

# リスク回避層を導入した制約緩和 A\*と適応的ウィンドウアプローチによる 高速で安全な経路計画手法

栢植 恵太<sup>†</sup> 佐久間 拓人<sup>†</sup> 加藤 昇平<sup>†</sup>

<sup>†</sup> 名古屋工業大学 大学院工学研究科 工学専攻

## 1 はじめに

近年、物流業界では 2024 年問題が課題となっており、自律移動ロボットの活躍している倉庫業でもその影響は大きい。そのため、より効率的な搬入のために、自律移動ロボットの経路計画手法の改善が求められる。ここで言う経路計画とは、ロボットに地図や目的地などの情報を与え、開始地点から目的地への経路を作成することであり、経路計画は大きく 2 つに分けられる。1 つ目は大域的経路計画 Global Path Planning (GPP) であり、静的かつ既知の環境下で最適経路を求めるために使用する。GPP は、動的環境や大規模な環境に適していない。しかし実環境では、人間など多くの動的障害物が存在し、倉庫や農業は大規模環境であるため、実用化には GPP の欠点を補う必要がある。2 つ目は局所的経路計画 Local Path Planning (LPP) であり、未知の動的環境でリアルタイム回避が可能である。しかし LPP には、局所最小解に陥りやすいという欠点がある。これら 2 つの要素を組み合わせたものが、2 つの利点を併せ持つハイブリッド経路計画 Hybrid Path Planning (HPP) である。HPP は大規模な動的環境下での大域的最適化と、動的回避を目的としている。しかし近年、HPP の研究は大域的経路計画・経路追従・障害物回避の同時実現には不十分である。本稿では、ZHONG ら [1] が提案した適応的ウィンドウアプローチと、CHAARI ら [2] の制約緩和 A\* アルゴリズムを組み合わせた HPP を構築し、HPP の性能向上を目指す。

## 2 提案システム

図 1 にシステムの概要図を示す。SLAM と呼ばれる地図作成技術により壁などの情報を読み込み、リスク回避層でコストを追加した上で経路生成を開始する。開始時・移動中に障害物を検出した場合はその障害物も含めてコストを追加し、順次コストマップを更新する。

本稿では、ハイブリッド経路計画システムの GPP に CHAARI らの開発した制約緩和 A\* を、LPP に適応的ウィンドウアプローチを使用し、HPP を構成した。また、制約緩和 A\* で計算時間を短縮しつつ、リスク回避層により経路の安全性確保を目的として、HPP の性能向上を検証する。

### 2.1 制約緩和 A\*

制約緩和 A\* は、Ammar ら [3] が提案した A\* の線形時間緩和版で、大規模なグリッドマップの経路計画問題を解くためのもの

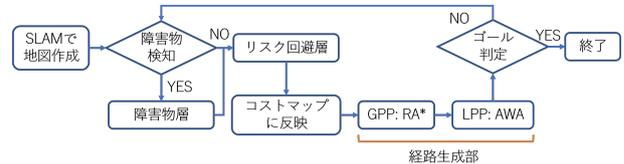


図 1: 提案 HPP の概要図

である。以降 RA\* と呼ぶ。従来の A\* ではノード  $n$  での正確なコストは何度も計算される可能性があるのに対し、RA\* では開始ノードからノード  $n$  に関連するノードへの最小移動パスのコストで近似される。したがって、どのノードも 2 回以上訪れることなく、最適経路の正確な近似を確立することができ、従来の A\* より少ない計算時間での経路計画が可能になった。式 (1), (2) にそれぞれ A\*, RA\* の評価関数を示す。

$$g(x, y) \leftarrow \min[\text{周囲 4 座標の } g(x, y)] + R(n) \quad (1)$$

$$g(x, y) \leftarrow g(x', y') + \text{ノードサイズ} \quad (2)$$

A\* の評価関数は周囲 4 座標の評価値の最小値にリスク関数  $R(n)$  を加えた形で、4 つの評価関数を呼び出す必要がある。対して RA\* では関数に含まれる評価関数が  $(x, y)$  の一ステップ前に通過した座標  $(x', y')$  のみであり、評価関数の呼び出し回数を 1 回に抑えることで計算時間短縮を図る。なお、RA\* の評価関数に含まれない  $R(n)$  はノードの通過可能判定に使用した。

### 2.2 リスク回避層

リスク回避層はロボットのオープンソースソフトウェアである ROS の標準機能であり、格子地図における壁や障害物の周囲に幅を持たせるようにコストを追加する。この層によって、大域的経路計画で作成する経路が壁などに衝突するリスクを減らすことができる。コストは以下の式 (3) に示すように、ロボットの内接半径を基準として壁に近づくほど大きく、離れるほど小さく追加する。なお、 $C_{ref}$  はコストの基準値であり、本稿では ROS 固有のパラメータとして 253 に設定している。

$$R(n) = \exp(-10 \times (r - \text{内接半径})) \times C_{ref} \quad (3)$$

$r$ : 内接半径からの距離,  $C_{ref} = 253$

### 2.3 適応的ウィンドウアプローチ

適応的ウィンドウアプローチは ZHONG らが提案した LPP 手法で、LPP の代表例、DWA の改変版である。適応的ウィンドウアプローチは、DWA における計画ウィンドウ、つまり LDS でロボットが未知の障害物を視認できる範囲を周囲の環境に応じて拡大・縮小する。周囲に障害物が少ないとき、計画ウィンドウを広げることでよりよい移動方向を探索し、逆に、周囲に障害物が密集しているとき、計画ウィンドウを縮小することで狭い空間で走行できる経路を探索する。

Fast and safe path planning method with Relaxed-A\* introducing risk aversion layer and Adaptive Window Approach

Keita TSUGE<sup>†</sup>, Takuto SAKUMA<sup>†</sup>, Shohei KATO<sup>†</sup>

<sup>†</sup> Dept. of Engineering, Graduate School of Engineering, Nagoya Institute of Technology

<sup>†</sup> Gokiso-cho, Showa-ku, Nagoya-city, Aichi 466-8555, Japan tsuge, sakuma, shohey@katolab.nitech.ac.jp

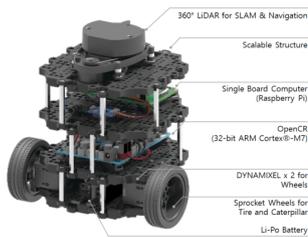


図 2: TurtleBot3 burger. [4]

### 3 シミュレーション実験

#### 3.1 実験環境

今回、大域的経路計画に採用した RA\*の性能を確認するため、局所的経路計画は従来手法の DWA に固定して RA\*と A\*アルゴリズムを比較した。図 2 は実験に使用した TurtleBot3 burger で、物理エンジンである gazebo のシミュレーション環境で実験した。また、事前地図には gazebo で作成した 8[m] × 14[m] の迷路状のマップを使用した。

#### 3.2 実験評価

実験の目標は、提案システムを用いた経路計画が、事前地図の壁及び障害物に衝突することなく目的地に到達し、改良前のシステムより性能が向上することとする。生成された経路を比較するための評価指標を以下に示す。

- 大域的経路の計算時間
- 大域的経路長
- 目的地への到達時間

#### 3.3 実験結果

図 3, 4 に迷路状の地図において、それぞれ RA\*, A\*によって生成された大域的経路を示す。赤線の両端に存在する赤矢印と黒い物体は、経路計画の目的地とロボットである。また、表 1 は RA\*, A\*それぞれ 12 回ずつ評価指標を測定した結果の平均と標準偏差である。3 つの指標に対して有意水準 0.05 の両側 t 検定をしたところ、すべての指標において RA\*と A\*の 2 者間で有意な差を確認できた。

計算時間に関して、平均値で見比べるとほとんど差がないように見える。しかし、マップの中央での GPP の計算時間がおよそ 0.01 秒未満であることが確認でき、これはスタート時点での計算時間の 10 分の 1 に満たない。そのため、より経路長が長くなるような環境ではさらに計算時間の差が広がり、大規模環境では大きな計算時間の短縮につながると考えられる。

経路長は約 40%短縮された。A\*が壁から離れた大回りな経路を生成したのに対し、RA\*は最短経路に近い壁に沿った経路を生成した。これは、RA\*においてリスク関数を通過可能か否かの判定のみに使用し、評価関数の計算から省いたために壁から最低限の距離だけ離れた経路を生成したと考えられる。

目的地への到達時間は約 9%短縮された。これは経路長が短縮されたことで移動時間が減少したことによるものだが、短縮率が経路長での 40%から減少した。LPP は GPP の生成した経路をそのまま進むのではなく追従するため、A\*による実際の移動経路が短くなったことで差が縮まったと考えられる。

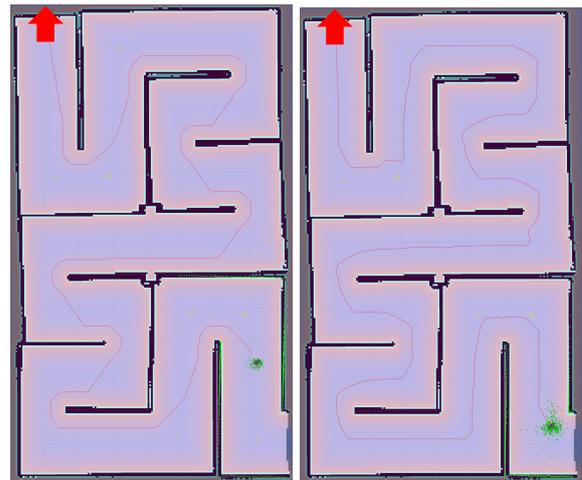


図 3: RA\*による経路

図 4: A\*による経路

表 1: 評価値の計測結果 (12 試行)

	RA*		A*		
	平均	SD	平均	SD	
GPP 計算時間 [s]	0.098	0.004	0.101	0.005	*
経路長	1110.6	1.5	1798.5	10.6	*
到達時間 [s]	186.1	0.9	203.6	0.8	*

\* $p < .05$

### 4 まとめ

HPP の安全性と計算時間短縮を目的として、制約緩和 A\*にリスク回避層を組み合わせた GPP の性能を検証し、計算時間の短縮と経路長短縮を実現した。今後は、適応的のウィンドウアプローチを実装し、DWA による経路との比較実験を予定している。

### 5 謝辞

本研究は、一部、文部科学省科学研究費補助金（課題番号 JP19H01137 および JP20H04018）、ならびに、国立研究開発法人情報通信研究機構委託研究の助成により行われた。

### 参考文献

- [1] ZHONG, Xunyu, et al: Hybrid path planning based on safe A\* algorithm and adaptive window approach for mobile robot in large-scale dynamic environment. Journal of Intelligent & Robotic Systems, 2020, 99.1: 65-77.
- [2] CHAARI, Imen, et al: Design and performance analysis of global path planning techniques for autonomous mobile robots in grid environments. International Journal of Advanced Robotic Systems, 2017, 14.2: 1729881416663663
- [3] AMMAR, Adel, et al: Relaxed Dijkstra and A\* with linear complexity for robot path planning problems in large-scale grid environments.
- [4] e-Manual TurtleBot3 Overview, <https://emanual.robotis.com/docs/en/platform/turtlebot3/overview/>, (最終閲覧 2023-11-26)