Journal of Computational Design and Engineering, 2021, 8(4), 1098–1114

https://doi.org/10.1093/jcde/qwab034
Journal homepage: www.jcde.org

RESEARCH ARTICLE

Automatic design system for generating routing layout of tubes, hoses, and cable harnesses in a commercial truck

Saekyeol Kim[1], Shinyu Kim[2], Taeheok Choi[2], Taejoon Kwon[2], Tae Hee Lee[1,2,*] and Kwangrae Lee[3]

[1]BK21 Four Educational and Research Program for Automotive-Software Convergence, Hanyang University, Seoul 04763, South Korea; [2]Department of Automotive Engineering, Hanyang University, Seoul 04763, South Korea and [3]Commercial Vehicle Engineering Data Management Team, Hyundai Motor Company, 150 Hyundaiyeonguso-ro, Namyang-eup, Hwaseong-si, Gyeonggi-do 18280, South Korea

*Corresponding author. E-mail: thlee@hanyang.ac.kr http://orcid.org/0000-0002-3876-1134

Abstract

Although many routing algorithms have been developed, it is difficult for designers in the automotive industry to adopt them because of the complicated preliminary steps that are required. This study presents a systematic framework for generating the routing layout of the tubes, hoses, and cable harnesses in a commercial truck. The routing layout design problem in a commercial truck is analysed and defined. For routing operations, a sequential graph-based routing algorithm is employed to rapidly provide a routing solution. Because a reference routing layout design does not exist in most engineering problems, a cell-based genetic algorithm combined with a modified maze algorithm is employed to generate a reference design. To consider the clamping condition of the routing components, a new fitness function in the genetic algorithm is implemented. The numerical study shows that the proposed routing algorithm provides a better reference routing layout design than the conventional algorithm. The proposed automatic design system was applied to the routing layout design problem of a commercial truck. It was demonstrated that the proposed framework satisfies all industrial practitioners' functional requirements and provides a systematic method of solving the routing layout design problem, considering all its characteristics.

Keywords: automatic design system; commercial truck; Dijkstra's algorithm; genetic algorithm; routing layout design; routing algorithm

1. Introduction

Achieving design automation is a priority in many industries where companies are required to produce high-quality products that satisfy the customers' requirements while complying with various governmental regulations. Despite these restrictions, companies are expected to lower the cost of their products. In addition, the current pandemic of the coronavirus disease (COVID-19) has increased the demand for an automatic virtual design and assembly system. As the design and production of engineering products are based on the global supply chain, many international manufacturers have suffered massive delays in design, assembly, and production during the shutdown

Received: 31 December 2020; **Revised:** 9 April 2021; **Accepted:** 17 May 2021

and stringent social distancing around the world. For instance, many global automotive manufacturers have suffered a massive shortage of electrical cable harnesses and their prototypes, which are essentially required in the layout design process, during the early stage of the COVID-19 pandemic. Design and assembly schedules were also unexpectedly and continuously delayed because the layout design of routing components is usually achieved through frequent interactions and off-line meetings between many design teams, who are in charge of different vehicular components. The automatic design system for generating the routing layout of tubes, hoses, and cable harnesses is one of the core technologies for resolving current challenges in the automobile industry.

The layout design automation of long and round products, such as pipe design in ships or cable harnesses in vehicles, has always been a research interest. Routing layout design is a process that finds a satisfactory route for these components, connected from their starting points to their end points under various design constraints. For instance, pipe routing layout design is a critical design step during the detailed design phase in ship design, because more than 50% of the total person hours are spent at this step (Park & Storch, 2002). An enormous amount of information and constraints must be managed to obtain the layout design of routing components. Material and production cost, assembly and installation cost, accessibility and maintenance, and space availability are vital factors that companies have to consider at this step. Although many skilled designers and engineers manage this routing operation, two critical and fundamental problems must be overcome. First, the routing layout design is highly dependent on the knowledge and experience of the designers and engineers. Based on their know-how and preferences, the design and its validation can vary and yield inconsistent results. Second, conventional design methodologies are manually carried out by practitioners using computer-aided design (CAD) software. This requires a time-consuming process of trial-and-error, massive collaboration between a large number of different teams, and intensive teamwork. Many routing algorithms have been proposed for finding a routing solution without involving manual operations. As most of these algorithms work for some simple problems, they are challenging to implement in the design process in the industry. For these reasons, some studies have developed an automatic pipe design system to solve these fundamental problems, and some useful results for the ship design industry were obtained (Asmara, 2013; Kim *et al.*, 2013).

The electrification of vehicles and the implementation of advanced electronic vehicle control systems have caused a rapid increase in the number of electrical cables and wires used in automobiles (Drotz & Huber, 2014). Thus, a free design space is no longer available for routing components. The conventional engines and battery-supplied electric powertrain components need to be accommodated in a densely packed vehicular system, as today's automobile companies are focusing more on electrified and hybrid solutions. Hence, the geometric interference with other automotive parts should be carefully evaluated for each routing component (Hermansson *et al.*, 2013, 2016).

Therefore, the routing layout design of various tubes, hoses, and cable harnesses has become an essential task in the automotive industry. Figure 1 shows some of the targeted components in a commercial truck. There are some challenges to the automobile industry. First, automobile companies lack skilled designers and experts because, to date, the routing layout design of these tubes, hoses, and cable harnesses has not attracted much attention in the industry. Manually creating the routing design of these components has become extremely inefficient, resulting in poor quality because vehicle development time has been considerably shortened. Because the routing layout is very sensitive to the design of other automobile components, frequent design change in these components causes a bottleneck in the design and manufacturing process. Moreover, routing layout design has been assigned the lowest priority because this procedure has been underestimated in the automobile industry. This has caused considerable wastage of person hours and has deteriorated the efficiency of the design and manufacturing procedure. Finally, routing is incredibly challenging because a reference routing layout design does not exist in most industrial problems.

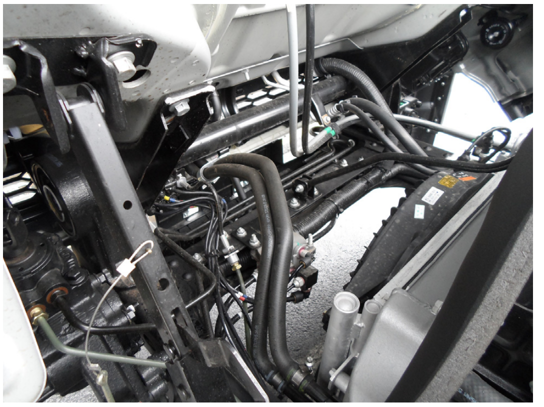


Figure 1: Tubes, hoses, and cable harnesses under the cab of a commercial truck.

Despite the importance of the routing layout design problem, only a few researchers have addressed it in the automobile industry. The assembly procedure of a wiring harness was optimized (Kobayashi *et al.*, 2014). Researchers from Volvo and Chalmers University of Technology found a new cab-chassis electrical wiring harness routing layout (Drotz & Huber, 2014). Dijkstra's and A∗ algorithms have been adopted to solve the route optimization problem of a flexible 1D component with manufacturing constraints (Hermansson *et al.*, 2016). The routing layout of the air conditioning (AC) hose found using this method was satisfactory. The A∗ algorithm significantly decreased the computational time, but this method required a user-defined function. In a previous work, our research team developed a graph-based routing algorithm (Kim *et al.*, 2021). The limitation of earlier studies is that they focused on predicting vehicular performance or proposing a new routing structure or algorithm. However, conventional methodologies cannot be easily generalized or adopted in industrial applications without knowledge or experience in routing algorithms.

This study presents an automatic design system for generating the routing layout of tubes, hoses, and cable harnesses of a commercial truck to address the issues mentioned above in the automotive industry. A commercial truck was selected as the target vehicle in this study because it has a relatively large number of tubes, hoses, and cable harnesses compared to other vehicular types. The number of these routing components is expected to significantly increase as vehicles adopt more electronics in the near future. In addition, a validation procedure for routing layout design, which is obtained from the routing operation, should be established because reference routing layout designs do not exist in the early design stage or during the vehicle development process. This study had three main objectives. The first objective was to develop an automatic design system for generating the routing layout of tubes, hoses, and cable harnesses of a commercial truck. This system would facilitate the route design process of these components, lower the dependence on the experts' know-how and the interactions between design teams, and reduce the time required for repetitive design changes during manual operations of the design, assembly, and manufacturing processes. The second objective was to develop a routing algorithm that provides a reference layout design for validating the routing results during the vehicle development process. The third objective was to apply the proposed automatic design system and routing algorithm to the engineering example of a commercial truck. The remainder of this paper is organized as follows: In the next section, a literature review of the routing algorithm and the automatic routing design system for various industrial applications is presented. In Section 3, the routing design problem for a commercial truck is described. Section 4 presents the proposed automatic design system, including the adopted routing algorithms. In Section 5, the proposed automatic design system is applied to six routing components in a commercial truck. The routing algorithm, which was developed for design validation, was also compared with the conventional routing algorithm. Finally, conclusions are presented in Section 6.

2. Literature Review

This section reviews various routing algorithms and some practical approaches for applying these techniques to an automatic system and designing the routes of components. Several automatic design systems have been developed to generate pipe routes in ships, pipe networks, and aircraft, and we believe that these can be extended to the routing problem of tubes, hoses, and electrical cable harnesses in automobile design.

An early study of the routing problem was primarily performed to solve the shortest collision-free path problem. Dijkstra's, Maze, network-based, Escape, and A∗ algorithms are some of the early research outcomes that provide the basic ideas and fundamental concepts of routing algorithms. Many routing algorithms have been developed over the past two decades based on these research studies. The aforementioned classical algorithms were reviewed in some doctoral dissertations (Park, 2002; Asmara, 2013).

As the routing algorithm is the core technology in the automatic design system for routing layouts, routing algorithms are briefly reviewed. The routing algorithms currently used for layout design consist of two basic steps: (1) definition of the design domain and (2) identification of the route solution in the defined design domain that satisfies the design constraints. The existing routing algorithms, which have been developed thus far, use numerous methods to define the design domain and identify the routing layout with the shortest length. In the first step, the design domain is defined by cells or grids using meshes, or graph or network based on vertices and edges. The second step determines the routing layout based on the design domain, and is accomplished using many different algorithms. The classical approaches, e.g. Dijkstra's algorithm, Maze algorithm (MA), or simple rule-based algorithm (Park, 2002), still comprise a large proportion of the algorithms used to accomplish this task, whereas heuristic algorithms such as the genetic algorithm (GA; Niu *et al.*, 2016; Sui & Niu, 2016), particle swarm optimization (Dong & Lin, 2017), and ant colony optimization (Qu *et al.*, 2016) algorithms have also been extensively adopted. In addition, a combination of these conventional methods has been developed to overcome their limitations (Liu & Wang, 2012). In our previous work, routing algorithms were reviewed in detail (Kim *et al.*, 2021).

The main problem with these routing algorithms is that they cannot be widely used in industry. The designers and practitioners not only need to understand the fundamental background theories of each routing algorithm but should also be able to carry out the entire pre- and post-processing. To enhance the applicability of routing algorithms, several researchers have developed routing design systems to improve the work efficiency of practitioners and designers. The development of a design system is one of the most demanding and challenging tasks. In the early stages of research, many researchers constructed a decision-making framework to support pipe designers. A knowledge-based prototype expert system for ship pipe design was developed (Kang *et al.*, 1999). A design framework that selects a satisfactory routing path for every pipeline in a ship engine room was proposed (Park, 2002); this methodology considered design constraints like material and installation costs. An automatic pipe model generation method based on the hull structural model was also proposed (Roh *et al.*, 2007), which automatically updated the pipe route design as the hull structural design was changed. An automatic pipe routing system was proposed for ship design (Kim *et al.*, 2013), which utilized a graph-based design domain and Dijkstra's algorithm to find the shortest pipe route. In addition, this system exported the routing results of the algorithm to the CAD system. A pipe routing methodology for ship design was developed and implemented into a prototype software package (Asmara, 2013). However, the knowledge of experienced ship pipe designers was reflected in the development of this methodology. Therefore, it is significantly challenging to employ it in the routing problems of other industries. Similar studies have been performed in other industries. An automatic intelligent system was developed for designing the routes of electrical wires and pipes in aircraft (Van der Velden *et al.*, 2007). Knowledge-based engineering methods and a grid-based A∗ algorithm were used to obtain the route design. Meanwhile, FE and CAD software were integrated to show the routing results and facilitate the designers' work. A routing software was developed that optimized natural gas pipe networks using GA (El-Mahdy *et al.*, 2010). This software yielded promising solutions that minimized network costs. Table 1 summarizes previously conducted studies related to the development of these systems. These systems were applied to solve some engineering design problems in shipbuilding and aircraft industries, e.g. the pipe route design in the ship engine room and the route design of electrical wiring harnesses and pipes in aircraft.

Although many routing algorithms have been proposed, a unified methodology that can be employed for routing problems across industries has not been established because the design objectives and the geometric and economic constraints are highly dependent on the characteristics of the routing layout problem and industry type. Moreover, routing algorithms are impractical because designers and engineers have no prior knowledge of routing algorithms. Some studies have been conducted to construct expert systems or automatic piping systems. Despite promising results in some industrial fields, a wide range of applications of these software is yet to be realized.

Our research team has developed a routing algorithm to solve the routing layout design problem in the automobile industry (Kim *et al.*, 2021). Although it improved the efficiency of the routing operation, validating the results during the vehicle

Table 1: Summary of routing layout design systems in related studies.

Type	Function	Application
Design expert system	Support pipe routing design in a CAD system (Kang *et al.*, 1999)	Shipbuilding industry
Design expert system	Support decision making, visualization by a CAD system (Park, 2002)	
Model generation system	Generation of a parametric pipe model in a CAD system (Roh *et al.*, 2007)	
Automatic design system	Implementation of the automatic pipe route design module into a CAD system (Kim *et al.*, 2013)	
Automatic design system	Prototype software package integrated with commercial software (Asmara, 2013)	
Automatic layout routing system	Integration of the layout route system with CAD system and FE software (Van der Velden *et al.*, 2007)	Aircraft industry
Network optimization system	Natural gas pipe network optimization (El-Mahdy *et al.*, 2010)	Pipe network

Figure 2: Targeted commercial truck.

development process remains challenging because a reference layout design does not usually exist. In addition, a vehicle product production cycle is too short for practitioners to learn the routing algorithm and integrate it with the various types of software used for routing layout design. Hence, the demand for an automatic routing design system in the automobile industry and a routing algorithm that generates a reference layout design is rapidly increasing.

3. Routing Layout Design Problem in a Commercial Truck

The routing layout design problem is commonly encountered in the automobile industry. Unfortunately, previous studies on routing algorithms or automatic design systems for automotive vehicles are extremely limited. To develop an automatic design system for generating the layout design of tubes, hoses, and cable harnesses of a commercial truck, some important characteristics of the routing layout design problem and the functional requirements for the routing design system should be investigated. The commercial truck used in this study is illustrated in Fig. 2.

3.1. Characteristics of the routing layout design problem

The characteristics of the routing layout design problem in a commercial truck should be carefully investigated by experts in the automobile industry. Some general characteristics of the routing layout design problem obtained after intensive discussions with the designers and engineers in the industrial field are summarized as follows:

1. There are many tubes, hoses, and cable harnesses that need to be routed in a limited design space.
2. There are many layout design candidates for each routing component because commercial trucks have a longer vehicular structure than other vehicle types.
3. The priority of the routing components in the layout design should be determined by the company's internal guidelines.
4. The profusion of automobile components should be considered as an obstacle to the routing process.
5. Branches and multiple terminals should be considered in the routing process.
6. The routing layout design must be obtained rapidly to deal with the frequent design changes of other automobile parts and collaborations with other design teams.
7. A reference routing layout design is required for validation during the vehicle development process.

The CAD model and routing components of the targeted commercial truck are shown in Fig. 3. The name of the commercial truck is undisclosed in this study because it is proprietary information belonging to the sponsoring company. Although some vehicular components are not shown in the model, it can be appreciated that all the listed characteristics should be considered in the routing algorithm.

3.2. Functional requirements for the automatic design system

The functional requirements of an automatic design system should be discussed with design practitioners. After intensive discussions with the designers and engineers in the industrial field, the following issues were selected as top priorities for incorporation into the automatic design system.

1. All layers of the system must be programmed using a single software (MATLAB) for the convenience of the practitioners.

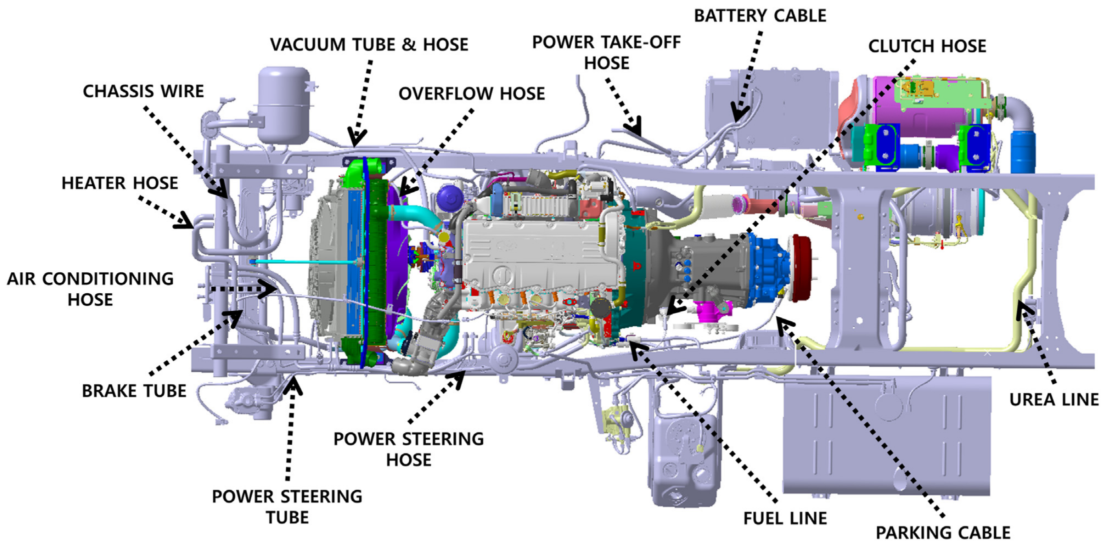


Figure 3: CAD model and routing components of the targeted vehicle (Kim *et al.*, 2021).

2. A model simplification module is required to reduce the complexity of the entire CAD model of the vehicle.
3. Practitioners should be able to obtain routing layout designs before the final CAD model of the vehicle is determined.
4. The starting point(s), end point(s), and intermediate points of the layout design and the priority order of the routes should be separately assigned by the practitioners (Excel).
5. A rapid routing design support module that assists the routing operation and a design validation module that provides a reference routing layout design must be included.
6. Output results of the routing algorithm should be exported to the CAD software.

4. Proposed Automatic Routing Design System

This section presents the proposed automatic design system and the implemented routing algorithms. The proposed framework for the automatic routing design system is shown in Fig. 4.

4.1. Definition of routing layout design problem

At the input layer, a simplified CAD model is created, and the priority order of the tubes, hoses, and cable harnesses is determined. The CAD model of an automotive vehicle cannot be directly used as an obstacle in the system layer because it is usually too complex. Additionally, it is not practical to use the entire model because the computational time drastically increases as the complexity of the model increases.

Figure 5 shows an example of model simplification (Black, 2017). Among the several existing CAD software, CATIA V5 R20 is selected because it provides valuable tools and is extensively used in the automobile industry. To simplify the CAD model, this study utilized the wrapping tool in the CATIA software. This tool wraps the original model using a virtual surface and generates a new CAD model based on the wrapped surface. The precision of the simplification was determined by the user's defined grain size. The routing design procedure becomes accurate but more time consuming as the grain size is reduced. Thus, the level of complexity and the number of obstacles are determined by considering the time limit. The priority order of the targeted components, e.g. tubes, hoses, and cable harnesses, is usually determined by the designers or the company's internal guidelines.

Subsequently, at the data layer, model information is extracted from the simplified CAD model, and the terminals and intermediate points for each routing component are determined. The intermediate points, which are determined by the designers, help to determine the routing path in the design support module. The simplified CAD model is exported in the STL format from CATIA and imported into MATLAB. The model information includes all its vertices and faces. Figure 6 shows an example of the simplified CAD models of the targeted commercial truck imported into MATLAB. In this example, the cooling module, engine, and transmission were simplified using a grain size of 5 mm. The location data of the terminals and the intermediate points are listed in Table 2. Terminals and intermediate points are abbreviated as T and I, respectively. The left-hand and right-hand sides are abbreviated as LS and RS, respectively. Finally, the design domain of the problem is defined based on the model information and the locations of the terminals and intermediate points. The routing problem can be redefined through the editor layer based on the results of the routing algorithms to consider more obstacles or routing components.

4.2. Routing algorithm for supporting routing operation

The purpose of the design support module is to rapidly provide practitioners with a routing solution. Thus, a sequential graph-based routing algorithm developed by our research team was adopted because it provides satisfactory results within a short time (Kim *et al.*, 2021). The results of this algorithm allow automobile designers and engineers in the industry to deal with frequent design changes in other automobile components, as well as numerous layout change requests from other teams.

Routing algorithms can be classified as graph-based and cell-based routing algorithms, depending on their design domains. The cell-based routing algorithm can provide a routing solution close to optimum when the cell size is very small. However, this requires a large amount of computational time, which is not suitable for supporting the routing operation of designers. On the other hand, the graph-based routing algorithm can easily

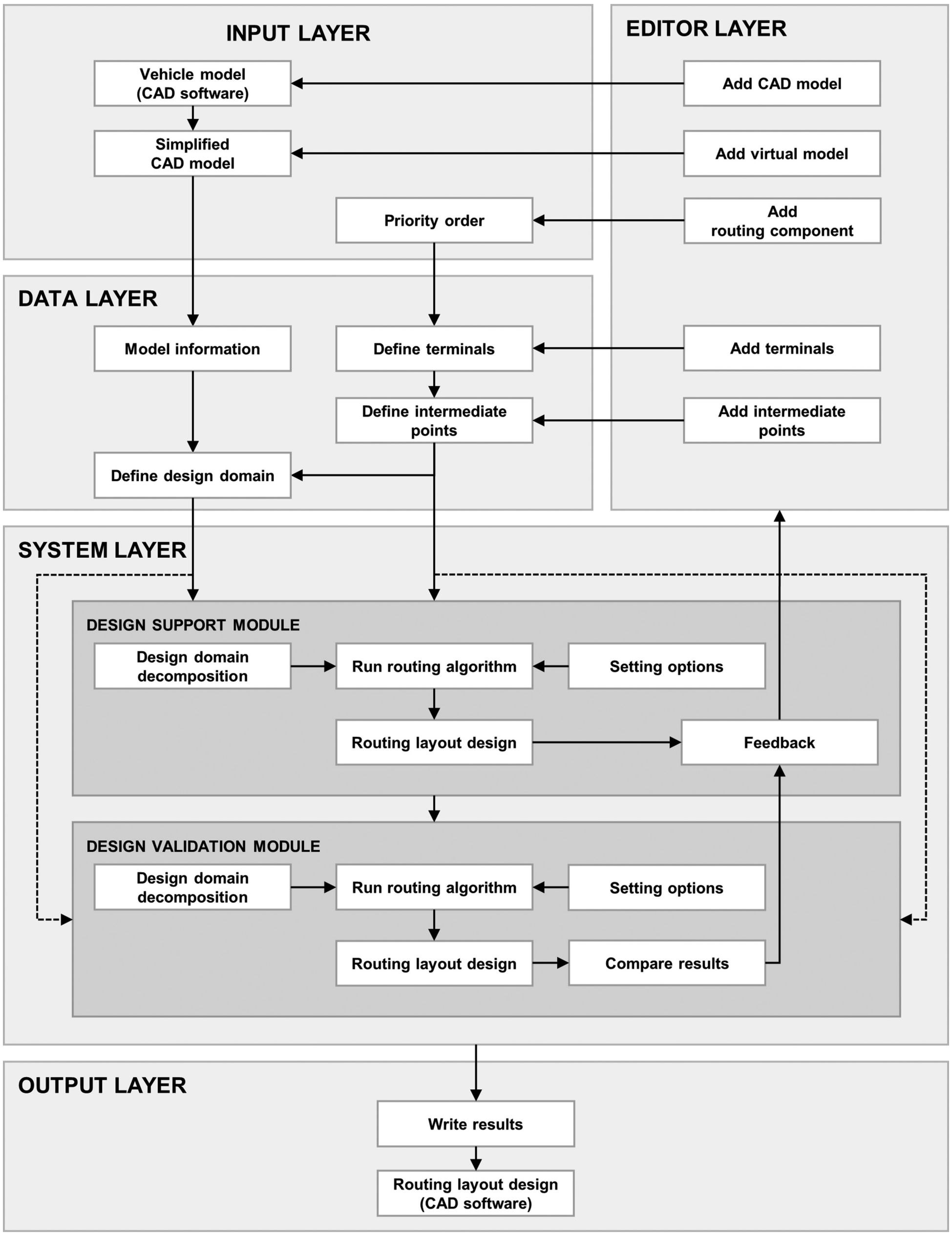


Figure 4: Framework of the proposed automatic routing design system.

control the computational time because vertices and edges are usually generated by user-defined rules. The sequential graph-based routing algorithm implements the following strategies to deal with the characteristics of routing problems in the automotive industry: (1) division of design domain, (2) design of experiment technique, (3) intermediate point, and (4) minimum spanning tree.

The graph plays the most crucial role in a graph-based routing algorithm because it determines the candidate paths for the routing layout design. Because the entire design domain is too large and obstacles in automotive applications have complicated shapes, a graph with a very large number of vertices is required to solve this routing problem. This drastically increases the computational time. Hence, the design domain is divided into several design subdomains to solve several routing problems instead of the original one. The graph for each design subdomain is much less complicated than the graph covering the entire design domain. This makes it possible to significantly reduce computational time.

Most graph-based routing algorithms propose their own rules to determine the vertices of a graph. This may be efficient, but it is highly dependent on routing problems and is not generally applicable. This routing algorithm implements the design of experiment (DoE) technique to provide a general approach for determining the vertices of a graph. Among various DoE techniques, a full factorial design was adopted to improve the uniformity of vertices in the design subdomain.

The terminals in automotive applications are usually located far apart. Intermediate points are adopted to facilitate the routing procedure in the algorithm. There are three main functions

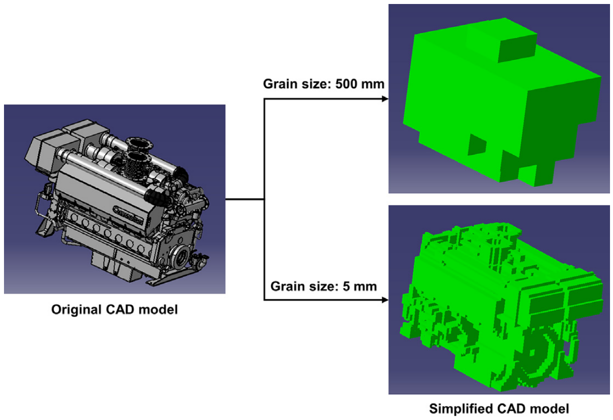


Figure 5: Example of CAD model simplification.

of the intermediate points. First, the design domain is divided into design subdomains based on these points. Second, the intermediate points, which are usually determined by the design experts, are assigned to facilitate the path searching process in particular areas in the design domain. Finally, they help solve the routing problem with multiple terminals and generate branches.

When two vertices are connected, Dijkstra's algorithm is applied to find the shortest path. Dijkstra's algorithm is not applicable when more than two vertices are to be connected. A minimum spanning tree, which is a subset of the edges of a graph that connect the specified vertices whose sum of distances is minimum, is employed. Prim's algorithm, which is widely adopted in the minimum spanning tree, is used to find the routing solution.

A flowchart of the routing algorithm is presented in Fig. 7. This routing algorithm divides the design domain into several design subdomains based on the terminals and intermediate points. Vertices are created using a full factorial design, and those that are located inside the obstacles are removed. Next, the edges are created using the remaining vertices. The terminals and intermediate points at each design subdomain are regarded as the remaining vertices during this procedure. The edges that pass through the obstacles are removed. Then, an undirected graph in the design subdomain is generated using the remaining vertices and edges. The routing solution is determined by Dijkstra's algorithm or Prim's algorithm using a minimum spanning tree. The steps are repeated for each design subdomain until the routing layout design is obtained. This procedure is operated sequentially according to the priority order of the tubes, hoses, and cable harnesses of the commercial truck.

The routing process is described using a simple routing layout design example to elucidate the routing algorithm. The routing problem, which is illustrated in Fig. 8, has three terminals and two intermediate points. The figures below represent the floor plan and the side view. The routing algorithm divides the design domain and determines the routing layout design according to the following rules. The first design subdomain is defined by using the first terminal and the first intermediate point. In this design subdomain, the vertices are created using a five-level full factorial design, where they are made equidistant by dividing the differences between the *x*-, *y*-, and *z*-coordinates of these two points, respectively. Vertices located inside the obstacle are excluded. A graph in the design subdomain is then created, and the distance between all remaining vertices is calculated using Dijkstra's algorithm, whereas the paths that pass through the obstacle are excluded. If there are only two points that should be connected, Dijkstra's algorithm provides the shortest route. If there are three or more points that should be connected, the shortest path is then obtained by Prim's algorithm using the minimum spanning tree. The following design subdomain is defined using the first and second intermediate points, and the same routing procedure is repeated until the last intermediate point. The final design subdomain is defined between the last intermediate point and the last terminal, and the same routing process is performed. The algorithm for supporting the routing operation is explained in detail in our previous work (Kim *et al.*, 2021).

4.3. Routing algorithm for generating reference layout design

The graph-based routing algorithm is used for supporting the routing operation of designers. In general, however, a reference routing layout design does not exist in engineering problems during the vehicle development process. Hence, it is not possible to validate the routing solution obtained from this module. A cell-based routing algorithm is usually time consuming and is inefficient for rapidly obtaining routing results. However, because it finds an optimal solution in the given design domain, it can provide an excellent reference routing layout design in engineering applications.

Among the various types of cell-based routing algorithms, a routing algorithm that combines the modified MA and GA

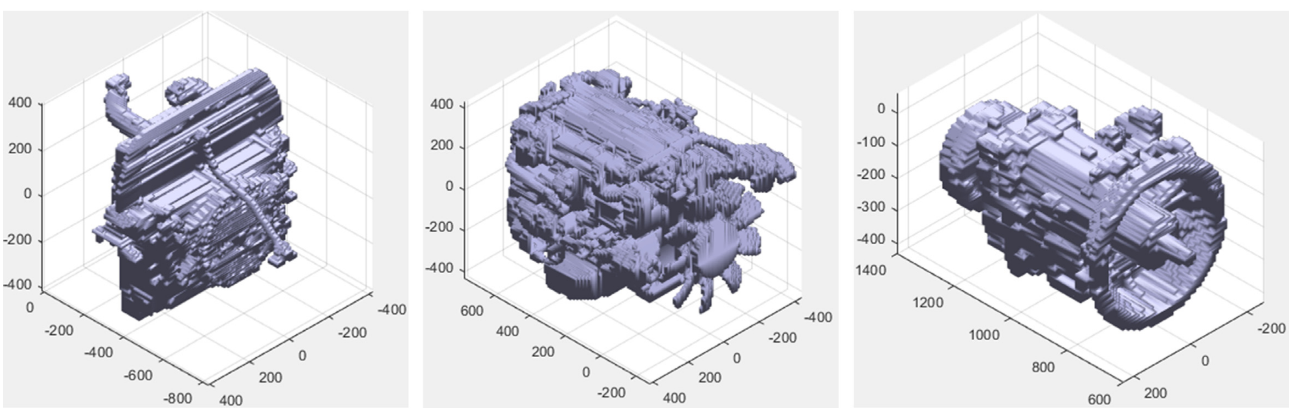


Figure 6: Example of simplified CAD models of commercial truck components imported into MATLAB.

Table 2: Location data of terminals and intermediate points (Kim *et al.*, 2021).

No.	Routing component	Type	Location data
1	Brake tube (LS)	T1	(−740, −290, −60)
		I1	(−740, −450, 10)
		T2	(−280, −450, 10)
2	Brake tube (RS)	T1	(−670, 280, −60)
		I1	(−500, 700, −100)
		I2	(−100, 600, −100)
		T2	(−240, 330, −100)
3	Heater hose (LS)	T1	(−905, 7.5, 160)
		I1	(−905, −330, −115)
		T2	(−555, −330, −115)
4	Heater hose (RS)	T1	(−877, 34,160)
		I1	(−877, 34, 450)
		I2	(−155, 0, 450)
		T2	(−155, 0, 260)
5	AC hose	T1	(−935, −52, 40)
		I1	(−935, −410, −40)
		T2	(−490, −410, −210)
		I2	(60, −410, 60)
		T3	(60, −190, 113)
6	Chassis cable harness	T1	(−884, 196, 160)
		I1	(−884, −203, 160)
		T2	(312, −203, −40)
		I2	(1330, −203, −202)
		T3	(1330, 345, −202)
		T4	(3900, −350, −135)

considering branch routing is adopted. Because the basic MA and GA are described in previous studies, they are only briefly explained in this section. The MA, which is also called Lee's algorithm, finds a possible routing solution based on a breadth-first search. It always provides an optimal solution if one exists. The green and yellow cells in Fig. 9a represent the terminals and obstacles, respectively, whereas the blue cells represent the design domain. This algorithm consists of two steps: wave propagation and backtracking. The wave propagation consecutively labels the neighboring cells from the starting point to the goal point. When the wave propagation is completed, each cell is labelled with a value, as shown in Fig. 9b. The backtracking process starts from the goal point and moves to a neighboring cell with a lower value. As this process randomly chooses the direction of backtracking, the MA provides multiple solutions. Random selection in the backtracking process increases the diversity of solutions, but it also increases the number of bends in the routing solution, as illustrated in Fig. 9b. In addition, the search area is limited to within the rectangular (2D problem) or cuboidal shape (3D problem) of the design domain determined by the locations of the two terminals. Thus, the routes cannot deviate away from the obstacles when they block all paths from the starting point to the goal point in the search area.

A modified MA has been developed to overcome the disadvantages of MA (Niu *et al.*, 2016; Sui & Niu, 2016). In the modified MA, two strategies are adopted: the auxiliary point and the priority direction. The routing solution from the modified MA passes through the auxiliary point. This point plays an important role when a specific area needs to be searched, and it enables the identification of a routing path outside the search area of the MA. Figure 10b illustrates how an auxiliary point works in the modified MA. The priority direction modifies the backtracking process of the MA. As the bending deteriorates the quality of the routing solution, the path direction does not change during the backtracking process until the backtracking path encounters an obstacle or the next neighboring cell is labelled with a higher value than the current cell. Figure 10c illustrates how the priority direction works in the modified MA. These two strategies can be compared with the basic MA, as shown in Fig. 10a.

Although the modified MA finds a better routing solution than the MA does, it does not always provide the best routing path because randomness is still inherent in the backtracking process. As the design domain increases, the probability of finding a good routing solution decreases drastically. The GA, which is one of the most popular heuristic optimization algorithms, has been adopted to find a better routing solution. Four core elements of the GA should be defined to solve routing problems: (1) route chromosome, (2) initial population, (3) fitness function, and (4) genetic operators. A route chromosome, which represents a routing path, consists of a series of neighboring cells from the starting point to the goal point. The information of each cell includes the x-, y-, and z-coordinates of its center point. The initial population of the GA is generated by the MA to maintain a high level of diversity in the routing solutions. The fitness function, which is a criterion for evolving and updating the initial population, is the objective function for design optimization. The total length of the route or the number of bends is usually selected as a fitness function. There are two typical genetic operators: mutation and crossover. A two-point mutation and crossover are used in the GA for routing problems. In the two-point mutation procedure, a route chromosome, called the "parent chromosome," is randomly selected, and two nodes on this chromosome are randomly chosen. A midpath between these two cells is generated using the modified MA and inserted into the chromosome. The evolved chromosome is called the "child chromosome." Figure 11 shows an example of how the two-point mutation works in the GA. In a two-point crossover, two route chromosomes are randomly selected from the popula-

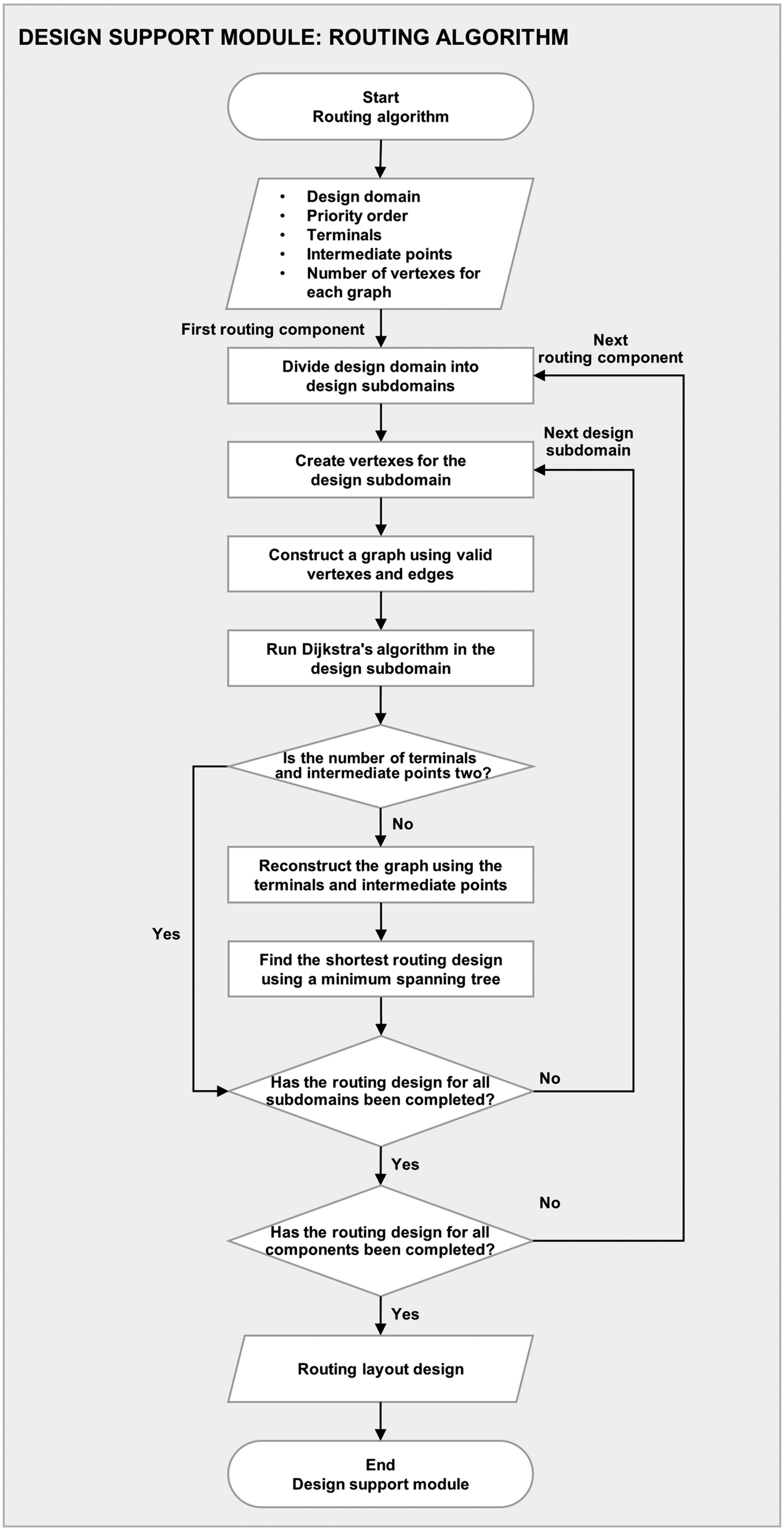


Figure 7: Flowchart of the routing algorithm incorporated in the design support module.

tion, and a crossover node is randomly chosen from each chromosome. A midpath between these two cells is generated using the modified MA and inserted into each chromosome. Figure 12 shows how the two-point crossover operates in the GA. Based on these genetic operators, the routes in the population evolve into better routing solutions by minimizing the fitness function.

The GA, which is combined with the modified MA, can deal with routing problems with two terminals. However, this algorithm cannot be adopted when there are more than two ter-

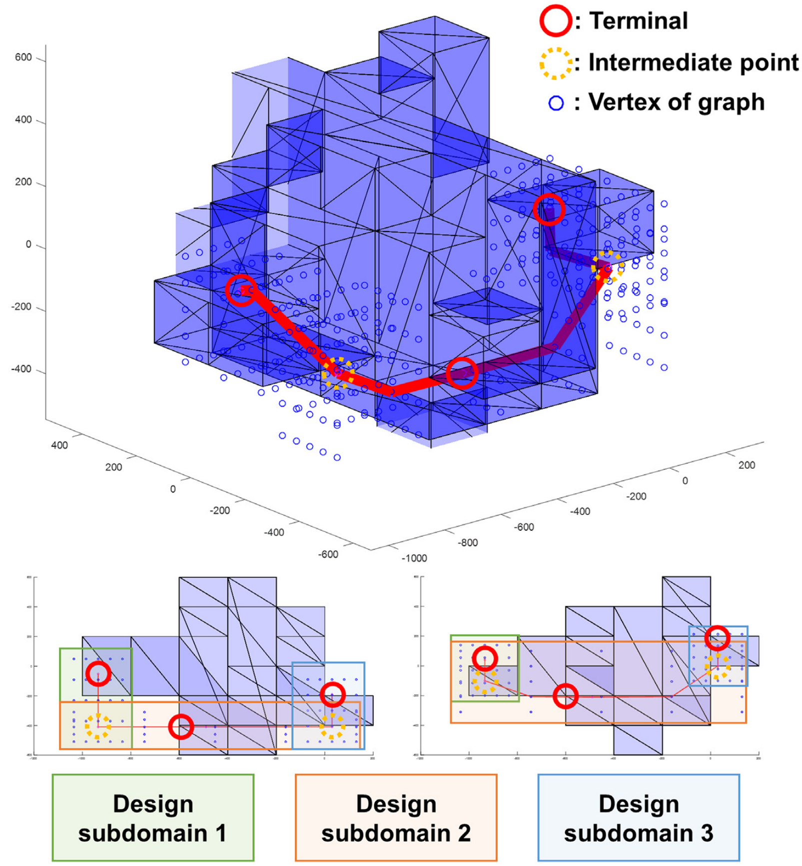


Figure 8: Example of the routing algorithm incorporated in the design support module.

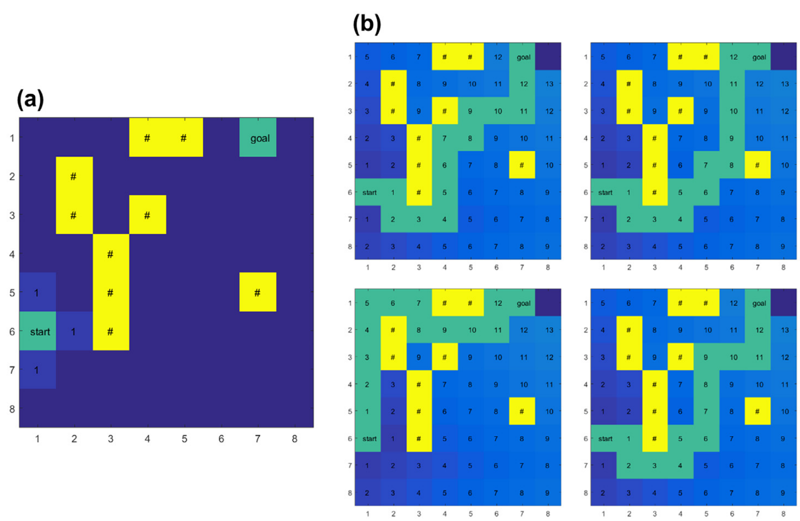


Figure 9: MA: (a) routing problem and (b) multiple solutions.

minals. The modified GA, which deals with branches, was proposed to solve this type of routing problem (Sui & Niu, 2016). In the modified GA, the route chromosome consists of branch routes that connect a terminal and a common starting point (CSP). Figure 13 illustrates m chromosomes in a population for a routing problem with n terminals. All branches are connected at the CSP. In the initial population, the point that minimizes the sum of the Euclidean distance between this point and each terminal at each branch is selected as the CSP. After determining the CSP, the initial routing path between the CSP and the terminal at each branch is determined by the modified GA. The fitness function is defined as the weighted sum of the total length of the route and the total number of bends in the route as follows:

$$f\ (\mathbf{x}) = w_1\, l\,(\mathbf{x}) + w_2 b\,(\mathbf{x})\,, \tag{1}$$

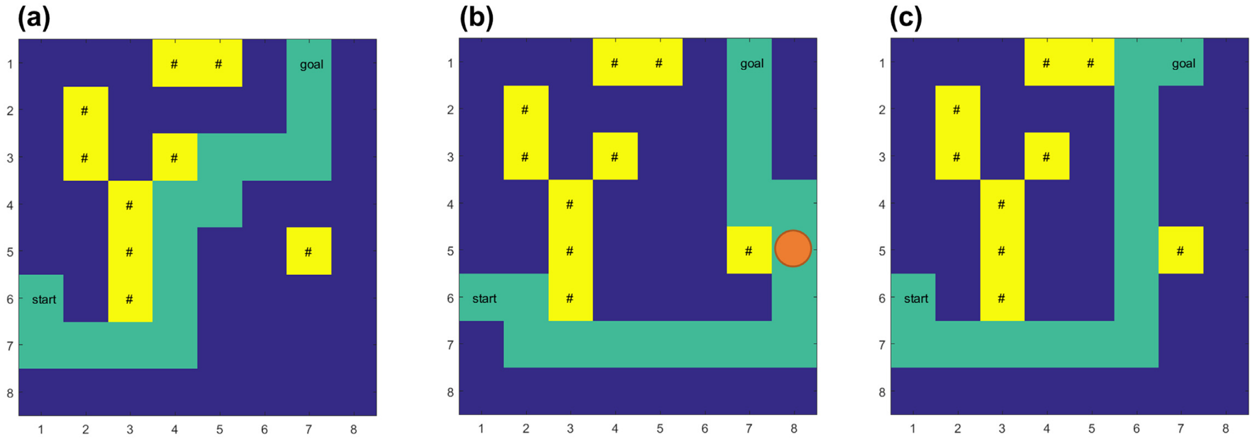


Figure 10: Modified MA: (a) without auxiliary point and priority direction, (b) with an auxiliary point, and (c) with priority direction.

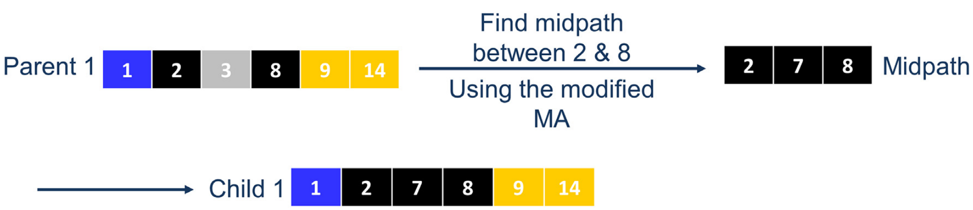


Figure 11: Two-point mutation operation in the GA.

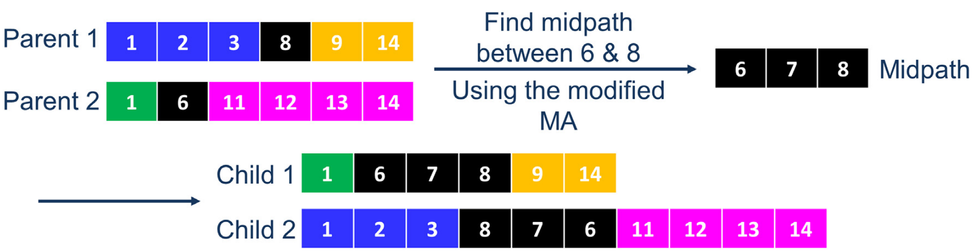


Figure 12: Two-point crossover operation in the GA.

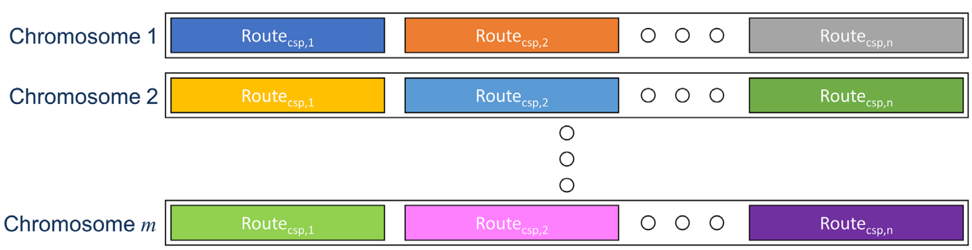


Figure 13: Chromosome of the modified GA.

where f is the fitness function, the components of vector $\mathbf{x}$ are coordinates of the cells in the route, l is the total length of the route, b is the number of bends in the route, and w_1 and w_2 are the weight or normalization factors. There are three genetic operators: two-point mutation, two-point crossover, and whole crossover. In the modified GA, the two-point mutation and crossover are identical, but they are consecutively carried out between the corresponding branches. The two-point mutation and crossover operations in the modified GA are shown in Figs 14 and 15, respectively. The whole crossover, which is shown in Fig. 16, randomly chooses two parent chromosomes and switches the route of the corresponding branches.

Although the conventional modified GA provides various reference routing layout designs for validation, they are far from practical solutions because their routes pass through the open design space. This leads to an increase in assembly and production cost in engineering problems. Maintaining routes as close as possible to obstacles is called the clamping condition. To re-

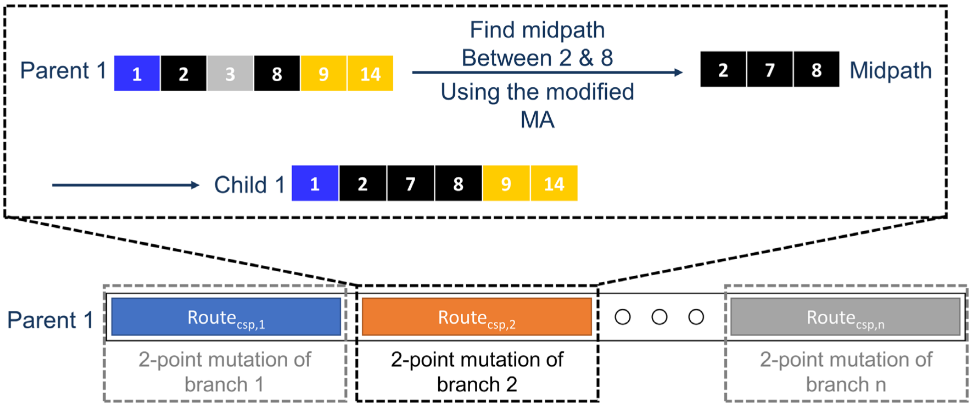


Figure 14: Two-point mutation operation in the modified GA.

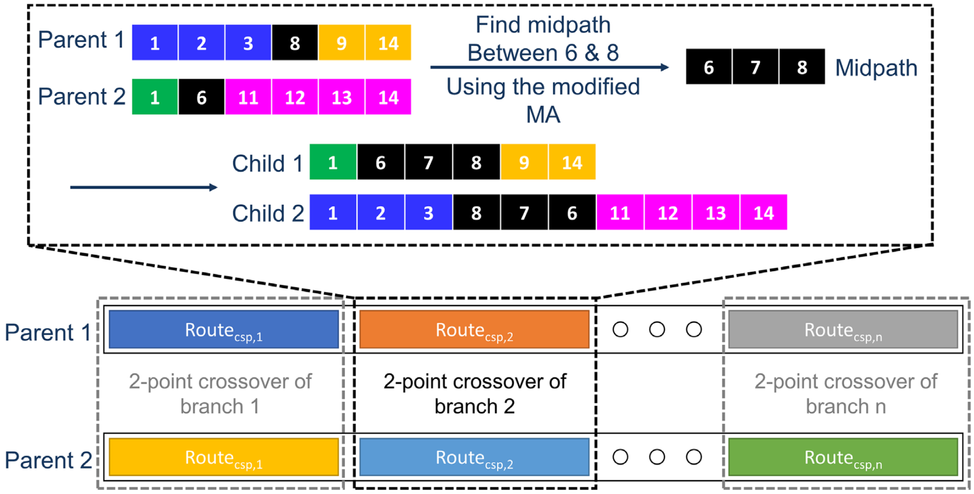


Figure 15: Two-point crossover operation in the modified GA.

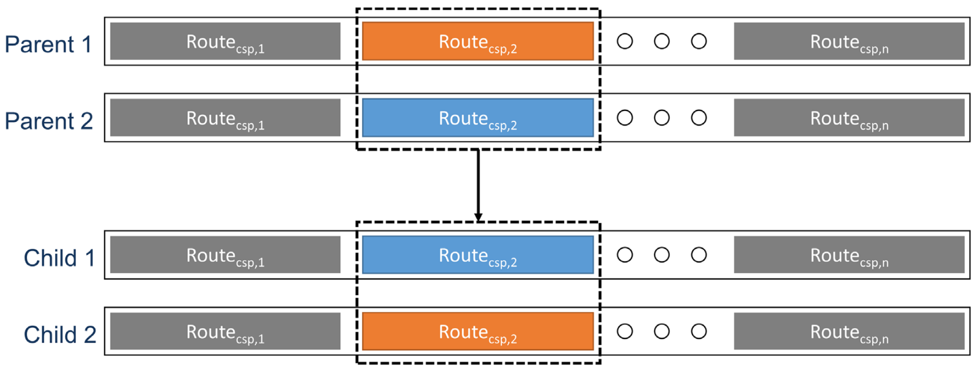


Figure 16: Whole crossover operation in the modified GA.

solve this issue, a new fitness function is proposed to consider the condition of the tubes, hoses, and cable harnesses in the algorithm. In this study, the fitness function was defined as follows:

$$f\ (\mathbf{x}) = w_1\, l\,(\mathbf{x}) + w_2 b\,(\mathbf{x}) + w_3 \sum_{i\,=\,1}^{n_c} \mathrm{d}\,(x_i)\,, \tag{2}$$

where f is the fitness function, the components of vector $\boldsymbol{x}$ are coordinates of the cells in the route, l is the total length of the route, b is the number of bends in the route, d is the shortest distance between each cell in the route and the closest obstacle, n_c is the number of cells in the route, and w_1, w_2, and w_3 are the normalization factors. The score map that considers the shortest distance between each cell in the design domain and the closest obstacle of an example with four goal points or terminals is shown in Fig. 17. As this score is minimized using the routing algorithm, the routes are located near the obstacles that satisfy the clamping condition. In this study, the weight factors are de-

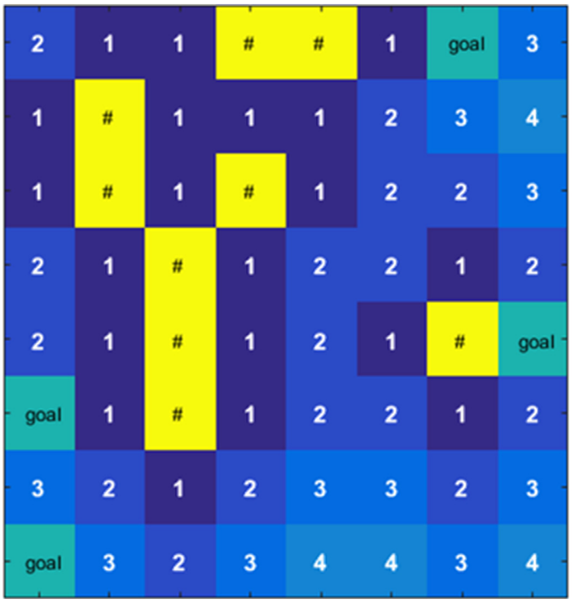


Figure 17: Example of the shortest distance between each cell in the design domain and the closest obstacle in a routing problem.

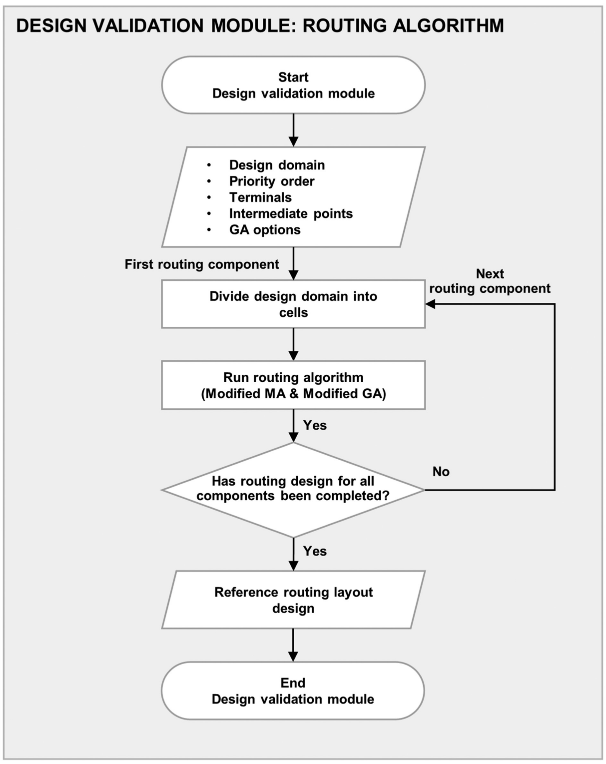


Figure 18: Flowchart of the routing algorithm incorporated in the design validation module.

fined as $w_1 = 1/l_g$, $w_2 = 1/\max(b)$, and $w_3 = 1/\max(d)$, where l_g is the grain size, and max(b) and max(d) are the maximum values that appeared during the optimization, respectively. The default values in MATLAB were used as the parameters for GA.

A flowchart of the proposed algorithm, which generates a reference routing layout design for validation, is shown in Fig. 18. The reference routing layout design from this algorithm may not be directly comparable to the routing results because it is obtained in a different design domain, and the route is always bent orthogonally. This design usually provides longer and more conservative routes than the routing results because of the clamping condition. However, the proposed algorithm provides an excellent reference routing layout design during the vehicle development process.

5. Application Results

The proposed automatic design system for generating a routing layout was applied to a commercial truck. Our collaborators in the industry provided a CAD model of the vehicle. The targeted routing components were carefully determined based on intensive discussions with the designers. In this study, six routing components were selected, and their priority order was determined as follows: brake tube (LS) and (RS), heater hose (LS) and (RS), AC hose, and chassis cable harness. The brake tubes and heater hoses had two terminals. The AC hose had three terminals, whereas the chassis cable harness had four terminals.

First, a model simplification of the targeted vehicle was performed. As the routing components were located near the front of the vehicle, only three critical parts were included in the design domain as obstacles: the cooling module, engine, and transmission. Figure 19 shows the model simplification of these parts using a grain size of 50 mm. The grain size was set to 50 mm for the routing operation and 20 mm for generating the reference layout design. The grain size was large in the graph-based routing algorithm to reduce computational time. The location coordinates of the terminals and intermediate points, which are listed in Table 2, were saved in an Excel file at the data layer in the priority order from the first to sixth component. The intermediate points were only considered in the graph-based routing algorithm. The model information consisted of vertices and faces of the simplified model. The vertex was defined by x-, y-, and z-coordinates, and all vertices were numbered. The face, which is a triangular face in this model, was defined by the numbers associated with the three vertices. The vertex and face information of the simplified model was saved in the Excel file, as shown in Fig. 20, and was then imported into MATLAB. The first and second sheets in Fig. 20 list the x-, y-, and z-coordinates of the vertices and the numbers of the three vertices of each face of the cooling module. The third and fourth sheets and the fifth and sixth sheets list those of the engine and transmission, respectively.

After the data were imported into the system layer, the developed graph-based routing algorithm was applied. The routing layout design results are illustrated in Fig. 21 and listed in Table 3. The routing algorithm was run using a workstation (Intel Core i7-4960X CPU, 3.60 GHz, 32GB RAM) with MATLAB 2020a. The routes of all components were successfully obtained by this module. The routing solutions of the brake tube (LS), heater hose (LS) and (RS), and AC hose were acceptable. The routing solution of the brake tube (RS) involved excessive deviation outside the chassis frame because the CAD models of obstacles became larger during the model simplification procedure. Finally, the route of the chassis cable harness was also acceptable, except for the rear part of the route, which would have been reasonable if it was located inside the chassis frame in the final routing layout. Thus, the routes of the brake tube (RS) and chassis cable harness had to be slightly redesigned in manual operation after its geometric information was exported from MATLAB to CATIA

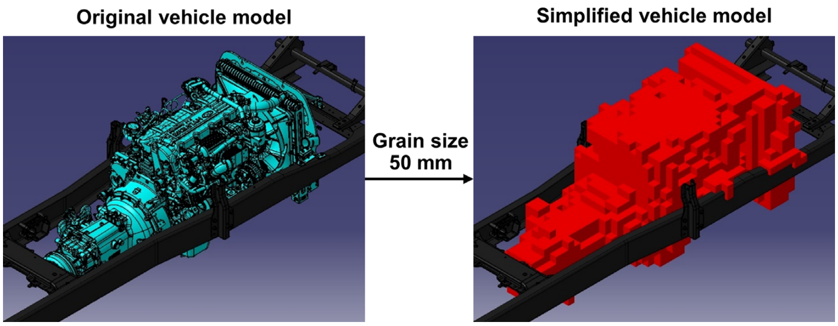
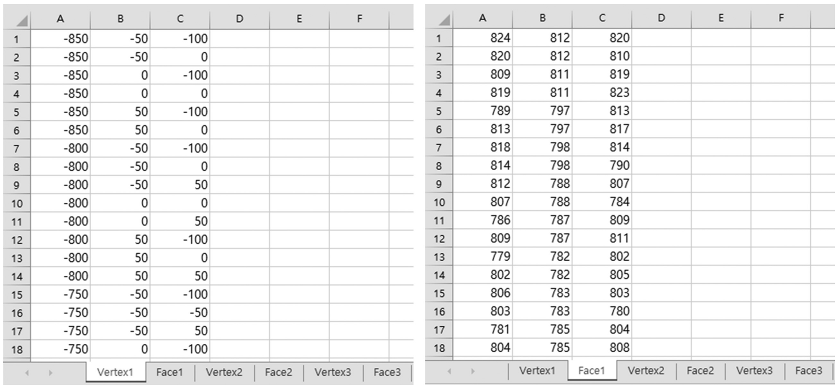


Figure 19: Simplification of the vehicle model at the input layer.



	A	B	C	D	E	F
1	-850	-50	-100			
2	-850	-50	0			
3	-850	0	-100			
4	-850	0	0			
5	-850	50	-100			
6	-850	50	0			
7	-800	-50	-100			
8	-800	-50	0			
9	-800	-50	50			
10	-800	0	0			
11	-800	0	50			
12	-800	50	-100			
13	-800	50	0			
14	-800	50	50			
15	-750	-50	-100			
16	-750	-50	-50			
17	-750	-50	50			
18	-750	0	-100			

Vertex1 | Face1 | Vertex2 | Face2 | Vertex3 | Face3

	A	B	C	D	E	F
1	824	812	820			
2	820	812	810			
3	809	811	819			
4	819	811	823			
5	789	797	813			
6	813	797	817			
7	818	798	814			
8	814	798	790			
9	812	788	807			
10	807	788	784			
11	786	787	809			
12	809	787	811			
13	779	782	802			
14	802	782	805			
15	806	783	803			
16	803	783	780			
17	781	785	804			
18	804	785	808			

Vertex1 | Face1 | Vertex2 | Face2 | Vertex3 | Face3

Figure 20: Model information in Excel at the data layer.

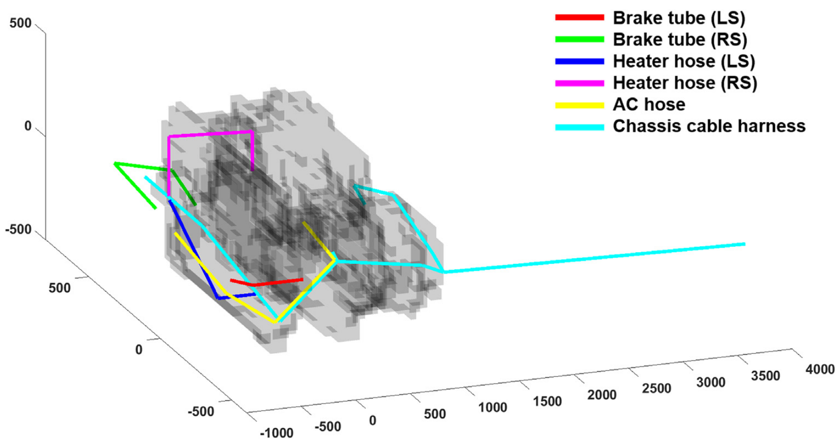


Figure 21: Routing layout design from the design support module.

Table 3: Numerical results of routing layout design.

Routing component	Routing layout design	Total length (mm) Reference routing layout design: Proposed algorithm	Total length (mm) Reference routing layout design: Conventional algorithm
Brake tube (LS)	634.6	740	740
Brake tube (RS)	1171.2	620	640
Heater hose (LS)	785.3	1000	1000
Heater hose (RS)	1252.8	1260	1240
AC hose	1682.1	2240	2260
Chassis cable harness	6857.3	7880	8020

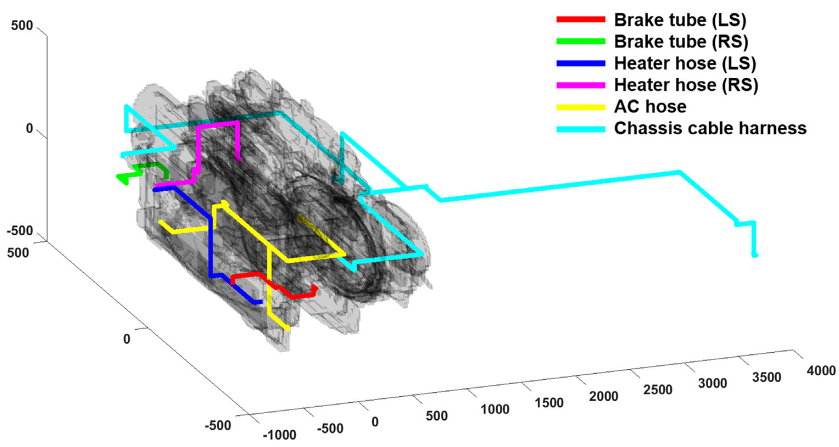


Figure 22: Reference routing layout design from the conventional cell-based routing algorithm.

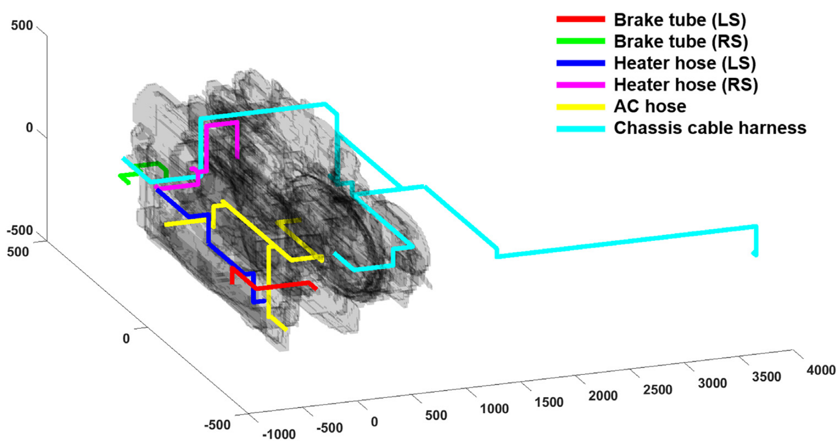


Figure 23: Reference routing layout design from the proposed cell-based routing algorithm in the design validation module.

in the output layer. The routing results were compared with the current routing design in our previous work (Kim *et al.*, 2021). In general, however, a reference routing layout design does not exist during the vehicle development process.

In this study, the proposed cell-based routing algorithm for a generating reference routing layout design was compared with the conventional algorithm, and the routing solution from the design support module was validated using the reference routing layout design. The routing results from the conventional and proposed cell-based routing algorithms based on the modified GA and MA are shown in Figs 22 and 23, respectively. The numerical results of these algorithms are listed in Table 3. Although there are some minor differences in the total length of some routing components, the two algorithms provided similar results, except for the chassis cable harness. The differences in their routes are shown in Fig. 24. As the conventional routing algorithm does not consider the clamping condition, the routing path of the chassis cable harness is located far from the obstacles, as shown in Fig. 24a. However, this is not a realistic option as it increases unnecessary assembly and production cost. On the other hand, as the proposed routing algorithm considers the clamping condition, it provided routes with higher quality, which are located near the obstacles, as shown in Fig. 24b.

Finally, the routing results from the design support module were compared with the reference routing layout design from the proposed routing algorithm. The results from the brake tubes, heater hoses, and AC hose were generally in good agreement with the reference layout, and the route of the chassis cable harness was generated on the left-hand side of the vehicle, but it passed over the engine in the reference layout. In the graph-based routing algorithm, the intermediate guided this route to the left-hand side of the vehicle. As the intermediate points were not used in the proposed cell-based routing algorithm to provide a greater degree of freedom, it found the shortest path above the engine. The total length of the route for each component from the design support module was shorter than that from the reference routing layout, except for the brake tube (RS). The total lengths of the brake tube (LS), heater hose (LS) and (RS), AC hose were approximately 14.24%, 21.47%, 0.57%, and 24.90%, respectively, shorter than the corresponding reference length. The total length of the chassis harness cable could vary slightly depending on the vehicular components behind the transmission, but it showed that the length was approximately 11.86% shorter than the reference length. The routes of the brake tube (RS) heavily deviated outside the vehicular frame compared to those of the reference routing layout. The route of the brake tube (RS) was approximately 88.90% longer than that of the reference routing layout. This is because the graph could not find the routing path based on the generated vertices. This issue can be solved by the final manual operation of the designers in the output layer. The final routing layout design of the application, however, is not included in this study because it is confidential

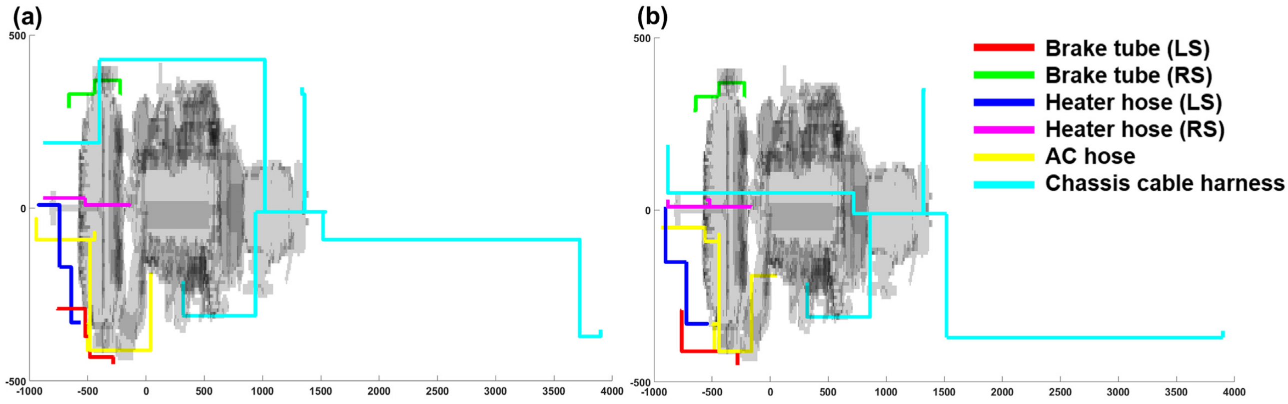


Figure 24: Reference routing layout design: (a) conventional algorithm and (b) proposed algorithm.

information that belongs to the sponsoring company. However, the authors believe that the results of this application demonstrated the effectiveness and high potential of the proposed automatic design system for generating routing layout design and showed that it worked successfully for an industrial problem.

6. Discussion and Conclusions

The routing layout design of vehicular components is one of the most demanding and challenging areas across industries. Although many studies have been carried out in other industries to resolve this issue, only a few studies have been conducted in the automobile industry. The demand for an automatic virtual design system for vehicular routing components is increasing because of the COVID-19 pandemic. Many automotive manufacturers have recently suffered problems in the design, assembly, and production processes, primarily because of the shortage in the supply of routing components and their prototypes, as well as changes in the working environment. Hence, conventional design methodologies based on know-how and trial and error are no longer favored.

In this study, we constructed an automatic design system for generating a routing layout that significantly improved the manual routing operation of the designers and engineers. It minimized the interactions between design teams and made it possible to design a routing layout without frequent route changes and trial and error using the CAD software. The authors of this paper, who are from both academia and industry, collaborated closely to propose a new automatic design system. The characteristics of the routing problem were dealt with in several steps, and the functional requirements for the design system, which were determined by practitioners in the automobile industry, were satisfied. An engineering routing problem of a commercial truck was solved to demonstrate the effectiveness of the proposed automatic design system. The sequential graph-based routing algorithm of the design support module provided a satisfactory routing layout solution for each routing component. To generate a reference routing layout design, a cell-based routing algorithm based on the modified GA and MA was adopted. A comparison between the proposed and conventional algorithms was also performed. It was demonstrated that the proposed algorithm provides a better reference layout design than the conventional one.

The authors acknowledge that the proposed automatic design system has several limitations. For example, it disregards some aspects of the routing layout design problem, e.g. assembly, production, and installation costs. Routing components such as tubes, hoses, and cable harnesses have their own characteristics. The bending radius and electromagnetic compatibility, for instance, are important factors in the routing layout design of wiring harnesses. The routing algorithms proposed in this study find the shortest routing path but cannot reflect the characteristics of each component. Because of the variety of routing algorithms and the absence of benchmark problems, a comparative study between routing algorithms was not done. The models of the obstacles were also simplified to reduce the computational time of the routing algorithms. This simplification may lead to a deterioration in the accuracy of the routing results. Therefore, some manual operations were required to obtain the final routing layout design in the last step. Despite these limitations, the authors believe that this study is a unique attempt, which suggests an automatic design system for solving routing problems in the automobile industry that significantly contributes to improving the design, assembly, and production process of commercial trucks. In future work, the aforementioned issues will be considered in developing the routing algorithm for each vehicular routing component. An intensive comparison study of the existing design approaches and routing algorithms will also be conducted to select the best algorithm in accordance with the characteristics of routing problems. In addition, the problem of the missing benchmark, which is very important for comparing the efficiency and accuracy of the developed algorithms, will be studied.

Acknowledgment

The first five authors greatly appreciate the support of this work by the Hyundai Motor Company. Such support does not constitute an endorsement by the sponsor of the opinions expressed in this article.

Conflict of interest statement

None declared.

References

Asmara, A. (2013). *Pipe routing framework for detailed ship design*. TUD Technische Universiteit Delft (Delft: VSSD).

Black C. (2017). *1990's prototype marine variant of the Cummins QSKV60 industrial diesel engine*. Grabcad Community. https://grabcad.com. (Accessed on 11.5.2018).

Dong, Z-r., & Lin, Y. (2017). A particle swarm optimization based approach for ship pipe route design. *International Shipbuilding Progress*, *63*(1–2), 59–84.

Drotz, M., & Huber, J. (2014). *Alternative harness routing and pass-through*. MS Thesis, Chalmers University of Technology.

El-Mahdy, O. F. M., Ahmed, M. E. H., & Metwalli, S. (2010). Computer aided optimization of natural gas pipe networks using genetic algorithm. *Applied Soft Computing*, *10*(4), 1141–1150.

Hermansson, T., Bohlin, R., Carlson, J. S., & Söderberg, R. (2013). Automatic assembly path planning for wiring harness installations. *Journal of Manufacturing Systems*, *32*(3), 417–422.

Hermansson, T., Bohlin, R., Carlson, J. S., & Söderberg, R. (2016). Automatic routing of flexible 1D components with functional and manufacturing constraints. *Computer-Aided Design*, *79*, 27–35.

Kang, S., Myung, S., & Han, S. (1999). A design expert system for auto-routing of ship pipes. *Journal of Ship Production*, *15*(1), 1–9.

Kim, S., Ruy, W., & Jang, B. S. (2013). The development of a practical pipe auto-routing system in a shipbuilding CAD environment using network optimization. *International Journal of Naval Architecture and Ocean Engineering*, *5*(3), 468–477.

Kim, S., Choi, T., Kim, S., Kwon, T., Lee, T. H., & Lee, K. (2021). Sequential graph-based routing algorithm for electrical harnesses, tubes, and hoses in a commercial vehicle. *Journal of Intelligent Manufacturing*, *32*(4), 917–933.

Kobayashi, M., Hirano, Y., & Higashi, M. (2014). Optimization of assembly processes of an automobile wire harness. *Computer-Aided Design and Applications*, *11*(3), 305–311.

Liu, Q., & Wang, C. (2012). Multi-terminal pipe routing by Steiner minimal tree and particle swarm optimisation. *Enterprise Information Systems*, *6*(3), 315–327.

Niu, W., Sui, H., Niu, Y., Cai, K., & Gao, W. (2016). Ship pipe routing design using NSGA-II and coevolutionary algorithm. *Mathematical Problems in Engineering*, *2016*, 7912863.

Park, J., & Storch, R. L. (2002). Pipe-routing algorithm development: Case study of a ship engine room design. *Expert Systems with Applications*, *23*(3), 299–309.

Park, J. (2002). *Pipe-routing algorithm development for a ship engine room design*. University of Washington.

Qu, Y., Jiang, D., Gao, G., & Huo, Y. (2016). Pipe routing approach for aircraft engines based on ant colony optimization. *Journal of Aerospace Engineering*, *29*(3), 4015057.

Roh, M., Lee, K., & Choi, W. (2007). Rapid generation of the piping model having the relationship with a hull structure in shipbuilding. *Advances in Engineering Software*, *38*(4), 215–228.

Sui, H., & Niu, W. (2016). Branch-pipe-routing approach for ships using improved genetic algorithm. *Frontiers of Mechanical Engineering*, *11*(3), 316–323.

Van der Velden, C., Bil, C., Yu, X., & Smith, A. (2007). An intelligent system for automatic layout routing in aerospace design. *Innovations in Systems and Software Engineering*, *3*(2), 117–128.