

Distributed algorithms for capturing a target using autonomous mobile robots

Yuto Kodama and Akihiro Fujiwara

Kyushu Institute of Technology

Iizuka, Fukuoka, 820-8502, Japan

E-mail: q676113y@mail.kyutech.jp, fujiwara@cse.kyutech.ac.jp

Abstract—Distributed autonomous mobile robots is a set of robots that move freely in a two dimensional plane. For the distributed autonomous mobile robots, a number of algorithms for shape formation problems have been proposed.

In this paper, we propose three kinds of distributed algorithms for capturing an escaping target using the autonomous mobile robots. In addition, we implement the proposed algorithms in the simulation environment, and verify performance of the proposed algorithms.

Keywords-autonomous mobile robots, distributed algorithm, capturing

I. はじめに

自律分散ロボット群 [1] とは、複数のロボットが自律的、協調的に動作することによって全体として一つの問題を解決するロボット群であり、耐故障性、拡張性に優れ、今後さらに発展していくと予測される研究テーマの一つである。

この自律分散ロボット群について、文献 [2][3] では自律分散ロボット群による通信ができない状況での追跡を行うアルゴリズムが提案されており、また、文献 [4] ではロボット群が特定の形状を形成する形状形成問題を解くアルゴリズムが提案されている。

そこで本研究では、複数の自律分散ロボット群が、逃走する1台のロボットを追跡し、捕獲するアルゴリズムを提案する。本研究の提案アルゴリズムは、それぞれの追跡ロボットが逃走するロボットに向かって移動する「単純アルゴリズム」、他の追跡ロボットの位置を参考にしながらターゲットを端に追い込んで捕獲を目指す「境界包囲アルゴリズム」、そしてこの2つのアルゴリズムを合わせた「ハイブリッドアルゴリズム」の3つである。

また、これらの提案手法をシミュレーション環境において実装し、自律分散ロボット群による追跡性能を検証する。

II. 自律分散ロボット群

A. 問題の定義

自律分散ロボット群とは、ロボット群全体を管理する機構を持たず、各ロボットがそれぞれ自律的に動作することにより、全体として目的を達成するロボット群のことである。

本研究、自律分散ロボットの集合 $R = \{r_0, r_1, \dots, r_{n-1}\}$ が、それぞれのアルゴリズムに従って、1台のターゲット r_t を追跡し、逃走ロボットに接触することを目的とする。なお、各自律分散ロボット（追跡ロボットとも呼ぶ）は後述のアルゴリズムより動作し、ターゲット（逃走ロボットとも呼ぶ）は人的に操作するものとする。

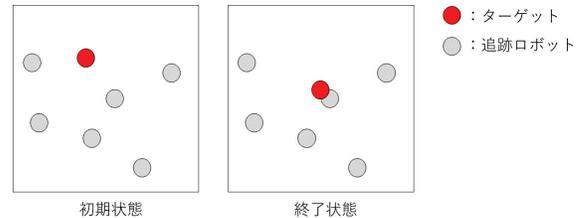


Fig. 1. 問題の例

Fig.1 に問題の入出力の例を示す。追跡ロボット群は、2次元平面上のランダムな位置から逃走ロボットの追跡を始める。1台以上追跡ロボットがターゲットに接触していれば、追跡終了とする。

本研究ではターゲットは追跡ロボットよりも高速に移動できるものとし、その速度は、追跡ロボットの速度の α 倍であるとする ($\alpha > 1$)。また、追跡ロボット群、及び、ターゲットの可動範囲は xy 座標系に平行な矩形領域であるとし、追跡ロボット群は、可動範囲の境界を認識できるものとする。

B. すべてのロボットに共通する定義

本研究におけるすべてのロボットは、2次元平面上で動作するものとし、共通の xy 座標系を持つものとする。また、2次元平面上の2点 $(x_1, y_1), (x_2, y_2)$ の距離 D を、以下のユークリッド距離を用いて定義する。

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

C. 追跡ロボットの定義

本研究における追跡ロボット群は、すべてのロボットに共通する定義に加えて、以下に示す条件を満たすものと仮定する。

- 追跡ロボットは互いに通信する能力はない。
- すべての追跡ロボットは同じアルゴリズムを実行する。
- 各追跡ロボットは、「待機」、「観測」、「計算」、「移動」という動作を1サイクルとして、サイクルを繰り返す。
- 各追跡ロボットは、以前のステップで得られた情報を記憶しておくことができない。
- 各追跡ロボットは、任意の観測において、自分の座標、全ての追跡ロボットの座標、ターゲットの座標、及び可動領域を知ることができる。

また、各ロボット $r_i (0 \leq i \leq n-1)$ の座標を点 $P_i = (x_i, y_i)$ とする。

D. ターゲットの定義

本研究ではターゲットの操作は人的に行われるものと仮定し、逃走時は常に人間の指定する場所を目的地として動作する。またターゲット r_t は1台であり、座標を点 $P_t = (x_t, y_t)$ とする。

E. ロボットの移動

各ステップにおけるロボットの移動方法を示す。追跡ロボットの場合、目的地は各ステップ毎に計算される座標であり、ターゲットの場合は、目的地は人間により指定される座標である。ただし、目的地への移動は1ステップで高々距離 d だけ移動できるものとする。

距離 d を x 方向、 y 方向それぞれに分解したものをそれぞれ dx, dy とする。座標 (x, y) にいる追跡ロボット r が、目的地 (X, Y) へ移動する場合の移動距離 dx, dy の導出方法を Fig.2 に示す。各追跡ロボットが1サイクルで移動できる距離は d なので、Fig.2 より、 dx 、及び、 dy は以下の式で計算できる。

$$dx = \frac{d * (X - x)}{\sqrt{(X - x)^2 + (Y - y)^2}}$$

$$dy = \frac{d * (Y - y)}{\sqrt{(X - x)^2 + (Y - y)^2}}$$

これより、 (x, y) にいるロボットが目的地 (X, Y) に向かって移動する場合、1サイクル後の位置 (x', y') は、目的地とのロボット間の距離を w とすると、以下の式で表される。

- $d < w$ の場合:

$$x' = x + dx$$

$$y' = y + dy$$

- $d \geq w$ の場合:

$$x' = X$$

$$y' = Y$$

III. 追跡アルゴリズム

本論文において、追跡ロボットは以下のサイクルを実行する。

待機: 追跡ロボット群は何もしない。

観測: 追跡ロボット群は、自分の座標、全ての追跡ロボットの座標、ターゲットの座標、及び可動領域の情報を入手する。

計算: 観測で得た情報をもとに計算を行い、目的地を設定する。

移動: 計算した目的地に向かって2.5節で説明の移動方法で移動する。

以下では、提案する3つのアルゴリズム（単純アルゴリズム、境界方位アルゴリズム、及び、ハイブリッドアルゴリズム）の概要を示す。

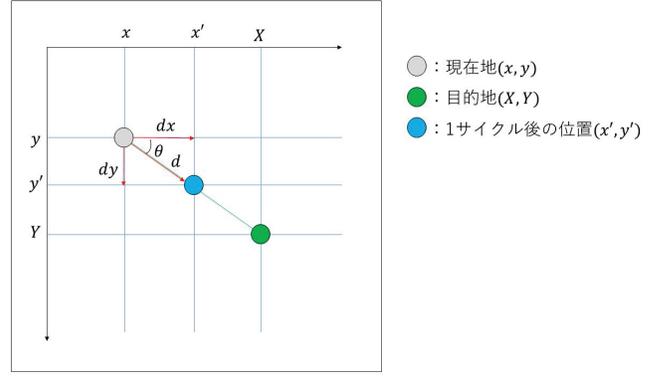


Fig. 2. 移動距離の算出

A. 単純アルゴリズム

単純アルゴリズムは、全ての追跡ロボットが、ターゲットを目的地として追跡するアルゴリズムである。各サイクルにおいて、以下のように目的地を設定することにより実行される。

(単純アルゴリズム)

観測において得られたターゲットの座標を (x_t, y_t) とする。このとき、目的地の座標 (X, Y) について、 $X = x_t, Y = y_t$ とする。

B. 境界包囲アルゴリズム

境界方位アルゴリズムは、追跡ロボット群により可動領域の端にターゲットを追い込むことを目的とするアルゴリズムである。本アルゴリズムは、以下のステップにより構成される。

(境界包囲アルゴリズム)

Step 1: 追跡ロボットの集合をターゲットを中心とする偏角順にソートする、このとき、偏角順にソートされた追跡ロボットの集合を $R_s = \{r_0, r_1, \dots, r_{n-1}\}$ とし、アルゴリズムを実行する追跡ロボットを r_i ($0 \leq i \leq n-1$) とする。

Step 2: 各ロボット r_i において、ターゲットと r_i を通る直線を引く。この直線により2つの半平面 P_L, P_R が得られる。(簡単のため、 P_L, P_R をそれぞれ左半平面、右半平面と呼ぶ。) 次に、 P_L, P_R に存在する追跡ロボットの数を計算し、それぞれ n_L, n_R とする。この n_L と n_R により、以下のような場合分けを行う。

- $n_L = 0$ の場合: このとき、半平面 P_L に存在するロボットは存在しない。そこで、ターゲットが可動領域の端から逃走することを防ぐために、以下のように仮目的地を設定する。

(1) ターゲットと自身を通る直線に対して、左方向に角度 90 をなし、自身を端点とする半直線 L を考える。

(2) 半直線 L において、自身から距離 d だけ離れた点を仮目的地 $P_s = (X_s, Y_s)$ とする。

上記の仮目的地の設定方法の概念図を Fig.3 に示す。これによりこの追跡ロボットを可動領域の端まで移動させる。

(3) 仮目的地 P_s が可動領域内の場合、目的地の座標 (X, Y) について、そのまま $X = X_s, Y = Y_s$

とする。仮目的地 P_s が可動領域外の場合は、以下の手順に従い、目的地の設定を行う。

- (3-1) ターゲットと自身を通る直線に対して、左方向に角度 θ をなし、自身を端点とする半直線 L_2 を考える。
- (3-2) 半直線 L_2 において、自身から距離 d だけ離れた点を目的地とする。

上記の目的地の設定方法の概念図を Fig.4 に示す。

このアルゴリズムにより、目的地を再設定することで、可動領域境界に到達した追跡ロボットはターゲットに向かって徐々に近づくようになる。

また、 θ の数値を大きくすると、ターゲットに向かって接近する移動距離が大きくなり、反対に数値を小さくすると、接近する移動距離が小さくなる。

- $n_R = 0$ の場合: $n_L = 0$ の場合と同様の方法で、半平面 P_R に目的地を設定する。
- それ以外の場合: ロボット r_i は、追跡ロボットの間隔を均等に保つことを目的として、以下の式のように、偏角順において前後のロボット r_{i-1}, r_{i+1} との中点を目的地 (X, Y) として設定する。

$$X = \frac{x_{i-1} + x_{i+1}}{2}, Y = \frac{y_{i-1} + y_{i+1}}{2}$$

なお、Fig.5 に本ステップの概念図を示す。

C. ハイブリッドアルゴリズム

本アルゴリズムは、単純アルゴリズムと境界包囲アルゴリズムを合成したアルゴリズムである。ターゲットに最も近い1台のロボットのみ単純アルゴリズムを実行することにより、ターゲットの逃走が困難になることを目的とする。(ハイブリッドアルゴリズム)

ターゲットに最も近い追跡ロボットは単純アルゴリズムを実行し、それ以外の追跡ロボットは、境界包囲アルゴリズムを実行する。なお、境界包囲アルゴリズムの実行においては、ターゲットに最も近い追跡ロボットは、アルゴリズムの実行対象から除かれるものとする。

IV. 実験結果と考察

提案アルゴリズムを Java アプレットとして実装し、被験者3名に対して、追跡ロボットがターゲットを捕獲するまでの時間を計測した。なお、計測結果の値は各被験者に5回ずつ施行を行ってもらった結果の平均値である。

図6に実験結果の一部を示す。Fig.6は各アルゴリズムにおいてターゲットを捕獲するまでの逃走時間を表すグラフである。この結果が示すように、単純アルゴリズムより、境界包囲アルゴリズムやハイブリッドアルゴリズムの方が捕獲時間が短くなっていることがわかる。各々のロボットでターゲットだけを見て目的地を決定するよりも、他のロボットを見て協力しながらターゲットの捕獲を目指す方が、捕獲にかかる時間を減らすことができると考えられる。

また、 θ の値については、境界包囲アルゴリズムは θ の値によって捕獲までの時間が多少変化しているが、ハイブリッドアルゴリズムの方は θ の値に関わらず捕獲までの時間が安定している。境界包囲アルゴリズムの方は角度を小さくすることで包囲網を狭める時間が短くなるので、捕獲までの時間に変化が出るが、ハイブリッドアルゴリズムの

方は、最終的にロボットを捕獲するのは単純アルゴリズムのロボットなので包囲網を作るまでの時間にそれほど依存しなかったためではないかと考えられる。

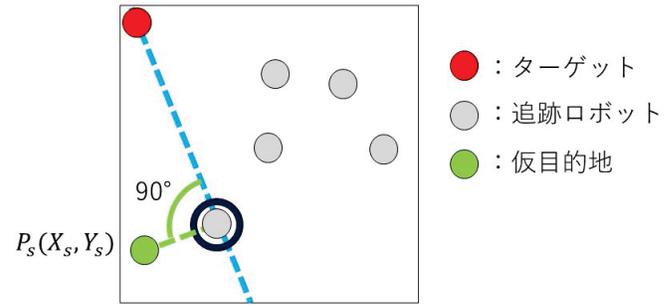


Fig. 3. 仮目的地の設定

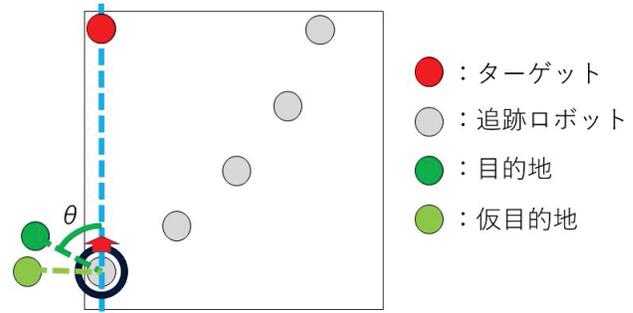


Fig. 4. 目的地の再設定

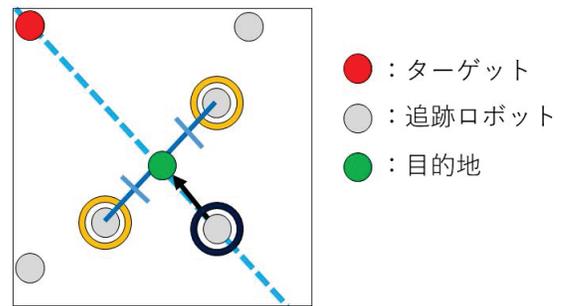


Fig. 5. 中点を目的地とするロボットの例

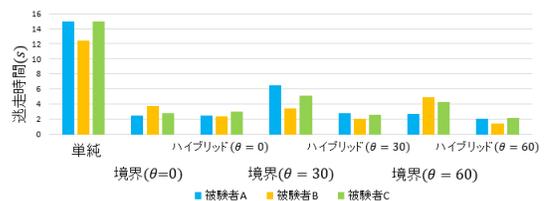


Fig. 6. アルゴリズムの実験結果

V. まとめと今後の課題

本研究では、自律分散ロボット群において、ターゲットの捕獲を目標とする捕獲アルゴリズムを提案した。中でもハイブリッドアルゴリズムはターゲットの捕獲に有用であるということを示すことができた。

今後の課題として、より実環境にあわせたシミュレーションの構築として、障害物の導入や、ロボットの視界範囲の設定などを行い、その上でのアルゴリズムの考案することなどが挙げられる。

REFERENCES

- [1] I. Suzuki and M. Yamashita: "A Theory of Distributed Anonymous Mobile Robots –Formation and Agreement Problems," *SIAM J. Computing* 28, 4, 1347–1363(1999).
- [2] 宮本淳史. "視界制限状況における自律分散ロボット群による追跡アルゴリズム", 九州工業大学卒業論文, 2009
- [3] 田崎隼斗. "自律分散ロボット群における障害物を考慮した追跡アルゴリズム", 九州工業大学卒業論文, 2015
- [4] 山中, 伊藤, 片山, 犬塚, 和田. "軸の方向に関する共有知識をもたない自律分散ロボット群に対する形状形成アルゴリズム", *電子情報通信学会論文誌*, Vol. J88-D1, No. 4, pp. 739-750, 2005.