

複数 AMR に向けた施設レイアウト最適化のための 高速なマップ評価手法の提案

○望月洋彰 杉山俊一郎 奥田裕之 鈴木達也 (名古屋大学)

Fast map evaluation for facility layout optimization with multiple AMRs

*H. Mochizuki, S. Sugiyama, H. Okuda, T. Suzuki (Nagoya Univ.)

Abstract: This study proposed an approximate cost prediction model for the facility layout optimization problem of an environment in which multiple autonomous mobile robots work, combining CNN and A* algorithm instead of path planning algorithm as a layout evaluation method. The proposed method improves the computational efficiency over the search algorithm and enables collision-aware cost prediction, and the combination of the A* algorithm achieves stable cost prediction.

Keywords: layout design, autonomous mobile robot, deep learning

1 緒言

我が国における少子高齢化の進行は深刻な社会課題となっており、特に製造業においては、働き手の減少が問題視されている。こうした人材不足を解消しつつ、生産性の向上および品質の安定化を図るため、ロボット技術を活用した工場の自動化が進められている。その一環として、自律走行ロボットである AMR (Autonomous Mobile Robot) を用いた部品搬送の自動化が行われている [1]。

AMR や AGV (Automated Guided Vehicle) に代表される移動エージェントは、指定された目的地まで荷物を運搬する役割を担うが、施設の配置によっては通路間の狭所や作業場の密集が生じ、迂回や衝突の可能性が高まることで、工場全体のコストや運用効率に悪影響を及ぼす [2]。このため、工場における施設配置 (レイアウト) は生産性に大きく影響を与える要因であり、Facility Layout Problem (FLP) の分野では、最適な配置を求めるためのアルゴリズムが提案されてきた [3]。その中でも、遺伝的アルゴリズム (Genetic Algorithm: GA) は、生物の進化過程を模倣し、レイアウトなどの個体群に対して適応度評価、選択、交叉、突然変異といった操作を繰り返すことで最適解を探索する手法として、FLP に広く用いられている。

複数エージェントが稼働する環境では、移動量である各エージェントの移動する経路長総和が工場全体の搬送コストや処理時間に直結するため、移動量を最小化することが必要である。この場合、GA における各レイアウトの評価には全エージェントの経路計画を行い、移動量を算出する必要がある。エージェント間で衝突を回避しながら経路を計画する問題はマルチエージェント経路計画 (Multi-Agent Path Finding: MAPF)

と呼ばれ、代表的な解法として Conflict-Based Search (CBS) [4] が知られている。しかし、CBS はマップサイズやエージェント数の増加に伴って解の候補が指数関数的に拡大し、計算時間が急増するという課題がある。さらに、FLP では多数のレイアウトに対して経路計画を繰り返し適用する必要があるため、GA の評価に計算時間がかかりすぎてしまい、解が現実的な時間で求まらないことがある。

計算時間の削減を目的として、機械学習を用いた経路計画手法も研究されている。例えば、強化学習と模倣学習を組み合わせた PRIMAL [5] では、障害物密度の低いマップで CBS よりも経路計画にかかる計算時間を短縮できるが、障害物やエージェントが密集すると計算精度および速度が低下する傾向がある。

そこで本研究では、経路計画そのものを実行せず、深層学習によって移動量を予測することで計算時間を削減することを目指す。特に、画像認識に優れる畳み込みニューラルネットワーク (Convolutional Neural Network: CNN) は、レイアウトにおける障害物配置やエージェント位置の空間的關係を特徴として捉えることが可能であり、エージェントの経路や衝突発生位置との関係性を学習できると考えられる。そのため本研究では、複数エージェント環境を対象として CNN を用いたコスト予測モデルを構築し、従来の経路計画によるコスト算出と比較することで、衝突回避下における全エージェントの移動時間総和の算出に要する計算時間の短縮を目的とする。

2 問題設定

本研究では、グリッドマップを対象とし、施設および各エージェントのスタート地点とゴール地点を配置する。ここで、障害物の情報を含む基盤をマップ、さ

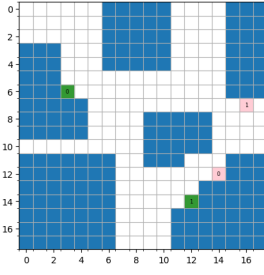


Fig. 1: map data

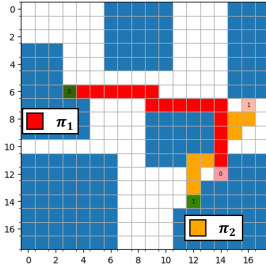


Fig. 2: path map

らにスタート・ゴール位置の情報を含めた Fig. 1 に示すような問題インスタンスをレイアウトと定義する。

このときマップは頂点集合 V と辺集合 E を用いて、無向グラフ $G = (V, E)$ として表現される。エージェント集合を $A = \{a_1, \dots, a_k\}$ とし、各エージェント a_i はスタートノード $s_i \in V$ とゴールノード $g_i \in V$ を与えられる。各エージェントは離散化された時刻 t においていずれかの頂点に存在し、時刻 $t+1$ で辺を通して隣接した頂点に移動する。時刻 t におけるエージェント a_i のノードを $\pi_i[t]$ とする。時刻 $T \in N$ を最大ステップ数とすると、 a_i の経路を $\pi = (\pi_i[0], \pi_i[1], \dots, \pi_i[T])$ とすると経路の組み合わせ $\pi = (\pi_1, \dots, \pi_k)$ は任意の $i, j \in \{1, \dots, k\}, i \neq j$ に対して、以下の条件を満たす必要がある。

- 初期条件: $\pi_i[0] = s_i$
- 終端条件: $\pi_i[T_i] = g_i$
- 頂点上での衝突の禁止: $(\pi_i[t] \neq \pi_j[t])$
- すれ違いの禁止: $(\pi_i[t] \neq \pi_j[t+1]) \vee (\pi_i[t+1] \neq \pi_j[t]) \quad \forall t \in \{0, 1, \dots, T-1\}$

これら制約を満たす π を求めることで Fig. 2 に示すような、全エージェントの経路を算出することが可能となる。ここで各エージェント a_i がゴール g_i に到達する最小時刻を $T_i = \min\{t \mid \pi_i[t] = g_i\}$ と定義する。本研究では、全エージェントの合計移動時間 $\sum_{a_i \in A} T_i$ を移動コストと定義し、レイアウトの評価指標とする。

2.1 従来の MAPF 解法

2.1.1 A*アルゴリズム

従来から広く使用されている経路計画アルゴリズムの一つに A*アルゴリズムがある。A*アルゴリズム (A*) はヒューリスティック関数を用いて、単一エージェントに対する最短経路を探索する手法である。A*ではノードと呼ばれる状態を展開し、探索を進める。各ノードにはエージェントの現在位置、ゴール位置、累積コスト、親ノードなどの情報が含まれる。展開されたノード

の中から、ゴールまでのコスト見積もりが最も小さいノードを優先的に選択することで、ヒューリスティック関数を用いないブラインド探索に比べ、高速に最適解へ到達することが可能である。ノード n からゴール状態までのコスト $f(n)$ は (1) 式で計算される。

$$f(n) = g(n) + h(n) \quad (1)$$

ここで、 $g(n)$ は初期状態からノード n までの経路にかかったコストである。 $h(n)$ はノードからゴールまでの推定距離を示すヒューリスティック関数である。ヒューリスティック関数には (2) 式のような 2 次元平面上でのゴール (x_g, y_g) までのユークリッド距離がよく用いられる。

$$h(n) = \sqrt{(x_n - x_g)^2 + (y_n - y_g)^2} \quad (2)$$

A*アルゴリズムは単一エージェントにおける経路計画手法であるため、複数エージェント間の衝突を考慮することができない。そのため、A*のような単一エージェントの経路計画とエージェント同士の衝突回避制約を組み合わせることで MAPF を解く手法が提案されている。

2.1.2 CBS(Conflict-Based Search)[4]

CBS (Conflict-Based Search) は、MAPF 問題に対して広く用いられている手法であり、高レベル探索と低レベル探索の 2 層構造から成る。低レベル探索では、各エージェントに対して A*アルゴリズムなどを用いた単一エージェントでの経路計画を行う。一方、高レベル探索では、各エージェント間の経路における衝突を検出し、それを回避するための制約を管理する。CBS はこの二つを交互に実行することで複数エージェント条件においても解を探索することができる。高レベルの探索では、constraint tree(CT) と呼ばれる探索木を用いる、高レベルの探索の CT 上の各ノード (CT ノード) では、エージェントのパス、総コスト、制約 (a_i, v, t) (時刻 t でエージェント a_i に頂点 v にいることを禁止) または (a_i, v_1, v_2, t) (時刻 t でエージェント a_i に頂点 v_1 から v_2 に移動することを禁止) の情報を持つ。

CBS は最適解が保証されるアルゴリズムであり、MAPF における代表的な解法である。しかし、レイアウト内に狭所が多い場合や、エージェントが密集している場合には、衝突回避のための制約数が急増することで、探索すべき解が急激に増加し、全体の計算時間が大きくなるという課題が存在する。

3 CNNを用いたマルチエージェントの移動コスト推定

従来の経路計画によって移動コストを算出する手法では、エージェント数や衝突回避制約の増加に伴い、経路探索アルゴリズムにおける計算負荷が大きくなる。そこで本研究では、計算時間の短縮を目的として、CNNを用いたコスト予測モデルを提案する。

3.1 モデル概要

本研究で提案するコスト予測モデルの構成を Fig. 3 に示す。本研究では、コスト全体ではなく、衝突回避によって発生するコスト増分（迂回コスト）に着目し、これを CNN で予測することで、モデルの学習対象を限定したコスト予測モデルを提案する。提案手法では、レイアウト入力に対して CNN によって迂回コストを推定し、A*アルゴリズムによって距離コストを推定し、両者の和を取ることでレイアウトのコストを算出する。

3.2 コストの定義

以下に提案手法で用いる最適コスト、距離コスト、迂回コストについて説明する。

最適コスト J

最適コストは、複数のエージェントが互いに衝突することなく目的地へ到達するための、経路長の総和の最小値として定義する。本研究においては、各レイアウトに対する正解ラベルとして、この最適コストを使用する。

距離コスト J_d

距離コストは、衝突を考慮せずに各エージェントが最短経路を用いて移動した場合の、経路長の総和として定義される。このコストは、単一エージェントに対して最短経路を求める A*アルゴリズムにより算出されるため、計算時間が速い。しかし、複数エージェント間の干渉による迂回経路の影響を反映できず、最適コスト以下の値が出力される。

迂回コスト J_c

迂回コストは、以下のように定義され、衝突回避や迂回によって発生する距離コストからのコスト増分を表す。

$$J_c = J - J_d \quad (3)$$

3.3 CNN(畳み込みニューラルネットワーク)

畳み込みニューラルネットワーク (Convolutional Neural Network, CNN) は、学習可能な畳み込み層を含む 4 層以上の層から構成されるディープニューラルネットワークであり、主に画像認識タスクに広く用い

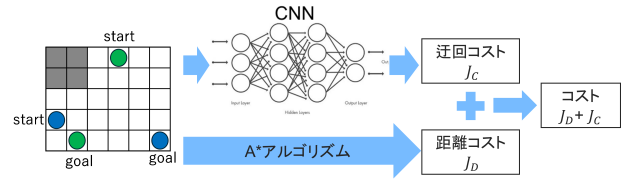


Fig. 3: proposed model

られる。代表的な CNN の構造として、AlexNet [6] や VGGNet [7] などが挙げられる。以下に、CNN を構成する各層および要素について述べる。

畳み込み層

畳み込み層は、それ以前の層において生成された複数チャンネルの特徴マップに対して畳み込み演算を適用し、フィルタに応答する新たな特徴マップを出力する層である。

プーリング層

プーリング層では、局所領域に対してプーリング操作により、特徴マップのダウンサンプリング処理が行われる。

全結合層

畳み込み層・プーリング層で得られた特徴を統合し、分類や回帰に用いる層である。

活性化関数

本研究では活性化関数として ReLU を使用する。

$$y = \max(0, x) \quad (4)$$

誤差関数

本研究では誤差関数として平均二乗誤差 (Mean Squared Error, MSE) を用いており、移動コストの予測値 \hat{y} と理想値 y に対して次式で表される。

$$L_{\text{MSE}}(\hat{y}, y) = \frac{1}{K} \sum_{n=1}^K (\hat{y} - y)^2 \quad (5)$$

勾配降下法

誤差関数の値を最小化するパラメータ最適化法として勾配降下法が用いられる。本研究では CNN に広く用いられる Adam (Adaptive Moment Estimation) [8] を用いてパラメータ更新を行う。

4 迂回コスト推定のための入力情報

4.1 レイアウトの作成方法

本研究では、迂回コストの推定に用いるレイアウトデータを生成するために、進化的アルゴリズム (Evolutionary Algorithm) の一種である遺伝的アルゴリズム (Genetic Algorithm, GA) を適用する。FLP は、二次元平面上に施設やスタートゴール位置を配置する組

合せ最適化問題であり、二次元パッキング問題に類似して NP 困難であるため、厳密解法では現実的な時間で解を得ることが難しい。GA は、多点探索により局所最適解に陥りにくく、評価関数の設計を通じて柔軟に制約（重複回避、境界外配置の禁止など）を組み込むことが可能である。また、大量のデータを短時間で生成することができ、並列計算とも親和性が高い。これらの理由から、本研究では GA を用いてレイアウトを生成する。

本研究で使用するレイアウトは、エージェントのスタート・ゴール位置および施設配置をランダムに決定することで生成する。施設数や形状、エージェント台数は固定とし、初期配置において施設間やエージェントのスタート・ゴールとの重複が存在する場合は再配置を行い、全ての構成要素が重ならないように調整する。その後、GA の操作を通じて新たなレイアウトを生成する。具体的には、複数のレイアウト（個体）から施設の配置位置やエージェントのスタート・ゴール位置といった要素を遺伝子として取り出し、交叉操作によって二つの個体を組み合わせることで、新しいレイアウトを構成する。また、一定の確率で突然変異操作を適用し、部分的に施設やエージェントの位置をランダムに入れ替えることで、重複のない多様なレイアウトを生成する。

4.2 入力情報

本研究では、CNN への入力として、Fig. 4 に示すように、 $H \times W$ 次元のマトリクスを各チャンネルに配置し、それらを重ね合わせて入力画像とする。このように情報を分離して入力することで、ネットワークは各情報を効果的に認識でき、学習および予測の精度向上が期待できる。

本研究で用いるチャンネルは、障害物の有無を示す障害物マップ $O(i, j)$ と、各エージェントの A^* 探索によって得られた最短経路を示す経路マップ $P_k(i, j)$ である。ここで経路マップを入力に含めるのは、単に障害物の有無だけでなく、各エージェントが最短経路として選択する移動パターンを CNN に与えることで、衝突が発生しやすい領域や迂回が必要となる箇所を学習しやすくするためである。

障害物マップ $O(i, j)$ 、経路マップ $P_k(i, j)$ は以下のように定義され、エージェント台数を n とすると、入力画像 X は次式のように表される。

$$O(i, j) = \begin{cases} 1 & \text{障害物が存在する場合} \\ 0 & \text{通行可能な場合} \end{cases}$$

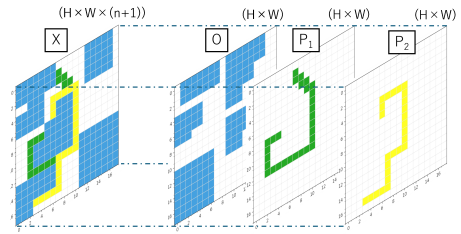


Fig. 4: input map (2 agents)

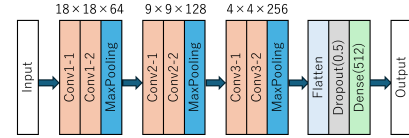


Fig. 5: CNN Model Architecture

$$P_k(i, j) = \begin{cases} 1 & A^* \text{ 経路上にある場合} \\ 0 & \text{経路上にない場合} \end{cases}$$

$$X = [O, P_1, P_2, \dots, P_n] \in R^{H \times W \times (n+1)} \quad (6)$$

5 提案手法の評価

5.1 評価時のモデル設定

本研究で用いた回帰モデル設定を Fig. 5 に示す。このモデルの畳み込み層は、カーネルサイズ 3、最大プーリング層ではカーネルサイズ 2、ストライド 2 を適用している。これらにより抽出された特徴マップを平坦化した後、ドロップアウト層、全結合層を経由して、線形回帰する構造としている。学習の際、ミニバッチサイズは 32、学習エポック数は 30 とした。入力マップはサイズを 18×18 に固定し、エージェントの台数は 2 台から 10 台まで変化させ、その影響を評価した。学習データには 90176 サンプル、検証データには 22545 サンプル、テストデータには 5771 サンプルを用いている。

5.2 評価結果と考察

テストデータ 5,771 サンプルに対して提案手法、 A^* 、および CBS による予測を行い、性能を比較した。評価指標として、最適コストとの差の絶対値の平均（平均誤差）および 1 サンプル当たりの計算時間を用い、結果を Fig. 6 および Fig. 7 に示す。さらに、エージェント台数が 10 台の場合の A^* および提案手法による予測コストの分布を Fig. 8 および Fig. 9 に示す。

提案手法、 A^* 、CBS を比較した結果、提案手法は CBS に比べて大幅な計算時間短縮を達成した。CBS では、エージェント数の増加に伴い衝突回避制約が増大

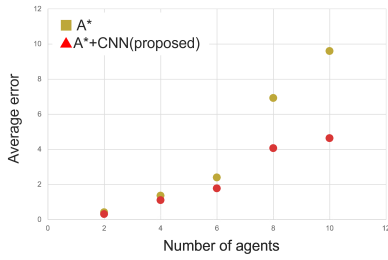


Fig. 6: average error

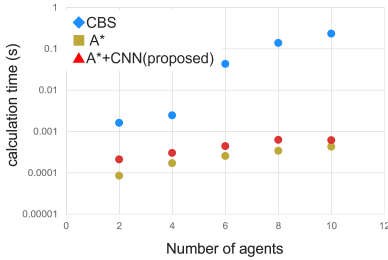


Fig. 7: calculation time

し、探索空間が指数関数的に拡大するため、計算時間が急激に増加する。一方、提案手法は CNN による推定を用いるため、計算時間は主として入力マップサイズとデータ数に依存し、エージェント数や衝突回避制約の増加による計算時間増加を抑制できる。この特性により、エージェント数が増加した高密度環境においても大幅な高速化が可能となった。精度の観点では、A* はエージェント台数の増加に伴い平均誤差が急激に拡大し、台数が多い条件では最適コストとの誤差が増加した。一方、提案手法は、台数が増えても迂回コストを適切に予測し、A* に比べ平均誤差を抑制できた。これは、距離コストを導入し、CNN の学習対象を衝突回避によるコスト増分に限定することで学習の複雑性を低減し、予測精度を向上させた結果であると考えられる。特にエージェント数が多数の状況では、迂回や待機が多く発生する。提案手法では A* 経路を入力として利用することで、経路の重なりから狭い通路を通過する台数や、衝突を回避するために必要な迂回量を学習できたと考えられる。さらに、すれ違いによってコストが増加しないパターンも取り込むことで、狭所・交差点・経路重複箇所といったコスト増分が生じやすい領域を効率的に捉えることが可能となった。

一方で、幅 1 マスの狭い通路を多数のエージェントが通過する場合には、衝突回避に必要な待機時間を十分に学習できていない場面が多く見られた。その結果、狭い通路が多数あるマップでは、迂回コストを過小または過大に推定する傾向がある。より正確な予測を行うためには、このような状況に対応できるよう、狭い

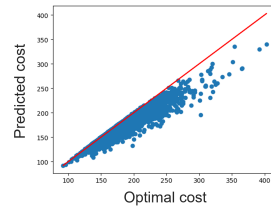


Fig. 8: A*

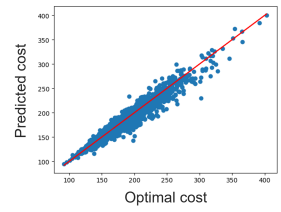


Fig. 9: A*+CNN

通路や密集領域を適切に反映した入力画像を用いて学習させることが重要であると考えられる。

6 結言

本研究では、複数 AMR の衝突を考慮したレイアウト評価値の算出を高速化するため、経路計画アルゴリズムに代わるコスト予測モデルを提案した。A* と CNN によるコスト予測を組み合わせることで、CBS と比較して大幅な計算時間の短縮を実現した。提案手法は、エージェント数が多く、密集度の高い環境においても効率的な予測が可能である一方で、狭い通路が多いマップではコストを過小あるいは過大に推定する傾向が確認された。今後は、狭い通路や密集領域を適切に反映した入力表現を設計することで、より高精度なコスト予測を目指す。

参考文献

- [1] 2023 年版 世界 AGV / AMR 市場の現状と将来展望. 矢野経済研究所, 2023.
- [2] JA Tompkins, JA White, YA Bozer, EH Frazelle, JMA Tanchoco, and J Trevino. Facility planning, 1996.
- [3] Hasan Hosseini-Nasab, Sepideh Fereidouni, Seyyed Mohammad Taghi Fatemi Ghomi, and Mohammad Bagher Fakhrazad. Classification of facility layout problems: a review study. *The International Journal of Advanced Manufacturing Technology*, Vol. 94, pp. 957–977, 2018.
- [4] Guni Sharon, Roni Stern, Ariel Felner, and Nathan R Sturtevant. Conflict-based search for optimal multi-agent pathfinding. *Artificial intelligence*, Vol. 219, pp. 40–66, 2015.
- [5] Guillaume Sartoretti, Justin Kerr, Yunfei Shi, Glenn Wagner, TK Satish Kumar, Sven Koenig, and Howie Choset. Primal: Pathfinding via reinforcement and imitation multi-agent learning. *IEEE Robotics and Automation Letters*, Vol. 4, No. 3, pp. 2378–2385, 2019.
- [6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, Vol. 25, , 2012.
- [7] Karen Simonyan. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [8] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.