

## A\*アルゴリズムによる 3 次元配線設計の効率化の検討 A Study on the Efficiency Improvement of 3D Routing Design Using the A\* Algorithm

廣岡 隆孝<sup>†</sup>      増井 秀之<sup>†</sup>  
Takanori Hirooka    Hideyuki Masui

### 1. はじめに

製造業における基板上のケーブル配線設計は、空間的制約や部品との干渉回避を考慮する必要があり、熟練作業者による試行錯誤に依存しているのが現状である。この作業は多大な工数を要し、納期遅延の一因ともなっている。こうした課題に対し、A\*と蟻コロニー最適化(ACO)を統合した手法 [1] や、ヒューリスティック評価を強化した MSSA\* [2]、配線の集約性と 3D 空間制約を両立するレイアウト最適化法[3]などが提案されている。しかし、これらは計算資源の消費や汎用性に課題がある。そこで本研究では、CAD 形状と配線制約を反映した 3 次元格子グラフ上で A\*を用い、敷設作業性に優れた経路の自動生成手法を提案する。

### 2. 問題設定と入力データ

本章では、本研究における問題設定と、使用する CAD データおよび配線接続情報について説明する。

#### 2.1 問題の概要

本研究の目的は、基板の CAD 情報を基に回路モジュールを結ぶ最適な配線経路を自動生成するアルゴリズムの構築である。配線方法には自由度があり、製造現場の要望に応じた設計を行う。主な要望は以下の通りである：

- 配線長を可能な限り短くする。
- 複数のケーブルが接近する場合、適切に結束する。
- 部品の周囲を迂回する配線経路を引く。

#### 2.2 CAD データ

本研究では、STP 形式の CAD データを使用する。STP ファイルは異なる CAD ソフトウェア間でデータを交換するための標準化された中間ファイルであり、設計データのスムーズなやり取りが可能となる。STP ファイルにはジオメトリ情報、材料情報、アセンブリ構造情報、部品名称などのメタデータが含まれる。従来では STP ファイルに対する前処理が必要であったが、本研究では STP ファイルを直接利用し、汎用性を確保している。

#### 2.3 配線接続情報

本節では、基板上の部品や回路モジュールなどを配線接続するために必要な 4 種類の情報について詳述する。

##### (a) 線材情報 (CABLE.csv)

各ケーブルについて、ケーブル名・直径・最小曲げ半径などの線材情報を定義したものである。

##### (b) 接続点情報 (CONNECTION.csv)

各配線に対するスタート地点とゴール地点の名称を定義したものである。

##### (c) 座標・方向情報 (CS.csv)

各配線のスタート地点・ゴール地点の XYZ 座標、及び接続点からの配線方向を定義したものである。

##### (d) 配線可能面定義情報 (config.yaml)

CAD データを構成する部品のうち、表面配線が許可される部品を定義する情報である。配線可能面として明示的に指定されていない部品は、全て配線不可能な部品とする。

### 3. 提案手法

本章では、対象となる基板上の部品を 3 次元配線する手法について提案する。手法は二段階の工程で構成される。第一段階として、CAD 形状と配線制約を反映した 3 次元格子状有向グラフを作成し、第二段階として A\*アルゴリズムを用いたグラフ上の経路生成を行う。

#### 3.1 3 次元格子状有向グラフの作成

配線の土台として、CAD 形状を反映した 3 次元格子構造を持ち、各ノードから上下・左右・前後の 6 方向に対して双方向にエッジが張られた有向グラフ[4] $G = (V, E, w)$ を考える。ここで、 $V$ は格子点に対応するノードの集合であり、 $E$ は有向エッジの集合である。また $w: V \times V \rightarrow \mathbb{R}_{\geq 0}$ はエッジの重み関数であり、双方向エッジで結ばれたノード対  $u, v \in V$  に対して、 $w(u, v) \neq w(v, u)$ となる場合を許容する。このグラフ $G = (V, E, w)$ を以下の手順で作成する。

##### (a) ノード判定 (配線可能ノード・配線不可能ノード)

CAD データを基に、XYZ 各軸の最大値と最小値を持つ直方体を考える。この直方体に対して一辺 $d$ の小立方体を格子状に配置し、各立方体の中心をノードとする 3 次元格子状有向グラフを作成する。各ノードは、配線可能面に対応する場合は配線可能ノード、配線不可能な部品に対応する場合は配線不可能ノードとして判定する。

##### (b) ノード判定 (クリアランスノード)

配線不可能な部品の周囲に、配線を避けるための幅 $d_c$ のクリアランス領域を定義する。これは、部品表面から距離 $d_c$ 以内の空間領域とする。この領域を抽出するため、配線不可能ノード判定前のグラフ $G_1$ を複製して $G_2$ を作成し、判定距離を $d + d_c$ に拡大して配線不可能ノードを判定する。 $G_1$ 上で配線可能、かつ $G_2$ 上で配線不可能であるノードをクリアランスノードとして判定する。

##### (c) ノード判定 (空中ノード)

(a)及び(b)で判定したノードの種類の内いずれにも該当しなかったノードは空中領域を表しており、このノードを空中ノードとして判定する。

##### (d) エッジの重み設定

ノードを種類別に分類後、各種類のノード間のエッジに配線のしやすさを反映した重みを設定する。重みは非対称とし、 $w(u, v) \neq w(v, u)$ となることを許容する。例えば、クリアランスノードから配線可能ノードへの接続は容易だ

<sup>†</sup> 三菱電機株式会社 Mitsubishi Electric Corporation

が、逆方向への移動は制約を加えたいため、本手法ではこのような特性を重みに反映する。一例を以下に示す。

表 1 終点ノード別のエッジ重み設定例

No.	エッジの終点ノード	重み
1	配線不可能ノード	$\infty$
2	空中ノード	100
3	クリアランスノード	10
4	配線可能ノード	1

### 3.2 A\*アルゴリズムによる経路探索

本節では、3次元格子状有向グラフ上でA\*アルゴリズム[5]を用いた3次元配線方法を述べる。なお本稿ではA\*アルゴリズムの説明は省略し、経路集約化のための拡張手法について説明する。具体的には、一度配線したエッジの重みをゼロに設定し、再度そのエッジを通る際に経路長が短縮されるようにする。

以上で述べた、グラフの作成から3次元配線経路生成に至るまでの提案手法の流れをアルゴリズム1に示す。

#### Algorithm: 提案手法

```

Input: CAD data, csv data(CABLE.csv, CONNECTION.csv, CS.csv), config.yaml
Output: P
1.  $G = (V, E, w) \leftarrow \text{create3DGrid}(\text{CAD data})$  # 3次元格子状有向グラフを作成
2.  $\text{connections} \leftarrow \text{preprocess}(\text{csv data})$  # 配線接続情報の前処理
3.  $V \leftarrow \text{classifyNodeTypes}(G, \text{config.yaml})$  # 各ノード種別を判定
4.  $w \leftarrow \text{assignNodeCosts}(G)$  # 各エッジに重み付与
5.  $P \leftarrow \text{empty list}$ 
6. for all (start, goal) in connections do
7.    $\text{path} \leftarrow \text{Astar}(G, w, \text{start}, \text{goal})$  # A*アルゴリズムによる経路探索
8.    $P.append(\text{path})$ 
9.    $w \leftarrow \text{setEdgeCost}(\text{path}, 0)$  # 一度配線したエッジの重みを0に設定
10. end for
11. return P

```

#### アルゴリズム 1. 提案手法

## 4. 実験

本章では、提案手法による3次元配線結果の一例を示す。

### 4.1 配線結果

あるCADデータを用いて3次元配線を試行したところ、32本全てのケーブルを配線することができた。図1に、経路集約化を適用しない場合(左)及び適用した場合(右)の配線結果を示す。

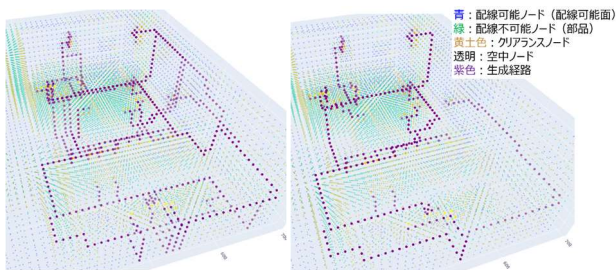


図 1 配線結果例 (左: 集約化無し、右: 集約化有り)

図1において、紫色の点列は各配線経路を示す。配線がクリアランス領域を避けており、また右図では接近した場所を通る経路の集約化が実現されていることが分かる。

### 4.2 計算処理時間

本節では、一連の処理に所要する時間<sup>1</sup>について述べる。表2にCADデータ毎の処理時間を示す。なお、表内の例1は図1で示す3次元格子状有向グラフでの配線結果を示している。

表 2 CADデータ毎の3次元配線設計所要時間

CADデータ	グラフ作成	A*処理	部品数	ノード数	配線数
例1	約11分	約2秒	492	23328	32
例2	約37分	約35分	63	300000	27
例3	約13分	約1分	169	129500	24
例4	約46分	約2秒	305	57645	37

表2より、部品数やノード数が多いほど3次元格子状有向グラフの作成所要時間が長くなり、また配線数が多いほどA\*アルゴリズムによる配線経路生成の所要時間が長くなる事が分かる。

## 5. まとめ

本稿では、基板のCADデータと配線接続情報を基に基板上の3次元配線設計を効率化する手法を提案した。CAD形状と配線制約を反映した3次元格子状有向グラフを作成し、A\*アルゴリズムを用いた経路生成を行うことにより、ケーブルの集約化や部品周辺領域の迂回配線など、敷設作業性の良い配線設計を効率化できることを示した。本手法はA\*アルゴリズムのみを用いることで計算資源を抑えつつ、STPファイルをそのまま利用できる構成により高い汎用性を有しており、従来手法における計算量および適用範囲の制約といった課題を克服している点の特徴である。今後の課題として、配線の曲げ回数考慮と複雑なCADモデルでの検証が挙げられる。

### 参考文献

- [1] F. Yang, P. Wang, R. Zhang, S. Xing, Z. Wang, M. Li, and Q. Fang, "A\*-Ant Colony Optimization Algorithm for Multi-Branch Wire Harness Layout Planning," *Electronics*, vol. 13, no. 3, p. 529, Jan. 2024. doi:10.3390/electronics13030529
- [2] C. Liu, L. Wu, G. Li, H. Zhang, W. Xiao, D. Xu, J. Guo, and W. Li, "Improved Multi-Search Strategy A\* Algorithm to Solve Three-Dimensional Pipe Routing Design," *Expert Systems with Applications*, vol. 240, Art. no. 122313, Apr. 2024. doi:10.1016/j.eswa.2023.122313
- [3] T. Karlsson, E. Åblad, T. Hermansson, J. S. Carlson, and G. Tenfält, "Automatic Cable Harness Layout Routing in a Customizable 3D Environment," *Computer-Aided Design*, vol. 157, Art. no. 103671, Nov. 2023. doi:10.1016/j.cad.2023.103671
- [4] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, \*Introduction to Algorithms\*, 3rd ed., MIT Press, 2009.
- [5] P. E. Hart, N. J. Nilsson and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," in *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100-107, July 1968.

1. 本実験は、Intel Core i7-1260P (最大 4.70 GHz)、32 GB RAM を搭載した Ubuntu 22.04 LTS (64 ビット) 環境で行った。