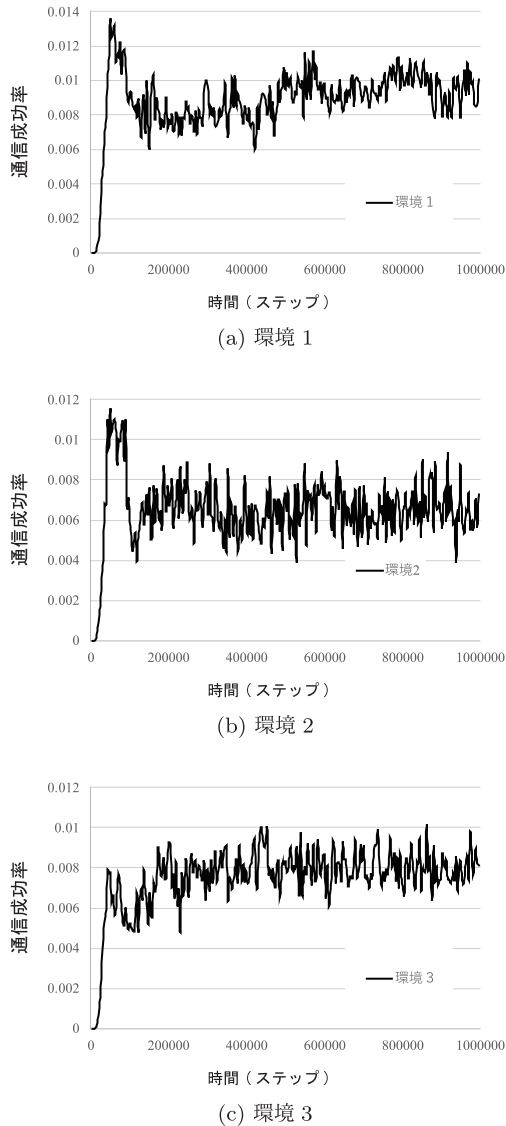


図 14 提案手法における隣接エージェントとの通信成功率
 Fig. 14 Success rate of communication with adjacent agent (proposed method).

5.4 考察

ここで提案手法が既存手法や常時通信可能手法と比べて効率を向上させた要因を考察する。分析の結果、効率向上の要因には大きく3つあり、それぞれの詳細を説明する。

5.4.1 重複領域の減少

第1の要因(要因1)は、5.3節でも述べたように、重複領域の減少である。重複があると各担当領域が広くなり、無駄な冗長性が生まれ回収効率が低下し、環境のゴミが増える。特に環境2の結果で示されたように、基地の前に汚れやすい領域のあるエージェントは担当領域が増えた分作業が分散し、担当領域内のゴミが増加した。実際に、図13に示すように、既存手法より担当領域の重複を低下させた提案手法は各環境において既存手法と少なくとも同等、あるいは明らかに清掃効率を向上させている。しかし、提案手法は重複領域のない常時通信可能手法よりも各環境にお

表 2 環境全体のゴミの残量の分散

Table 2 Variance of remaining dirt in each environment.

環境	手法	分散値
環境 1	既存手法	5.67
	提案手法	2.43
環境 2	既存手法	84.98
	提案手法	36.81
環境 3	既存手法	13.83
	提案手法	8.49

いて効率を向上させており、その理由の説明にはさらに別の要因もあることが示唆される。

5.4.2 環境の特徴：重複領域と重点領域の位置関係

第2の要因(要因2)として、その環境の特徴による例外的効果が考えられる。この要因は特に境界領域に汚れやすい場所がある環境3に影響している。前述したように、一般に担当領域の重複は効率の低下を招く。しかし、環境1と3では、重複領域の2手法間の差は同程度である(図13(a), (c))が、図4, 10を比較すると環境1では提案手法が効率を向上させているのに対し、環境3では同程度の効率となっている。同様に環境2と3の比較においても環境2の方が重複領域の差は小さいが(図13(b), (c)), 効率は環境2の提案手法の方が大きく向上している(図7, 10)。環境3のように汚れやすい領域が担当領域の境界付近にあり、かつその領域については単一エージェントだけでは十分ではない場合、重複が結果として負荷分散につながり、効率化の要因となりうる。このため、重複のない常時通信可能手法より、提案手法の効率が向上し、環境3においては既存手法も提案手法と同程度の効率となったと考えられる。

しかし、この要因は環境への依存度が大きい。領域分割の基本的な利点は担当領域の分割による競合の防止、効率の向上である。しかし、継続的な活動および多様な環境への適応性を考えると、従来の領域分割手法は柔軟性において課題があった。要因2は環境の一部重複による効率向上の可能性を示しているが、既存手法は環境2においては常時通信可能手法より効率が低下している。それに対し、提案手法はどの環境においても高い効率を示しており、図13(c)が示すように、環境3においても同等な効率の既存手法よりも重複領域のサイズは小さく抑えている。つまり、要因2の効果を利用しつつ、提案手法が環境によらず効率向上ができた要因が存在する。

5.4.3 領域拡大による相互援助効果

第3の要因(要因3)は、提案手法による領域拡大の際に負荷の高い方向へのみ拡大することにより、暗黙的に負荷の高いエージェントを助ける「適度な相互援助」の行動である。効率的な巡回のためには、領域分割によって各エージェントに負荷が適切に分担されることが望ましい。負荷の分散度合いを見るために、表2に既存手法と提案手法そ

それぞれの、環境全体のゴミの残量の分散値を示す。この表から、すべての環境において提案手法はゴミの残量の分散値が小さく、平均として環境を綺麗に保つだけではなく、巡回問題において重要な評価指標である均一的な清掃もできていることが分かる。分散値を低下させた理由は、環境2のように汚れやすい領域に応じて担当領域のサイズに差が発生した場合は明らかである。しかし、このような環境要因による担当領域サイズの差がなくても、提案手法は分散値を低下させ効率的な清掃を実現している。たとえば環境1, 3では、環境の特徴上、各担当領域に含まれるゴミが発生しやすい領域の比率はほぼ公平であり、図5, 図11から分かるように、4エージェントの担当領域の大きさに差はない。しかし、図4, 7, 10の比較からも分かるようにいずれの場合も、提案手法の効率が最も高い。

このように提案手法が環境内のごみ残量の分散値を低下させ、効率的かつ均一的な清掃を行えた理由を説明する。たとえば、環境1では、常時通信可能手法では基本的に重複領域はなく、環境3(要因2)のように境界領域に特にゴミの発生しやすい部分もないが、提案手法は常時通信可能手法より効率が上がる。常時通信可能手法では、拡大行動の直後に通信により重複領域を取り除く。そのとき、拡大力である $\xi(i, t)$ の値がエージェント間で相互に変わる*1。その結果、拡大した領域を取り返し合うというような揺らぎがわずかながら発生する。十分に学習時間が経過した後も拡張力は相互に上下に変化し、その拡大した領域は次のタイミングまで保存される。他方、提案手法では通信して得た近隣の拡張力の通信時点での値 ξ_j と自己の現時刻での拡張力 $\xi(i, t)$ を比較する。したがって、提案手法でも担当領域の重なりは発生するが、それはつねに自分より負荷が高いと推定した方向のみに向かう。さらに担当領域拡大により自己の拡張力も下がるため、通信が長期間できなくてもある時点で拡大行動は抑制される。これにより、ゴミの残量の分散値も減少する。通信ができなくても、負荷が高いと考えられるエージェントをそのエージェントの領域に対し一定の範囲で領域拡大することで助けながらも過度な援助にならないように拡大を停止することで、結果的に効率を向上できたと考えられる。

同様な理由により環境2の結果も説明できる。つまり提案手法では、既存手法に比べ不要な重複領域を減少させただけでなく、重複領域は発生させるものの、それが負荷の高いエージェントを助けるように働く。領域拡張行動は拡大力の差に基づくため、その差がなくなったときは拡大を止める。結果として、提案手法は拡大行動による重複は防がないが、適切な方向への適度な拡大行動により、効率化が達成できたと考えられる。なお現状では、障害物や環境

の形状、エージェントの能力に差があるときの相互援助は確認しておらず、これは今後の課題とする。

6. まとめ

本研究では、領域分割を用いたマルチエージェント巡回清掃において、通信範囲の制限によって生じる冗長性が引き起こす清掃効率の低下を回避する手法を提案した。既存研究[3]から、通信ができる機会が限られている環境では、通信による交渉だけでは重複領域を防ぐのには間に合わず、その環境全体の清掃効率の低下につながっていた。そのため、各エージェントが不要な重複領域の発生を抑えることが必要である。他方、すべてを独立に判断させるとノードの取りこぼしや清掃効率の低下を招くおそれもある。そこで本研究では、通信を行った際に隣接エージェントの情報を記録し、次に通信が可能となるまでの間、他者の清掃状況や負荷度合いを推定し、それを考慮しながら領域拡大を行うことで、整合性のある分業を行う手法を提案した。評価実験の結果、一部の負担の大きいエージェントの担当領域の過剰な拡張を防いだことで従来手法に比べて効率的かつ、より均等な領域分割が実現でき、提案手法の有用性を示せた。特に提案手法では、通信を常時可能とした場合でも発生する「担当領域の境界付近の取り合いによる振動」を和らげるように拡大方向を決定するため、既存研究だけでなく、常時通信可能とした手法よりも効率が向上した。

今後の課題としては、エージェントの多様性と環境の多様性を導入することと環境の大規模化が考えられる。領域分割法は隣接したエージェントどうしがやりとりをしていくものであり、そのような大規模環境では通信の制限もあるので、別のエージェントを仲介する形で領域のやりとりが必要と考えられる。また常時通信可能とした場合にも、同様な考えを導入し、さらに効率化ができると考えられる。

謝辞 本論文の修正にあたり、建設的なコメントをいただいた査読者の方々に感謝いたします。なお本研究は、JSPS 科研費(17KT0044)の助成を受けたものです。

参考文献

- [1] Sugiyama, A. and Sugawara, T.: Improvement of Robustness to Environmental Changes by Autonomous Divisional Cooperation in Multi-agent Cooperative Patrol Problem, *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection*, Demazeau, Y., Davidsson, P., Bajo, J. and Vale, Z., (Eds.), pp.259-271, Cham, Springer International Publishing (2017).
- [2] Sea, V., Kato, C. and Sugawara, T.: Coordinated Area Partitioning Method by Autonomous Agents for Continuous Cooperative Tasks, *Journal of Information Processing*, Vol.25, pp.75-87 (online), DOI: 10.2197/ipsjip.25.75 (2017).
- [3] 吉村 祐, 杉山歩未, 菅原俊治: 通信範囲に制限を持つマルチロボットの巡回清掃における効率的な領域分割法の提案, 電子情報通信学会技術研究報告(人工知能と知

*1 $\xi(i, t)$ の書き換えは、基地に戻ったときのみ行う。しかし、拡大行動は1巡回あたり1回であり、次の拡大行動のときは書き換えることに注意。

識処理研究会), Vol.115, No.478, pp.7-12 (2016).

[4] Ahmadi, M. and Stone, P.: Continuous area sweeping: A task definition and initial approach, *Proc. 12th Int. Conf. Advanced Robotics (ICAR 2005)*, pp.316-323, IEEE (2005).

[5] Othmani-Guibourg, M., Fallah-Seghrouchni, A.E., Farges, J.L. and Potop-Butucaru, M.: Multi-agent patrolling in dynamic environments, *2017 IEEE Int. Conf. Agents (ICA)*, pp.72-77 (online), DOI: 10.1109/AGENTS.2017.8015305 (2017).

[6] Chevaleyre, Y.: Theoretical Analysis of the Multi-agent Patrolling Problem, *Proc. Intelligent Agent Technology*, pp.302-308 (2005).

[7] Yoneda, K., Sugiyama, A., Kato, C. and Sugawara, T.: Learning and relearning of target decision strategies in continuous coordinated cleaning tasks with shallow coordination, *Web Intelligence*, Vol.13, No.4, pp.279-294 (online), DOI: <http://dx.doi.org/10.3233/WEB-150326> (2015).

[8] Sampaio, P.A., Ramalho, G. and Tedesco, P.: The Gravitational Strategy for the Timed Patrolling, *Proc. 2010 22Nd IEEE Int. Conf. Tools with Artificial Intelligence - Volume 01, ICTAI '10*, pp.113-120, IEEE Computer Society (online), DOI: 10.1109/ICTAI.2010.24 (2010).

[9] Nasir, A., Salam, Y. and Saleem, D.Y.: Multi-Level Decision Making in Hierarchical Multi-agent Robotic Search Teams, Vol.1 (2009).

[10] Ahmadi, M. and Stone, P.: A multi-robot system for continuous area sweeping tasks, *Proc. 2006 IEEE Int. Conf. Robotics and Automation (ICRA 2006)*, pp.1724-1729 (2006).

[11] Wiandt, B., Simon, V. and Kokuti, A.: Self-organized graph partitioning approach for multi-agent patrolling in generic graphs, *IEEE EUROCON 2017 - 17th Int. Conf. Smart Technologies*, pp.605-610 (online), DOI: 10.1109/EUROCON.2017.8011183 (2017).

[12] Elor, Y. and Bruckstein, A.: Multi-agent graph patrolling and partitioning, *Proc. 2009 IEEE/WIC/ACM Int. Joint Conf. Web Intelligence and Intelligent Agent Technology - Volume 02*, pp.52-57, IEEE Computer Society (2009).

[13] Kato, C. and Sugawara, T.: Decentralized Area Partitioning for a Cooperative Cleaning Task, *Proc. 16th Int. Conf. Principles and Practice of Multi-Agent Systems (PRIMA-2013)*, pp.470-477 (2013).

[14] Jain, U., Tiwari, R., Majumdar, S. and Sharma, S.: Multi Robot Area Exploration Using Circle Partitioning Method, *Int. Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012), Procedia Engineering*, Vol.41, pp.383-387 (online), DOI: <https://doi.org/10.1016/j.proeng.2012.07.188> (2012).

[15] Mao, T. and Ray, L.E.: Frequency-Based Patrolling with Heterogeneous Agents and Limited Communication, *Proc. 2011 Int. Conf. Artificial Intelligence (ICAI 2011): WORLDCOMP '11*, pp.3-9 (2011).

[16] Hahnel, D., Burgard, W., Fox, D. and Thrun, S.: An Efficient FastSLAM Algorithm for Generating Maps of Large-Scale Cyclic Environments from Raw Laser Range Measurements, *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS 2003)*, Vol.1, pp.206-211 (2003).

[17] Wolf, D. and Sukhatme, G.: Mobile Robot Simultaneous Localization and Mapping in Dynamic Environments, *Autonomous Robots*, Vol.19, No.1, pp.53-65 (2005).

[18] Liu, Y. and Sun, Y.: Mobile robot instant indoor map building and localization using 2D laser scanning

data, *2012 Int. Conf. System Science and Engineering (ICSSE)*, pp.339-344 (online), DOI: 10.1109/ICSSE.2012.6257203 (2012).

[19] Hennes, D., Claes, D., Meeussen, W. and Tuyls, K.: Multi-robot Collision Avoidance with Localization Uncertainty, *Proc. 11th Int. Conf. Autonomous Agents and Multiagent Systems - Volume 1, AAMAS '12*, Richland, SC, IFAAMAS, pp.147-154 (2012).

[20] von der Osten, F.B., Kirley, M. and Miller, T.: Robust Anticipatory Stigmergic Collision Avoidance in Multi-agent Systems, *Proc. 2014 Int. Conf. Autonomous Agents and Multi-agent Systems, AAMAS '14*, Richland, SC, IFAAMAS, pp.1403-1404 (2014).



吉村 祐

2016年早稲田大学基幹理工学部情報理工学科卒業。2018年同大学大学院基幹理工学研究科情報理工・情報通信専攻修士課程修了。マルチエージェントシステム、セキュリティ問題等に関心を持つ。



杉山 歩末 (学生会員)

2014年早稲田大学基幹理工学部情報理工学科卒業。2016年同大学大学院基幹理工学研究科情報理工・情報通信専攻修士課程修了。現在、同専攻博士課程に在学中。日本学術振興会特別研究員 (DC-1)。マルチエージェントシステム、行動科学等に関心を持つ。人工知能学会、IEEE各会員。



菅原 俊治 (正会員)

1982年早稲田大学大学院理工学研究科数学専攻修士課程修了。同年日本電信電話公社武蔵野電気通信研究所基礎研究部入所。1992~1993年マサチューセッツ大学アマースト校客員研究員。現在、早稲田大学理工学術院基幹理工学研究科情報理工・情報通信専攻教授。マルチエージェントシステム、分散人工知能、機械学習、インターネット等の研究に従事。博士 (工学)。日本ソフトウェア科学会、電子情報通信学会、人工知能学会、IEEE、ACM、AAAI各会員。