

エルボおよびベンドを考慮した配管設計アルゴリズム

学正会員 安藤 悠人* 正会員 木村 元*

An Automatic Piping Algorithm including Elbows and Bends

by Yuto Ando, Student Member Hajime Kimura, Member

Summary

Nowadays, the pipe arrangement has been enabled to be more efficient and economical by development and spread of CAD(Computer-Aided Design). However, it is difficult to design a piping layout automatically because there are many regulations and functional design rules which must be satisfied. We propose an automatic routing method for simple pipes including elbows and bends. In a practical design of a piping layout, there are many bends connecting straight eccentric pipes which have gaps within the pipes' diameter. However, no precedence automatic piping algorithm has been taken into account pipelines with such bends. The proposed method finds piping routes making use of not only elbows but the bends in order to minimize costs of the path connecting start point to goal point, while avoiding obstacles such as structures, equipments and the other circuits. In our approach, we regard the piping route design problem as a routing problem in a directed and weighted graph. Note that the nodes in the proposed graph have state variables not only locations but directions of the pipes. This graph can easily express the bends as simple edges, and then the routing algorithm can easily handle the bends. In addition, the presented method has specifications that the sizes of each cell, which is generated by decomposing of a free space, are not restricted within the diameter of the pipe. The routing algorithm uses Dijkstra's method to provide candidate paths. The efficiency of the proposed method is demonstrated through several experiments

1. 緒 言

近年造船業界では、情報処理技術の発達に伴い、これを利用することで作業の省力化を図ってきた。しかし配管設計作業は未だ熟練技術者の経験に頼る部分が多く、設計作業の自動化には至っていない。この原因として、先行研究や既存の自動配管システムの経路探索性能が不十分であることが挙げられる。そこで本研究では、配管設計作業の自動化へ向け、配管設計問題をグラフ上の経路探索問題へ帰着することにより最適な経路を見つけ出す新しい手法の提案を行う。さらに、本論文では空間を局部的に斜めに通る S 字型のパイプピースをベンドと呼称し、提案手法では空間を直角に曲がるエルボだけでなく、ベンドも含めた経路探索を行うことが可能となる。

配管設計問題における先行研究でのアプローチとして、空間を離散化せずにパイプの経路を探索する手法と、空間を離

散化して探索する手法が存在する。ここで、空間を離散化する手法とは、設計対象となる空間をパイプ直径以上の幅を有するメッシュで分割するアプローチを意味する。まず、空間を離散化しないアプローチを適用している研究例としては、鷹・山口ら¹⁾、池平ら²⁾の研究が挙げられる。これらのアプローチでは経路探索の際の計算量を抑えることができるが、複雑な経路に対応できないといった問題が生じる。一方で、空間の離散化を行うアプローチを適用している研究として、Asmaraら³⁾、Pauloら⁴⁾、Parkら⁵⁾、伊藤ら⁶⁾の研究が挙げられる。これらのアプローチは、各メッシュに異なるコストを設定することにより、作業員用の経路を避けつつ壁際にパイプを配置することが可能になるなどの優れた特徴を有している。しかし、先行研究での手法は、空間を分割するメッシュの幅がパイプの径に依存しているため、実際の配管設計問題を取り扱う際に、始点・終点や障害物などがパイプの径に依存した位置に存在するとは限らないなどの理由から、経路の自動生成が不可能となる場合が考えられる。そこで本研究では、空間を分割するメッシュの幅がパイプの径に依存しない、より実用的な手法を提案する。

さらに本研究の独自のアプローチとして、経路中に空間を

* 九州大学大学院工学研究院海洋システム工学部門

局所的に斜めに通る S 字の経路を探索の考慮に入れる。このような経路を本論文ではベンドと呼称する。このようなパイプピースは実際の配管経路において、エルボのみでは現実的な対応がとれない個所に使用されている。

自動設計において、ベンドの考慮を行わない場合、Fig.1 が示しているような迂回経路を生成する可能性がある。Fig.1 の経路は、パイプ直径がパイプの始点・終点の縦方向のギャップよりも大きい場合に獲得される解を示している。このような条件の場合、経路中の直径を均一に保つ形状制約を守るため、獲得される解は迂回した経路となっている。そこで提案手法では、Fig.1 のようなエルボだけでは対応が現実的でない解を避け、Fig.2 のようなベンドを使用した現実的な解の自動生成を試みる。本論文では、いくつかのシミュレーションを通じて本手法の有用性を検証する。

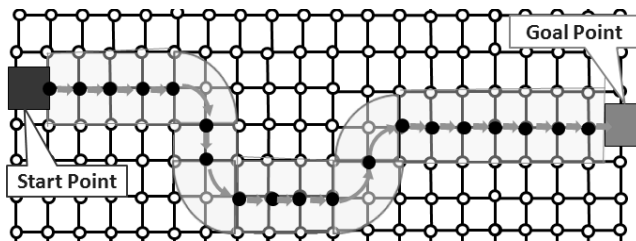


Fig.1 An example of a pipe route with only elbows

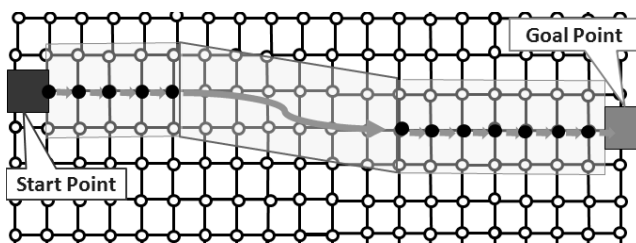


Fig.2 An example of a pipe route with a bend

2. 配管経路設計問題のモデル化

配管設計を行う際に、以下の条件を仮定する：

【設計対象空間】パイプを配置可能な空間の範囲が予め与えられているものとする。本研究ではこの空間を直方体に限定している。

【対象とするパイプ】本問題設定では、分岐を持たず、かつ始点から終点まで直径が変化しない1本のパイプを対象とする。複数パイプを設計対象とする場合は、配置を終えた他のパイプを障害物とみなして1本のパイプの経路設計を繰り返すことで対応する。パイプラックを使用することで、太さの異なる複数本のパイプの高さを揃える等については5.3節にて述べる。

【パイプの配置可能な方向】パイプは、局所的に使用されるベンド部分を除いて、設計対象である直方体の各辺に平行な方向へ配置されるものとする。また、進行方向を変える際はJIS規格に対応した90度のエルボを使用するものとする。配管設計においては、保守管理や工作の容易さ、およびパイプ支持や機器配置の都合上、上記の設定が基本となっている。

【パイプの始点・終点・直径】対象とするパイプの始点の位置座標とそこからパイプが伸びている方向ベクトル、および終点の位置座標とそこからパイプが伸びている方向ベクトルが与えられているものとする。またパイプの直径も予め与えられている。

【障害物の幾何情報】設計対象空間内に存在する構造部材や機器などの位置情報が与えられているものとする。設計対象のパイプは、これらとの干渉を許されない。

【通路空間】乗組員用通路として使用すると予想される設計対象空間内での領域が予め与えられているものとする。この領域内へパイプを通すことは、極度な迂回経路となる場合を除いて、極力回避しなければならない。通路空間に関しては、本論文での実験では取り扱っていないが、通路空間にコストを設定することで、特定の空間へのパイプの通過を防ぐ手法は、Pauloら⁴⁾、伊藤ら⁶⁾により既に示されている。

【パイプラック空間の支持】予めパイプを通すことを前提として用意された空間や、パイプラックが設置しやすい天井付近や壁付近の空間が支持されているものとする。このような空間を本研究では「パイプラック空間」と呼ぶ。設計対象空間内にパイプラック空間を設置することで、パイプを壁際に設置することが可能となる。設計対象のパイプは、このパイプラック空間を通ることが望ましい。

以上の条件のもとで、以下のパラメータを探索する：

【パイプ経路】設計対象である直径が変化せず、分岐の無い1本のパイプの始点と終点を結ぶ経路は、途中のエルボとベンドの座標リストで表される。

パイプ経路の設計目標は、以下のとおりである：

- 1) パイプ長が短いこと
- 2) エルボおよびベンドの数が少ないこと
- 3) 通路空間中を通ることを極力回避すること
- 4) 遠回りにならない限りパイプラック空間を通ること

これらの配管設計問題を単目的最適化問題へ定式化するため、パイプの長さ按比例したコスト、エルボおよびベンドのコスト、通路空間やパイプラックを通る場合のコストをそれぞれ与えておくものとする。パイプ経路探索アルゴリズムは、上記コストの合計が最小になる最適な経路を探索する。

3. 配管経路探索アルゴリズム

3.1 対象空間のメッシュ分割によるグラフ表現

本研究では経路探索アルゴリズムとして、先行研究で提案しているダイクストラ法を適用したアルゴリズムを使用する⁷⁾。このアルゴリズムでは配管経路探索問題を重み付きグラフ(ネットワーク)上の経路探索問題に帰着させることで最小コストの経路を探索している。また、このアルゴリズムは、設計対象空間を格子状に分割する際に、重み付きグラフのノード(節点)を特徴付ける状態量として、格子分割した各格子の位置座標に加え、その格子を通るパイプの方向を考慮することにより、格子間隔がパイプの直径に依存しないといった優れた特徴を有している。さらに本研究では先行研究⁷⁾での配管経路探索アルゴリズムをエルボだけでなくベンドも考慮する仕様に拡張している。また、ノード間を結ぶリンク(エッジ)は、ノード間の距離に比例したコストにより特徴付けられる。さらに、エルボもしくはベンドを使用する場合は、それらのパイプピースを使用しているリンクに材料費を示す追加コストが加算されている。

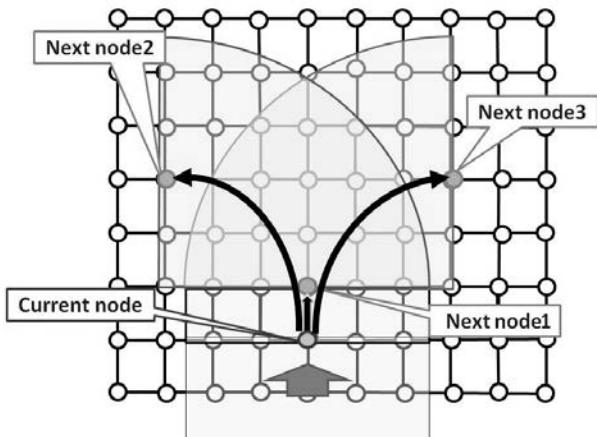


Fig.3 A grid partitioning network model in case of straight and elbows

Fig.3 は経路中において、直進およびエルボを配置して左右に方向を変える場合のノード間の遷移先例を示している。Fig.3の Current node で示されているノードは、探索中のパイプが下方から現在そのノードまで伸びていることを示している。パイプは Current node で示されたノードの下方にある長方形であり、空間を分割したメッシュの幅よりも大きな直径を有している。本手法では Current node におけるノードの位置座標およびパイプの方向をダイクストラ法で扱う 1つの状態量とする。Current node から直進する場合に移動可能なノードは、パイプの進行方向に沿って、1つ先に位置している Next node1 である。この場合は、パイプをメッシュ

の寸法分だけ伸ばすことに相当するので、その距離に比例したコストが加算される。さらに Current node から直進した場合に障害物との干渉判定が行われる。干渉判定は、パイプに内接する最小の直方体と障害物との干渉を、それぞれの座標を比較することで行う。干渉しないと確認された場合、新たな遷移先候補となる。

次に、エルボを経由して左右に曲がる場合での遷移先ノードは Next node2 および Next node3 で示されている。これらのノードは直管とエルボを配置することで遷移可能な最も近い位置に存在する。エルボを使用する場合の干渉判定は、エルボに内接する最小の直方体と障害物との干渉判定を行う。さらに、コストとしてノード間のマンハッタン距離にエルボに応じた値を付与する。以上のような処理を行うことで Current node からの遷移先ノードを獲得していく。遷移先のノードまでの距離をその都度計算していくので、メッシュ幅が等間隔でなくとも処理が可能である。

3.2 ベンドのグラフ表現方法の提案

本手法では、先行研究では考慮されていなかったベンドを経由した場合の遷移先例も探索している。本論文で呼称しているベンドは、パイプ曲げ加工機を使用して直管を任意の角度で曲げることで工作可能なパイプピースである。

まず、ベンドをグラフ表現するにあたり、Fig.4で示しているようにベンドを上下に位置する曲管部分と、その間に位置する直管部分の3つのパーツに分割して考える。

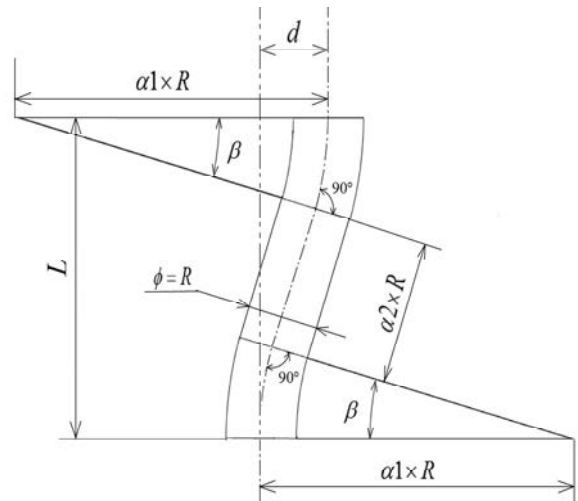


Fig.4 A bend part composed of three primitives

ここでベンドの工作上的制約から決まる曲げ半径係数 $\alpha1$, $\alpha2$ 及び設計上の要求から決まる d を与えることで、ベンドの曲がり角度 β を以下の式で求める。

$$\beta = \arcsin \left\{ \frac{d - 2 \times \alpha1 \times R}{\sqrt{(2 \times \alpha1)^2 + (\alpha2)^2}} \right\} + \arcsin \left\{ \frac{2 \times \alpha1}{\sqrt{(2 \times \alpha1)^2 + (\alpha2)^2}} \right\} \quad (1)$$

さらに、式(1)により与えられた β を用いて、バンドの設置に必要な最小の距離である L を以下の式より求める。

$$L = (2 \times \alpha_1 \times R \times \sin \beta) + (\alpha_2 \times R \times \cos \beta) \quad (2)$$

ただし、本研究では $\alpha_1 = 5$, $\alpha_2 = 0$, $d < R$ と設定した。ここで、 $d < R$ と設定している理由は1章における Fig.1, 2 で示したように、 $d < R$ の条件時でのバンドを考慮しない場合における非現実的な解の生成を避けるためである。また、 α_1 , α_2 の設定に関してはバンドの加工に用いる機械の制約に応じて設定すればよい。よって、本実験での設定は1つの典型例を示している。

本手法は、式(2)により求めた L 以上の距離に存在し、かつ最も近い場所に位置するノードに対してバンドの適用も考慮しながら始点・終点間を結ぶ経路の探索を行う。

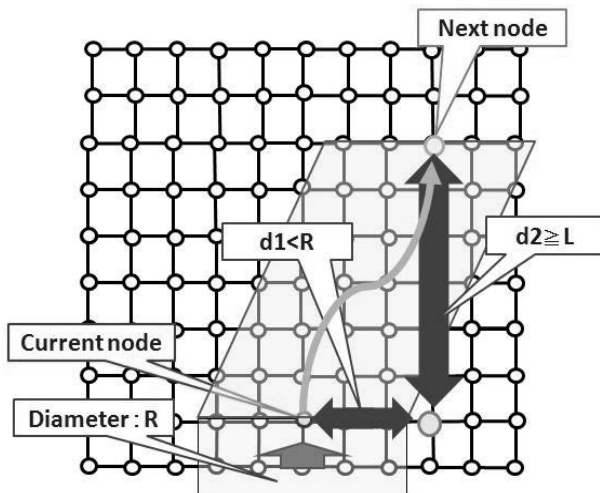


Fig.5 A grid partitioning network model in case of bends

Fig.5 は 2 次元空間におけるバンドを経由した場合におけるノードの状態遷移例である。Fig.5 において Current node で示されたノードは、探索中のパイプ経路が下方から上方へ伸び、現在そのノードまで伸びていることを示しており、パイプは空間を分割したメッシュの幅より大きな径を有している。この Current node から右にずれるバンドを経由することで到達可能となるノードが図中の Next node である。Next node を探索する際には、バンドを配置するための空間中に障害物があるかどうかの干渉判定を行う。この場合の干渉判定は、バンドを上半部に内接する最小の直方体と下半部に内接する最小の直方体に置き換えて行う。そして物理的に配置可能な場合のみ、これらのノードを新しいリンクで結びネットワークを構成していく。この新しいリンクには、エルボを配置する場合と同様に、ノード間を接続したパイプの長さおよびバンドに応じたコストを寄与する。

4. 計算機シミュレーション実験

4.1 実験設定

本手法の動作を確認するため、障害物として 10 個の障害物が配置された、 x 座標 0~16m, y 座標 0~3m, z 座標 0~3m の設計対象空間でシミュレーションを行う。パイプの始点は座標(0.5, 1.75, 1.5)であり x 軸正方向へ伸びる。パイプの終点は座標(15.75, 1.5, 1.5)であり x 軸正方向からパイプが伸びてきて終点の座標にて止まる。この設計対象空間を 0.25m 間隔で格子分割し、始点・終点および障害物の位置を変化させずに、パイプの直径のみを変化させた場合にどのような経路が獲得されるのかについて確認する。パイプの直径は格子間隔以下の 0.2m のほか、格子間隔以上である 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9m の計 8 種類について、経路中におけるバンドの出現を考慮している場合としていない場合のそれぞれ 2 パターンずつ、合計 16 パターンのシミュレーションを行う。また、コスト設定としては、直管の単位長 (1m) をコスト 1 とし、エルボのコストとしてはパイプ長に相当するコストに加え、エルボ 1 つあたりにコスト 0.1 を加算した。またバンドのコストについてもパイプ長に相当するコストに加え、バンド 1 つあたりにコスト 0.3 を加算した。これらのコスト設定は実験における一例であり、実問題に対応する場合はそれぞれのパイプピースの値段を設定できる。なお、詳しい障害物の配置については Appendix に記載する。

4.2 実験結果

計算環境は、OS に Windows7 を使用し、CPU は Intel Core2 Duo 2.8Ghz, メモリーは 2.00GB, プログラム言語は Java version 1.6 を用いた。

経路中におけるバンドの出現を考慮している場合における、本手法で獲得された経路を Fig.6 ~ Fig.13 に示す。また、取り扱ったグラフにおけるノードの数、経路中のエルボの数、バンドの数、始点から終点までのコスト合計、経路探索に要した時間についてパイプの直径ごとに Table 1 にまとめた。経路中にバンドの出現が確認されたのは、直径がそれぞれ 0.3, 0.4, 0.5, 0.6, 0.9m の場合であり、これらの解は本手法によって初めて得られたものである。またパイプ直径が格子間隔以下である 0.2m の場合は、その他の場合と比較して数倍の探索時間が必要であることが確認された。さらに、すべての直径においてバンドを考慮した場合の方が、考慮しない場合よりも探索時間が長かったが、基本的に数秒程度の差であり、バンドを考慮することで探索時間が大幅に増加することはないと考えられる。

Table 1 Features of obtained solutions with 0.25(m) meshes of x, y, and z direction

	Diameter [m]	Num. of Nodes	Num. of Elbows	Num. of Bends	Total Costs	Time [s]
Without Consideration of Bends	0.2	43070	9		17.9	1260
	0.3	37766	9		19.4	306
	0.4	35441	9		19.9	298
	0.5	35441	9		19.9	288
	0.6	19007	8		22.3	68
	0.7	17237	9		22.4	53
	0.8	17010	9		22.4	50
	0.9	15446	15		27.5	44
With Consideration of Bends	0.2	43070	9	0	17.9	1285
	0.3	37878	7	1	19	447
	0.4	35467	7	1	19.5	387
	0.5	35408	8	1	19.6	373
	0.6	18997	7	1	21.5	80
	0.7	17397	9	0	22.4	68
	0.8	17192	9	0	22.4	63
	0.9	15171	14	1	26.7	45

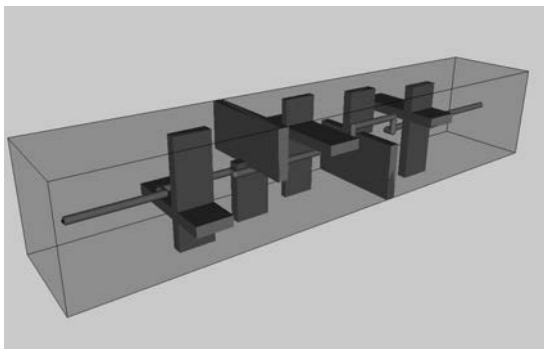


Fig.6 An obtained route with 0.2(m) diameter of the pipe

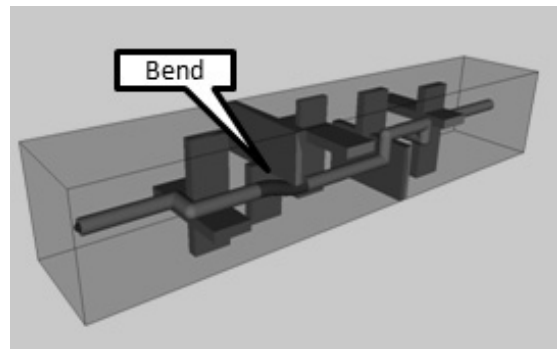


Fig.8 An obtained route with 0.4(m) diameter of the pipe

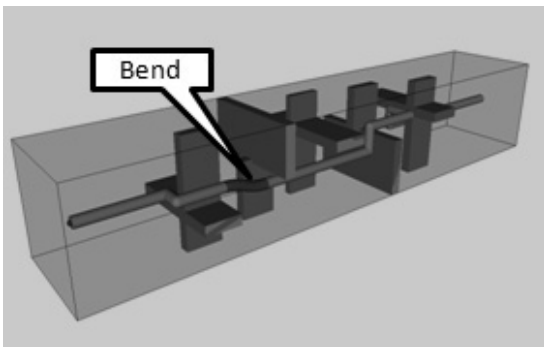


Fig.7 An obtained route with 0.3(m) diameter of the pipe

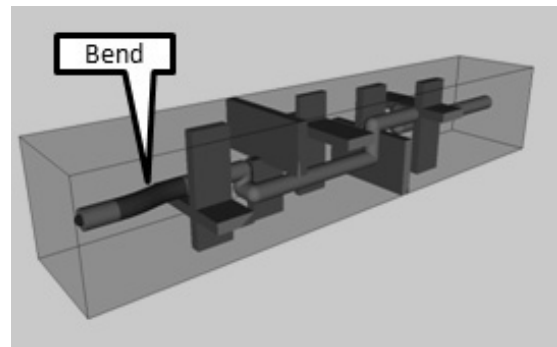


Fig.9 An obtained route with 0.5(m) diameter of the pipe

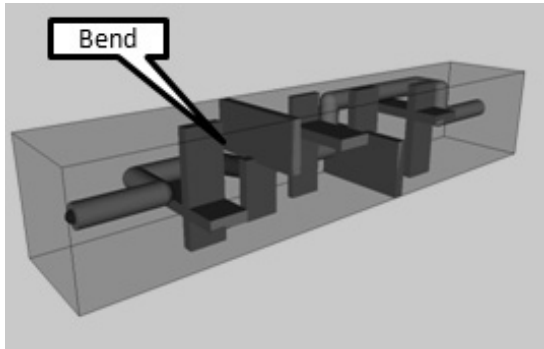


Fig.10 An obtained route with 0.6(m) diameter of the pipe

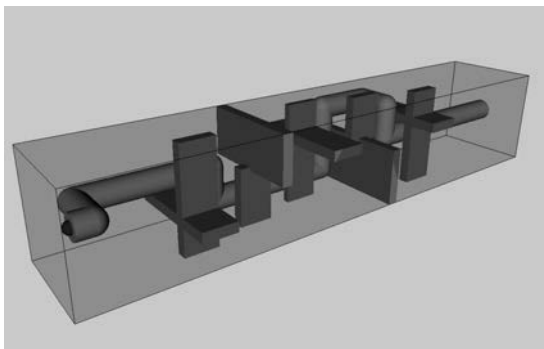


Fig.11 An obtained route with 0.7(m) diameter of the pipe

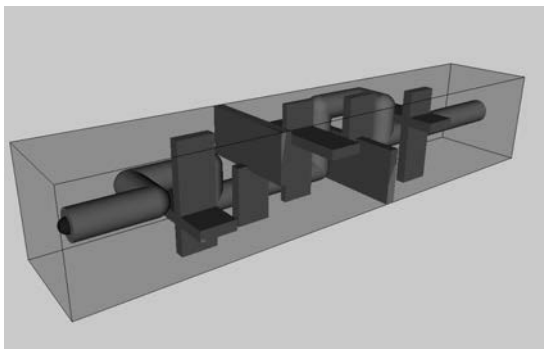


Fig.12 An obtained route with 0.8(m) diameter of the pipe

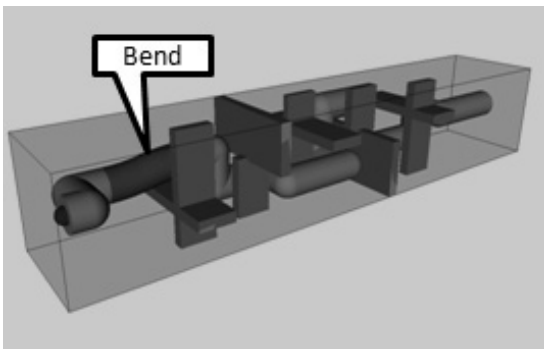


Fig.13 An obtained route with 0.9(m) diameter of the pipe

5. 考察

5.1 実験結果について

Fig.7 ~ Fig.13 で示されている経路は、パイプ直径がメッシュ幅よりも大きいという観点から本手法によって初めて得られた解である。また、経路中にベンドが確認されたのは、直径がそれぞれ 0.3, 0.4, 0.5, 0.6, 0.9m の場合であった。Fig.7 ~ Fig.10 および Fig.13 の配管経路中において、矢印で示されている部分がベンドであり、ベンドを考慮していない従来手法では得られなかった解である。Fig.6 で示されているケースでは、パイプの径が空間を分割しているメッシュ幅よりも小さく、経路のコストも全経路の中で最小であることが確認される。これに対して、パイプの径が大きくなるにつれて、コストが大きくなっている。よって、この実験により、本来は大きく迂回していた経路に対して、ベンドを適用することで各パイプ直径に応じた制約条件を満たす最適な経路を獲得することが、Table 1 より確認できる。

また、Fig.7 ~ Fig.10 および Fig.13 におけるベンドの出現個所はいずれも始点付近であることが確認される。これは経路の自由度が始点付近で限られるためであると考えられる。よって、本手法では経路の始点・終点付近のほうが、経路の中盤よりもベンドが現れる可能性が高いと考えられる。

最後にパイプの直径が小さくなるにつれて経路探索時間が増加していることが Table 1 より確認できる。経路中におけるノードの数が増加していることから考えて、これはパイプの直径が小さくなることで、設計対象空間内での経路探索を行う領域が増加したためであると考えられる。よってパイプの直径が小さい場合の処理時間短縮を目的とした配管経路探索アルゴリズムの改良を計画している。本手法において経路探索アルゴリズムとして活用しているダイクストラ法は、グラフ問題の探索において隣接した全てのノードについて探索を行う横型探索に分類されるが、この横型探索を適用した場合、解の生成に多くのメモリと時間を要する。よって、今後の方針として、ダイクストラ法をさらに改良した A*アルゴリズムの適用を検討している。

5.2 メッシュ間隔における問題について

本実験とは計算環境、設計空間のサイズ、障害物および始点・終点の位置は変えずに、x 軸方向における格子分割幅を 0.5m とした場合の実験結果を Table 2 に示す。

この実験で獲得された経路は、格子幅 0.25m の場合と比べ、全く同様の経路であった。しかし、探索に要した時間が、特にパイプの径がメッシュ幅よりも小さい場合に大幅に短縮されたことが Table 1 と Table 2 を比較することから確認さ

れる。この結果より、設計空間における各方向について適切なメッシュ間隔を設定することで、経路探索に要する時間を短縮することが可能であると考えられる。よって、様々な設計対象空間に対して、どの程度のメッシュ間隔を設定すれば妥当であるかに関して、今後の研究が必要である。

Table 2 Features of obtained solutions with 0.5(m) meshes of x axis, 0.25(m) meshes of y and z direction

Diameter [m]	Num. of Elbows	Num. of Bends	Total Costs	Time [s]
0.2	9	0	17.9	213
0.3	7	1	19.0	146
0.4	7	1	19.5	126
0.5	8	1	19.6	126
0.6	7	1	21.5	24
0.7	9	0	22.4	23
0.8	9	0	22.4	22
0.9	14	1	26.7	12

5.3 パイプラックを含んだ経路探索

パイプラックを含んだ設計対象空間での動作確認を行うための単純な追加実験を行った。x 座標 0~3m, y 座標 0~2.5m, z 座標 0~2m の設計対象空間内に、始点 (0.5, 2.0, 0.5) および終点 (2.75, 2.0, 0.5) を配置し、これらを結ぶ直径 0.6m の配管経路を探索する。x 座標 1.3~1.7m, y 座標 0.75~2m, z 座標 0~0.75m の範囲を障害物として設定し、x 座標 0~3m, y 座標 0~0.75m, z 座標 0~2m の範囲をパイプラック空間とする。この実験では、パイプラック空間を考慮する場合と、考慮しない場合の 2 通りの実験を行う。

コスト設定としては、エルボのコストを 1 つあたり 0.1、バンドのコストを 1 つあたり 0.3、直管のコストを単位長さ (1m) あたり 1.0 に設定した。また、パイプラックを通過する部分の配管のコストは 10 分の 1 に軽減されるよう設定した。このパイプラック空間での割引率に関してはユーザーが設定できる仕様となっている。Table 3 に経路の詳細を示す。また、パイプラックを設定しない場合の獲得経路を Fig.14、設定する場合の獲得経路を Fig.15 にそれぞれ示す。計算環境は 4 章でのシミュレーション実験と同様である。

Table 3 Features of obtained solutions with and without pipe-rack area

	Diameter [m]	Num. of Elbows	Total Length [m]	Total Costs
Without pipe-rack	0.6	4	3.25	7.25
With pipe-rack	0.6	4	5.25	5.2

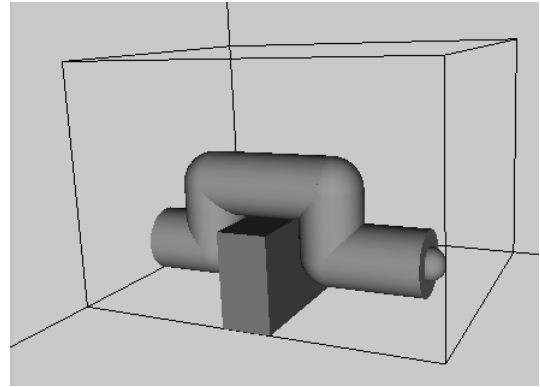


Fig.14 An obtained pipeline without pipe-rack area

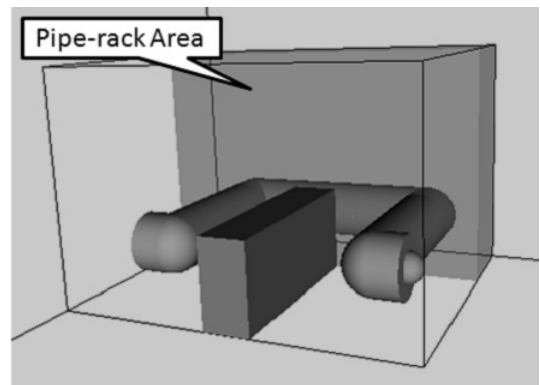


Fig.15 An obtained pipeline with pipe-rack area

Fig.15 において矢印で示されている空間がパイプラック空間である。Table 3 内の比較より、パイプラック空間を設定し、通過した場合の方が、パイプの総長が長くともコストとしては低い値を示していることが分かる。今後の課題として、現段階ではパイプラック空間をとパイプ経路が少しでも干渉した場合、経路がパイプラック空間を通ったとみなしているが、パイプラックと経路の干渉の度合いを測る仕様に変更することを検討している。さらに、今回の実験ではパイプラック空間での割引率を 0.1 に設定しているが、最適な割引率に関してもさらなる検討が必要である。

このようなパイプラック空間を既に設定された経路の横に配置することで、実際の配管経路のように複数本のパイプを束にすることや、パイプラック空間のメッシュ幅を変化させることでパイプの高さを揃えることが可能となる。

5.4 経路探索における問題について

本実験とは異なった実験条件下で、Fig.16 のようなパイプが自身と干渉した解が得られるケースが確認された。提案手法ではダイクストラ法で扱う 1 つのノードとして、パイプの座標および方向を用いおり、Fig.16 が示しているような自身と干渉しているケースでは探索の最中に座標は同じだが方向は違う 2 つのノードが獲得されている。しかしダイクスト

ラ法はその構造上, 探索が終わるまでどのノードが使用されるか分からないので, 座標が同じだからといってそこから先の探索を打ち切ることが不可能である. よってパイプが自身と干渉している解が得られた後で, その解を自動的に除去するようなアルゴリズムの改良が必要である.

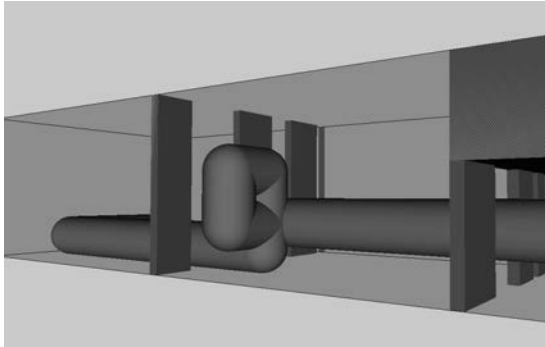


Fig.16 An obtained self interfered route

6. 結言

本論文では, 配管設計作業の自動化に向けて, 障害物が多数存在する複雑な空間内で分岐のない 1 本のパイプの最適な経路を見出すため, S 字型のパイプピースを経路中に考慮する手法を提案した. また, 本研究で使用している経路探索アルゴリズムは, 空間を格子分割する際のメッシュの幅がパイプの径に依存しないアプローチを採用することで, より実問題に対応した経路案を獲得できる仕様となっている. さらに, 経路探索の際にダイクストラ法を適用しているため, 獲得される経路案は最小コストが保証されている. これらの手法を Java プログラムに実装し, 設計対象, 始点・終点, 障害物の配置は変化させずに, パイプの径のみを変化させた場合における例題へ適用することで, それらの獲得された経路がパイプの各直径に応じた最適解となっていることを検証した.

謝 辞

本研究は, 日本学術振興会科学研究費補助金基盤研会 (B) 課題番号 23360388 より一部補助を受けた.

参 考 文 献

1) 蔦 敏和, 山口 太郎: 機器配置・配管の自動化と 3 次元表示, 情報・システム・利用・技術シンポジウム論文集,

vol.27, pp.327-330 (2004 年 12 月)

- 2) 池平 怜史: 船舶分野における配管設計の自動化手法に関する研究, 九州大学, 博士論文 (2010 年)
- 3) Asmara, A. and U. Nienhuis: Automatic piping system in ship, proceedings of the 5th International Conference. on Computer and IT Application (COMPIT), Mar. Ind. 2006. Leiden.
- 4) T. M. Paulo, and V. J. A. S Lobo: A Tool for Automatic Routing of Auxiliary Circuits in Ships, Proceedings of 14th Portuguese Conference on Artificial Intelligence (EPIA 2009), Aveiro (Oct. 2009)
- 5) J. H. Park, and R. L. Storch: Pipe-routing Algorithm Development: Case Study of A Ship Engine Room Design, Journal of Expert Systems with Applications, Vol.23, No.1, pp.299-309 (2002)
- 6) 伊藤 照明, 福田 収一: 遺伝的アルゴリズムを用いた配管経路探索法の検討, 人工知能学会誌, Vol.13, No.5, pp.791-802 (1998 年)
- 7) 木村 元: ダイクストラ法を用いた配管設計アルゴリズム, 日本船舶海洋工学会講演論文集, Vol.11, pp.121-124, 2010 年 11 月.

Appendix

4 章の実験における障害物の座標リスト

	x_min [m]	x_max [m]	y_min [m]	y_max [m]	z_min [m]	z_max [m]
Area	0	16	0	3	0	3
Obstacle1	3	4	0	3	0.8	1.1
Obstacle2	3	4	1.8	2.1	0	3
Obstacle3	7	8	0	3	1.9	2.2
Obstacle4	7	8	0.9	1.1	0	3
Obstacle5	11	12	0	3	1.9	2.2
Obstacle6	11	12	1.9	2.2	0	3
Obstacle7	5	6	1.3	1.6	0	1.7
Obstacle8	5.3	5.6	0	3	1.6	3
Obstacle9	9	10	1.3	1.6	1.6	3
Obstacle10	9.3	9.6	0	3	0	1.7