ALEKSANDER KNIAT, M.Sc.
Technical University of Gdańsk
The Faculty of Ocean Engineering
and Ship Technology

On the possible application of wave and hierarchical algorithms in routing pipes in three dimensional space

SUMMARY

The principles of both hierarchical and wave algorithms are presented and their possible applications discussed. The aim of both algorithms is to automatically route pipes in 3D space in order to facilitate the process of creation of the virtual 3D model of piping installations.

Solutions proposed by the algorithms are based on the P&ID (Piping and Instrumentation Diagram) and known positions of appliances to be connected to the pipes. A brief comparison of the wave and hierarchical algorithms as well as results of experiments made with them are presented.

INTRODUCTION

The principles of both hierarchical and wave algorithm are presented and their possible applications are discussed. The advantages and disadvantages of both algorithms are summarized and the hybrid algorithm which provides switching between the wave and hierarchical algorithms is considered. Results of the hybrid algorithm applied to a more complex exemplary installation are shown at the end.

To the author's knowledge the problem of pipe route optimization in 3D space is not yet solved in the CAD/CAM programs for piping.

This is probably due to difficulty of the optimization problem. Classification of the Steiner's problem as „NP-complete" in the graph theory implies that an optimum solution could not be found by a deterministic algorithm in a short time.

WAVE ALGORITHM

Principles of the algorithm

The described algorithm is a very simplified version of the algorithm presented in [1]. This algorithm produces a tree which is very close to the optimum solution. Simplicity and accuracy are its advantages.

The idea of the algorithm is based on the wave propagation phenomena. The wave propagates on the shortest way from its source to destination. The wave algorithm works on an array which contains vertices of the grid. The connection points of pipes can be placed only in cells of this array. The cells where obstacles exist are blocked for wave propagation. The wave or waves propagate from one or more connection points or from already routed pipe branches. When the wave meets connection point, routed pipe branch or another wave, a new pipe branch is routed (Fig.1 a, b, c, d). If the wave cannot meet another connection point, another wave or routed pipe branch and there are still not connected points, the algorithm fails. This happens when the configuration of obstacles prevents from making connections.

This is the shortest connection between two points. Fig.2 a, b, c and d show how a branch between an already routed pipe branch and a connection point is found.

For the demonstration purposes the principles of the wave algorithm were shown in 2D space. Routing in 3D space requires a three dimensional array. The connection point coordinates must include Z coordinate. The wave must propagate in six directions and direction priorities must include all of them.

Sequential (one-by-one) routing the pipes is a particular feature of the wave algorithm. The sequence of connections taken into account during optimization may have serious influence on the result.

Definition of 3D space and allowed placement of connections

The space in which the connection points are placed is to be divided into elementary cubes - grids. The size of an elementary cube is to be chosen so as to allow only one connection point in a cube. To obtain better accuracy it is possible to farther divide cubes into smaller parts.

The wave propagates from any connection point into all six directions „flooding" more and more new cubes - grids. A route between two points is formed of the edges which connect centres of neighbour grids. At most six perpendicular edges come to the centre of one grid from six neighbour grids. So the final solution is a tree which vertices can be of 6th degree at most.

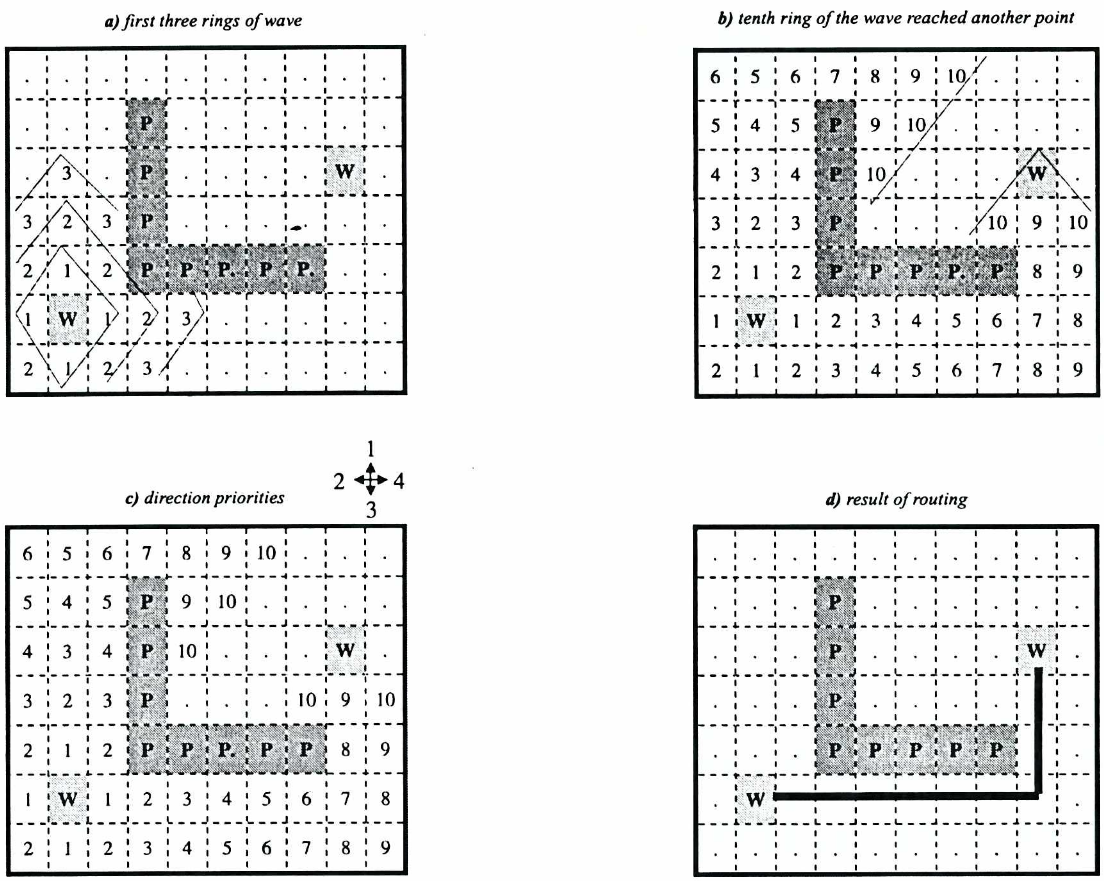


Fig.1. Process of routing between two points
(1,2,3 ... etc. - rings of wave, W - point, P - obstacle)

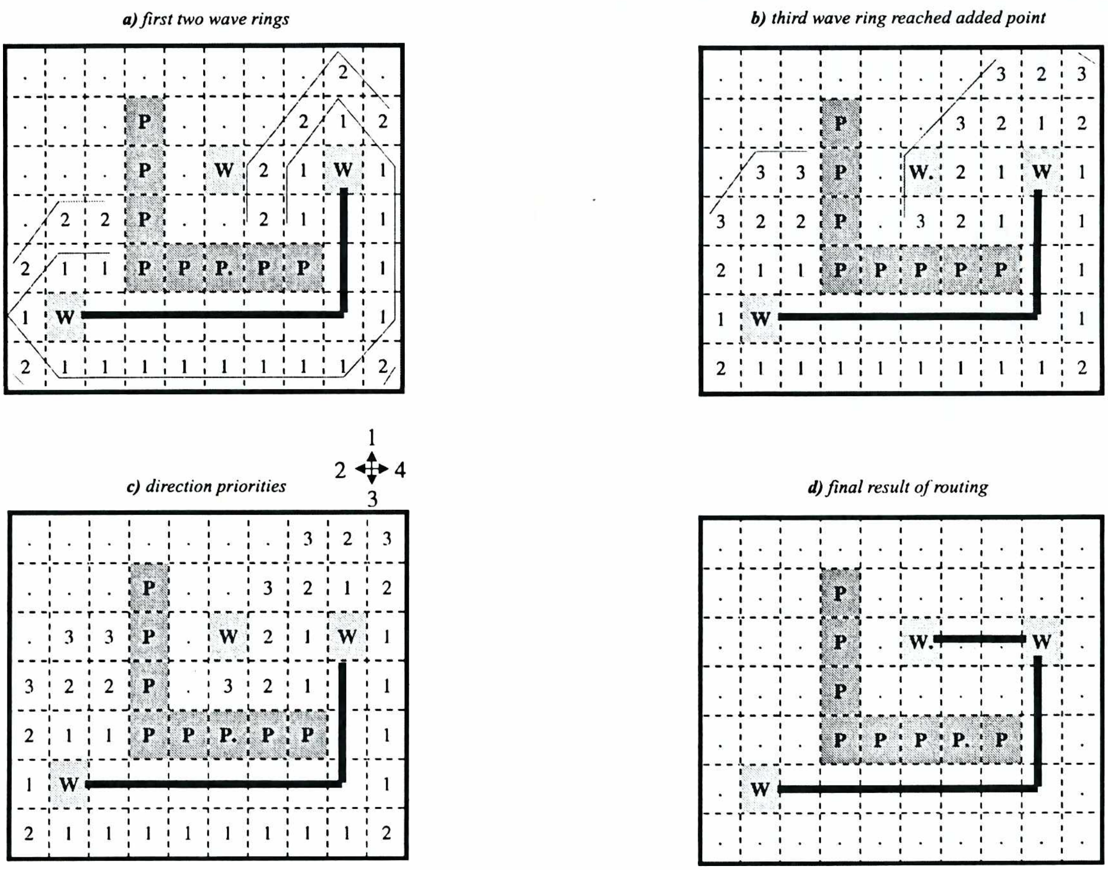


Fig. 2. Process of routing between a branch and connection point
(1,2,3 ... etc. - rings of wave, W - point, P - obstacle)

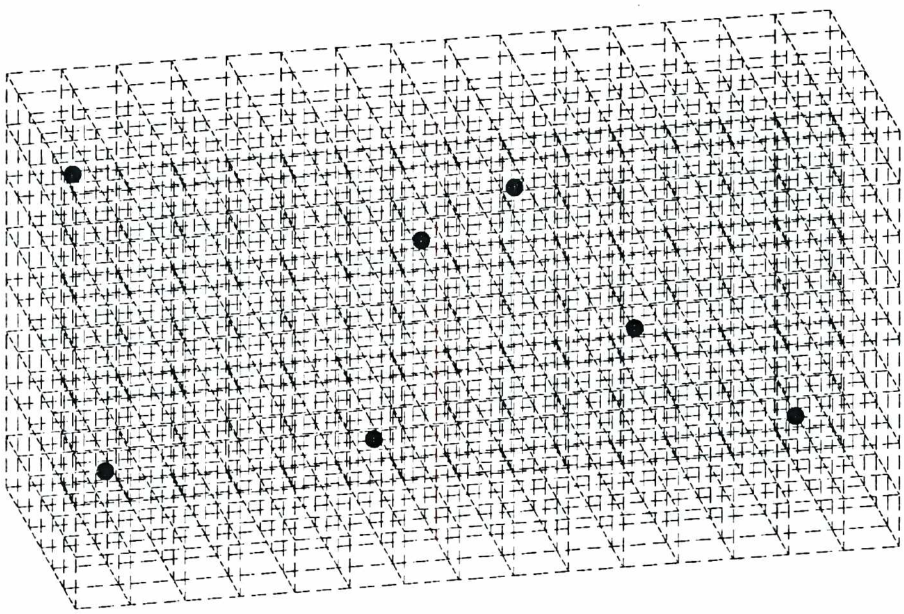


Fig.3. Example of 3D space for routing between 7 connection points (Connection points are marked by spheres. Dashed lines are edges between grids)

HIERARCHICAL ALGORITHM

Principles of the algorithm

The hierarchical algorithm for IC (Integrated Circuit) and PCB (Printed Circuit Board) layout routing was described in [2] and [3]. Only orthogonal (horizontal and vertical) route placement on planar space with rectangular limits was allowed. The purpose of the algorithms was to find an optimum Steiner tree or trees for one or more nets of the connection points.

The algorithm described in [2] divides space into four parts by using horizontal and vertical lines. Thus a 2x2 matrix is created. Routing takes place in this matrix. Finding the optimal tree in 2x2 matrix is relatively simple. Then each of four cells of the matrix is farther divided into four parts. The process continues generating more and more accurate solutions.

The allowed configurations of vertices in 2x2 matrix and possible solutions are as follows :

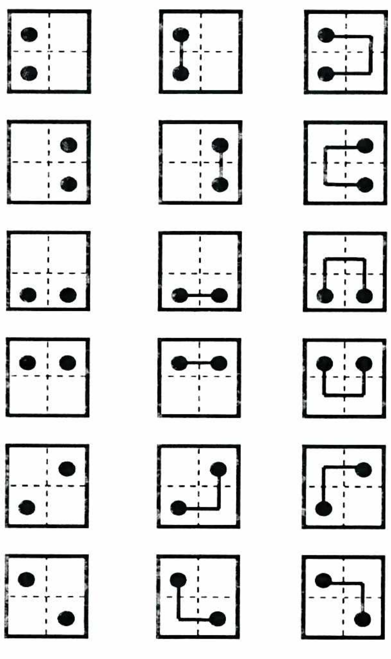


Fig.4. Possible 2-vertex configurations in 2x2 matrix and all routing solutions for these configurations

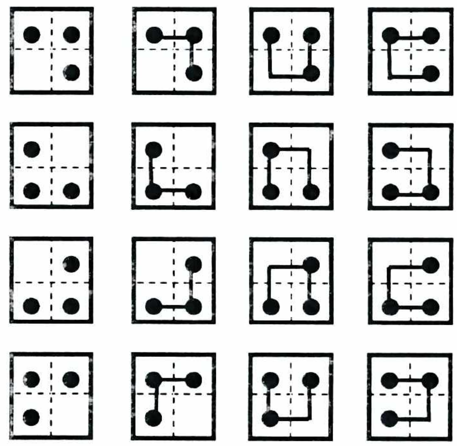


Fig.5. Possible 3-vertex configurations in 2x2 matrix and all routing solutions for these configurations

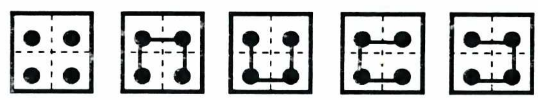
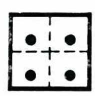


Fig.6. Possible 4-vertex configurations in 2x2 matrix and all routing solutions for these configurations

The choice of the solution for a particular configuration of vertices in a 2x2 matrix depends on the cost of the solution. Because each edge has its cost the cost of the solution is a sum of the costs of edges. The cost of an edge can be its length. The algorithm always minimises the cost of the solution.

An example is given below. In the first step the whole routing area is divided into four parts (Fig.7) :



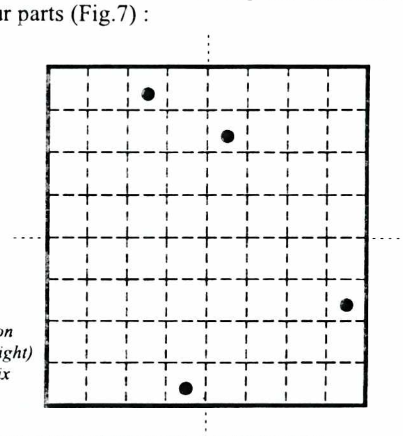


Fig.7. Initial vertex configuration with dividing lines (on the right) and equivalent 2x2 matrix (on the left)

The solution may be as follows (Fig.8) :

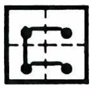
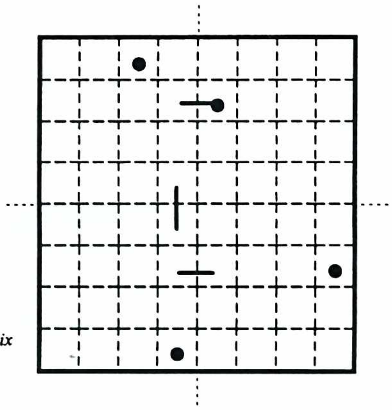



Fig.8.
The set of edges in 2x2 matrix for the first division

In the second step each cell of the initial 2x2 matrix is again divided into four cells (Fig.9) :

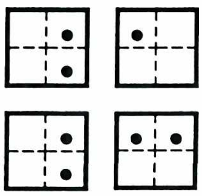
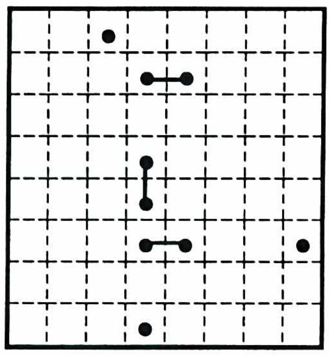



Fig.9.
Division into 2x2 matrices

Here is the choice of edges in the 2nd step (Fig.10) :

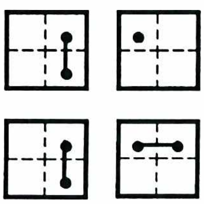
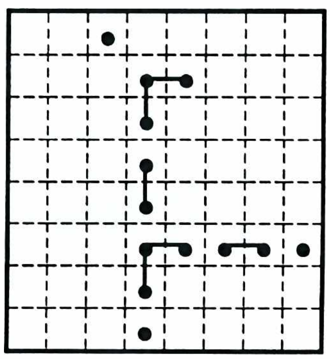



Fig.10.
Four solutions in four 2x2 matrices in the 2nd step

In the 3rd step each cell is again divided into four parts and edges are placed. Now all the edges form the Steiner's tree. The algorithm stops at this step because there is no need to continue divisions any more.

Here is the final solution (Fig.11) :

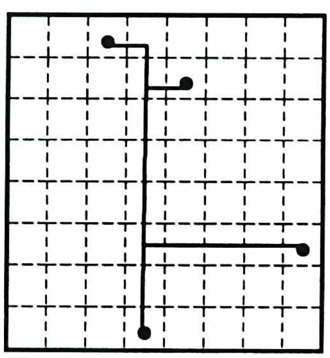


Fig.11.
Steiner's tree created by hierarchical algorithm

Definition of 3D space and allowed placement of connections

Three orthogonal planes cross each connection point. These planes are parallel to X-Y, Y-Z and X-Z planes of the co-ordinate system. Outer planes form walls of bounding cube. All connection points are inside the cube and the solution will be placed in the cube as well. Inside the cube there are also many points and edges on the crossings of the inner planes. The points which are not connection points will be called Steiner points and the edges will be used to form a solution. Each route between the connection points will consist of the edges. The solution is the set of the edges which join all the connection points and the sum of edge lengths is minimum. The solution is a tree with vertices of 6th degree at most.

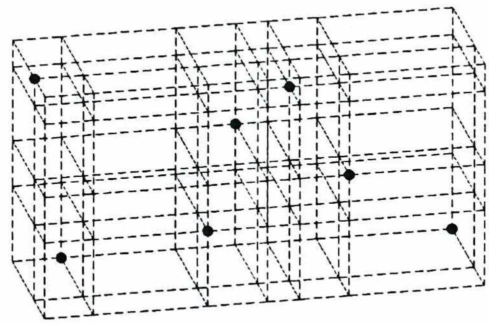
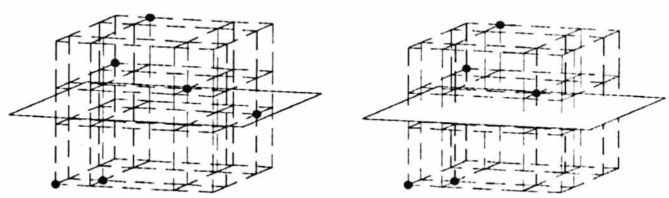


Fig.12. Example of 3D space for 7 connection points
(Connection points are marked by spheres, dashed lines represent edges, other points on crossings of the dashed lines are Steiner points)

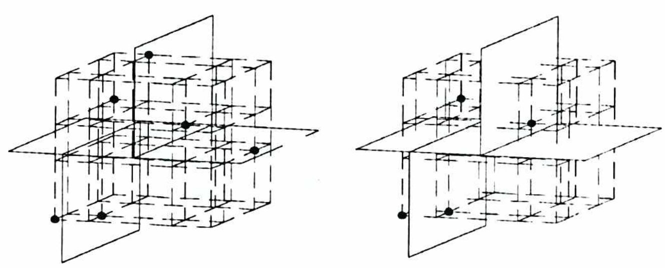
The general principles of the algorithms in 3D space remain unchanged in comparison to those in 2D space. The routing process takes place in a simplified matrix and step-by-step dividing of 3D space results in more and more accurate solution.

Description of hierarchical algorithm in 3D space

Division of 3D space into two parts is assumed. The division is performed with one plane parallel to one of the X-Y, Y-Z i Z-X planes of the co-ordinate system. Each division can be made with a plane parallel to another plane of the co-ordinate system. The dividing plane must cross edges and not include either connection or Steiner points.



First division



Two next divisions with planes perpendicular to the first one

Fig.13. Method of dividing 3D space into two parts
***On the left**: a view with all visible edges*
***On the right**: a view with the edges hidden behind the division planes*

A connection of two points will be made with the edge which crosses the division plane. It is the simple connection within 2x1 matrix.

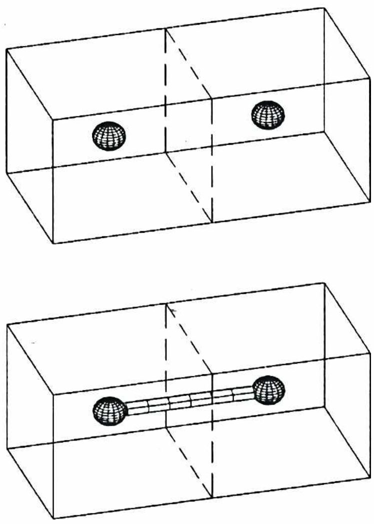


Fig.14. 2x1 matrix
In the upper figure *: possible point locations*
In the lower figure *: only one allowed edge*

COMPARISON OF WAVE AND HIERARCHICAL ALGORITHMS

Comparison

The presented comparison shows how different the two algorithms are. The idea is to utilize their advantages. It is possible to make the hybrid algorithm which switches between the hierarchical and wave algorithms. The hybrid algorithm starts with the hierarchical one which is faster but less accurate. If the hierarchical algorithm fails the hybrid algorithm switches to the wave algorithm to find exact solution. However the space for the wave algorithm is limited to a part of the entire space in which the hierarchical algorithm has failed.

Property	Wave algorithm	Hierarchical algorithm
Computational complexity	Square Number of operations carried out by the algorithm increases according to the second power of the number of connection points	Linear Number of operations carried out by the algorithm increases according to the number of connection points
Time of execution	Long	Very short
Quality of results	Very good Results are very close to optimum	Acceptable Total length of connection lines is about 20% worse than the optimum
Possibility of dividing big routing areas into smaller parts - divide and conquer method	Not acceptable Because wave must spread in whole space	Principle of the algorithm In each step 2x2x2 matrix is an independent problem to be solved
Accuracy of space representation	The whole space is divided into elementary cells – grid. Doubling the accuracy causes eightfold increase of the number of cells	Algorithm divides space that far that is necessary for connection points
Empty areas	Not different from areas occupied by connection points, filled with wave rings	If edges not cross them they are omitted by the algorithm and no memory is allocated for them
Obstacles	Easy to implement with accuracy of grid. They do not increase computational complexity	Difficult in implementation and cause computational complexity much higher
Implementation problems	Managing memory which is very much consumed by the program	Recognising obstacles and routing in areas closed with obstacles
Sequence of routed installations	Has great impact on results	Of minor importance possible simultaneous routing of several installations
Parallel solving fragments of the problem	Not possible	Possible Could increase algorithm efficiency in parallel or distributed environment

Examples

A simplified P&ID diagram of the fuel heating block located inside the ship machinery room is shown in Fig.15. Piping in this block will be automatically routed with the use of the hybrid algorithm.

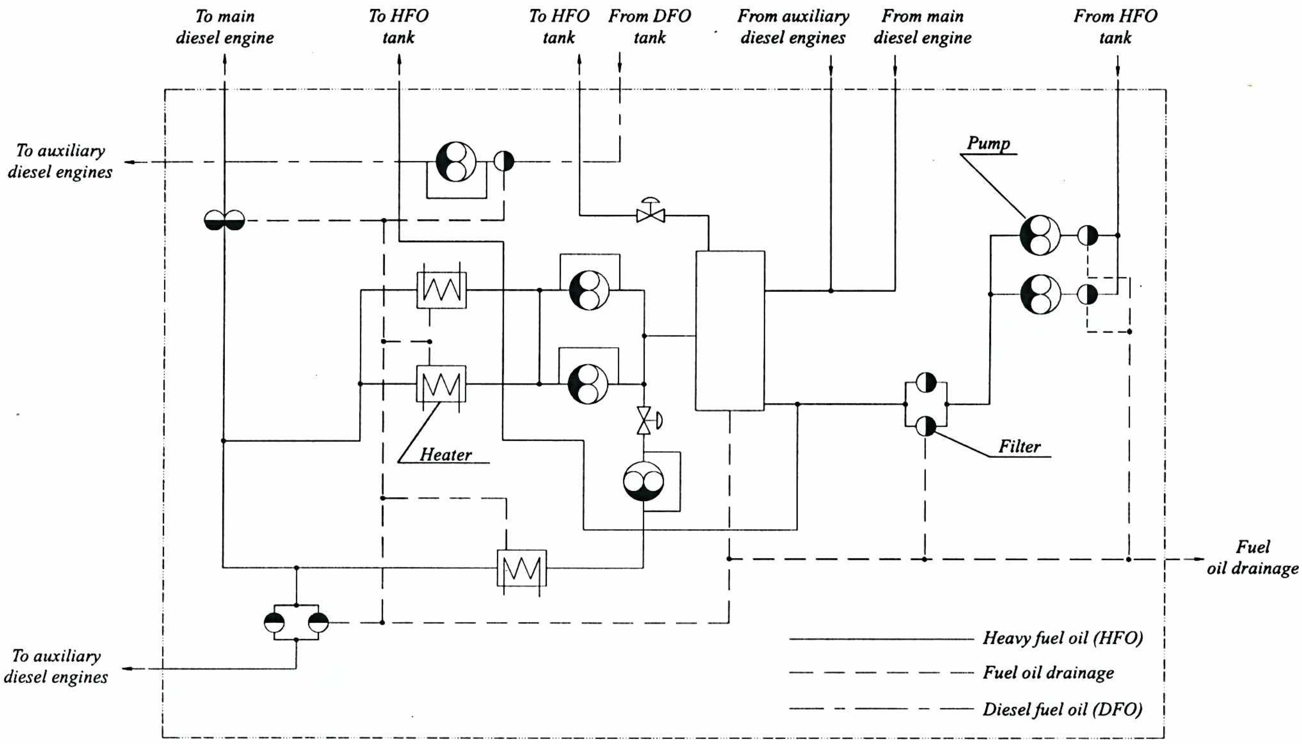


Fig.15. Simplified P&ID diagram of fuel heating block within ship machinery room

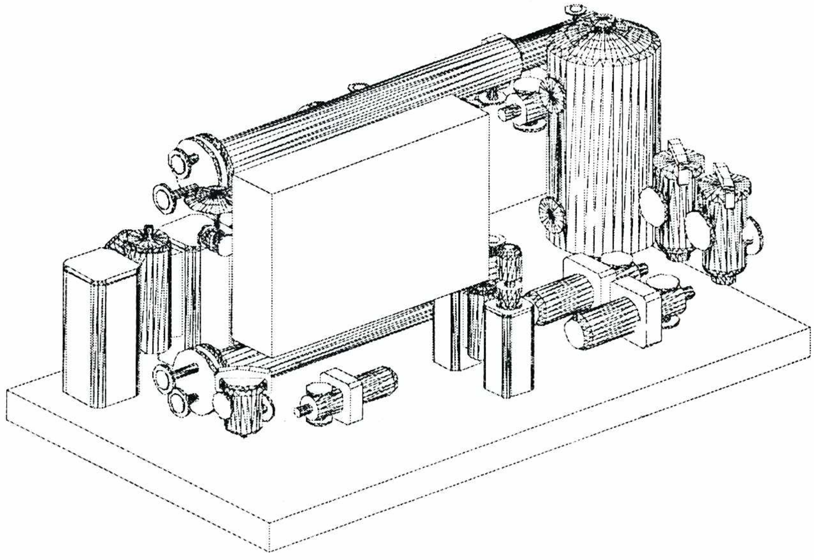


Fig.16. Arrangement of the appliances of the fuel heating block (a part of ship's machinery room)

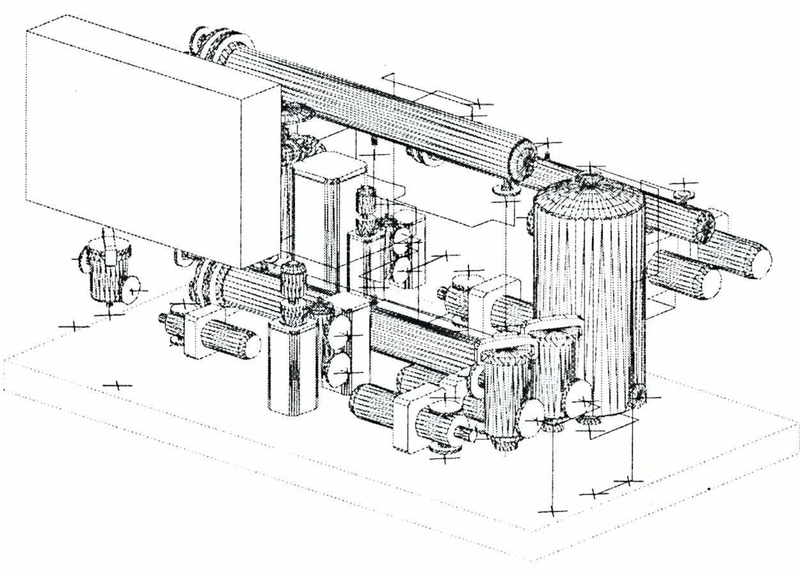


Fig. 17. Partially routed pipes of the fuel heating block

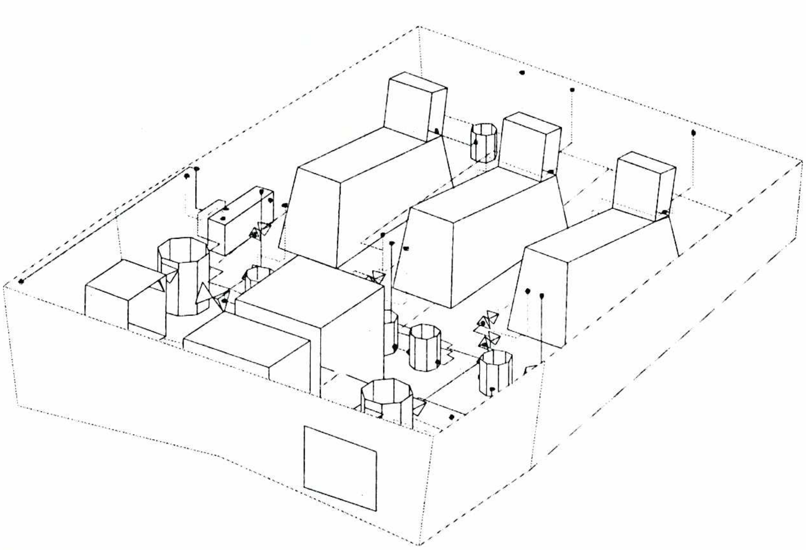


Fig.18. Routing pipes of some installations within double bottom of a small naval boat

Placement of the appliances in 3D space was arranged manually before starting the algorithm. All appliances had to be represented as the solids of natural size. The more detailed are models of the appliances the more accurate routing is obtained. However the more detailed models of the appliances cause the algorithm to work longer and require more memory. Some compromise must be make during modelling to assure good accuracy and obtain results in a reasonable time. Fig.16 shows how the appliances have been placed in 3D space.

The pipe routing example presented in Fig.17 took 4 h 48 min by using the PC of 166 MHz processor and 300 MB virtual memory. At the assumed 100 mm accuracy the algorithm managed to route 47% of all pipes. The algorithm failed to route the remaining connections because of space limitation and not sufficient accuracy.

Fig.18 shows an example of the automatically routed pipes within double bottom of a small naval boat. Not all installations were chosen to be routed.

In this example all pipes of the chosen installations were routed because the space was less packed with appliances. However 200 mm accuracy was not satisfactory.

SUMMARY

The research carried out in 1997 and 1998 showed that the combination of the wave and hierarchical algorithms shortened the time of automatic routing and enabled more complex piping to be tested. However the hybrid algorithm which is a combination of the wave and hierarchical one was not accurate enough. The main remaining problems are :

- difficulties in modelling and recognizing the obstacles
- not satisfactory accuracy of the hierarchical algorithm
- difficulties in predicting the instant of switching between the hierarchical and wave algorithms

The results of the examplary pipe routing were highly limited by the hardware and software used for carrying out the calculation. More powerful computational tools are required to process the same examples with a higher accuracy or test any larger examples.

The author will appreciate critical remarks and suggestions about practical application of the presented CAD algorithms.

Appraised by Wojciech Tarnowski, Prof.,D.Sc.

BIBLIOGRAPHY

1. Rubin F.: „The Lee Path Connection Algorithm". IEEE, Vol. c-23, No 9, September 1974
2. Burstein M., Pelavin R.: „Hierarchical Wire Routing". IEEE Transactions on Computer-Aided Design, Vol. CAD-2, No 4, 1983
3. Hachtel G.D., Morrison Ch. R.: „Linear Complexity Algorithms for Hierarchical Routing". IEEE Transactions on Computer-Aided Design, Vol. 8, No 1, 1988