移動障害物を回避するための軌道生成アルゴリズム

Trajectory Planning Algorithm for a Mobile Robot to Avoid Moving Obstacles

○ 学 新原諒子 (慶大) 学 井上貴公 (慶大) 学 今村元 (慶大)
学 小島佑太 (慶大) 正 山崎信行 (慶大)

Ryoko NIIHARA, Keio University, niihara@ny.ics.keio.ac.jp
Takayuki INOUE, Keio University,
Hajime IMAMURA, Keio University,
Yuta KOJIMA, Keio University,
Nobuyuki YAMASAKI, Keio University

Nowadays, we can see the use of mobile robots in various environments close to human life such as offices, hospitals. So it is important to solve the problem of trajectory planning under the environment which has moving objects like human beings. This paper proposes an algorithm of trajectory planning combined potential method and particle control for a mobile robot to avoid moving objects. With this algorithm, a mobile robot can move safely under the continuous dynamic environment that has moving objects.

Key Words: Mobile Robot, Obstacle Avoidance, Potential Method

1 序章

最近では、製造業ばかりでなく医療、福祉、災害救助、エンターテインメントなど幅広い分野でロボットの導入が検討されている。用途が拡大するにしたがい、ロボットにも様々な機能が要求される。そのひとつに環境の認識や障害物の回避がある。これらの機能なくしてロボットが複雑な環境下で安全に動作することは望めない。

これまで、静的、すなわち自分（ロボット）以外の障害物が固定されている既知の環境を対象とした障害物回避問題については様々な方法が考案されている。代表的な方法として、ロードマップ法、セル分割法、ポテンシャル法などがある [?]。しかし、実環境には人間のような移動する障害物が存在するため、動的に変化する環境に対する柔軟に適應できる軌道生成が不可欠である。そこで、本論文では、ポテンシャル法と Particle Control [3] の考え方に基づき、より現実世界に近い動的、すなわち障害物の位置が時間とともに変化する環境の中で、移動ロボットが障害物に衝突することなく安全な軌道を生成することのできる軌道生成アルゴリズムを提案する。

以下では、まず2節でポテンシャル法について、3節では Particle Control について説明する。続く4節で本論文で提案するアルゴリズムについて説明し、5節で提案アルゴリズムのシミュレーション評価について述べ、6節まとめるを行う。

2 ポテンシャル法

ポテンシャル法では、擬似的に山や谷のようなポテンシャル場を作る。そして、その斜面の勾配に従って仮想的な力を働かせることにより、ロボットを目標地点に到達させるという方法である。

このポテンシャル場は、ゴールに向かってロボットを引っ張る引力ポテンシャルと、障害物からロボットを遠ざけようとする斥力ポテンシャルから構成される。ポテンシャル場はポテンシャル関数によって決定される。そこで、本論文では、Kahtib [2] らによって提案されたポテンシャル関数を用いる。ゴールにより作ら

れる引力ポテンシャル U_d は次式により定義される。

$$U_d(\mathbf{x}) = \frac{1}{2} k_p (\mathbf{x} - \mathbf{x}_d)^2 \quad (1)$$

次に、障害物からの斥力ポテンシャル U_o は次式で定義される。

$$U_o(\mathbf{x}) = \begin{cases} \frac{1}{2} \eta \left(\frac{1}{\rho(\mathbf{x})} - \frac{1}{\rho_0} \right)^2 & \rho(\mathbf{x}) \leq \rho_0 \text{ のとき} \\ 0 & \rho(\mathbf{x}) > \rho_0 \text{ のとき} \end{cases} \quad (2)$$

ここで、 $\mathbf{x}$ はロボットの現在の位置を表すベクトル、 $\mathbf{x}_d$ は目的地を表すベクトル、 η は正の重み係数、 $\rho(\mathbf{x})$ は $\mathbf{x}$ から障害物への最近距離を表し、 $\rho = 0$ は正の定数である。 U_o は 0 以上で、ロボットが障害物領域に近づく程無限大に近くなり、ロボットから障害物への距離が ρ_0 以上のとき 0 になる。従って、全体のポテンシャル関数 U は次式で定義される。

$$U = U_d + U_o \quad (3)$$

このとき、式 (3) で表されるポテンシャル関数の最急降下方向 $\mathbf{d}$ は、 x, y の 2 次元平面の場合、

$$\mathbf{d} = -\Delta U = - \left[\frac{\partial U}{\partial x} \quad \frac{\partial U}{\partial y} \right]^T \quad (4)$$

と表せる。時刻 t でのロボットの位置 $\mathbf{x}_t$ の最急降下方向 $\mathbf{d}$ を時刻 t から $t+1$ までのロボットの軌道とすると、時刻 $t+1$ でのロボットの位置 $\mathbf{x}_{t+1}$ は、

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \alpha \cdot \mathbf{d}(\mathbf{x}_t) \quad (5)$$

と表せる。 α は、1[step] の降下量を決める係数である。

このように、ポテンシャル法では、各時刻 t でのロボットの位置において最急降下方向を求めるだけで、軌道を生成することができる。そのため、計算時間が短く、ロボットを動かしながらオンラインで計算できる。

アルゴリズム

- 与えられたゴールの位置, 障害物の位置, 大きさを基にポテンシャル関数 U を決定する.
- ロボットの初期位置からゴールするまで, 以下の (a), (b) の操作を繰り返す.
 - 時刻 t におけるロボットの位置 $\mathbf{x}_t$ から式 (4) を用いて, 最急降下方向 $\mathbf{d}(\mathbf{x}_t)$ を出す.
 - 最急降下方向 $\mathbf{d}(\mathbf{x}_t)$ をロボットの軌道とし, ロボットを移動させる. 次の時刻 $t+1$ におけるロボットの位置 $\mathbf{x}_{(t+1)}$ は, 式 (5) により次式で表される.

$$\mathbf{x}_{(t+1)} = \mathbf{x}_t + \alpha \cdot \mathbf{d}(\mathbf{x}_t) \quad (6)$$

3 Particle Control

Particle Control [3] は, ロボットの周りに重みを持つ多数の粒子を配置し, 各粒子に対して軌道を生成し, その重み平均をとることにより最適な軌道を生成するという手法である.

アルゴリズム

- ロボットの初期位置 x_0 から初期分布に従い, 粒子の初期位置 $\mathbf{s}_0^{(i)}, \dots, \mathbf{s}_0^{(N)}$ を生成する.
- 各粒子 $i = 1, \dots, N$ について以下の (a), (b) のステップを実行する.
 - 時刻 $t = 0, \dots, T$ について次の操作を実行し, 時刻 0 から T までの軌道を計算する.
 - 粒子の位置 $\mathbf{s}_t^{(i)}$ より, 時刻 t から $t+1$ までの軌道 $\mathbf{r}_{s,t}^{(i)}$ を計算する. すると, 時刻 t から $t+1$ までの粒子の位置 $\mathbf{s}_{t+1}^{(i)}$ は次式で表される.

$$\mathbf{s}_{t+1}^{(i)} = \mathbf{s}_t^{(i)} + \mathbf{r}_{s,t}^{(i)} \quad (7)$$

- 評価関数 h により, 粒子 i の初期位置から時刻 T までの位置 $\mathbf{s}_{0:T}^{(i)}$ を評価し, 粒子の重み $w^{(i)}$ を計算する.

$$w^{(i)} = h(\mathbf{s}_{0:T}^{(i)}) \quad (8)$$

- 各粒子 i の重み $w^{(i)}$ の和 $W = \sum_{i=1}^N w^{(i)}$ を計算する.
- 時刻 $t = 1, \dots, T$ について, 各粒子 i の経路 $\mathbf{r}_{s,t}^{(i)}$ の重み付き平均を計算し, 時刻 t から $t+1$ までのロボットの経路 $\mathbf{R}_t$ を求める.

$$\mathbf{R}_t = \frac{1}{W} \sum_{i=1}^N (w^{(i)} \cdot \mathbf{r}_{s,t}^{(i)}) \quad (9)$$

式 (9) より, 時刻 $t+1$ のロボットの位置 $\mathbf{x}_t$ は, 次式で表される.

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{R}_t \quad (10)$$

4 ポテンシャル法と Particle Control を組み合わせた軌道生成アルゴリズムの提案

この節では, 移動障害物が存在し時間とともに動的に変化する複雑な環境において, 移動ロボットが安全に走行するための, ポテンシャル法と Particle Control を組み合わせた軌道生成アルゴリズムを提案する.

ロボットの位置の周囲に多数の粒子を分布し, それぞれの粒子についてポテンシャル法を用いて数ステップ後まで軌道を生成する. そして, 各粒子の軌道の長さや障害物の近さ (危険度) で評価を行い, 粒子の重さ (重要度) を決定し, それを用いてロボットの軌道を生成する.

4.1 各粒子の軌道生成

式 (5) より, 時刻 t から $t+1$ までの各粒子 i の軌道 $\mathbf{r}_{s,t}$ は次式で定義される.

$$\mathbf{r}_{s,t}^{(i)} = \alpha \cdot \mathbf{d}(\mathbf{s}_t^{(i)}) \quad (11)$$

式 (7) より, 時刻 $t+1$ における各粒子 i の位置 $\mathbf{s}_{t+1}^{(i)}$ は, 次式で定義される.

$$\mathbf{s}_{t+1}^{(i)} = \mathbf{s}_t^{(i)} + \alpha \cdot \mathbf{d}(\mathbf{s}_t^{(i)}) \quad (12)$$

ここで, $\mathbf{d}(\mathbf{s}_t^{(i)})$ は, 時刻 t の粒子の位置 $\mathbf{s}_t^{(i)}$ でのポテンシャル関数の最急降下方向, α は 1[step] の降下量を決める係数である.

4.2 各粒子の重み

本論文では, ロボットが障害物に当たらずに最短経路もしくは最短時間でゴールすることが重要となる. ここで, 経路とポテンシャル関数の関係に着目する. ポテンシャル関数によりゴールに近い位置のポテンシャルは小さい. よって, 経路上のポテンシャルの総和を比較したとき, 小さいほど短い距離でゴールできる経路であることが分かる. また, 障害物の位置に近い位置のポテンシャルは大きくなる. よって, 経路上のポテンシャルの総和がおおきいことは, 障害物に近く危険な経路であることも示している. 従って, 経路上のポテンシャルの総和は, 小さいほどより最適な経路だと言える.

そのため, 各粒子の重み $w^{(i)}$ は, 各粒子 i のポテンシャル和の中から一番大きいポテンシャル和 $\max(p^{(i)})$ を導き, $\max(p^{(i)})$ から各粒子のポテンシャル和 $p^{(i)}$ を引き算することで決定することとする.

$$w^{(i)} = \max(p^{(i)}) - p^{(i)} \quad (13)$$

4.3 粒子の軌道の重み付き平均

ロボットの軌道は各粒子の経路の重み付き平均により計算する. 各粒子 i の軌道 $\mathbf{r}_{s,t}$ の重み付き平均を計算し, 時刻 t から $t+1$ までのロボットの軌道 $\mathbf{R}_t$ を求める.

$$\mathbf{R}_t = \frac{1}{W} \sum_{i=1}^N w^{(i)} \alpha \cdot \mathbf{d}(\mathbf{s}_t^{(i)}) \quad (14)$$

4.4 アルゴリズム

- 各粒子 i の軌道 $\mathbf{r}_{s,t}$ とポテンシャル $p^{(i)}$ を計算する. そこで, 各粒子 $i = 1, \dots, N$ について以下の (a), (b), (c) の

ステップを実行する. 以下では軌道を生成するときの時刻を毎回 0 とする.

- (a) ロボットの初期位置 x から正規乱数により, 粒子 i の初期位置 $s_0^{(i)}$ を計算する.
- (b) 粒子 i の初期位置 $s_0^{(i)}$ をポテンシャル関数 U に代入することで, その位置のポテンシャル $p_0^{(i)}$ を計算する.

$$p_0^{(i)} = U(s_0^{(i)}) \quad (15)$$

- (c) 粒子がゴールに着くか, 時刻 t が 0 から, 決められた時刻 T になるまで (i) (ii) を繰り返す.

(i) ポテンシャル法によって, 各粒子の軌道を求める.

$$s_{t+1}^{(i)} = s_t^{(i)} + \alpha \cdot \mathbf{d}_t^{(i)} \quad (16)$$

- (ii) 時刻 $t+1$ の粒子の位置 $s_{t+1}^{(i)}$ でのポテンシャル $U(s_{t+1}^{(i)})$ を計算し, 時刻 t までの軌道上のポテンシャルの総和 $p_t^{(i)}$ と足し合わせることで, 時刻 $t+1$ までのポテンシャルの総和 $p_{t+1}^{(i)}$ を求める.

$$p_{t+1}^{(i)} = U(s_{t+1}^{(i)}) + p_t^{(i)} \quad (17)$$

2. 求めたポテンシャルの総和 $p^{(i)}$ をもとに, 各粒子 i について重み $w^{(i)}$ と各粒子の軌道の重み付き平均 $\mathbf{R}_t$ を計算する. そこで, 各粒子 $i = 1, \dots, N$ について以下の (a), (b), (c) のステップを実行する.

- (a) 各粒子 i のポテンシャルの総和の中から一番大きいポテンシャルの総和 $\max(p^{(i)})$ を導き, $\max(p^{(i)})$ から各粒子のポテンシャルの総和 $p^{(i)}$ を引くことで, 重み $w^{(i)}$ を計算する.

$$w^{(i)} = \max(p^{(i)} - p^{(i)}) \quad (18)$$

- (b) 各粒子 i の重みの和 $W = \sum_{i=1}^N w^{(i)}$ を計算する.

- (c) 粒子の軌道 $\mathbf{r}_{s,t}$ と各粒子の重み $w^{(i)}$ を用いて, 軌道の重み付き平均を計算し, ロボットの軌道 $\mathbf{R}_t$ を求める.

$$\mathbf{R}_t = \frac{1}{W} \sum_{i=1}^N (w^{(i)} \cdot \mathbf{r}_{s,t}^{(i)}) \quad (19)$$

3. ロボットを計算した軌道 $\mathbf{R}_t$ に従い, 移動させる.
4. ロボットが目的地に到達するまで 1, 2, 3, 4 を繰り返す.

5 評価

この節では, 静的環境, 及び移動障害物を含む動的環境の 2 つの場合について, ロボットの障害物回避シミュレーションの結果を述べる. 今回は, 提案アルゴリズムを用いた場合とポテンシャル法を用いた場合のシミュレーション結果を検証し, 提案アルゴリズムの有用性を示す.

5.1 静的環境でのシミュレーション

5.1.1 シミュレーション条件

障害物やゴールの位置はわかっている, 障害物の大きさはわかっているものとする. 環境は, スタート座標, ゴール座標, 障害物の中心座標を表 1 のように設定する. また, シミュレーシ

ョンでは, ロボットは生成した軌道とまったく同じ軌道を進むこととする.

また, 粒子の数は 250 個とし, ロボットの位置から平均 (0.0, 0.0), 分散 0.25 である正規分布にしたがい決める. そして, $T = 1, \dots, 30$ で, それぞれシミュレーションを行い, 軌道の長さを計測する.

スタート	(0.0, 10.0)
ゴール	(0.0, 70.0)
障害物 1	(-5.0, 47)
障害物 2	(3.0, 37.0)

表 1: シミュレーション条件 1

5.1.2 シミュレーション結果

生成された軌道を図 1 に示す. ポテンシャル法による軌道の長さは 72.5 となった. 提案手法による軌道の長さは, $T = 1$ から $T = 30$ までの平均すると, 70.1 となった.

図 2 に粒子の軌道の先読みステップ数と軌道の長さの関係を示す. 先読みステップ数が増えるにしたがって軌道の長さが短くなっているのがわかる.

図 1: 生成された軌道 ($T=10$)

図 2: T に対する軌道の長さ

5.2 移動物体を含む環境でのシミュレーション

5.2.1 シミュレーション条件

移動障害物の初期位置と速度、ゴールの位置はわかっていて、障害物の大きさはわかっていないものとする。環境は、スタート座標、ゴール座標、障害物の初期位置座標と速度を表 2,?? のように設定する。また、シミュレーションでは、ロボットは生成した軌道とまったく同じ軌道を進むこととする。

また、粒子の数は 250 個とし、ロボットの位置から平均 (0.0,0.0)、分散 0.25 である正規分布にしたがい決める。そして、 $T = 1, \dots, 30$ で、それぞれシミュレーションを行い、軌道の長さを計測する。

スタート	(0.0, 10.0)
ゴール	(0.0, 60.0)

表 2: シミュレーション条件 2

	初期位置	速度
障害物 1	(70.0, 55.0)	(-1.0, -0.5)
障害物 2	(30.0, 40.0)	(1.0, -1.0)
障害物 3	(0.0, 30.0))	(1.5, 0.2)

表 3: 障害物の初期設定

5.2.2 シミュレーション結果

ポテンシャル法による軌道と提案手法による軌道を図 3 に示す。ポテンシャル法では軌道生成に失敗しているのに対し、提案手法では移動障害物にぶつからずにスタートからゴールまで軌道生成できている。図 4 に粒子の軌道の先読みステップ数と軌道の長さの関係を示す。静的環境では、先読み回数が増えると軌道の長さが短くなっていたのに対し、動的環境では、先読み回数が増えても軌道の長さの短縮は見られなかった。

図 3: 生成された軌道 (T=10)

6 まとめ

本論文では、移動障害物の存在する動的環境下でも移動ロボットが障害物に衝突せずに安全に走行することを可能にするための、

図 4: T に対する軌道の長さ

ポテンシャル法と Particle Control を組み合わせた軌道生成アルゴリズムを提案した。今後は、ロボットや障害物の大きさや形も考慮したアルゴリズムへと発展させ、移動ロボットに実装し実験評価を行うことで、実環境における有効性を示していきたい。

謝辞

本研究の一部は独立行政法人新エネルギー・産業技術総合開発機構 (NEDO)、文部科学省グローバル COE プログラム「環境共生・安全システムデザインの先導拠点」に依るものであることを記し、謝意を表す。

文献

[1] 太田順, 倉林大輔, 新井民夫. ”知能ロボット入門-動作計画問題の解法-”. コロナ社, 2001.

[2] O. Kahtib. Real-Rime Obstacle Avoidance for Manipulators and Mobile Robots. *International Journal of Robotics Research*, Vol. 5, No. 1, pp. 90–98, 1986.

[3] Lars Blackmore, Asker Bektassov, Masahiro Ono, and Brian C. Williams. Robust, Optimal Predictive Control of Jump Markov Linear System using Particles. *LECTURE NOTES IN COMPUTER SCIENCE*, 2007.