

制約付きロバストマルチチーム形成におけるエージェントの重要度を考慮した割り当て手法

寺澤 遼†

藤田 桂英‡

†東京農工大学 工学部 情報工学科

‡東京農工大学大学院 工学研究院 先端情報科学部門

1 はじめに

ミッション指向のチーム形成問題 [1] は、限られたリソースを与えられた中で、相互に作用するミッションを達成するために、可能な限り最良のチームを形成する問題である。しかし、チーム形成問題を実世界の問題に適用しようとした場合に、考慮すべき要素が複数存在する。例えば、救助隊の編成において、ミッション中にエージェントが負傷する可能性がある環境では、エージェントを失った場合に、ミッションを継続できる能力が重要である。また、ロボットが複数の会社から提供される場合に、同じチームに競合他社のロボットを配属することが憚られる可能性がある。

本研究では、ミッション指向の制約付きロバストマルチチーム形成問題 (MOCRMTF) と呼ぶ新しいフレームワークを定義する。この問題は、沖本ら [2] が提案した、ミッション指向のロバストマルチチーム形成問題に制約を新たに加えたものである。制約とは、エージェントの集合から、同じチームに配属することを禁止するものであり、ミッションの遂行能力以外の要素をチーム形成問題で考慮することができるようにするためのものである。

チームから k 個のエージェントを削除しても、与えられたミッションを達成できるチームが残っている場合、チームのロバスト性は (負ではない整数 k に対して) k となる。マルチチームの要素である全てのチームのロバスト性が k であるとき、マルチチームのロバスト性は k となる。チームのロバスト性を固定し、ロバスト性の目標を満たす最もコストが小さいチームを見つける最適化問題を考えることができる。

本研究では、MOCRMTF のフレームワークを定義し、ロバスト性を固定した最適化問題に対するアルゴリズムを提案する。実験では、提案したアルゴリズムが、探索するエージェントの順番により、実行時間が変化する結果が得られた。

2 提案手法

ロバスト性を固定した最適化問題のためのアルゴリズムは、ロバスト性、制約を満たしたマルチチームの内、最もコストが小さなものを出力することを保証している。なお、このアルゴリズムでは、探索木を作成するために、エージェントの順序が辞書的に与えられていることを前提としている。

このアルゴリズムでは、はじめのエージェントをルートノードとして、順番にエージェントを子ノードとして、各チームへの割り当てまたは、どのチームにも割り当てないものとして探索木を形成する。割り当てを行う場合には四つのチェックを行い、不合格だった場合に探索空間の剪定を行う。以下に詳細を述べる。

スキルのチェック

エージェントがチームのミッションに関連するスキルを持たず、チームに貢献できない場合、そのチームに割り当てず探索木を剪定する。

制約のチェック

エージェントの制約を確認し、同じチームに割り当てることが禁止されている相手が、チーム内に既に割り当てられていた場合、そのチームに割り当てず探索木を剪定する。

コストのチェック

マルチチームのコストが、チームにエージェントを追加することにより、現在の解のコストを超えた場合には、そのチームに割り当てず探索木を剪定する。

ロバスト性のチェック

エージェントのあるチームに割り当てた場合の、現在の部分的な割り当てのロバスト性を確認する。このチェックでは、現在の割り当てがロバスト目標を満たすマルチチームとなる可能性を検出する。 t 個のチームのロバスト性を増加させるためには、少なくとも t 体のエージェントが必要である。したがって、現在の部分的なチームについて、有効なマルチチームを得るために必要なエージェントの最小数を計算することができる。未探索のエージェントの数が、この値よりも少ない場合、有効なマルチチームが見つからないことを意味する。そこで、チームに割り当てず探索木の剪定を行う。また、チェックの際に、現在の割り当てがロバス

Allocation Method Considering Agent Importance in Constrained Robust Multi-Team Formation

†Department of Computer and Information Sciences, Faculty of Engineering, Tokyo University of Agriculture and Technology

‡Division of Advanced Information Technology and Computer Science, Institute of Engineering, Tokyo University of Agriculture and Technology

ト目標を見たしていた場合、以下の操作を行う。

現在の部分的なチームの割り当てがロバスト目標を満たしている場合、割り当てを行う際にコストをチェックしているため、現在の解よりもコストが小さいことは保証されている。したがって、解を現在の割り当てに更新し、残りの子ノードの探索を行う。

3 実験

3.1 実験設定

今回の実験では、エージェントのコストを固定化、つまり、どのチームに割り当てても、同じコストとなることとした。これは、エージェントと目的地の距離の最小化を目指すような問題ではなく、エージェントを購入する費用を最小化するような問題へ適用することを考慮した問題設定となる。以下に実験設定をまとめる。

- エージェントの数:18
- エージェントのコスト:[10, 11, ..., 100] の範囲内から一様ランダム
- ミッションチームのミッション: チーム毎に三つ。また、同じミッションを持つ別のチームは存在しない。
- エージェントのスキル: 一様ランダムに選ぶが個数は次の項に従う。
- 各エージェントのスキルの数:[1, 2, ..., 5] の範囲内でランダムに選ばれる
- 制約数:10
- ミッションチームの数:2

実験では、ロバスト目標とエージェントの割り当て順を変更する。ロバスト性は [1, 2, 3] の範囲で変更し、エージェントの割り当て順はランダムな順、コスト順、スキル数順、制約数順を用いる。シミュレーションは同一条件で 30 回ずつ行った。

3.2 実験結果

表 1 に平均実行時間を示す。スキルの昇順/降順により実行時間に大きく差が発生するのは、エージェントの重要度が関連していると考えられる。ロバスト性が小さいほど、必要なエージェント数が少なくなるため、割り当てのパターン数に対して、ロバスト性の条件を満たすパターンの割合は大きくなる。これは、ロバスト性が小さいほど、条件を満たすパターンが見つけやすくなることを意味する。スキル数が多いエージェントはスキル数が少ないエージェントよりも、ロバスト性を満たしやすく、ミッションチームにとって重要なエージェントといえる。ロバスト性が小さい場合には条件を満たす割合が多く、スキル数が多い重要なエー

表 1: 平均実行時間 (単位:秒)

ロバスト性	1	2	3
ランダム	1.06	16.40	29.13
コスト昇順	1.17	19.57	33.55
コスト降順	0.96	8.10	24.25
スキル数昇順	0.87	11.28	12.66
スキル数降順	1.17	16.94	56.33
制約数昇順	0.92	15.83	49.68
制約数降順	0.93	10.37	12.66

ジェントから割り当てることで、すぐに条件を満たすことができるため、コストのチェックを行えるようになり、実行時間が短くなることが考えられる。

ロバスト性が大きい場合には、条件を満たす割合が少ないため、探索を十分に進めなければ、条件を満たすことが難しい。今回のアルゴリズムでは、割り当てが遅い後ろのエージェントほど、先に複数のチームに割り当てるため、スキル数の多い重要なエージェントを後ろにすることで、ロバスト性を満たしやすくなり、実行時間が短くなると考えられる。

また、コストが大きいエージェントよりも小さいエージェントの方が、ミッションチームに重要であるといえるが、実験結果では、コストを降順で割り当てた方が実行時間が短くなっている。これは、コストの重要度はスキルの重要度に比べて低く、コストのチェックがある場合に、より探索木の根に近い部分で剪定することによる、探索空間の減少の影響の方が大きいためだと考えられる。

4 まとめ

本研究では、制約付きロバストマルチチーム形成において、フレームワークを定義し、ロバスト目標を固定した最適化問題に対するアルゴリズムを提案した。エージェントの割り当ての順番で実行時間が変化することから、エージェントには重要度があり、重要度を考慮し、割り当て手法を決定することで、実行時間が短くなることがわかった。

参考文献

- [1] Somchaya Liemhetcharat and Manuela M. Veloso. Modeling and learning synergy for team formation with heterogeneous agents. In *AAMAS 2012*, pages 365–374, 2012.
- [2] Tenda Okimoto, Tony Ribeiro, Damien Bouchabou, and Katsumi Inoue. Mission oriented robust multi-team formation and its application to robot rescue simulation. In *IJCAI 2016*, pages 454–460, 2016.