A NEW PIPE ROUTING APPROACH FOR AERO-ENGINES BY OCTREE MODELING AND MODIFIED MAX-MIN ANT SYSTEM OPTIMIZATION ALGORITHM

Y.-F. Qu*

Engineering Training Center
Shanghai Polytechnic University
Shanghai, China

School of Mechanical Engineering
Shanghai Jiaotong University
Shanghai, China

D. Jiang **X.-L. Zhang**

School of Mechanical Engineering
Shanghai Jiaotong University
Shanghai, China

ABSTRACT

Aero-engines usually contain a lot of pipes and cables which have an important influence on product performance and reliability. In this paper, a new pipe routing approach for aero-engines is proposed. First, an adaptive octree modeling method is presented according to the characteristics of the layout space. After considering three types of engineering constraints, the total length of pipelines, the total number of bends and the natural frequency of pipelines are modeled as the optimal objective. Then, a Modified Max-Min Ant System optimization algorithm (MMMAS), which uses layered node selection and dynamic update mechanism, is proposed for pipe routing. For branch pipelines, ant colony searches in groups and parallel to improve the solution quality and speed up the convergence greatly. Finally, numerical comparisons with other current approaches in literatures demonstrate the efficiency and effectiveness of the proposed approach. And a case study of pipe routing for aero-engines is conducted to validate this approach.

Keywords: Pipe routing, Octree model, Layered node selection, Ant colony optimization.

1. INTRODUCTION

Authors shall adapt this template to the preparation of submission.

Pipe routing, including single pipes with two terminals and branch pipes with multiple terminals, poses a considerable challenge to complex product developments such as aero-engines design, factory layout, ship building and other machinery products [1]. Aero-engines is the most crucial part of an airplane, and the piping system is one of the most complex parts of aero-engine design. In an aero-engine, there are about 50 accessories, hundreds of clamps and 100-250 two-terminal and multi-terminal external pipes in the narrow space between engine casing and nacelle [2]. These pipes connect engine components, accessories and aircraft to transmit fluid and ensure the engine works. Meanwhile, piping system plays an important role on aero-engines security, reliability and ease of operation. Statistics have shown that approximately 50% of engine accidents in the air are caused by the pipes and accessories of aircraft engines such as CF6 and CFM56 of General Electric [3]. Therefore, academic researchers and practitioners in engineering disciplines extensively pay attention to pipe routing algorithms [4].

During the past decades, much attention has been paid to two-terminal pipe routing problem [5, 6]. The maze algorithm can guarantee a solution, if one exists, but it requires a lot of memory space [7]. The genetic algorithm usually results in unavailable paths and lot of additional modifications [8, 9]. Liu and Wang proposed a modified particle swarm optimization approach for automatic generation of near-optimal pipe routes in constrained aero-engines 3D rotational space [1]. Ren *et al* proposed a new pipe routing method for aero-engines based on genetic algorithm [6]. This method simplified the searching space and added initiation and direction guideline to avoid converging to the local optimal value. From 21st century, some researchers began to study branch pipe routing [10]. Park and Storch took branch

* Corresponding author (purple1234@163.com)

DOI : 10.1017/jmech.2016.86
Copyright © 2016 The Society of Theoretical and Applied Mechanics
https://doi.org/10.1017/jmech.2016.86 Published online by Cambridge University Press

pipes into consideration in the cell-generation method, where branch pipes were end-forked and middle-forked forms [11]. Kim *et al*. presented a network routing algorithm for pipe auto-routing system including branch pipelines [12]. Liu and Wang adopted an attraction operator and an energy function in a discrete particle swarm optimization algorithm to plan the shortest collision-free connecting branch pipe routing based on escape graph (DPSO-RBPR) [13].

Even though these methods can provide feasible solutions, current research has at least two limitations: 1) the characteristics of layout space and pipe routing have not been considered in these modeling methods; 2) these branch terminals are connected sequentially or some assumptions about branch shapes are made in advance.

In this paper, the optimization algorithms for single and branch pipe routings are proposed. In Section 2, pipe routing problems are modeled in three parts. An adaptive octree modeling is constructed according to the size of the layout space and pipe diameter to save computation time and storage space. A Modified Max-Min Ant System algorithm (MMMAS) is proposed for single and branch pipe routing in Section 3. Layered node selection, dynamic update mechanism and grouped parallel search are presented to reduce the duplicate search and speed up the convergence. In Section 4, numerical simulations are conducted to comparison the proposed algorithm and the conventional algorithms. And a case of aero-engine piping system further demonstrates the effectiveness of the proposed algorithm. Section 5 finally concludes this study.

2. PROBLEM MODELING

Pipe routing approaches, including single pipelines with two terminals and branch pipelines with multiple terminals, are composed of space modeling, constrains modeling, fitness function and optimization algorithms. In this section, the above 3 steps are described. Optimization algorithms will be described in Sections 3 and 4. In space modeling section, an adaptive octree modeling is used for 3D layout space. In constraints modeling section, potential value P_i is assigned to each subspace in octree model for satisfying three engineering constraints. Based on the total length of pipeline and the total number of bends, the fitness functions of single and branch pipe routings are set in Section 2.3.

2.1 Space Modeling Based on Adaptive Octree Modeling

Space modeling is an abstract expression of 3D space environment information. It is the key to obtain the environmental information of modeling space quickly and accurately. Fan annual mesh and octree model are adopted for the layout space model of aero-engines.

The layout space is a fan annulus space between engine casing and nacelle, so fan annual mesh is adopted [14]. A cylindrical coordinate system is established, where its coordinate origin is the center of the engine front face, and the axis of engine is the axis Z. In the coordinate system, all obstacles are expanded in accordance with the minimum gap and safety requirements. Two endpoints of each pipe need be extended pipe diameter in direction of the end face. Then, the sizes of the layout space are approximated as R_0, θ_0, Z_0, which are multiples of pipe diameter in r, θ, z directions.

Octree model is adopted for more efficiency, since the shape of many obstacles in the layout space of aero-engines is regular. The octree is defined as follows: the root of the octree is the whole layout space (R_0, θ_0, Z_0). (R_0, θ_0, Z_0) is the multiple of pipe diameters in r, θ, z directions. The layout space is divided into 2^3 approximately equal subspaces in r, θ, z directions recursively. The size of subspaces is (R_i, θ_i, Z_i) and i is the division time and the layer of the octree. The relation of (R_{i+1}, θ_{i+1}, Z_{i+1}) and (R_i, θ_i, Z_i) is shown in Eq. (1).

$$\begin{cases} R_{i+1} = \text{int}(R_i / 2) \text{ or } \text{int}(R_i / 2 + 0.5) \\ \theta_{i+1} = \text{int}(\theta_i / 2) \text{ or } \text{int}(\theta_i / 2 + 0.5) \\ Z_{i+1} = \text{int}(Z_i / 2) \text{ or } \text{int}(Z_i / 2 + 0.5) \end{cases} \tag{1}$$

Where "int" is the integral symbol. If (R_i, θ_i, Z_i) is divided by 2 exactly, (R_{i+1}, θ_{i+1}, Z_{i+1}) is equal to (R_i/2, θ_i/2, Z_i/2). If (R_i, θ_i, Z_i) is not divided by 2 exactly, (R_{i+1}, θ_{i+1}, Z_{i+1}) of the inside, left, before subspace is equal to (R_i/2+1, θ_i/2+1, Z_i/2+1).

Each subspace belongs to one of three regions: black regions "0" are completely in obstacles; white regions "1" are without any obstacles; gray regions "2" are partly in obstacles. If subspaces belong to white regions or black regions, the division will be stopped. If subspaces belong to gray regions, the division will be continued till all subspaces meet the division precision--the pipe diameter. If subspaces are still gray regions when their sizes are equal to the pipe diameter in one direction, subspaces became layout planes and they will be divided into 4 approximately equal subspaces based on pipe diameter recursively. If subspaces are still gray regions when their sizes are equal to the pipe diameter in two directions, subspaces became layout curves and they will be divided into 2 approximately qual subspaces based on pipe diameter recursively. Figure 1 shows an example in octree space structure of a layout space. The size of the layout space is (4ϕ, 12ϕ, 8ϕ). Figure 2 shows the octree storage structure of the layout space in Fig. 1. The layout space is divided into 4 layers in its octree, whose sizes are (2ϕ, 6ϕ, 4ϕ), (ϕ, 3ϕ, 2ϕ), (ϕ, 2ϕ, ϕ) or (ϕ, ϕ, ϕ), (ϕ, ϕ, ϕ).

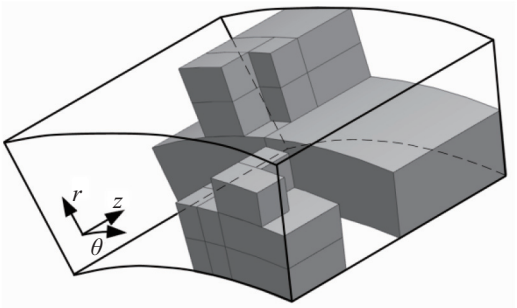


Fig. 1 Obstacles in octree space structure of a layout space.

https://doi.org/10.1017/jmech.2016.86 Published online by Cambridge University Press

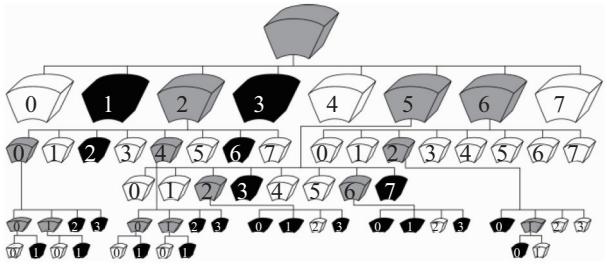


Fig. 2 Obstacles in octree storage structure of a layout space.

In octree modeling, the entire tree is stored by a location sequence number list. The location sequence number is a string of octal numbers, where each number stands for the branch in each hierarchy. These branches of the octree are sorted according to the inside and outside, left and right, before and after positions. The nodes of the octree can record the coordinates of the inside, left, before corner point of the corresponding subspace. The conversion formula between the location sequence number $N_1N_2\ldots N_n$ of a subspace and the coordinate value (R, θ, Z) of the inside, left and before corner of the subspace is shown in Eq. (2). Based on the coordinate value of the inside, left and before corner and the size of the layout subspace, the coordinate values of other seven corners can be gotten.

$$\begin{cases} R = \sum_{i=1}^{n} \left(\text{int}\left(\dfrac{N_i}{4} \right) \times \dfrac{R_{i-1}}{2} \right) \\ \theta = \sum_{i=1}^{n} \left(\text{int}\left(\dfrac{N_i \%4}{2} \right) \times \dfrac{\theta_{i-1}}{2} \right) \\ Z = \sum_{i=1}^{n} \left(\text{int}(N_i \%2) \times \dfrac{z_{i-1}}{2} \right) \end{cases} \tag{2}$$

Where the symbol % is for taking the remainder. In Fig. 2, ($3R_0/4$,0,$3Z_0/4$) and ($R_0/4$, $\theta_0/2$, $Z_0/8$) are the inside, left and before corner coordinates of 55 and 2410 subspaces, respectively. 55 and 2410 are the sequence numbers of two subspaces. 55 represents that the subspace is divided twice. The subspace belongs to white regions. 2410 represents that the subspace is divided four times. The subspace belongs to black regions.

2.2 Constraints Modeling Based on Potential Value

In octree modeling, potential value P is assigned to each white subspace for satisfying engineering constraints. By potential value P, three types of engineering constraints will be satisfied: placing pipes next to the bottom of the layout space; placing pipes along the boundaries of obstacles; placing pipes away from dangerous areas, such as high temperature areas.

Firstly, since location sequence number lists of white subspaces are related with the height and position of subspaces, P of white subspaces are changed with their location sequence number lists. The shorter and smaller the location sequence number lists are, the lower corresponding subspaces and the larger P are. The height of the subspace can be derive $\sum_{i=1}^{n} \text{int}(N_i/4)$. Secondly, P of white subspaces will change with their types. White subspaces have four types: S1-includes several terminals; S2-has several black neighbor subspaces; S3-has no any black neighbor subspaces; S4-has several special neighbor subspaces which are in dangerous areas, such as high temperature areas. P of white subspaces will increase from S1 to S4. ST = $\{d,2d,3d,4d\}$, d is the relative importance parameter. As a result, P is derived by Eq. (3).

$$P = \text{ST} \sum_{i=1}^{n} \text{int}(N_i/4) \tag{3}$$

2.3 Fitness Function

In optimization algorithms, fitness functions and constraints can be transformed into each other. In pipe routing, there are multi objectives and multi constraints. Since the route is made of the corresponding octree nodes in different layers, all nodes will be changed into the lowest layer nodes. First, fitness function consists of two parts shown in Eq. (4).

$$\begin{cases} Min(fitness\ function_F = aL + bB) \\ \text{s. t.}: P \le \text{PR};\ S \le \text{SR} \end{cases} \tag{4}$$

Where a, b are two relative importance parameters, $a + b = 1$, $a > 0$, $b > 0$. L is the length of the route, which is decided by node's numbers and distances in the route. B is the total number of bends in the route. If two coordinate values of two neighbor nodes of a node in the route are different, the node T_t is a bend. If the whole pipe routing is composed of several pipes, L can be dealt with partly and different parts L_s are set different weights W_s. L is derived by Eq. (5).

$$\begin{cases} L = \sum W_s L_s \\ L_s = \sum (|r_{T_t} - r_{T_{t-1}}| + |\theta_{T_t} - \theta_{T_{t-1}}| + |z_{T_t} - z_{T_{t-1}}|) \end{cases} \tag{5}$$

Where $(r_{T_t}, \theta_{T_t}, z_{T_t})$ and $(r_{T_{t-1}}, \theta_{T_{t-1}}, z_{T_{t-1}})$ are the coordinates of node T_t and node T_{t-1} respectively. P is the total of potential values in the whole route. PR is the experimental value. When the route has some branch points, S is the total number of branch points. SR is the total number of branch points decided by designers.

Vibration analysis of piping system has been studied early on some literatures [15, 16]. The natural frequency of piping system should be adjusted to outside 20% of the engine speed frequency for avoiding resonance. As a continuous elastic structure, the actual pipe is so thin that it can be discretized into beam elements. The start, branch, turning and end points of the pipeline are treated hinged support. When damping is neglected, the natural frequency of the pipe is only related to its mass and stiffness. Consequently the vibration equation is shown in Eq. (6).

$$[M]\{\ddot{u}\} + [K]\{u\} = 0 \tag{6}$$

Where $[M]$, $[K]$ is the structural mass, stiffness matrix

https://doi.org/10.1017/jmech.2016.86 Published online by Cambridge University Press

of the pipeline respectively. $\ddot{u}$, u denotes acceleration vectors, displacements respectively. Its harmonic solution is $\{u\} = \{\psi\}\sin(wt + \alpha_0)$, w is the circular frequency, the equation of which is reorganized as $([K] - w^2[M])\{\psi\} = 0$.

The objective function F_v is the evaluation of pipeline vibration property, as shown in Eq. (7).

$$F_v = 1/(|1 - w_v / W| - 0.2) \tag{7}$$

Where W is the excitation frequency, w_v is the natural frequency of the pipeline, 0.2 is the smallest gap between pipeline natural frequency and the excitation frequency.

3. THE MODIFIED MAX-MIN ANT SYSTEM OPTIMIZATION ALGORITHM

Ant Colony Optimization algorithm (ACO) transfers some core behaviors of real ants to an algorithm for tackling complex combinatorial problems [17]. Among four classic forms of ACO, Max-Min Ant System optimization algorithm(MMAS) has better performance than other ant colony optimization algorithms [18]. The paper proposes the Modified Max-Min Ant System optimization algorithm (MMMAS) for pipe routing as shown in Fig. 3. The main process is summarized as follows:

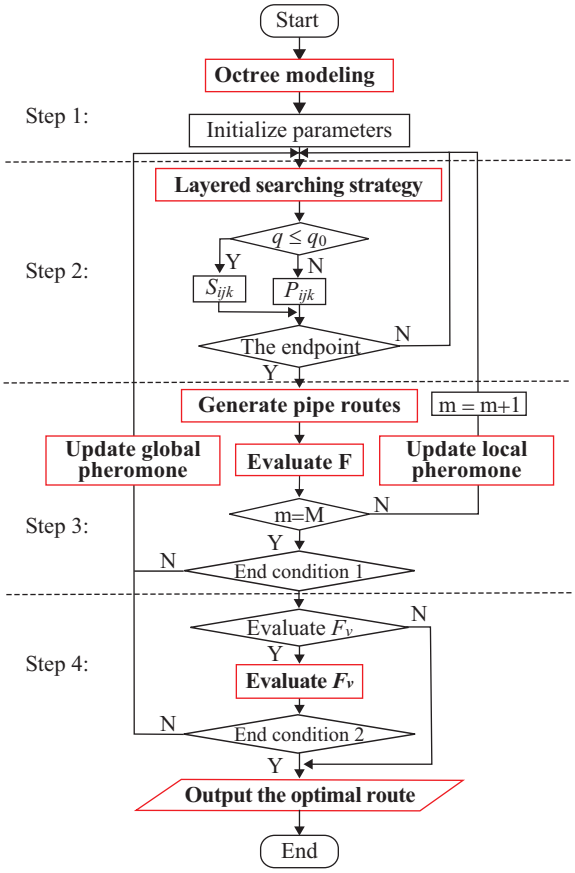


Fig. 3 The flowchart of Modified Max-Min Ant System optimization algorithm.

Step 1: create the octree model of the layout space and initialize the parameters of the optimization algorithm.

Step 2: layered neighborhood search to find all the neighbor white nodes of the current node; select the next node according to the pseudo-random-proportional rule. The detailed explanation is in section 3.1.

Step 3: Transform different layer nodes selected into the same layer for generating a route and evaluating the fitness function according to section 2.3; dynamically update the local and global pheromone. The detailed explanation is in section 3.2.

Step 4: The natural frequency of the pipeline is calculated only if need, output the optimal route and finish the algorithm.

3.1 Layered Node Selection

In octree models, a pipe route is described as node sequences. There are two fundamental search operations: layered neighborhood search and pseudo-random-proportional selection.

In layered neighborhood search, the search begins from the nodes in the same layer as the current node based on the location sequence number. Six neighbor nodes in inside, outside, left, right, before and after direction will be selected first, whose location sequence numbers in the octree are gotten by the current node. According to the regions of the neighbor nodes, there are four options: (1) white regions -- be put in a set $allowed_m$ of the optional nodes; (2) black regions -- be neglected; (3) gray regions -- the lower layer will be searched until the new nodes belong to white regions or black regions; (4) not defined -- the higher layer will be searched until the new nodes belong to white regions or black regions. All white neighbor nodes are put in the set $allowed_m$. Figure 4 shows all white neighbor nodes of a current node, where different white neighbor nodes are in different layers.

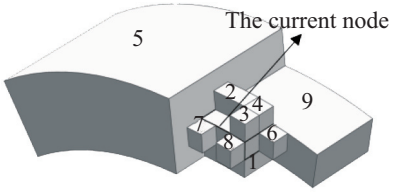


Fig. 4 Neighbor white nodes.

In pseudo-random-proportional selection, the next node v_j is searched according to Eqs. (8) and (9), which can make a balance between exploration of new paths and exploitation of accumulated knowledge.

$$S_j^m(t) = \begin{cases} org\max\limits_{k \in allowed_m} \left\{[\tau_j(t)]^\alpha \cdot [\eta_j^m]^\beta\right\} & q \le q_0 \\ P_j^m(t) & Otherwise \end{cases} \tag{8}$$

$$P_j^m(t) = \begin{cases} \dfrac{[\tau_j(t)]^\alpha \cdot [\eta_j^m]^\beta}{\sum_{k \in allowsed_m} \{[\tau_j(t)]^\alpha \cdot [\eta_j^m]^\beta\}} & if\ \ j \in allowed_m \\ 0 & Otherwise \end{cases} \tag{9}$$

https://doi.org/10.1017/jmech.2016.86 Published online by Cambridge University Press

Where α, β are two parameters which determine the relative importance of pheromone intensity $\tau_j(t)$ and heuristic information $\tau_j(t)$ ($0 \le \alpha, \beta \le 1$). q_0 is the proportion of occasion when the greedy selection technique is used, $q_0 \in (0,1)$. q is a uniform random number, $q \in [0,1]$. α, β, q_0 are experience constants. Each ant has its tabu-list that records its nodes visited to avoid revisit. Pheromone intensity $\tau_j(t)$ will be described in detail in Section 3.2.

Table 1 shows 8 octants of the neighbor nodes v_j decided the coordinates of the current node v_i and the end node v_e. Table 2 shows heuristic information $\eta_j(t)$ of the white neighbor nodes v_j in 8 octants. The axial, circumferential and radial direction is prior to other directions. The smaller radius is prior to the larger one. The final probability of white neighbor node is again calculated according to proportion after rejecting all obstacle nodes.

Table 1. 8 octants of the neighbor nodes

Octants	r	θ	z
1	$r_i \le r_j \le r_e$	$\theta_i \le \theta_j \le \theta_e$	$z_i \le z_j \le z_e$
2	$r_i \le r_j \le r_e$	$\theta_j \le \theta_i \le \theta_e$	$z_i \le z_j \le z_e$
3	$r_i \le r_j \le r_e$	$\theta_j \le \theta_i \le \theta_e$	$z_j \le z_i \le z_e$
4	$r_i \le r_j \le r_e$	$\theta_i \le \theta_j \le \theta_e$	$z_j \le z_i \le z_e$
5	$r_j \le r_i \le r_e$	$r_i \le r_j \le r_e$	$\theta_i \le \theta_j \le \theta_e$
6	$r_j \le r_i \le r_e$	$r_i \le r_j \le r_e$	$\theta_i \le \theta_j \le \theta_e$
7	$r_j \le r_i \le r_e$	$r_i \le r_j \le r_e$	$\theta_i \le \theta_j \le \theta_e$
8	$r_j \le r_i \le r_e$	$r_i \le r_j \le r_e$	$\theta_i \le \theta_j \le \theta_e$

Table 2. Priority of the next direction

Octants	Next direction								
	r+	r	r-	θ+	θ	θ-	z+	z	z-
1	2/6	2/6	2/6	2/6	3/6	1/6	2/6	3/6	1/6
2	2/6	2/6	2/6	1/6	3/6	2/6	2/6	3/6	1/6
3	2/6	2/6	2/6	1/6	3/6	2/6	1/6	3/6	2/6
4	2/6	2/6	2/6	2/6	3/6	1/6	1/6	3/6	2/6
5	1/6	2/6	3/6	2/6	3/6	1/6	2/6	3/6	1/6
6	1/6	2/6	3/6	1/6	3/6	2/6	2/6	3/6	1/6
7	1/6	2/6	3/6	2/6	3/6	1/6	1/6	3/6	2/6
8	1/6	2/6	3/6	2/6	3/6	1/6	1/6	3/6	2/6

3.2 Dynamic Update Mechanism

After finishing search, an ant gets some interconnected nodes in different layers. These node should be unified into the same layer according to the following three rules: (1) the nodes in the lower layers are prior; (2) the nodes having the smaller coordinates are prior; (3) the nodes in the higher layer are unified by their neighbor nodes. According to 5 connections with their two adjacent nodes, 9 ways to deal with the nodes in higher layers show in Fig. 5.

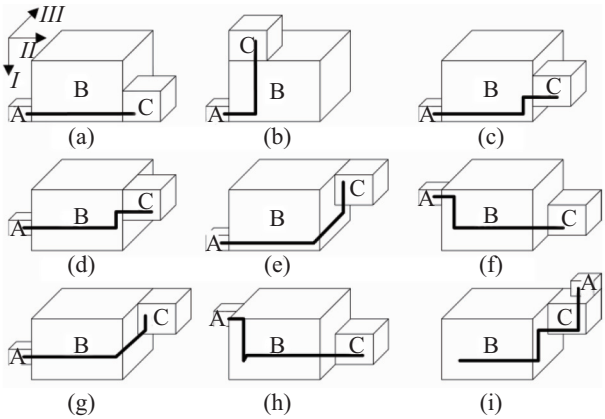


Fig. 5 9 ways to deal with the nodes in the higher layers.

(a) Two coordinates of 3 adjacent nodes have the same values;

(b) – (d) One coordinate of 3 adjacent nodes has the same value;

(e) Two coordinates of 2 adjacent nodes have the same value;

(f), (g) One coordinate of 2 adjacent nodes has the same value;

(h), (i) No adjacent node has the same coordinator value.

In order to simplify the description, *I*, *II*, *III* are used to describe the coordinators of nodes and *I*, *II*, *III* have no correspond with r, θ, z. Three adjacent nodes are represented by A, B, C. The classification is based on the coordinates of three neighbor nodes as follow.

After having a route, pheromone intensity will be updated. Firstly, after that an ant find a route, pheromone intensity in the nodes just passed will be updated according to the local update rule in Eq. (10). The rule enhances the search ability of ants on the other nodes and expands the search space. Secondly, once all ants have finished their routes, pheromone intensities on all nodes are updated with the global update rule according to Eqs. (11) and (12). The rule strengthens the attraction of this nodes and increases the convergence speed of the algorithm. The whole rule ensures that pheromone intensities are in a reasonable limitation [19, 20]. The influences of the best and worst ants are kept for improving the information feedback.

$$\tau_j(t+1) = \gamma \tau_j(t) + \tau_0 \tag{10}$$

$$\tau_j(t+1) = \Big[\Big(\rho\tau_j(t) + \sum_{m=1}^{B} w_m \Delta\tau_j^m(t) - \sum_{m=1}^{W} w_{M-m+1} \Delta\tau_j^{M-m+1}(t)\Big) d(j,av)/D\Big]_{\tau_{\min}}^{\tau_{\max}} \tag{11}$$

$$\Delta\tau_j^m = \begin{cases} Q/F_m & \text{if } j \in \text{route done by ants} \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Where τ_0 is the initial quantity of pheromone in all the nodes. γ, $\rho \in (0,1)$, two experience constants, are the local pheromone residual parameter and the global one. d(j,av) is the distance of the current node v_j and the virtual node v_{av} with the average coordinates of all the terminals. D is the longest distance among the terminals.

https://doi.org/10.1017/jmech.2016.86 Published online by Cambridge University Press

d(j, av)/D is the dynamic operator of pheromone intensity, which can ensure that pheromones are updated with the positions of nodes. That is to say, those nodes in outer region get more pheromone intensities than those in central region.

According to the value of fitness function, all ants are listed in descending order. In order to absorb the experience of the previous iterations better, there are only pheromones of the best ants and the worst ants considered in Eq. (11). B, W is the number of the best ants and the worst ants respectively. w_m, w_{M-m} are the weights of different fitness functions. $\Delta\tau_j^m$ is the pheromone quantity of ant m. Q is the total quantity of pheromone laid by ants in one route. F_m is the fitness function. After updating, pheromone intensities $\tau_j(t)$ are in the limit of $\tau_{\min}$ and $\tau_{\max}$. τ_0 is the initialization of pheromone intensity in every node. τ_0, $\tau_{\min}$, $\tau_{\max}$ are three empirical constants. The dynamic update mechanism can prevent excessive accumulation of pheromone in central region in order to reduce the duplicate search and ensure the diversity of solution.

3.3 Grouped Parallel Search

If a pipeline has several terminals, its route will have branches and every terminal should be assigned an ant. Then the ant colony is grouped and a group of ants search in parallel for a branch route. The generating process of a branch route is shown in Fig. 6. The following is detailed description.

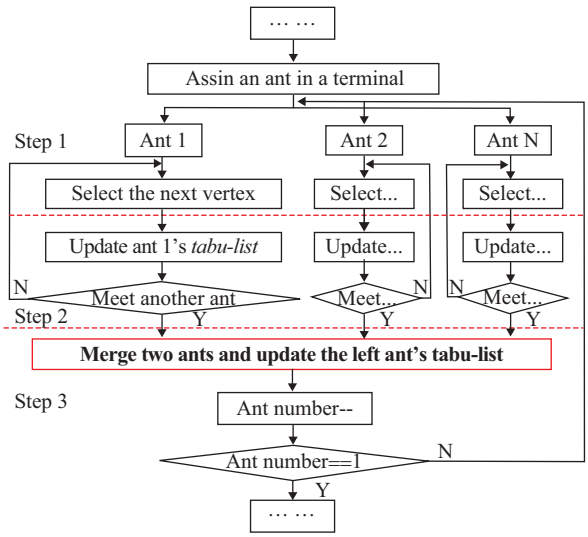


Fig. 6 The generating process of a branch route.

Step 1 The number of ants in a group is decided by the number of terminals. All ants in a group search in parallel and select their next nodes according to layered neighborhood search and pseudo-random proportional selection in Section 3.1.

Step 2 After the next nodes will be added into the corresponding ants' *tabu-lists*, the algorithm judges whether an ant meets another ant by all ants' *tabu-lists*. If the next node of ant 1 is not in ant 2's *tabu-list*, ant 1 does not meet ant 2. The algorithm will turn to ***Step 1*** and ant 1 will keep searching its next node. If the next node of ant 1 is in ant 2's *tabu-list*, ant 1 meets ant 2. The algorithm will continue to ***Step 3***.

Step 3 Ant 2's *tabu-list* is added into ant 1's *tabu-list* and ant 2 is deleted. The number of ants is decreased. Judge whether only one ant left in the group. If the number of ants is above 1, the algorithm will turn to ***Step 1*** and ant 1 will keep searching its next node. If only one ant left, a branch route is found.

4. NUMERICAL SIMULATION

4.1 Single and Branch Pipe Routing

The approach is implemented by UG and MATLAB software. The layout space is developed in UG/NX 7.5 and the algorithms are programmed in MATLAB 7.0. And the layout results are imported to the CAD model in UG. Here the pipeline diameter is abstracted into 1. The size of the layout space is (8, 24, 16). Depending on the size of the layout space and simulation experiences, these parameters of the algorithm are set in Table 3.

Table 3 The parameters in the algorithm

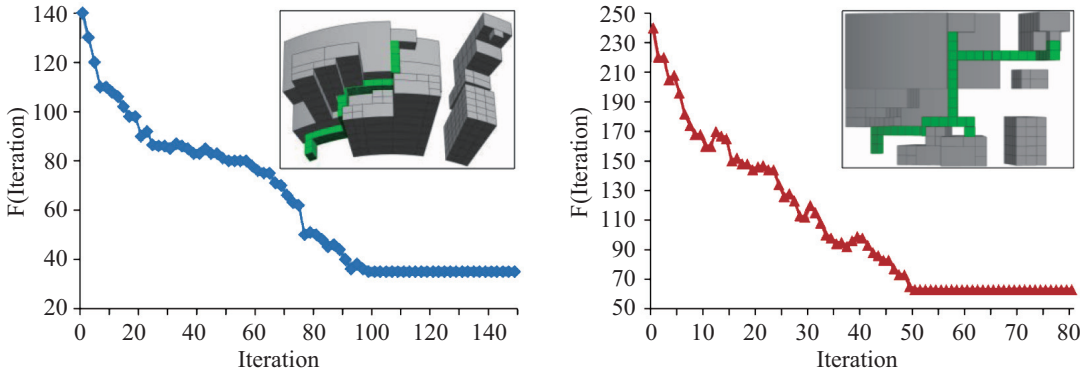
Parameters	Values
Fitness function parameters	a = 0.2, b = 0.8
Number of ants/groups	M/G = 100
The proportion of occasion	q_0 = 0.4
Pheromone residual parameters	ρ = 0.5, γ = 0.6
The relative importance	α = 0.6, β = 0.4
Initial pheromone value	$\tau_{ijk}(0)$ = 10
The total quantity of pheromone	Q = 100~2000
Pheromone limitation	$\tau_{\min}$ = 2, $\tau_{\max}$ = 200
The experimental potential value	**PR** = 60
the best **B** and the worst **W** ants/group	**B** = 20, **W** = 5

Two examples of Modified Max-Min Ant System optimization algorithm (MASMM) are shown in Fig. 7. Figure 7(a) shows a single pipe routing and its convergence curve. Figure 7(b) shows a branch pipe routing and its convergence curve. There are 3 terminals and a branch point in the branch pipeline. The proposed approach converges to 34.7 and 62.5 in 100 and 50 iterations. Two pipe routes not only avoid obstacles, follow around obstacles as close as possible, but have shortest length and less number of turns. The optimal route and obstacles have different grays. Although the case only is a simplification of the actual layout, the layout result can show that the proposed approach can get the optimal pipe routes under given conditions.

4.2 MMMAS Versus MPSO and DPSO

To further demonstrate the effectiveness of the proposed approaches, the approach is compared among MPSO [1] in single pipe routing and DPSO [13] in branch pipe routing. Table 4 shows the performance comparison between the proposed approach MMMAS and MPSO on 6 cases with 2 terminals. Table 5 shows

https://doi.org/10.1017/jmech.2016.86 Published online by Cambridge University Press



(a) A single pipe routing (b) A branch pipe routing

Fig. 7 Two examples of the Modified Max-Min Ant System optimization algorithm.

the performance comparison between the proposed approach MMMAS and DPSO on 8 cases with more than 4 terminals.

The first column of Table 4 is the number of obstacles. The first two columns of Table 5 are the number of terminals and obstacles. The last two columns of Table 4 and 5 shows running time and fitness function produced by MMMAS and MPSO or DPSO respectively. The fitness functions of two approaches are equal in most cases. With the increase of obstacles or terminals, the running time of MMMAS is less than that of MPSO and DPSO. MMMAS takes about 29 seconds to lay a single route in the layout space with 20 obstacles. MMMAS takes about 145 seconds to lay a branch route with 50 terminals. So the proposed approach has better performance and is more practical for pipe routing application.

Table 4 The comparisons among MMMAS and MPSO.

Obstacle Number	Running Time(s)		Fitness function	
	MMMAS	MPSO	MMMAS	MPSO
1	3	3	216.6	216.6
4	9	10	256.6	256.6
7	10	12	316	316
10	16	20	346.4	352.4
15	22	26	392.6	392.6
20	29	35	407.4	410.2

Table 5 The comparisons among MMMAS and DPSO.

Terminal Number	Obstacle Number	Running Time(s)		Fitness function	
		MMMAS	DPSO	MMMAS	DPSO
4	1	4	5	277.4	277.4
4	4	10	11	295	304.4
5	4	14	16	322	322
7	7	18	27	361.4	361.4
12	10	18	36	524.4	562.2
20	10	36	68	577.8	577.8
30	15	90	106	728.4	738.4
50	20	145	247	900.2	900.2

4.3 A Case Study

The Modified Max-Min Ant System optimization algorithm is applied for single and branch pipelines in an aero-engine as shown in Fig. 8. The size of the aero-engine is diameter 806 mm, length 1212 mm. 12 accessories are kept as obstacles, which are approximated to the directions of three coordinate axes. Two intersection obstacles should be combined. The optimization propose of this case can be expressed by Eq. (13).

$$\begin{cases} Min(fitness\ function_F = 0.2L + 0.8B) \\ \text{s. t.}: P \le 1000;\ S \le 10 \end{cases} \tag{13}$$

There 10 different specification pipes are laid out, including 14 single pipelines and 3 branch pipelines. All pipelines are rounded at turning nodes. These pipelines are laid along the axis, circumference and radial direction. The total routing time is about 20 minutes. The average time is about 60 seconds for routing a single pipeline and 180 seconds for routing a branch pipeline. Their natural frequencies of pipeline A and B are calculated. The parameters and natural frequencies of pipe A and B are shown in Table 6. Their natural frequencies are out of 20% limitation of engine agitation frequency 428.3 Hz. The average time is about 5 seconds for vibration performance of one pipeline. The results further show that the algorithm is effective.

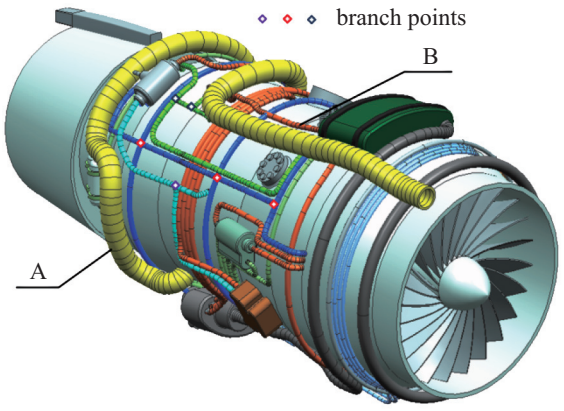


Fig. 8 Pipe routing layout.

https://doi.org/10.1017/jmech.2016.86 Published online by Cambridge University Press

Table 6 The parameters and natural frequencies of pipelines

Pipeline	Outside diameter	Inside diameter	Material	1 order	2 order	3 order	4 order
A	20	16	20	527.2	561.2	574.3	677.8
B	5	3	T7	321.4	334.4	427.7	486.4

5. CONCLUSIONS

This paper presents an approach for efficiently generating the optimal single and branch pipelines in aero-engines, which mainly includes adaptive octree space modeling and the Modified Max-Min Ant System optimization algorithm(MMMAS).

The 3D rotational space of aero-engines is modeled by fan annual mesh. And then its octree model is created according to the size of the layout space and the multiple of pipe diameter, which can improve the search efficiency and save the storage space. In MMMAS algorithm, layered node selection and dynamic update mechanism are set to improve search ability. For branch routes, ant colony will adapt grouped parallel search for faster convergence speed. Numerical comparisons with other approaches and a case study of an aero-engine demonstrate the effectiveness of the proposed algorithm. Future research should be on taking more engineering constraints into consideration, such as pipes should be laid to reduce the fluid loss.

ACKNOWLEDGEMENTS

The work was financially supported by the National Aviation Special Fund Project of China (Grant NO. 2014ZB57).

REFERENCES

1. Liu, Q. and Wang, C., "Pipe-Assembly Approach for Aero-Engines by Modified Particle Swarm Optimization," *Assembly Automation*, **30**, pp. 365-377 (2010).
2. Yin, Y. H., Li, D. X., Bi, Z. Chen, H. and Zhou, C., "Novel Human-Machine Collaborative Interface for Aero-Engine Pipe Routing," *IEEE Transactions on Industrial Informatics*, **9**, pp. 2187-2199 (2013).
3. Yin, Y. H., Zhou, C. and Zhu, J. Y., "A Pipe Route Design Methodology by Imitating Human Imaginal Thinking," *CIRP Annals-Manufacturing Technology*, **59**, pp. 167-170 (2010).
4. Liu, Q. and Wang, C. E., "A Modified Particle Swarm Optimizer for Pipe Route Design," *The 11th IEEE International Conference on Computational Science and Engineering-Workshops*, Washington, U.S. (2008).
5. Ito, T., "Piping Layout Wizard: Basic Concepts and Its Potential for Pipe Route Planning," *Methodology and Tools in Knowledge-Based Systems*, pp. 438-447 (1998).
6. Ren, T., *et. al*., "A New Pipe Routing Method for Aero-Engines Based on Genetic Algorithm," *Journal of Aerospace Engineering*, **228**, pp. 424-434 (2014).
7. Lee, C. Y., "An Algorithm for Path Connections and Its Applications," *IRE Transactions on Electronic Computers*, **10**, pp. 346-365 (1961).
8. Ito, T., "A Genetic Algorithm Approach to Piping Route Path Planning," *Journal of Intelligent Manufacturing*, **10**, pp. 103-114 (1999).
9. Fan, X. N., Lin, Y. and Ji, Z. S., "The Ant Colony Optimization for Ship Pipe Route Design in 3D Space", *Proceedings of the 6th IEEE World Congress on Intelligent Control and Automation*, **1**, pp. 3103-3108 (2006).
10. Rezaei, G., Afshar, H. M. and Rohani, M., "Layout Optimization of Looped Networks by Constrained Ant Colony Optimisation Algorithm," *Advances in Engineering Software*, **70**, pp. 123-133 (2014).
11. Park, J. H. and Storch, R. L., "Pipe-Routing Algorithm Development: Case Study of a Ship Engine Room Design," *Expert Systems with Applications*, **23**, pp. 299-309 (2002).
12. Kim, S. H, Ruy, W. S. and Jang, B. S., "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**, pp. 468-477 (2013).
13. Liu, Q. and Wang, C. E., "A Discrete Particle Swarm Optimization Algorithm for Rectilinear Branch Pipe Routing," *Assembly Automation*, **31**, pp. 363-368 (2011).
14. Bai, X. L, Zhang, Y., Xie, H. L. and Ren, T., "Spatial Information Extraction for Automatic Layout Routing in Complex Product Design," *Chinese Control and Decision Conference*, Guilin, China (2009).
15. Ouyang, X. P., Gao, F., Yang, H. Y. and Wang, H. X., "Two-Dimensional Stress Analysis of the Aircraft Hydraulic System Pipeline," Proceedings of the Institution of Mechanical Engineers, *Part G: Journal of Aerospace Engineering*, **226**, pp. 532-539 (2012).
16. An, C. and Su, J., "Dynamic Behavior of Pipes Conveying Gas–Liquid Two-Phase Flow," *Nuclear Engineering and Design*, **292**, pp. 204-212 (2015).
17. Dorigo, M. and Blum, C., "Ant Colony Optimization Theory: A Survey," *Theoretical Computer Science*, **344**, pp. 243-278(2005).
18. Hu, X. B., "Research on Principles, Theory and Ap-

https://doi.org/10.1017/jmech.2016.86 Published online by Cambridge University Press

plication of Ant Colony Optimization." Ph.D. Dissertation, *Chongqing University*, Chongqing, China (2004).

19. Dorigo, M. and Gambardella, L. M., "Ant Colonies for the Travelling Salesman Problem," *BioSystems*, **43**, pp. 73-81 (1997).
20. Bullnheimer, B., Hartl, R.F. and Strauss, C., "A New Rank Based Version of the Ant System-a Computational Study," *Central European Journal for Operations Research and Economics*, **7**, pp. 25-38 (1999).

(Manuscript received February 24, 2015,
accepted for publication June 15, 2016.)

https://doi.org/10.1017/jmech.2016.86 Published online by Cambridge University Press