Simulation Optimization of Police Patrol District Design
Using an Adjusted Simulated Annealing Approach

Yue Zhang, Donald Brown
Predictive Technology Laboratory,
Department of Systems and Information Engineering,
University of Virginia, Charlottesville, Virginia, USA
Email: Yue Zhang - yz5yf@virginia.edu

Keywords: Police Patrol, District Design, Simulated Annealing, Simulation Optimization, Discrete-Event Simulation

Abstract
In this study, a simulated annealing approach is developed to find optimal police patrol district design using a discrete-event simulation to evaluate the performances of districting plans, such as average response time and workload variation among districts. Similar to the solution neighborhood definition in an existing simulated annealing approach, the new solution (districting plan) in each iteration is developed by making changes to the current solution. Instead of changing the assignment of only one atom (atomic geographical unit), the new approach makes relatively big changes to a current districting plan through a cutting and merging process of current districts. A case study on Charlottesville, VA is conducted. Experimental results show that the new approach uses fewer iterations to reach optima so it is well-suited for discrete-event simulation. It evaluates districting plans in a more detailed way but uses longer computational times than Police Car Allocation Model (PCAM). In addition, the new approach is more robust for the adjacency pattern where atoms have relatively fewer adjacent neighbors. The adjusted simulated annealing approach is also compared with a response surface method of optimizing district design. Experimental result shows that the simulated annealing approach provides better solution. It is also an automatic method and thus better for practical application.

1. INTRODUCTION
1.1. Police Patrol District Design
Police patrolling is an indispensable function of police department [22]. It responds incidents, deter and prevent crimes and is an important urban emergency service [18]. It is a highly labor intensive service [21] and uses more than half of the personnel resources [8]. Police departments need to find patrol operation strategies that maximize performances at minimum cost. Police patrol district design is one of the important decisions that affect the effectiveness of patrol operations. Better districting plans lead to lower response times, more equal division workload, officer's familiarization with their assigned area, more efficient use of personnel, more visible police presence et al. [9]. Two relatively important performance measures of district design are the average response time and the variation of workload among different districts [6]. The objective of the district design optimization problem can be modeled as finding optimal or near-optimal district plans that minimize both measures. Usually, police patrol districts are designed based on atomic geographical units, such as census blocks, police beats or police reporting district [20]. In the redistricting procedure, these atoms are partitioned to several groups (districts) with the constraints of contiguity and compactness. This graph partitioning problem has been shown to be NP-hard [2][10]. It is not feasible to evaluate the districting plans using field experiment due to the issues of safety, risks, costs, public relations and law enforcement [11]. In addition, a large portion of inter-sector dispatches (cross-boundary responding) [13] makes the estimation of average response time and workload variation more difficult. Some existing Operation Research and GIS methods include: p-median clustering method [17], an interchange heuristic method [1], a simulated annealing approach [6] and a GIS based maximal covering method [5]. These approaches can produce good alternatives for police district design but their evaluations of districting plans do not consider the important factors (stochasticity of the temporal and spatial pattern of Calls For Services (CFS) incidents, patrol and dispatching rule, traffic condition, road network, cross-boundary responding et al.) altogether. However, simulation method can model these factors affecting the patrolling and responding activities of police cars in a flexible manner and provide more detailed evaluation of district designs. In addition to district design, more recent literature related to police patrol assignment and routing operations can be found in [7] [14] [23] [4].

1.2. Related Work
Our previous work [24] used agent-based simulation in GIS environment to evaluate districting plans, which was developed based on a RepastCity prototype in [16][15] using Java Repast. The animation of the simulation model was validated by the local police department and the evaluation of
district plans is considered to be close to reality. It provides high-fidelity evaluation of performance measures but is not computationally efficient. The evaluation time of the agent-based model is relatively long because the agents (police cars) read geographical information and update their status (speed, position, etc.) each tick. Instead of tick-by-tick evaluation of police cars’ state variables in agent-based model, the police patrol discrete-event simulation in [26] updates police cars’ status only when they get calls, arrive at scenes, leave scenes and return to patrol states. The travel times are modeled using Euclidean distance and responding speed affected by overall traffic conditions at different time of day. The generation of CFS incidents is the same as that used in agent-based model. The discrete-event model can be considered as a simplified version of agent-based model. It runs much faster than the agent-based model but provides similar districting plan evaluation [26].

To optimize the police patrol district design using simulation for evaluation, a parameterized districting algorithm [24] was proposed to generate districting plans by partitioning the atomic geographical units with the constraints of contiguity and compactness of districts. The pattern of layout of districting plans can be represented by districting parameters, which describe the locations of district “seeds” (A seed is one atomic geographical unit). Districting plans can be generated by “seeds-growing” algorithm automatically. Then, an iterative searching procedure based on experimental design and response surface methodology was developed in [25] to study the relationship between districting parameters and performance measures of districting plans. Using improved districting parameters, good plans can be generated efficiently. This optimization procedure can be used for both agent-based and discrete-event simulation model. Optimal or near-optimal districting plans found in both models show similar pattern.

In this study, a more automatic simulation optimization method was developed based on a simulated annealing approach [6]. Instead of using Police Car Allocation Model (PCAM) for district design evaluation, our approach uses discrete-event simulation model. The new approach defines the solution neighbourhood by cutting and merging districts in current districting plan. Experimental results are compared with the response surface optimization method.

2. DISCRETE-EVENT SIMULATION

2.1. Introduction

Similar to Hypercube Queuing Model (HQM) [12], each patrol car has two states: idle (0) and busy (1) in the police patrol discrete event simulation model. The state of the whole system is represented as a binary sequence of server statuses. While the basic HQM provides a well-defined framework for modeling emergency response systems, the size of the problem grows exponentially with the number of servers. Solving each instance requires solving a linear system with an exponential number of variables [3]. [3] demonstrates that Monte Carlo discrete-event simulations based on HQM converge to the steady-state probabilities estimated by HQM very quickly. Therefore, discrete event simulations provide an alternative method for solving for the HQM steady-state probabilities. The discrete-event model can more easily be extended to simulate complex situations, such as multiple cars responding, different priorities of CFS incidents, different CFS arrival rate at different times of day, as well as various patrol and dispatch rules.

2.2. Case Study Data

The Charlottesville Police Department (CPD) provided historical CFS incidents data for this case study. The city of Charlottesville is a mid-size city centrally located in the state of Virginia, USA. The city has a diameter of about 7 miles and a year-round population of about 40,000, which swells to about 66,000 during the academic year due to the presence of a major university. The city uses eight city patrol districts, with one car routinely assigned to each patrol district. Usually, the CPD dispatches the nearest available car to the scene of a CFS. Figure 1 shows the trends of CFS inter-arrival time (related to the demand for police patrol), service time (time on scene) and response time (highly correlated with traffic condition) over the 24 hour period based on 330,000 CFS incidents. These historical CFS incidents are spatially joined to a grid network (323 grids) and the spatial CFS density plot is shown in Figure 2. It can be seen that the CFS incidents are more likely to happen in the central area and along a major road. These grids are used as basic geographic units when developing the districting algorithm. In Charlottesville, the smallest existing atomic geographical unit is census blocks but Charlottesville only has about 30 census blocks, which provide limited number of possible partitioning plans. For scientific study, we need to increase the granularity of base geographical units and the 323 grids can provide more possibilities of district designs. The districting method developed based on
these grids can generalize well to large city which has several hundreds of atoms.

2.3. Implementation

We developed the simulation model in Java 1.6 SE using pseudocode provided in [3, 19] provides the method we used to calculate the expected locations of CFS and patrol cars in the city. The inputs for the simulation model are: 1) CFS inter-arrival time, 2) service time, 3) CFS probability for each atom, 4) Geographical information (atoms and district plan), 5) responding speed.

The discrete-event simulation model tracks the occurrence of four types of events: 1) arrival events of CFS, 2) patrol car arrival at CFS, 3) patrol car departure from CFS, 4) patrol car arrival at base (idle position). The program uses an event list to store these events ranked by scheduled time. The earliest event is taken out of the list and corresponding actions are executed. Usually, the actions include updating police cars’ status, scheduling the following events and putting them into the event list. The model generates CFS incidents based spatial (Figure 2) and temporal (Figure 1) pattern from historical data. When a CFS occurs in the simulation model, the nearest idle patrol car is dispatched by changing the server availability status from idle (0) to busy (1). The idle police car’s current position is generated by randomly selecting an atom centroid in patrol district based on the police car’s preventive patrol frequency. The travel time can be estimated by the Euclidean distance between the car’s current position and the CFS incident location divided by the responding speed. Then the event of police car arrival at scene can be scheduled and put into the event list. When police car arrives at scene, service time is randomly generated from the exponential distribution estimated from historical records. Similarly, the event of police car departure can be scheduled. When police car leaves the scene, the travel time is estimated in the similar way. The patrol car status returns to idle (0) once the car returns to its base location within the patrol sector after each event.

The input parameters of the simulation model are well calibrated. The districting plan is fixed at current district design in Charlottesville. The inter-arrival time and service time are modeled in the hour-by-hour manner based on the historical data (Figure 1). The responding speed is also set hourly so that the response time in simulation output is equal to the historical means of that hour. In this way, the generation of CFS incidents and overall queuing statistics in simulation are calibrated to actual situation.

3. SIMULATED ANNEALING APPROACH

3.1. Existing Method

The existing simulated annealing approach [6] starts with an initial districting plan as current solution. In each iteration of the simulated annealing process, the neighborhood solution is defined by making slight changes on the current solution. In the current districting plan, one atom is randomly selected on the border between two adjacent districts. After swapping the atom between the two districts, the new plan is generated. However, not each new plan can be considered feasible to replace the current solution. The algorithm has several conditions and constraints to constitute feasibility. These constraints include: 1) The ratio of areas of the biggest and smallest districts should not exceed a specified bound; 2) The atoms of any district should be connected to each other; 3) Compactness of any district (the ratio of the longest Euclidean path and the square root of the area) should not exceed a specified bound; 4) The district which receives a new atom should not have a protrusion (convexity constraint). Similarly, the district that loses an atom should not have an indentation. More specifically, the following three situations are considered infeasible situations that violate the convexity constraints: “(i) an exchange which causes an atom to be too far away from the closest atom in its district; (ii) an exchange which causes an atom to be adjacent to only one other atom in its district; (iii) if A is the exchanged atom and B is an atom adjacent to A in the receiving district, then B must be adjacent to at least four atoms (including A) in its district.” [6]

These constraints ensure the solution generated is implementable. The size constraint avoids the possibility of generating districts that are too large or too small. The contiguity constraint keeps patrol cars from crossing district boundaries. The compactness and convexity constraints avoid gerrymandering and ensure that a district has a relatively round rather than long and slender shape. In the Buffalo police case study, 409 R-districts (atomic geographical unit) are partitioned into 5 districts. The simulated annealing algorithm begins with the current district design in Buffalo and runs for 400K iterations. The actual computational time is 2 hour in a 450 MHz per-
In our study, it is of great interest to integrate the existing simulated annealing algorithm with the discrete-event simulation for district design evaluation. However, the current simulated annealing approach cannot be applied directly in Charlottesville police case study. First of all, the discrete-event simulation takes much longer time to evaluate a districting plan than PCAM. The PCAM uses less than 0.018 second to evaluate a district design while the discrete-event model uses 5 seconds. The estimated total computational time is 23 days for 400K iterations, which is not realistic in practice. Secondly, the convexity constraints cannot be applied to atoms in the form of grid network. The convexity constraints are so strong that the solution tends to get stuck in a local optimal. For example, the convexity constraint (ii) alone will lead to the situation shown in Figure 3. It can be seen that the layout or structure of districting plan does not change so much after 50K iterations of simulated annealing process. The initial plan is a bad plan and the final plan is still far from the optimal configuration based on previous study. Similar problem occurred if convexity constraint only includes situation (iii) even if the “at least four atoms” condition is relaxed to 3 or 2 atoms. These convexity constraints work well in Buffalo case because the base atomic units are “R-districts” and most of them have more than 4 adjacent atoms. However, in Charlottesville case, most atom grids are squares which have at most 4 adjacent atoms. The convexity constraints limit the alternatives of possible changes and thus confine the solution in local optima. If we remove the convexity constraints (ii) and (iii), gerrymandering problem will occur. Therefore, small adjustments on the existing simulated annealing approach may not work in Charlottesville grid network case. In the next section, an adjusted simulated annealing algorithm is described, which works for atoms in grid network (and other types of atoms) and is well-suited for discrete-event simulation evaluation.

### 3.2. Adjusted Simulated Annealing Algorithm

The basic simulated annealing process in the new algorithm is the same as the existing approach in Section 3.1. The major adjustment made for the original approach is the definition of solution neighborhood. In existing method, the new districting plan is generated by changing only one atom based on current plan. In adjusted algorithm, relatively “big” changes are made to current solution in each iteration.

#### Algorithm 1 Adjusted Simulated Annealing Algorithm

1. **Merge and cut random 2 adjacent districts**
   - Randomly select two adjacent districts from all pairs
   - Merge them into one and randomly cut it into 2 districts
   - Find the “center of mass” of the combined district
   - Select a random angle and cut through the center
   - Conduct contiguity check. If it fails, cancel the changes
   - Repeat this process on all pairs of districts until there is a successful “merge and cut”
   - If all pairs fail, the program goes to the next step
2. **Cut one “bad” district and merge two smallest districts**
   - “Bad” districts: high CFS probability, bad compactness
   - Cut one “Bad” district into two districts
   - Choose the “smallest” two adjacent districts to merge
   - Conduct contiguity check. If it fails, cancel the changes
3. **Seeds-growing procedure**
   - Find “center of mass” of each district
   - Find atoms closest to “centers of mass” and use them as “seeds” in the “re-growing” procedure
   - Mark All atoms as “unassigned”
   - Assign each seed to a district
   - An iterative process of seeds growing
     - Develop the district with min CFS probability
     - The district’s unassigned neighbor atoms are assigned to the nearest district
     - Repeat this process until it stops
     - The smallest district cannot grow anymore (all its neighbor atoms are assigned to districts)
   - An iterative process for remaining unassigned atoms
     - From unassigned atoms adjacent to all current districts, randomly select one atom
     - Assign it to the nearest adjacent district
     - Repeat this process until all atoms are assigned

In step 1 of the algorithm (Figure 4), relatively big changes are made to current districting plan by merging and cutting 2 adjacent districts. To ensure the randomness of the algorithm, the two adjacent districts are randomly selected and the cutting angle is also randomly chosen. When cutting the merged district, the $x$ and $y$ coordinates of the “center of mass” are calculated by taking the weighted sum of all atoms’ coordi-
nates based on CFS probabilities. In this way, the two districts after cutting will have almost equal CFS probabilities. Due to various shapes of the combined district, the random cutting process may produce districts that violate the contiguity constraint. If that happens, the changes made by the merging and cutting process are canceled. Since it is important to make a big change to the current plan, the merging and cutting procedure is tried on all pairs of adjacent districts until there is a successful one.

Since the merging and cutting process of step 1 is in a highly random manner, the derived districting plan may have very large districts or districts with bad compactness. If the district’s cumulative CFS probability or compactness exceeds an pre-specified upper bound, it is considered as a “bad” district. If there is no “bad district”, step 2 can be skipped. If there are many “bad” districts, the worst district is cut into 2 districts. The cutting procedure is similar that in step 1. Usually, the worst district has gerrymandering issue and the cutting tends to produce districts with contiguity violation. So a pre-specified number of trials of cutting can be conducted with random cutting angles. In this way, the chance of successful cutting is increased. Based on previous study, the districting plans with lower variation of CFS probabilities among all districts tend to have better performances on average response time and workload variation. Step 2 can help balance the CFS probabilities among districts so that the search for optimal plans are kept in a solution set that has relatively low CFS probability variation. Step 2 can also break the gerrymandering district into 2 districts and in most cases each of them has better compactness.

Step 1 makes big random changes to the original districting plan and step 2 adjusts the plan so that districts become more compact and the CFS probabilities are balanced among districts. However, the random cutting and merging process may make the district boundaries serpentine. Step 3 of the algorithm uses the “seed-growing” procedure to smooth the district boundaries while keeping the original pattern or structure of the districting plan (Figure 5). The “center of mass” of each district serves as a “seed” and these seeds ultimately grow to districts. In each iteration of the growing procedure, the district with the lowest cumulative CFS probability has the priority to acquire adjacent neighbor atoms. So, the CFS probabilities can be balanced among districts. When districts grow, the neighbor atoms are usually assigned to the nearest adjacent district. In this way, the shape of the district tends to be round and thus the compactness of the districting plan can be guaranteed.

4. CASE STUDY

4.1. Charlottesville Police Case

As is noted in Section 2.2., the goal of Charlottesville case study is to find an optimal or near-optimal districting plans that minimize both average response time and workload variation among districts. The districting plan is drawn by partitioning the 323 atoms in a grid network (Figure 6) into 8 districts with the constraints of contiguity and compactness. The adjusted simulated algorithm described in Section 3.2. is applied for the optimization of districting designs. The discrete-event simulation described in Section 2. is used for evaluation of district plans.

The discrete-event simulation outputs two measurements (average response time and workload variation) for a districting plan while simulated annealing algorithm only needs one single evaluation score. So it is necessary to develop a procedure to convert two simulation measures into one evaluation score. This procedure is developed based on 3000 random plans generated by parameterized districting algorithm described in [24]. These plans have various layouts and structures and include very good and very bad plans. They are generated before the simulated annealing algorithm and are used as a reference to develop the evaluation score. First, these plans are evaluated by discrete-event simulation so that the two measures of each plan are obtained. Then, the evaluation score of each plan is calculated based on weighted sum of standardized performance measures (0 for worst and 1 for best). The weights for average response time and workload variation is [0.4, 0.6] since reducing workload variation is more important to Charlottesville police department. The evaluation score is 0 for the worst plan and 1 for the best plan. Lastly, the relationship between simulation measures and one evaluation score can be obtained by running regression analysis on these sample plans. Thus, this relationship can be used to convert two simulation measures into one evaluation score. Based on this relationship, if the districting plan generated by simulated annealing algorithm is better than the best plan in the random samples, the evaluation score will be greater than 1. If it is worse than the worst plan in the samples, the evaluation score will be less than 0. In most cases, the evaluation score is between 0 and 1.

The adjusted simulated annealing algorithm, instead of minimizing the objective value of a solution, maximizes the evaluation score of districting plans. Correspondingly, the probability of inferior solution acceptance $p$ in the simulated annealing algorithm is given by: $p = \exp((v(s) - v(s_0))/T)$
where $v(s)$ is the evaluation score of the new solution $s$ while $v(s_0)$ is the evaluation score of the current solution $s_0$. $t$ is the temperature of the current iteration. The parameters in the simulated annealing are set in the following way. In step 2 of the algorithm, if the cumulative CFS probability of a district is greater than 0.16, this district is considered as “bad” district and will be cut into two smaller districts. We set this upper bound to be greater than 0.125, which is the balanced CFS probability for 8 districts. For compactness, if the ratio of the square root of a district’s area to its longest Euclidean path is greater than 1.5, this district will be “bad” district. These parameters can be configured based on practical requirements and considerations of police departments of different cities. The initial and the ending temperatures are determined by a temperature experiment, which is similar to the procedure described in [6]. In the experiment, the temperature is kept constant and simulated annealing algorithm runs for a large number of iterations. The average evaluation score is calculated for all the new districting plans generated in the process. Repeat this procedure for different levels of temperatures. The annealing curve in Figure [6] shows the increasing trend of average evaluation score when the temperature changes from high to low. It can also be seen that the average score at temperature 1000 and 10000 are slightly higher than 100. Since we desire more randomness on the prospective solutions, 100 is a better choice of the initial temperature. Similarly, since the evaluation score at temperature 1e-6 is slightly higher than 1e-7, the ending temperature can be set as 1e-6.

In the simulated annealing process, temperature is exponentially reduced from 100 to 1e-6. Experimental result shows that exponential reduction function is better than the linear one. The starting districting plan is selected from the 3000 random plans generated by the parameterized districting algorithm. A very bad plan is selected as the starting plan to show the robustness of the algorithm. It has very bad performance on average response time and workload variation, whose evaluation score is only 0.066. The simulated algorithm runs for 3000 iterations using 4 hours in a personal computer. Most of the time is spent on simulation evaluation of plans and the process of changing a current solution is relatively quick. The trends of evaluation scores of new plans, current plans and best plans in the simulated process are shown in Figure [7]. It can be seen that the evaluation score of best plan increases very quickly at the first several iterations and has a stable improvement in the rest of the process. The reason why the algorithm can jump out of the bad plans very quickly is that the algorithm cuts large districts into small districts and very small districts are merged into median-sized districts so that the CFS probabilities of districts are balanced. Districting plans that has lower variation of CFS probabilities among districts tend to have better performances measures. The final best plan has a evaluation score that is greater than 1, which is better than the best plan in the set of random plans. The current plan changes in a highly random manner in the first half of the process due to the high temperature and high probability of accepting inferior solutions. The current plan tends to be stable in the rest of the process because the temperature gradually decreases which lowers the acceptance probability. The evaluation score of new plans shows drastic fluctuation in the whole process because the solution neighborhood is defined to make relatively big changes to current plan.

The computer program records all the districting plans generated in the simulated annealing process. Figure [8] shows the top 4 plans after ranking them by the evaluation score. It can be seen that they all have similar patterns. Three small districts are near the downtown area of the city where the CFS probability is relatively high and the other five districts are on the periphery of the city where the CFS probability is lower than the central region. This pattern of districts found by simulated annealing algorithm is very similar to the one found in previous study [24]. Compared with the districting plan generated in the old simulated annealing approach [6], the compactness of some districts is not very good and the boundaries of districts are not very smooth and sometimes convex. However, the top plans generated by new algorithm can provide general structure of districting plans for the po-
lice departments. The final police patrol districts can be drawn by replacing grid boundaries by nearest roads or boundaries of existing geographical units such as police beats or census blocks.

4.2. Comparative Study

Figure 9 shows the comparison of districting plans generated by the adjusted simulated annealing algorithm, response surface optimization approach and parameterized districting plans. Each point represents a districting plan which is evaluated by the same discrete-event simulation. Since the goal is to minimize both average response time and workload standard deviation, the points on the left bottom part of the graph represent better districting plans. The blue points are the 3000 plans generated by randomizing districting parameters of the parameterized districting plans [24]. These plans include very good and very bad plans. The performance measures of them have a large range so that the random search is very inefficient. The green points are districting plans generated by a response surface optimization methodology. An iterative search procedure is conducted in a series of simulation experiments on districting parameters in parameterized districting algorithm. In this procedure, important districting parameters are identified and optimized by experimental design and response surface methodology. Using improved districting parameters, the range of performances is relatively small and in the top proportion of random plans. So, the response surface optimization method can generate good districting plans efficiently. The red points represent plans generated in the adjusted simulated annealing approach. The range of performance of these plans is much smaller than the random plans but is bigger than that in response surface method. So, the search efficiency of the response surface method is better than the simulated annealing approach. But it should be noticed that the searching procedure in response surface method is a iterative process. One iteration of experimental design depends on the statistical analysis of simulation output of the previous step. On the contrary, the simulated annealing algorithm searches districting plans in an automatic manner. The user only needs to set the parameters of the simulated annealing algorithm and let the program run. After the program is running for a pre-specified number of iterations, districting plans can be generated automatically. There is no need to analyze the intermediate result in the middle of the process. In addition, Figure 9 shows that the optimal plans in simulated annealing algorithm are slightly better than those in response surface method. More red points are in the Pareto frontier of plans. Therefore, we conclude that the adjusted simulated annealing approach is better for practical use than the response surface optimization method.

5. CONCLUSION

In this study, a simulated annealing searching algorithm is proposed to find optimal or near-optimal police patrol districting plans. The police patrol discrete-event simulation is used to evaluate the average response time and workload variation among districts. The adjusted approach defines the solution neighborhood in a new way. Instead of changing the assignment of only one atom, the new approach makes relatively big changes to current solution through a cutting and merging process of districts in current district plan. The experimental result of the Charlottesville case study shows that the new approach uses fewer iterations to reach good solutions, which is well-suited for discrete-event simulation evaluation of districting plans. Another benefit of the new approach is the robustness for the connectivity and adjacency pattern of atoms. The old method’s convexity constraint works well for atoms in the form of “R-Districts” in the Buffalo case but it is too strong and cannot work well for atoms in grid network where atoms have relatively fewer adjacent neighbors. The consequence is that the current plan changes very little after many iterations. Small adjustment on the convexity constraint does not work well either. In the new approach, the adjacency pattern of atoms has little influence on the algorithm. The adjusted simulated annealing approach is also compared with the response surface optimization method based on a parameterized districting algorithm in previous study. Experimental results show that the simulated annealing approach searches solutions over a bigger range of districting plans. It is an automatic procedure and the user does not need to conduct statistical analysis during the process. Thus, it is better for practical application. Experimental results also show that it can generate better plans than response surface method.

REFERENCES


