

ロボカップレスキューシミュレーション上でのMax-Sum アルゴリズムの適用に関する検討

福重 芳樹^{*1} 幸塚 義之^{*2} 伊藤 暢浩^{*1}
Yoshiki Fukushima Yoshiyuki Koduka Nobuhiro Ito

^{*1}愛知工業大学情報科学部情報科学科
Faculty of Information Science in Aichi Institute of Technology

^{*2}愛知工業大学大学院経営情報科学研究科
Graduate School of Business Administration and Computer Science

The aim of the RoboCup Rescue Simulation is to find effective strategies for dealing with disaster situations. And it is study faster of robotics and AI, contribution to society. The system provides a platform for exploring ideas for a multi-agent system. There are some research problem in RCRS, we think about the task allocation problem in this paper. In this paper, we regard as DCOP the task allocation problem of fire brigade, applying the max-sum algorithm. By using this algorithm, it is possible to allocation for task efficiently.

1. はじめに

近年頻発する地震災害に対する取り組みの一つに RoboCup Rescue Simulation(以降, RCRS) がある. RCRS の目的は災害救助を題材としたマルチエージェントシステムを通して AI やロボティクスなどの研究を促進することと, その成果による社会貢献である.

RCRS とは, 地震災害による火災や建物の倒壊, 道路の閉塞が発生した仮想都市において, 災害救助隊が市民の救助や火災の消火及び道路の閉塞の除去をおこなうシミュレーションである. このシミュレーションでは消防隊, 啓開隊, 救急隊, それらを統括する司令部を仮想都市に配置し, エージェントとして自律行動をおこなわせ, 救助戦略や協調行動を通して地震災害による被害の軽減を目指している.

実際の災害救助活動では災害救助隊の数は有限である. そのため有限である災害救助隊を効率よく災害現場に向かわせることができれば, 多くの人命を救助することができる. これは RCRS でも同様であり, このような問題は条件の複雑なタスク割り当て問題とみなすことができる. 消火活動の場合, 1つの火災に多くの消防隊が向かうと他の消火活動が間に合わなくなってしまう. つまり, 火災ごとに適切な数のエージェントが割り当てられる必要がある. このようなマルチエージェントの協調問題は, 分散制約最適化問題によって定式化することができる. そこで, 本研究では, 分散制約最適化問題として消防隊による消火問題に注目し, その解法の1つである Max-Sum アルゴリズムの有効性を検証する.

本研究の目的は, RCRS に Max-Sum アルゴリズムを適用しその有効性を検証することである. 対象とする分散制約最適化問題を複数の火災に対する消防隊の割り当て問題とすれば, 消防隊の利得を定義して, これを最大化するようにすれば良い. これを実現するためには利得の計算式を設計する必要がある.

そこで解を求めるための評価関数を定義し, 次に評価関数に用いる変数を決定する.

2. 分散制約最適化問題 (DCOP)

DCOP は文献 [2] より, 以下のように定義される. エージェントの集合 S , 変数の集合 X , 制約の集合 C . 利得関数の集合 O により定義される. エージェント i は自身の変数 x_i をもつ. x_i は離散有限集合 D_i に含まれる変数値をとる. 制約 (i, j) は x_i と x_j の間に制約があることを示す. 制約で関係する2変数についての, ある割り当て $\{(x_i, d_i), (x_j, d_j)\}$ の利得は, 二項利得関数 $r_{i,j} : D_i \times D_j \rightarrow \mathbb{R}$ により定義される. 全ての変数への割り当て Z に関して, $R(Z) = \sum_{(i,j) \in C, \{(x_i, d_i), (x_j, d_j)\} \subseteq Z} r_{i,j}(d_i, d_j)$ を二項利得関数の合計値として, 問題の最適解 Z^* は $R(Z)$ を最大化する割り当てである. また $R(Z^*)$ を最適値と呼ぶ.

DCOP には, 厳密解法と非厳密解法があり, 厳密解法は最適解を得られるが計算量が膨大になってしまうため RCRS に適用するには不向きである. 一方非厳密解法は最適解が得られると決まっている訳ではないが近似解を得ることができ, 計算も高速であるため RCRS の環境に向いている. そのため本稿では非厳密解法を用いる. 非厳密解法の代表例には, DSA[3] や Max-Sum アルゴリズムが挙げられる.

DSA は, そのエージェントの状態のみをメッセージとして送信するため, 通信コストは低い. しかし, 近傍のエージェントの状態のみに基づいて状態を決定するため局所最適解に陥りやすく, 解の精度は低くなってしまう.

Max-Sum アルゴリズムも同様に近傍のエージェントから受け取ったメッセージを基に状態を決定するが, そのメッセージは送信元のエージェントの近傍の情報も含まれた評価関数である. そのため DSA よりも広範囲の情報を得ることができ, 解の精度も向上する. そのため, 本研究では Max-Sum アルゴリズムを使用する.

3. Max-Sum アルゴリズム

Max-Sum アルゴリズムは情報理論の分野で用いられている確率伝搬に基づく手法である. Max-Sum アルゴリズムは関数ノード F と変数ノード X からなるグラフ上でメッセージを伝搬する.

Max-Sum アルゴリズムは, 変数ノード X が各エージェントの状態を, 関数ノード F がその利得を求める評価関数を持ち, 各エージェントがメッセージの交換をおこなうことで「評

連絡先: 福重 芳樹, 愛知工業大学, 愛知県豊田市八草町八千草 1247, TEL:(0565)48-8121, FAX:(0565)48-0277, minaduki1234@gmail.com

価値関数を最大化するような状態の集合を計算する」ことを実現する。

一般的な分散制約最適化問題のグラフ表現を、関数ノード F と変数ノード X からなるグラフとして表す必要がある。

一般的な分散制約最適化問題のグラフは図 1 のようにノードがエージェント、エッジが制約を表す。対して Max-Sum アルゴリズムは、図 2 のように各エージェントは関数ノード F と変数ノード X を持つ。このとき、メッセージの伝搬はエッジで接続されているエージェント同士でしかおこなわれない。

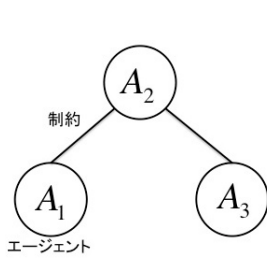


図 1: DCOP のグラフ構造

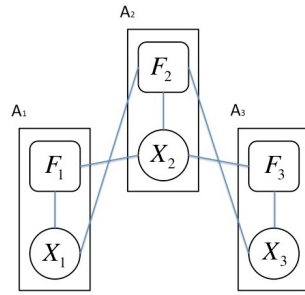


図 2: Max-Sum アルゴリズムのグラフ構造

4. RCRS での消火活動

RCRS での火災には以下のような特徴がある。

- 火災が発生している建物に近い建物へと火災が広がる (延焼)
- 火災が発生している建物の面積によって延焼の速さが変化する
- 火災が発生している建物に対して消火活動をおこなうと延焼を防ぐことができる
- 消防隊が火災に対して放水する量によって火災の弱まり方が変化する
- 面積が大きい建物での火災は消火に多くの水が必要になる
- 火災が発生している建物に入ると火傷を負い徐々にダメージを受ける

また、消火活動をおこなう際に割り当てを決定する方法には以下の 2 種類がある。

- 各消防隊からの情報を基に司令所が割り当てを決定
- 各消防隊同士が通信をおこなうことで割り当てを決定

本研究では、上記の割り当て決定方法を実装し、実験をおこない比較する。ここでは司令所が割り当てを決定する方法での実験結果について考察を示す。

5. RCRS 環境への適用

本研究では、Max-Sum アルゴリズムを RCRS に適用するための手法として、消防隊エージェントの消火能力を最大化する問題を解く。

はじめに、Max-Sum アルゴリズムは静的環境での動作をおこなうので、RCRS のような動的環境に対応させる必要がある。本研究では各ステップ毎に Max-Sum アルゴリズムの計算をおこなうことで対応する。

エージェントを $a \in A$ とし、各火災を $f \in F$ として表し、エージェントの割り当てを $y_f = \{a | a \in A, a \rightarrow f\}$ ただし、 $a \rightarrow f$ はエージェント a が火災 f に割り当てられていることを示す、割り当ての集合を $\mathbf{Y} = \{y_f | f \in F\}$ として表すと、 $u(\mathbf{Y})$ は火災ごとの利得の合計となる。つまり、本研究での目的は、利得を最大化する評価関数の割り当て集合 $\mathbf{Y}$ を見つけることである。

$u(\mathbf{Y})$ は各火災を消火することで得られる利得 $u_f(\mathbf{Y})$ の合計となる。よって、 $u(\mathbf{Y}) = \sum_{f \in F} u_f(\mathbf{Y})$ と表すことができる。ここで、 $u_f(\mathbf{Y}) = e_f(\mathbf{Y}) - r_f(\mathbf{Y})$ とすると、第 1 項は火災 f に対する最大消火能力、第 2 項は火災 f を消火するために必要なコストとして表せる。

$e_f(\mathbf{Y})$ を評価する際、火災はすべて同じわけではないため、各火災に価値 v_f を与える。 v_f は燃焼度から得る。RCRS では燃焼度は 1 3 の整数値をとり、火災が発生したばかりの状態では 1 をとり、時間が経つにつれて 2, 3 へと変化していく。また、燃焼度が低い建物ほど他の建物へと燃え広がりやすいため、燃焼度が低いものを優先して消火しなければならない。よって、 $v_f = (3/\text{燃焼度})$ とする。燃焼度の最大値を燃焼度で割ることで無限小数になることを防ぎ、計算をおこないやすくしている。次に、複数の消防隊を同じ火災に割り当てる場合を考えると、 $e_f(\mathbf{Y})$ は v_f と f に割り当てられているエージェントの数を乗算することで表せる。ここで割り当てられているエージェントの数を $n_f(\mathbf{Y})$ とすると、 $e_f(\mathbf{Y})$ は

$$e_f(\mathbf{Y}) = s \cdot v_f \cdot n_f(\mathbf{Y})$$

として表すことができる。ここで s は定数である。 $r_f(\mathbf{Y})$ に関しては、火災 f に割り当てられた消防隊のコストを合計すればよい。

$$r_f(\mathbf{Y}) = \sum_{a \in A} r_{af}(y_f)$$

ここで $r_{af}(y_f)$ は火災 f にエージェント a が取り組むときのコストを表す。コストは火災までの距離を d_{af} とする。よって消防隊毎のコストは以下のように表すことができる。

$$r_{af}(y_f) = \begin{cases} \nu d_{af}(y_f) & a \text{ がタスク } f \text{ をおこなう} \\ 0 & \text{上記以外} \end{cases}$$

ν は定数を表す。

これで定数を決定すれば Max-Sum アルゴリズムを用いて割り当てを決定することはできるが、特定の火災に協力できるエージェントの数には限界があるため、火災の面積をある定数 w で割った値 t_f を閾値として定義する。この閾値より多くのエージェントを火災に割り当てた場合のペナルティを以下に示す。

$$\kappa \cdot \{\max(0, n_f(\mathbf{Y}) - t_f)\}^\gamma$$

上式のすべての式をまとめると以下ようになる。

$$u(\mathbf{Y}) = e_f(\mathbf{Y}) - \sum_{a \in A} r_{af}(y_f) - \kappa \cdot \{\max(0, n_f(\mathbf{Y}) - t_f)\}^\gamma \quad (1)$$

次に、実験をおこなって $s, w, \nu, \kappa, \gamma$ の値を決定する。

6. 定数の決定

実験には RCRS 環境を使用する。ここでは RCRS に DCOP を適用するための式の決定をおこなう。そのため本研究では (1) 式の消火能力を最大化しなければならない。

まず式の評価をおこなうため DCOP を最適化問題として単純化する。そのため消防隊同士が通信をして割り当てを決定するのではなく、司令所が各消防隊に割り当てを送信することでタスク割り当て問題の解を得る。ここで、司令所はマップ全体を把握できるものとする。

定数は $\nu, s, w, \kappa, \gamma$ があるが、 s, ν は利得とコストの大きさを合わせるための係数であるため、エージェント毎にコストの違いが出やすい 3 桁に設定する。そうすると、 s は 100, ν は 10^{-3} と設定できる。残りの w, κ, γ は実験で値を求める。

まず w は、火災の面積に対して必要なエージェントの数を決定するために、ランダムに初期出火を発生させその火災の面積に対して消防隊を何体割り当てれば消火ができるかを調べる。検証には VC のマップを使用した。

表 1: 必要面積に対するな割り当て数

火災の面積	必要エージェント数 (試行回数 20 回)
0-100	1
100-200	1-2
200-300	2-4
300-400	4-5
400-500	6-7

表 1 より、面積を 70 で割った値が消火に必要なエージェントの数に近いことがわかる。よって $w=70$ とする。

次に γ, κ はエージェントを必要最低限な数を割り当てたあとに余分に割り当てた場合のペナルティである。5 章で述べたように消火に協力できるエージェントの数は限られているため、多く割り当てても消火活動が早く終わる訳ではない、そのため火災までの距離が近くタスクに割り当てられていないエージェントを 1 体だけ割り当ててもよいとする。必要最小限のエージェントが一つの火災を消すのに約 10 ステップほどかかるため、5 ステップ以内に到着できるエージェントがいる場合にのみ余分に割り当てをおこなうことで消火を早く終わらせるようにする。

エージェントは 1 ステップでコスト換算約 30 の移動ができるため、5 ステップ以内に到着できる距離はコスト換算 150 以下になる。すなわち、 κ は、余分に割り当てるエージェントのコストが 150 を超えるとき割り当てないようにする。すなわち、コストと κ を足した値が最大消火力 300 を超えるように設定する必要がある。よって、 κ は 150 となる。 γ はエージェントが 2 体以上余分に割り当てられることがないようにするために、消火をすることによって得られる利得を超えるように設定する必要がある。1 体の消防隊の最大値消火力は 300 となるため、 $150 \times 2^\gamma$ が 300 を超えるように設定すると、 $\gamma \geq 2$ 。よって $\gamma=2$ とする。Max-Sum アルゴリズムに用いる $e_f(\mathbf{Y}), r_{af}(y_a), (\kappa \cdot [\max(0, n_f(\mathbf{Y}) - t_f)]^\gamma)$ の式は、

$$e_f(\mathbf{Y}) = 100 \cdot v_f \cdot n_f(\mathbf{Y})$$

$$r_{af}(y_f) = \begin{cases} 10^{-3} d_{af}(y_f) & a \text{ がタスク } f \text{ をおこなう} \\ 0 & \text{上記以外} \end{cases}$$

$$\kappa \cdot \{\max(0, n_f(\mathbf{Y}) - t_f)\}^\gamma = 150 \cdot \{\max(0, n_f(\mathbf{Y}) - t_f)\}^2$$

$$\text{ただし, } t_f = \frac{\text{面積}}{70}$$

となった。

7. 実験

作成した式を消防隊司令所に実装し割り当てを決定する。その際消防隊が火災に対して効率的に割り当てられているかを比較実験をして調べる。比較対象として、2013 年 RCRS の世界大会優勝チームの GUC ArtSapience(以下、GUC)を用いる。GUC は k-means++ と c-means という 2 つのクラスタリング手法を組み合わせで割り当てを決定している。これらは、DCOP とは異なる手法であるため、比較対象に向いていると考えられる。

実験環境を以下に表す。

- エージェントは消防隊と消防隊司令所のみ
- 無線通信の範囲は無制限
- 司令所は火災の位置を把握できる
- エージェントの配置場所や出火場所是不変

マップの情報を表 2 に、火災の情報を表 3 に表す。また、提案手法と GUC の実験結果を表 4 と表 5 に示す。

表 2: マップの設定 1

マップ	初期出火数	消防隊	マップの面積
Kobe	3	23	170,000
VC2	2	4	180,000
Berlin	3	26	3,080,000

表 3: マップの設定 2(火災)

マップ	出火面積 1	出火面積 2	出火面積 3
Kobe	266	93	196
VC2	32	24	-
Berlin	663	1083	453

各マップでは 2~3 箇所出火が火災が発生するものとし、その面積は表 3 の出火面積 1~3 に示したものとなる。

表 4: 実験結果:提案手法による割り当て

マップ	出火 1 に 割り当てた数	出火 2 に 割り当てた数	出火 3 に 割り当てた数	合計
Kobe	5	2	3	10
VC2	2	1	-	3
Berlin	1	1	4	6

表 5: 実験結果:GUC による割り当て

マップ	出火 1 に 割り当てた数	出火 2 に 割り当てた数	出火 3 に 割り当てた数	合計
Kobe	9	3	6	18
VC2	1	2	-	3
Berlin	0	1	7	8

表 6: 提案手法 Score

マップ	Score
Kobe	1.998
VC2	1.998
Berlin	1.984

表 7: GUC Score

マップ	Score
Kobe	1.999
VC2	1.996
Berlin	1.981

8. 考察

実験の結果, 提案手法と GUC は双方とも Kobe と VC2 のマップでの消火活動を完了した. また, 表 4,5 から Kobe のマップでの火災に対して提案手法のほうが割り当てる消防隊の数が減っていることがわかる. これより, 提案手法は GUC より効率よくタスク割り当てが行えていることがわかる.

また, Berlin のマップでは双方の消防隊は消火活動を終えることができなかった. そして, 表 3 より, それぞれの出火面積が大きいため割り当てるエージェント数が増えることが予測できる. また, 表 4 より, 実際に割り当てているエージェントの数を見ると割り当てているエージェント数が少ないことがわかる. 以上より提案手法は割り当てがうまく行えていないことがわかる. これは, 火災の近くに消防隊が少なかったため消火能力よりコストが大きくなってしまったからであると考えられる. しかし, Kobe や VC2 のマップでは割り当てがおこなえているため, マップの大きさに合わせて消火能力とコストに係数を乗算することで対応することができると考えられる.

係数を求めるために VC2 のマップを使用したため, VC2 のマップ面積を 1 とした場合 Berlin のマップは約 17 倍となる. 提案手法では距離について考えていたため, この平方根を取り整数部である 4 がマップの長さの倍率となる. よって e_f と κ に 4 を乗算した結果割り当てるエージェント数と Score は表 8, 表 9 のようになった. この結果より, VC のマップの面積を 1 としてシミュレートするマップの平方根を乗算することで他のマップにも対処可能であることが分かった.

表 8: 提案手法による割り当て

マップ	出火 1 に 割り当てた数	出火 2 に 割り当てた数	出火 3 に 割り当てた数
Berlin	5	4	7

表 9: 提案手法 Score

マップ	Score
Berlin	1.987

9. まとめと今後の課題

本研究では, DCOP として RCRS を解く前段階として, Max-Sum アルゴリズム式の提案及び実験をおこなった. 今回提案した式で最適化問題として割り当て問題を解くことが可能となった. よって, 消防隊司令所に実装した計算式を用いた割り当て決定方法を各消防隊エージェントに適用することで DCOP を解くことが可能となる.

しかし, 提案した式は火災が発生した時点での解を得るため, 精度を上げるためには消防隊エージェントが火災現場に到達するのにかかるステップ数を基に火災の広がり方を予測して割り当てを決定することや, マップの規模に応じて計算式変化を加えること, 救急隊や啓開隊といったエージェントとの協調行動を考慮する必要がある.

Max-Sum アルゴリズムを動的環境に適応させると, 一度送信したメッセージでも再び送信してしまうことがあるため無駄な通信が発生してしてしまう. そのため, 無駄な通信をさせないための手法も考える必要がある.

参考文献

- [1] A.Petcu and N.R.Jennings A.Farinelli, A.Rogers. Decentralised coordination of low-power embedded devices using the max-sum algorithm in Seventh International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-08),2008.
- [2] 津本天太, 桜井裕子, 横尾真, 井上克己. 多目的分散制約最適化問題における厳密/非厳密解法の提案. 電子情報通信学会論文誌 D Vol.J96-D No.12 pp.2929-2938.
- [3] Guandong Wang Weixiong Zhang and Lars Wittenburg. Distributed stochastic search for constraint satisfaction and optimization. In AAI-02 Workshop on Probabilistic Approaches is Search,2002.
- [4] Dina Helal, Ahmed Abouraya, Noha Khater, Mina Fahmy, Fadwa Sakr, Salma Osama and Abdullrahman Elhussen RoboCup 2013 Rescue Agent Simulation Competition GUC ArtSapience Team Description Paper