Motion Planning for Obstacle Avoidance of Life Support Robots in Dynamic Environments

1. 緒言

要介護者の支援のために、人型の生活支援ロボットを開発 している. 飲食の提供や掃除など生活支援動作を行うには, 動的環境下でもロボットが安全に移動することが不可欠で ある.人とロボットが混在する動的環境下において,生活支 援ロボットは人の動きを阻害せず安全に自律走行すること が必要である.生活環境は居間や廊下など大小さまざまな空 間がある.これらの環境で人とロボットが自由に行き来する と、問題となる2種類の環境が存在する.(I) 死角からロ ボットがいきなり現れロボットと人が衝突する環境.(Ⅱ) ロボットが人の行き先を塞ぎ,お互いに身動きの取れない環 境. そこで,環境Ⅰ,Ⅱに対して,有効な局所経路計画を提 案する.環境Iの場合は、死角から現れた動的障害物(人) と衝突回避するため,動的障害物の数秒後の位置を予測する 局所経路計画 Iを提案する.シミュレーションにより従来の DWA アルゴリズムと比較し、提案手法 I の優位性を示す. また環境Ⅱの場合は、ロボットが人に道を譲るため、動的障 害物の速度を予測する局所経路計画Ⅱを提案する.シミュレ ーション及び実機実験で提案手法Ⅱの有効性を示す.

自律走行の関連研究 2.

自律走行には、ある環境に対して目的地までの経路を計画 する大局的経路計画と、ロボットの近傍の障害物のみを考慮 する局所的経路計画がある.大局的経路計画は、目的までの 最短経路を計画できるダイストラクタ法や A*アルゴリズム がよく知られている.大局的経路計画は環境全体で経路計画 するため、動的障害物に対して繰り返し処理を行うのは、計 算時間の観点から困難な場合がある.そのため,動的障害物 には局所的経路計画で衝突回避することが望まれる.局所的 経路計画には、人工ポテンシャル法、Vector Field Histogram, DWA アルゴリズム⁽¹⁾等が挙げられる.しかし,これらの局所 的経路計画には、動的障害物が考慮されていない. DWA ア ルゴリズムは,軌道候補群を生成し,静的障害物に衝突しな い軌道を残す. その軌道軌道候補群から評価関数を基に最適 解を求める.提案手法Ⅱは、静的障害物だけでなく、予測し た動的障害物の数秒後の障害物にも,衝突しない軌道を残す.

3. 全方向移動型ロボット

使用した全方向移動型ロボットの外観を図1に示す.全方 向移動型ロボットは四輪駆動でメカナムホイールを搭載し ており,任意の方向へ移動可能である.車体前方部分に測域 センサーUST-10LX を搭載している.



Fig.1 Omnidirectional robot

知能機械工学コース 知能ロボティクス研究室 1255011 後藤 遼太

動的障害物の位置予測を行う局所的経路計画I 4

自律走行の構成は、大局的経路計画、局所的経路計画、リ カバリー動作,障害物認識の4つに分けた.障害物認識では, 測域センサーからのデータを基に障害物を抽出し,動的障害 物の行動予測を行う. そのデータをコストマップに変換し局 所的経路計画に提供する. コストマップは障害物をグリッド 表現した地図が階層構造になっており, 各階層の地図が呼び 出され, Master Layer にコスト値を埋めていく.

本研究は、数秒後の動的障害物をも反映させるため、予測 された動的障害物の位置と大きさを表現した Prediction Obstacles Layer をコストマップに追加した (図 2). Prediction Obstacles Layer を作るため、動的障害物の認識と障害物予 測の必要がある.



(b) Three types of obstacles (a) Costmap layers Fig.2 Image of layered Costmaps considering obstacle prediction

4.1 動的障害物の認識

測域センサーのデータを基に周囲環境を表現する方法は, 主に2種類に分類でき、グリッド表記とベクトル表記がある. グリッド表記は障害物を格子状で区切られたセルで表現さ れる.ベクトル表記では、線分、円、楕円、矩形などの単純 な幾何学模様に障害物をモデル化する.障害物表現方法は, 測域センサーから得られる Laser Scan, ベクトル表記である 半径と中心座標で表された円形障害物, グリッド表現のコス トマップとした (図 3).



(a) Laser Scan

(c) Costmap

続いて,障害物認識手法の流れを図4に示す.Laser scan か ら円形障害物を抽出する.抽出された障害物は複数個あり, それらを個別に追従し, Kalman Filter にかけることで, 各円 形障害物の位置と速度を推定する.障害物は等速モデル(CV model)とし、位置と速度から数秒後の位置を予測する.最後

Fig.3 Obstacle data





に予測された円形障害物を Costmap に変換する. 障害物抽出 と障害物追従は, M. Przybyła の手法を用いた⁽²⁾.

4.2 障害物予測

離散系で円形障害物の状態方程式を定義する. X((k+1)T) = AX(kT) + BU(kT) + W(kT) (1) Y(kT) = CX(kT) + V(kT) (2) Tはサンプリング周期を、W(kT)、V(kT)は、ホワイトガウ

スノイズを表す.ここで,状態変数X(kT)を定義する. $X(kT) = [x(kT) \dot{x}(kT) y(kT) \dot{y}(kT) r(kT)]^T$ 各行列A, B, Cは

$$A = \begin{bmatrix} \Gamma_x & 0 & 0 \\ 0 & \Gamma_y & 0 \\ 0 & 0 & \Gamma_r \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, B = 0$$
$$\Gamma_x = \Gamma_y = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \Gamma_r = 1$$

Kalman Filter により状態変数が推定される.推定された速度,位置から等速モデルにより障害物を予測する.

$$\begin{vmatrix} \hat{x}(kT + T_{predict}) \\ \hat{y}(kT + T_{predict}) \\ \hat{r}(kT + T_{predict}) \end{vmatrix} = \begin{bmatrix} \hat{x}(kT) \\ \hat{y}(kT) \\ \hat{r}(kT) \end{vmatrix} + \begin{bmatrix} \hat{x}(kT) \\ \hat{y}(kT) \\ 0 \end{bmatrix} T_{predict}$$
(3)

ここで、 $T_{predict}$ は予測時間を、予測 $\hat{x}(kT + T_{predict}), \hat{y}(kT + T_{predict}), \hat{r}(kT + T_{predict})$ は T_{precit} 秒後の障害物の位置、半径を表す.

5 障害物の速度予測をする局所経路計画Ⅱ

提案手法IIは、図5のようにロボットが動的障害物の経路 を阻害している生活環境を想定している.廊下など狭い空間 に動的障害物が存在し、動的障害物の奥に目的地があった場 合、ロボットは物理的に経路を見出すことができず、大局的 経路計画は失敗し停止する.また動的障害物にとっても、ロ ボットが道を塞ぎ身動きが取れない.そこで、ロボットが大 局的経路計画の失敗時にリカバリー動作として発動する局 所経路計画IIを提案する.提案手法IIは、動的障害物が動線 を確保するまでロボットが後退し続ける.局所経路計画IIの 終了後、動的障害物が狭い空間に存在しなくなったことで、 大局的経路計画が成功し経路を見つけ出し走行が再開され る.



Fig.5 Environment with robots blocking the path

局所経路計画 II の流れを図6に示す.動的障害物の経路を 確保するために,動的障害物の移動する候補として複数の位 置候補を用意する.次に動的障害物の軌道を予測し,軌道に 最も近い位置候補を選択する.動的障害物が進行する逆の方 向に,ロボットの軌道を選択する.ロボットが位置候補から 十分に離れると停止する.位置候補の選択とロボットの軌道 選択を繰り返し,動的障害物が位置候補を通り過ぎたときア ルゴリズムが終了する.

また動的障害物の位置と速度は、節4.1の障害物認識によって得られた推定値を使用する.



Fig. 6 Flowchart of Clear obstacle paths

絶対座標とロボット,壁との座標関係を図7に示す.壁の 座標の原点は出口となる広い空間と狭い空間の境界線上に 置いた.



Fig.7 Coordinate relationship between robot and wall $\Sigma(x, y, 0)$:絶対座標系 $\Sigma(x^r, y^r, 0^r)$:ロボットの座標系 $\Sigma(x^w, y^w, 0^w)$:壁の座標系 $\alpha: x \geq x^w$ の成す角度

5.1 位置候補

位置候補点と位置候補を定義する.位置候補点は点0^wを 基準とした半径r_{origin}の円周上の点とし、広い空間上に広が るように設定する.また、位置候補は、位置候補点を中心 とした半径r_{margin}の円形内部とする.位置候補点と位置候 補を図8に示す.



5.2 軌道の選択

ロボットに命令する速度ベクトルの速さと向きを定義す る.ここでは、命令する速さを定義する.動的障害物とロ ボットが衝突しないために、動的障害物とロボットの距離 が近いとき、ロボットは動的障害物と同じ速さで逃げるよ うに移動する.逆に動的障害物とロボットの距離が遠いと き、動的障害物がどのような経路を選択するのか予測でき ないため、ゆっくり動くようにする.プロダクションルー ルより表現すると

$$\begin{array}{l} \text{if } d \leq d_1 \text{ then } v_u = v_o \\ \text{if } d_1 < d < d_2 \text{ then } v_u = a(d - d_1) + v_o \\ \text{if } d_2 < d \text{ then } v_u = v_{min} \\ d_1 \triangleq R_{robot} + R_{obstacle} \\ v_o \triangleq \sqrt{\hat{v}_x^2 + \hat{v}_y^2} \\ a \triangleq \frac{v_{min} - v_o}{d_2 - d_1} \end{array}$$

ただし、dはロボットと動的障害物の距離を、 v_u は命令す る速さを、 v_o は動的障害物の速さを表している。 R_{robot} 、 $R_{obstacle}$ はロボットの半径、動的障害物の半径を示す。 d_1 は 衝突距離を表しており、 d_1 より dが小さくなる場合、衝突す る.また d_2 はロボットが動き始める間隔を意味する。また v_{min} は制御入力可能なロボットの最低速度である。速さと距 離の関係を図9に示す。



Fig.9 Relationship between distance and speed

続いて命令速度の方向を定義する. 軌道候補は, 右斜め後 ろと左斜め後ろの 2 つとした. 壁に対して平行な y^w 軸方向 の速度ベクトル v_y^w が, 動的障害物が左右どちらに移動するの か判別する指標とする. $v_y^w > 0$ のとき, ロボットから見て 動的障害物は左に移動している推定できるため, ロボットは 右斜め後ろの方向を選択し, 動的障害物とは反対方向にロボ ットが移動する. 6 シミュレーション

6.1 シミュレーション環境

提案手法を評価するために、2つの生活環境をシミュレーション環境 Gazebo で用意した.(I)壁や扉により、ロボットと動的障害物が初期位置において目視できない死角領域ある環境.(II)廊下と居間が繋がっている空間など、狭い空間と広い空間の狭間における環境.



Fig.10 Simulation scenarios

6.2 衝突回避実験 I の条件

死角領域のある環境 I で衝突回避実験を行う. ロボットと 動的障害物の軌跡が交わるように,動的障害物の速度と位置 を決める. ロボットの最大速度は 0.23[m/s],動的障害物は等 速で 0.5[m/s]とした. 初期位置の関係は,ロボットを原点と し,動的障害物は(x,y) = (2.5[m], 4.0[m])とした. ロボット の目的地は(x,y) = (4.0[m], 0.0[m])とした. 予測時間は, 3.2[s], 1.6[s]とした. 予測時間を 2 つ用意すると, 1 つの動的 障害物に対して 2 個,円形障害物が予測される. 予測する障 害物を増やすことで,ロボットは動的障害物の進路方向に進 みにくくなる.

6.3 シミュレーション結果

従来の DWA アルゴリズムと,動的障害物を予測した DWA アルゴリズムとのシミュレーション結果を比較する.動的障 害物とロボットが近づいたときの様子を図 11 に示す.従来 の DWA アルゴリズムでは,動的障害物の進行方向へロボッ トが進んでしまい,衝突した.一方で提案手法は,動的障害 物の進行方向にロボットが進まず,動的障害物が完全に通り 過ぎてからロボットが移動した.また動的障害物を予測した ときの様子を図 12 に示す.青色の円柱は動的障害物を,ピ ンク色の円形は予測された円形障害物を表している.また, 緑色で囲まれた四角形はロボットを,その中心から伸びてい る緑色の曲線は軌道を表している.動的障害物も考慮した DWA アルゴリズムは,予測された障害物も避けるように, 局所的経路計画が行われた.





(a)Without prediction obstacles layer Fig. 11 DWA algorithm



Fig.12 Costmap considering obstacle prediction

6.4 衝突回避実験Ⅱの条件

初期状態でロボットが出口を塞いでいる環境IIにおいて, ロボットに目的地を与え,問題なく自律走行可能かシミュレ ーションを行う.配置を絶対座標で表現する.ロボットの初 期位置は(1.0[m], 0.0[m]),目的地を(4.0[m], 0.0[m])とする.

動的障害物の初期位置は (3.0[m], 0.0[m]) とする. 動的障害物の軌跡を図 13 に示す. 動的障害物の軌道は初期位置から点 a (2.0[m], 0.0) まで $(v_x^o, v_y^o, \omega^o) = (-5.0[m/s], 0.0[m/s], 0.0[rad/s])$ で等速直線運動で動かし, 点 a から $(v_x^o, v_y^o, \omega^o) = (-5.0[m/s], 0.0[m/s], 0.5[rad/s])$ で等速円運動で曲線を描くように設定した.



Fig.13 Dynamic obstacle trajectory

6.5 衝突回避実験Ⅱの結果

動的障害物とロボットとの距離dを図14に,軌道選択の基準となるvyvを図15に,ロボットと動的障害物の経路結果を図16に示す. (距離dは衝突判定の指標になる)





Fig.16 Robot and dynamic obstacle path results 図 15 を見ると、動的障害物の推定値 v_y "は、2~4 秒間にか けて、真の値に収束していない.これは、障害物認識におい てカルマンフィルタにより速度推定を行っているが、動的障 害物の予測を等速モデルと仮定しているため、動的障害物の 急な速度変化に対応することができない.しかし、図 14 か ら、衝突することなく 0.519[m]以上の衝突余裕をもって動線 確保動作と自律走行が成功していることがわかる.動的障害 物の速度を正確に精度よく推定するのことは、環境 II におい て動的障害物がどのような運動をするのか事前に把握でき ないため困難である.動線確保動作は、動的障害物の速度を 基準とするのではなく、壁に対して平行なy"軸方向の速度 v_y "の正負を基準にしたため、精度よく速度を求めることを必 要としない.故に、速度推定の値が収束していなくとも、衝 突回避をしながら後退することができる.

7 実機実験

環境Ⅱを再現した実機実験を行った.実機実験結果を図17 に示す.初期状態においてロボットは人が進む道を阻害して いるが,動線確保動作によって,経路を確保していることが わかる.人が通り過ぎてから,ロボットは再度走行を行い, 目的地に到達した.



Fig.17 Collision Avoidance Experiment Results

8 結言

本報告は、生活環境下で人とロボットが混在する、環境 I、環境Ⅱを問題提起した.環境Iにおいては、動的障害 物を予測した DWA アルゴリズムと従来の DWA アルゴリ ズムとを比較実験し優位であることを示した.環境Ⅱにお いては、シミュレーション及び実機実験で提案手法である 障害物の速度予測をする局所経路計画の有効性を示した.

文献

- W. B. D. Fox and S. Thrun, "The dynamic window approach to collision avoidance," IEEE Robot. Automat. Mag., vol. 4, no. 1, pp. 23–33, May 1997.
- (2) M. Przybyła, "Detection and tracking of 2d geometric obstacles from LRF data," in Proc. IEEE 11th Int. Workshop Robot Motion Control, 2017, pp. 135–1