

非同期な無記憶ファットロボットのための 凸包充填アルゴリズム

上村 智哉^{1,a)} 山内 由紀子^{1,b)}

概要: 本稿では, 2次元空間中に置かれた単純多角形であるオブジェクトの周囲に体積を持つロボット群を配置し, オブジェクトの凸包を形成する凸包充填問題を解く分散アルゴリズムを提案する. 各ロボットは匿名でメモリを持たず, 非同期的に観測・計算・移動を行う. 提案アルゴリズムでは, ロボットの体積が視界を遮断する影響を考慮し, ロボット同士の衝突を回避しながらオブジェクトの欠損部分にロボット群を配置する.

Distributed algorithm for filling the convex hull of an object by asynchronous oblivious fat robots

TOMOYA UEMURA^{1,a)} YUKIKO YAMAUCHI^{1,b)}

Abstract: We propose a distributed algorithm for filling the convex hull of an object by a group of robots each of which has a volume. The object is a simple polygon in 2D space. Each robot is anonymous and memory-less, and performs sensing, computation, and movement asynchronously. The proposed algorithm sends the robots to cracks with considering obstruction for sensing and avoiding collisions during movement.

1. はじめに

自律的に計算, 移動を行うモバイル計算主体群の分散協調がロボティクス, 理論計算機科学, 制御理論といった多様な分野で注目を集めている. 本稿では, モバイルロボット群モデル [1,3] に着目し, 体積を持つモバイルロボット群によって与えられたオブジェクトの凸包を充填し, 欠損部分を補充する凸包充填問題を考える.

モバイルロボット群モデルは, アドホックネットワークや無線センサーネットワークだけでなく, 分子ロボットや生物群, 化学反応系の分散計算モデルとして注目されており, 集合, 形状形成, 探索, 分割などの基本的な振舞いを実現する分散アルゴリズムやその限界が示されている [2]. これらのアプリケーションに共通する特徴は, 個々の計算主体は識別不能で低機能なことである. 本研究では, 非同

期的に動作する, メモリを持たない匿名のロボット群を想定する.

モバイルロボット群モデルに関する理論研究の多くは体積を持たないロボットを想定しているが, 本研究では円盤状の体積を持つファットロボットを想定する. ファットロボットは観測や移動の障害物となるため, 協調動作を実現するには衝突回避や同期などの工夫が必要となる. ファットロボットについては, 全ロボットを集める集合問題を解くアルゴリズムなどの研究が行われている [3]. 本研究では, 非同期な無記憶ロボット群が凸包充填問題を解くための分散アルゴリズムを提案する.

2. モデル

本稿では先行研究 [3] のファットロボットモデルを用いる. ロボットは二次元連続平面上を移動する半径1の不透明な円盤とする. このような体積を持つロボットをファットロボットと呼ぶ. ロボットは匿名 (識別不能) で, 共通のアルゴリズムに従って自律的に移動する. ロボット同士は

¹ 九州大学 工学部 電気情報工学科
Department of Electrical Engineering and Computer Science, School of Engineering, Kyushu University
a) uemura@tcs.inf.kyusyu-u.ac.jp
b) yamauchi@inf.kyushu-u.ac.jp

通信できない。さらにロボットは共通の座標系を持たず、各ロボットが固有の局所座標系を持つ。局所座標系の原点はそのロボットの現在位置(円盤状の体積の中心)であり、軸の向きは一致していないが、単位長は共通であるとする。さらに、ロボット群は時計回りの方向に合意を持つ。

ロボットは、観測-計算-移動の3つのフェイズからなるサイクルを繰り返す。

- 観測フェイズ：各ロボットが自身の局所座標系を用いて、観測フェイズ中のある時刻における他のロボットの位置を観測する。ロボットの円周とある点を結んだ線分上に他のロボットやオブジェクトが存在しない場合、ロボットはその点を観測でき、そうでない場合、観測できない。ロボットが他のロボットの円周上のある点を観測できる場合、そのロボットの中心の位置を観測できるとする。
- 計算フェイズ：観測フェイズで得られた情報をもとに移動先を計算する。このように、以前のサイクルの観測結果や計算結果などを計算に用いないロボットを無記憶ロボットと呼ぶ。
- 移動フェイズ：計算フェイズで得た移動先に移動する。ロボットは移動中に他のロボットと衝突した場合、即座に移動を終了する。ロボットは最小移動距離 δ を移動した後は、任意のタイミングで停止する可能性がある。しかし、目的地までの距離が δ 以下である場合、ロボットは必ず目的地に到達する。ロボットは δ の数値を知らないとする。

全てのロボットが任意のタイミングでサイクルを実行する実行モデルを非同期モデルと呼ぶ。本稿は非同期モデルを想定する。

凸包充填問題では、2次元空間に置かれた不透明なオブジェクトを考える。ロボットは凸包内部に入ることにはできない。オブジェクトは単純多角形であり、さらに、その凸包とオブジェクト自身の集合差は三角形の集合であるとする。これらの三角形すべてをファットロボットで充填する問題を凸包充填問題と呼ぶ。全ての三角形欠損部分が以下の3つの条件を満たしている時、オブジェクトの凸包充填が完了したと言う。

- 三角形内部を充填する各ロボットがオブジェクトの境界もしくは他のロボットと接しており、移動することができない。
- 欠損部分の、凸包の辺と共通部分をもつロボットは中心が三角形内部に存在し、他のロボットと接しているため三角形内部の任意の点に向かって動くことができない。
- 欠損部分の、凸包の辺に新たなロボットを追加しても、そのロボットの体積の中心は三角形内部もしくは、凸包の辺上に配置できない。

3. 凸包充填アルゴリズム

非同期な無記憶ファットロボットによって、凸包充填問題を解くアルゴリズムを提案する。

ある三角形領域に着目し、その三角形領域の角を図1に示すように A, B, C とする。ロボットには時計回り方向の合意があるので、頂点 A, B, C を合意することができる。線分 AB および線分 BC の長さは2以上とし、 $\angle ABC = \alpha$ とする。提案アルゴリズムでは、 $\frac{\pi}{3} \leq \alpha < \pi$ の三角形領域を考える。

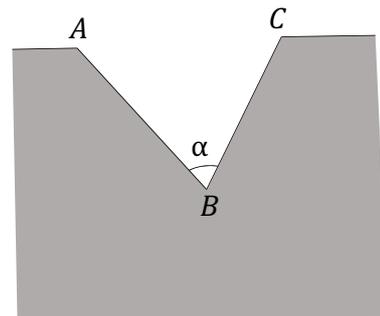


図1 三角形領域

提案する凸包充填アルゴリズムの概要を示す。まず、オブジェクト周辺に存在するロボット群は、三角形領域を観測可能であれば充填対象となる三角形領域の辺のうち、時計回りに後方の辺を延長した半直線上に整列する(図2)。

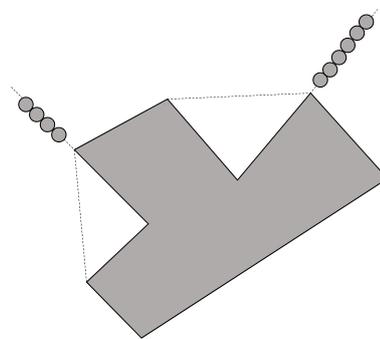


図2 ロボット群の整列.

整列したロボットは、頂点 C に最も近いロボットから順に三角形領域を充填する。この時、ロボットは辺 BC に沿って整列し、積み上がる(図3)。

三角形領域を観測できないロボットは、オブジェクトの周囲を時計回りに巡回し、三角形領域を探し出す(図4)。

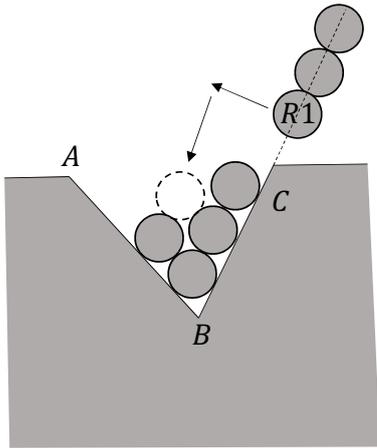


図 3 ロボット R1 の移動経路.

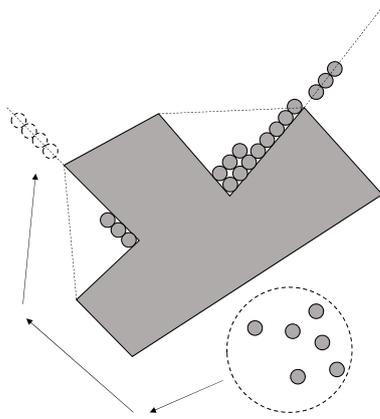


図 4 三角形領域の探索.

4. まとめと今後の課題

本稿では、オブジェクトの欠損部分が全て三角形であり、その角度 α が $\frac{\pi}{3}$ 以上の場合の凸包充填アルゴリズムを提案した。

今後の課題としては、一般のオブジェクトに対する凸包充填を解決できるように、三角形領域の角度 α が $\frac{\pi}{3}$ 未満の場合や、オブジェクトの欠損している部分の形状が任意の多角形である場合を考慮することが挙げられる。

参考文献

- [1] Masafumi Yamashita, Ichiro Suzuki, Characterizing geometric patterns formable by oblivious anonymous mobile robots. *Theoretical Computer Science*, 411, pp2433–2453, 2010.
- [2] Paola Flocchini, Giuseppe Prencipe, Nicola Santoro, *Distributed Computing by Oblivious Mobile Robots*, Morgan & Claypool Publishers, 2012.
- [3] Jurek Czyzowicz, Leszek Gasiemiec, and Andrzej Pelc, Gathering few fat mobile robots in the plane. *Theoretical Computer Science*, 410(6–7), pp. 481–499, 2009.