A Decision Support System for Tobacco Distribution Partition Optimization Based on Immune Co-Evolutionary Algorithm

Zhi-Hua Hu, Bin Yang, You-Fang Huang
Logistics Research Center, Shanghai Maritime University, Shanghai 200135, China
Email: zhihuah0115@yahoo.com.cn

***Abstract*—The tobacco distribution in China is organized by the tobacco company in the unit of city. The distribution cycle is commonly a week with five fixed distribution districts started from the center depot, and the routes in each district are fixed for the vehicles and drivers. This method with low efficiency and high cost has continued for several decades because of the poor technology and economic reasons. In this paper, we propose a feasible and optimal method for the tobacco distribution partition balance problem (TDPBP) by breaking the fixed partitions. An immune co-evolutionary algorithm (ICEA) is proposed to search the optimal partitions. Moreover, the decision support system (DSS) for partition balance is designed. Linfen city in China as a real-world case is proved that the DSS can discover the efficient distribution planning. The comparisons among three solutions, 'Fixed', 'the DSS' and 'Pure VRP', further prove that the DSS is an effective decision support tool for TDPBP.**

***Index Terms*—decision support system, partition optimization, vehicle routing problem, immune co-evolutionary algorithm, Geographic Information System**

I. INTRODUCTION

In this study we focus on a DSS aiming at TDPBP in China. TDPBP with Chinese specific characters is seldom studied before. In China, most cities still adopt fixed districts and fixed routes to organize tobacco distribution. The distribution cycle is five workdays of a week hence the region of city is correspondingly divided into 5 districts. Although the old method dealing with fixed districts is convenient and simple, the efficiency is low and the distribution cost is high because of unbalanced workload. It is urgent to break the fixed districts by optimized partitioning methods. The proposed DSS helps the decision makers (DMs) to build optimal balanced partitions.

In logistics system, partitioning is also a technique for vehicle routing problem (VRP) to choose targets for a single tour. A lot of attention has been given to the problem of determining efficient routes within a given district. However, more significant long-term savings can be achieved if the borders of the districts are optimally determined. Although VRP is widely studied, the studies are limited to the ideal parameters and the objectives. In particular, it is assumed that all parameters are given and the objectives are concrete, such as the total cost. And the VRP scheduling methods produce different routes passing though different customers for every execution. 'Pure VRP' increases the cost to organize the transportation and delivery. Therefore, a periodic balanced partition is a better choice. For each districts, the vehicles, distance and time are minimized and balanced. Moreover, the district should form a better geographical shape without overlapping by each other.

The remaining sections are organized as follows. In Section II, the background knowledge including related literatures and the problem is introduced. In Section III, the architecture of the DSS is proposed. In Section IV, Linfen city is studied to show the decision processes and the comparison shows the effectiveness of the DSS. Finally, conclusions and future researches are discussed in Section V.

II. BACKGROUND

A. Literature Review

a. Partitioning optimization

Partitioning process divides a region into districts or assigns the detailers to different clusters to achieve some minimal/maximal objectives and balance the workload. Over the last four decades, some researchers from different fields have developed models, algorithms and applications concerning the techniques to group the elementary units of territory into larger districts. Typical literatures are summarized as follows: electoral or political districts definition in a country [1-4]; School districting [5, 6]; defining electrical power zones [7]; jails location districting [8]; waste collection districting and routing [9]; areas definition in metropolitan Internet networks for installing hubs [10]; areas definition for manufactured and consumer goods [11]; districting of salt spreading operations [12]; districting for home care [13] and urban emergency services [14].

b. Vehicle routing problem

The common areas for routing optimization are the traveling salesman problem (TSP) [15] and the vehicle routing problem (VRP) [16-18]. They consider similar factors, such as minimizing transmitting distance, time or cost. Algorithm 'inver-over' [19] is reported to be an very effective method for TSP. Algorithm 'sweep' [20] is a popular and fast heuristic algorithms for VRP.

doi:10.4304/jcp.5.3.432-439

c. Immune algorithm

Artificial immune system (AIS) is a computational intelligence paradigm inspired by the biological immune system, and has been applied successfully to a variety of optimization problems [21-25]. Most of them are implemented based on three immune principles, the clonal selection [26], negative selection [21] and immune network [27]. Immune inspired co-evolutionary models are new mechanisms for algorithm design [28].

d. Co-evolutionary algorithm

Co-evolutionary algorithm is believed as the development of traditional evolutionary algorithms, which behaves in a complicated and counterintuitive ways [29]. Many inspirations from biology, physics, chemistry, economics, sociology, anthropology, psychology and others are adopted as co-evolutionary mechanisms [29]. And the application areas vary in an adequate broad range [28, 30, 31]. It appears to have advantages over traditional evolutionary methods in dealing with the larger searching space, non-intrinsic or complex objective measures and searching space with complex structures.

e. Multi-criteria decision making

A general MCDM model with competing objectives defined as functions of decision variable set X, can be represented as Eq. (1):

$$\text{Minimize:} \quad f = \left(f_1(x), f_1(x), \cdots, f_k(x)\right) \in Y \tag{1}$$

$$\text{Subject to:} \quad g_i(x) \le 0, i = 1,2,\cdots,m$$

$$h_i(x) = 0, i = 1,2,\cdots,p$$

$$x_i^L \le x_i \le x_i^U, i = 1,2,\cdots,n$$

Where, $x = (x_1, x_2, \cdots, x_n) \in X$ is the decision vector, $f(x)$ is the objective vector, X denotes the decision space and Y the objective space. MCDM uses the concept of domination referred as Pareto optimality. For the problem defined in Eq. (1), considering two decision vectors, $a, b \in X$, a dominates b (denoted as $a \prec b$, or else denoted as $a \prec^c b$) if and only if Eq. (2) is satisfied:

$$(\forall i)\left(f_i(a) \le f_i(b)\right) \wedge (\exists i)\left(f_i(a) < f_i(b)\right) \tag{2}$$

A decision vector which is not dominated by any other decision vector is called *Non-Dominated* or *Pareto Optimal*. The family of all non-dominated solutions is denoted as *Pareto-Optimal Set* (*Pareto set*) or *Pareto-Optimal Front*.

B. The TDPBP in China

a. The current situations

In most cities of China, the tobacco distribution is organized in the unit of city and the city area is divided into five districts. The districts are fixed and partitioned by political districts because of poor transportation condition and unavailable GIS technology.

Another situation is that a few companies have been set up VRP algorithms based systems for distribution scheduling. However the solution fails to solve the problem and the DMs fail to control the distribution time and managerial cost. The disadvantages of 'real time VRP' are prominent: (1) the ultimate 'dynamic' scheduling increase the time to find routes and locate detailers for drivers so that the real working time exceeds greatly the eight hours standard; (2) the managerial cost is increased; (3) the service quality to detailers is decreased.

Therefore, both the "fixed" approach and the "dynamic VRP" scheduling fail to solve the problem optimally, hence the balanced partition method is proposed as a solution.

b. Multi-criteria decision making

TDPBP must take into account the different constraints and objectives. In TDPBP, three important criteria must be optimized: (1) number of routes; (2) traveling distance of all routes; (3) working time of all routes. And four balance objectives among districts must be satisfied: (1) routes number; (2) traveling distance of all routes; (3) working time of all routes; (4) total tobacco demands.

In another aspect, the balanced districts should support effective routing, which is measured by number of tours, traveling distance and time, and complied with capacity and time restrictions, etc.. In the managerial and operational view the districts should be 'geographically' compact and not overlapped with each other.

Although the partition approach breaks the fix districts and routes, it should not change them too frequently. The balanced districts should be so robust that each district can resist minor changes of the number of detailers, the requirements and other operational principles or elements.

III. The Decision Support System for TDPBP

A. The Multi-Criteria Decision Making Model

The road map of the city is modeled as a graph from GIS. The partition problem is then to group road nodes into clusters, and each cluster is corresponding to a district. The elementary unit is therefore the vertex of the graph, while a pair of contiguous elementary units defines an edge of the graph or a road on the map. The basic information can be collected by GIS system.

Only the vertices with associated detailers are considered. Therefore, the graph of original road map is simplified as $G = (V, E)$, where $V = \{v_0, v_1, v_2, \cdots, v_{P_v}\}$ represents the set of vertices with associated detailers and $E = \{e_1, e_2, \cdots, e_{P_e}\} \subseteq V \times V$ represents the set of edges, where $e_k =< v_i, v_j >$ represents two vertices v_i and v_j, $v_i, v_j \in V$, v_0 is the depot. $Q = \{q_1, q_2, \cdots, q_{P_v}\}$ represents the tobacco requirements of vertices summed from detailers. Another vector $N = \{n_1, n_2, \cdots, n_{P_v}\}$ is the number of detailers of vertex. The unloading time of each detailer is defined by $P_{unloadt}$. On the arc set E, two property vectors are defined to represent the length and time of the arcs: $L = \{l_1, l_2, \cdots, l_{P_e}\}$ and $T = \{t_1, t_2, \cdots, t_{P_e}\}$.

A cluster matrix is introduced to represent the relation between vertex and district. In the problem, the number of districts is $P_d = 5$, and the districts is defined by $D = \{d_1, d_2, \cdots, d_{P_d}\}$. Therefore, the cluster matrix can be defined as $C = [c_{ij}]_{P_d \times P_v}$, where if v_j belongs to district d_i , $c_{ij} = 1$, else $c_{ij} = 0$. Every vertex belongs to only one cluster. Therefore, d_i can be defined as a set of v_j :

$$d_i = \{v_j | c_{ij} = 1, v_j \in V, c_{ij} \in C\}.$$

For each district, the vehicle routing procedure is performed to get the distribution solutions. The capacity of the vehicles is uniform and defined by $P_{capacity}$ and the working time limit of one tour is set to $P_{worktime}$. For each district, the optimized routes set is denoted as $R_i = \{r_1^i, r_2^i, \cdots\}$. For each route, the time on road, the unloading time, the served detailers, the route length and tour time can be computed by Q, N, L, T and $P_{unloadt}$. $Q(r_j^i)$, $T(r_j^i)$ and $L(r_j^i)$ represent the loading capacity, working time and tour distance. Therefore, for district d_i , the values can be defined and calculated: (1) the loaded tobacco $Q(d_i) = \sum_{r_j^i \in R_i} Q(r_j^i)$ is the total demands of the vertices; (2) the required vehicles or the number of routes $R(d_i) = |R_i|$; (3) the working time $T(d_i) = \sum_{r_j^i \in R_i} T(r_j^i)$ is the work time of all routes including time on the road and unloading; (4) the total distance $L(d_i) = \sum_{r_j^i \in R_i} L(r_j^i)$ is the sum of all routes' length. And the routes number, total traveling distance and work time of the partition D are denoted by $R(D)$, $L(D)$ and $T(D)$.

Based on the above definitions, the partition balancing is defined as Eq. (3):

$$Minimize\ f = (f_1, f_2, \cdots, f_7) \quad (3)$$

(1) $f_1 = R(D) = \sum_{d_i \in D} R(d_i) = \sum_{d_i \in D} |R_i|$

(2) $f_2 = L(D) = \sum_{d_i \in D} L(d_i) = \sum_{d_i \in D} \sum_{r_j^i \in R_i} L(r_j^i)$

(3) $f_3 = T(D) = \sum_{d_i \in D} T(d_i) = \sum_{d_i \in D} \sum_{r_j^i \in R_i} T(r_j^i)$

(4) $f_4 = Std(R(d_i)) = Std_{d_i \in D}(R(d_i)) = Std_{d_i \in D}(|R_i|)$

(5) $f_5 = Std(L(d_i)) = Std_{d_i \in D}(L(d_i)) = Std_{d_i \in D}\left(\sum_{r_j^i \in R_i} L(r_j^i)\right)$

(6) $f_6 = Std(T(d_i)) = Std_{d_i \in D}(T(d_i)) = Std_{d_i \in D}\left(\sum_{r_j^i \in R_i} T(r_j^i)\right)$

(7) $f_7 = Std(Q(d_i)) = Std_{d_i \in D}(Q(d_i)) = Std_{d_i \in D}\left(\sum_{r_j^i \in R_i} Q(r_j^i)\right)$

Subject to

$V = \{v_0, v_1, v_2, \cdots, v_{P_v}\}$,

(1) $N = \{n_1, n_2, \cdots, n_{P_v}\}$,

$Q = \{q_1, q_2, \cdots, q_{P1}\}$

$E = \{e_1, e_2, \cdots, e_{P_e}\} \subseteq V \times V$,

(2) $L = \{l_1, l_2, \cdots, l_{P_e}\}$,

$T = \{t_1, t_2, \cdots, t_{P_e}\}$

(3) $C = [c_{ij}]_{P_d \times P_v}, \sum_{d_i \in D} c_{ij} = 1, v_j \in V$

(4) $D = \{d_1, d_2, \cdots, d_{P_d}\}, d_i = \{v_j | c_{ij} = 1, v_j \in V, c_{ij} \in C\}$

(5) $Q(r_j^i) \le p_{capacity}, r_j^i \in R_i, d_i \in D$

(6) $T(r_j^i) \le p_{worktime}, r_j^i \in R_i, d_i \in D$

B. ICEA Based Optimization Approach

The model in the previous section is a multi-objective optimization problem (MOOP). We design ICEA for TDPBP. Through native functions and the cooperative cytokine networks, the immune system is dynamically stabilized. In the following, the native function models including clonal selection, negative selection and immune memory, the cooperative searching techniques including cooperative variation, balanced variation, and cooperative communication techniques including feedback model are designed in the proposed algorithm.

a. General framework of ICEA for TDPBP

The general framework of ICEA for TDPBP is defined in Algorithm 1, which shows two main stages: the construction of initial partitions (Step 1 and Step 2) and the cooperative searching among districts (Step 3~7).

Algorithm 1. ICEA for TDPBP

Step 1: Generated balanced sample point for each district;

Step 2: Cooperative construction of districts with balanced demand;

Step 3: Perform clonal selection on superior partition solutions;

Step 4: Proliferate to generate new partition;
Perform variance by affinity measures on the districts;

Step 5: Deal with the new solution: elimination or accepted by immune memory;

Step 6: Use feedback: **if** Pareto optimal solution is produced, **Goto Step 4**

Step 7: **If** termination conditions are not satisfied, **Goto Step 3**.

b. Construction of balanced initial partitions

In this stage, the initial balanced district partition set (SET_{Pi}) is generated. The set size is set: $P_{initsets}$. The element in the set is $D_{Pi} = \{d_1, d_2, \cdots, d_{P_d}\}$, where $|d_i| = 1$, d_i contains one vertex, called sample point of the district. For only one vertex in d_i, v_{d_i} is used to represent the vertex in d_i. Then, on d_i a measure $S(d_i)$ is defined by $S(d_i) = \min_{d_j \ne d_i}(l_{v_{d_i} v_{d_j}})$. $S(d_i)$ is the shorted distance from d_i to other components in D_{Pi} . The balance of the sample points is defined by the maximization and balance of distance among them. The distance of D_{Pi} is $L(D_{Pi}) = \sum_{d \in D_{P_i}} S(d)$, and the balance of distance is measured by $B(D_{Pi}) = STD_{d \in D_{P_i}}(S(d))$.

Then, the left vertices are chosen to add to the initial districts. Inspired by immune system, immune unit is introduced and affinity between vertices is defined. In immune system, some immune cells will gather together to form a group to detect or destroy the antigen. As an intuitive rule, the passing vertices from the depot to the far vertex should be scheduled in the same route with the vertex. Then, the vertices in the same shortest path form the depot are defined to construct an immune unit. The affinity between the vertices in the unit is high and set 1. We define this type of affinity between vertices in the same unit as Unit Affinity. Then, a matrix is defined to present the Unit Affinity: $A_{unit}=\left[a_{ij}\right]_{P_v\times P_v}$, if v_i and v_j are in the shortest path from the depot v_0, $a_{ij}=1$, else $a_{ij}=0$. In another aspect, if two vertices positioned nearer tends to belong to the same district. Distance Affinity ($A_{distance}$) is defined to present this relation: $A_{distance}=\left[a_{ij}\right]_{P_v\times P_v}=\left[1/\left(1+l_{ij}\right)\right]_{P_v\times P_v}$. The two affinity measures are adopted for cooperative district construction, as shown in Algorithm 2.

Algorithm 2. Construction of balanced initial districts

Step1: Generated balanced sample point for each district for P_{geni} times;

While $|SET_{Pi}|<p_{initsets}$ and less than P_{geni} times

Randomly generate a new $D_{Pi}=\{d_1,d_2,\cdots,d_{P_d}\},|d_i|=1$;

If $\left(L(D_{Pi}),B(D_{Pi})\right)$ is a Pareto optimal solution to $D\in SET_{Pi}$

$D_{Pi}\to SET_{Pi}$, D_{Pi} is added to SET_{Pi};

Eliminate the existing dominated ones;

End

End

Step2: Cooperative construction of districts with balanced demands.

For each $D_{Pi}\in SET_{Pi}$

While there is undistracted vertex

Computer $Q(d_i)$ on D_{Pi} and choose district d with the lowest value;

Choose the vertex v with the highest $A_{distance}$ to d;

$v\to d$: Add v to d;

Add the vertices v' with $\left(A_{unit}(v,v')=1\right)\wedge\left(v'\notin D\right)$ to d;

End

c. Cooperative searching among districts

In the first stage, the initial districts with balanced demands are constructed, whereas other objectives ($f_1\sim f_7$) are not optimized. In the second stage, all objectives and constraints are considered to generate Pareto optimal solution.

Several concepts and procedures inspired by immune system are employed to design the cooperative searching algorithm in Algorithm 3. The partition solutions initialized from the first stages are used to construct the partition reservoir, denoted as $SET_P\leftarrow SET_{Pi}$ (Immune Memory). The superior partitions generated later are added to it. And the procedure of immune clonal selection is performed on the superior ones in SET_P to generate new ones. The ratio of clonal proliferation is P_{clonal}. The new superior ones update SET_P. The proliferation is followed by immune variation to change the cell structure, the components of the partition. District Affinity ($A_{district}$) is defined to present the relation between one vertex and the district: $A_{district}(v,d)=\sum_{v'\in V}\left(A_{distance}(v,v')+A_{unit}(v,v')\right)$. The vertex with lower $A_{district}$ has the higher probability ($P_{exclude}(v,d)=1/\left(1+A_{district}(v,d)\right)$) to be excluded. The districts cooperate to achieve the optimal balance status. Whereas there are three criteria to be minimized and four to be balanced, these criteria compete to further the optimization process. If a vertex with lower $A_{district}$ is excluded from a district and absorbed by another district, the relation between new partition (n) and original (o) has three types: (1) $f_n\prec f_o$, the new partition dominates the original one; (2) $f_o\prec f_n$, the original one dominates the new one; (3) $f_n\prec^c f_o$, the un-dominated solution is generated. Correspondingly, there are three strategies.

Algorithm 3. Cooperative searching among districts

Step 1: Clonal Selection for P_{gens} times

Choose a Pareto optimal partition solution from SET_P

Step 2: Proliferation and variance for P_{clonal} times:

Step 2.1: Randomly choose the objective $f_i, i\in\{1,2,\cdots,7\}$;

Step 2.1: Choose the district d^+ with highest value on f_i;

Step 2.3: Choose the $v\in d^+$ with the highest $P_{exclude}$;

Step 2.4: Choose another d^- with the higher $A_{exclude}$ to v;

Step 2.5: Move v from d^+ to d^- to construct a new partition;

Step 2.6: **If** $f_n\prec f_o$: the new partition replace the original one; **Goto Step 1**;

If $f_o\prec f_n$: the new partition is discarded; **Goto Step 2**;

If $f_n\prec^c f_o$: the new partition is added to SET_P; **Goto Step 3**;

Step 3 Clonal elimination: find the dominated solutions in SET_P and eliminate them

Step 4: **If** the termination condition is not satisfied, **Goto Step 1**

C. Architecture of the DSS

The solution is based on the historical sales data and a professional GIS based system. The problem is model as a MCDM model and its Pareto optimal solutions are searched by ICEA. DMs use the GIS based system to choose the optimal partitions and make decisions.

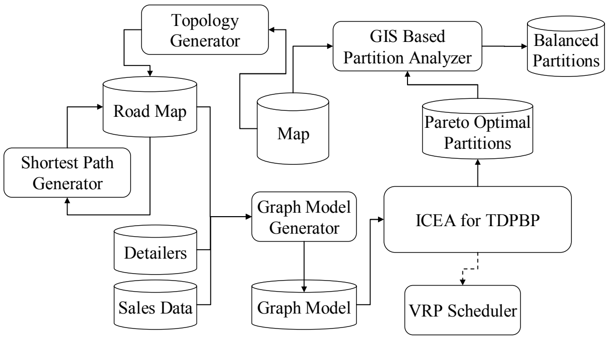


Figure 1. The architecture of the DSS

The DSS is composed of six main procedures, as shown in Figure 1.

(1) Topology Generator loads the road layers by the GIS engine and then attracts roads to generate crossing nodes and road segments.

(2) Shortest Path Generator generates roads between all pairs of road nodes at a single run based on the faster shortest path algorithms (e.g. 'Floyd').

(3) Graph Model Generator generates a graph based on three types of data: road map, detailers and sales data.

(4) ICEA is used to solve the MCDM model.

(5) VRP Scheduler is implemented based on the algorithm 'sweep', the result of which is then improved by a TSP algorithm 'inver-over'.

(6) GIS Based Partition Analyzer represents the generated balanced partitions to DMs.

In the implementation of the DSS, the trivial version of MapInfo Xtreme 2005 for Windows is used as the GIS Map engine.

IV. REAL-WORD CASE STUDY

A. Case Description and the Problem Scale

As a demonstration, eight political districts of Linfen city in China distributed from the center depot were divided into five districts. The tobacco distribution center owned *30* vehicles with the capacity of *4500* to delivery about *500,000* tobaccos to more than *10,000* detailers through the fixed districts and routes in every week. However, the loading ratio of the vehicles was lower than *75%*. And the demands of districts were not balanced so that the workload in a week waved in a wide range. The working time of the drivers was hard to control varying from *3* to *12* hours. The pre-processing and statistic results representing the scale of the problem are summarized in Table 1.

TABLE 1. PRE-PROCESSING AND STATISTIC RESULTS

Pre-processing or statistic items	Results
Radius (center is the depot) (*km*)	130
Road segments (arcs)	3026
road crossing nodes (vertices)	2184
P_v: Road crossing nodes associated with detailers	814
$N(D)$: Detailers number	10080
$Q(D)$: Average tobacco demand/Week	472509

B. Parameters Setting

The parameters are summarized in Table 2. Corresponding to five workdays of a week, the city areas are planed to be partitioned into five balanced districts. All vertices are connected bi-directionally. For a partition, VRP scheduling for every district of the partition solution is done by the fast 'sweep' algorithm [20] followed by an effective TSP optimization algorithm [19] for each generated tour. These two basic algorithms show promising performance and indeed ensure the entire performance of partition balancing.

TABLE 2. THE PARAMETERS SETTING

Parameter	Corresponding value
P_d : the number of districts	5
P_v : numbers of vertices	814
P_e : number of arcs	661782
$P_{capacity}$: capacity of vehicle	4500
$P_{worktime}$: working time limit for a tour (hour)	8
$p_{unloadt}$: unloading time per detailer (minute)	3
$P_{initsets}$: size of the initial balanced partition set, DSi	10
P_{geni} : generations for initial partitions searching	1000
P_{gens} : generations for clonal selection and proliferation	2000
P_{clonal} : proliferation copies	7
VRP scheduling approach	'Sweep' algorithm
TSP improvement method	'Inver-over' algorithm

C. Multi-Criteria Decision Making Processes

a. Pareto optimal partitions generation

The Pareto optimal partitions are generated by ICEA, which can be separated into two stages: (1) initialization: cooperative search of initial partitions with only five sample points; (2) immune searching: cooperative search for Pareto optimal partitions.

In these two stages, the variances of the initial partitions and Pareto partitions show the evolutionary processes of the algorithm. After the searching, only the non-dominated partitions are left, which is "Pareto selection". In the real application, the Pareto partitions are also abundant, the parameters of previous processes are readjusted or additional rules are employed to reduce the solutions for the final purpose of decision supporting. In this study, these rules are employed: (1) the shape of the districts and the routes, (2) the relations between districts and routes, and the geography conditions including valleys, rivers and hills. The process by heuristic rules and GIS based rules is named by "Reduction by Domain Knowledge". Finally, in the "interactive decision" (see Section IV.C.c), the system assists the DMs to include or exclude the solutions. In Figure 2 the varying of partitions and Pareto partitions are shown based on a 30 times test. In the 'Initialization' state, about 1000 partitions only with sample points are generated to search the optimal 10 solutions. In the

'Immune searching' stage, because of the 2000 clonal proliferation generations and the 7 clonal copies settings, more than 10,000 non-dominated partitions are generated, whereas most of them are dominated by the new ones except about 3,000 partitions as a average. At the end of co-evolutionary searching, only about 1,000 Pareto optimal partitions are left for interactive decision. After "Reduction by Domain Knowledge", about 20 partitions are chosen for the final decision. After the final interactive decision only one balanced partition is left.

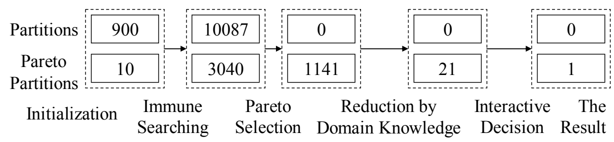


Figure 2. Partitions in different stages

The main body of the proposed algorithm is the second stage. A large number of partitions are generated by cooperative searching among districts of superior partitions. In Figure 3, the varying curves of the five objectives are shown. The other two objective including the total and standard deviation of tours number have little prominent varying. In order to show the tendencies in one figure, the value of f_2, f_3 and f_6 are adjusted by multiplying constants so that the curve shapes are kept but more apparent. All the five objectives endure a un-restrict decreasing curve to achieve better solutions. The curve of STD of demand drops apparently because in the initiation stage the A_{unit} is utilized for vertices agglutination. The varying of time and length is not so synchronous for the different speed of roads. From Figure 3 it can be drawn that the proposed method is an effective solution for the multi-criteria partition balancing problem. In Table 3, the final Pareto partitions are shown with seven objectives.

b. Interactive decision making

Although ICEA can produce Pareto optimal partitions as many as possible, most of them can not satisfy the DMs and at last only one partition is chosen. In this problem, there are rules can help DMs to make decision: (1) district, routes, vertices and detailers presenting tools; (2) queries between districts and other map layers; (3) interactive detail adjusting tools (e.g. moving of specific detailers or vertices between districts), etc..

In Figure 4, a chosen balanced partition (see Figure 4(b)) is compared with the original fix partition (see Figure 4(a)).

c. The DSS via pure VRP scheduling

Why not use the VRP scheduling method directly for whole area? The reasons includes: (1) the real distribution work is organized by the unit of five workdays of a week; (2) the "real time" VRP scheduling method generates different solutions for every time so that the drivers take more time to locate the detailers and to find the roads; (3) the "absolutely" dynamic solutions increase the managerial cost for cooperation among departments. In fact, the "dynamic VRP" scheduling method is a previous fail solution in the real case. Therefore, the periodic partition balance method is proposed to solve the problem. In the real application, the optimal partition solution and the routes of each district will be fixed for 3~5 months. The new detailers are associated to the nearest vertices. When the capacity of a route exceeds the limit, the detailers of the vertex with lower affinity to the route (defined like $A_{district}$) will be moved to another route, and as a second solution a 'free' vehicle is set to deal with the excessive demands.

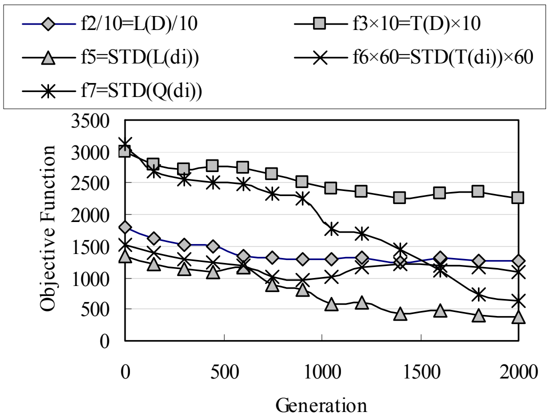
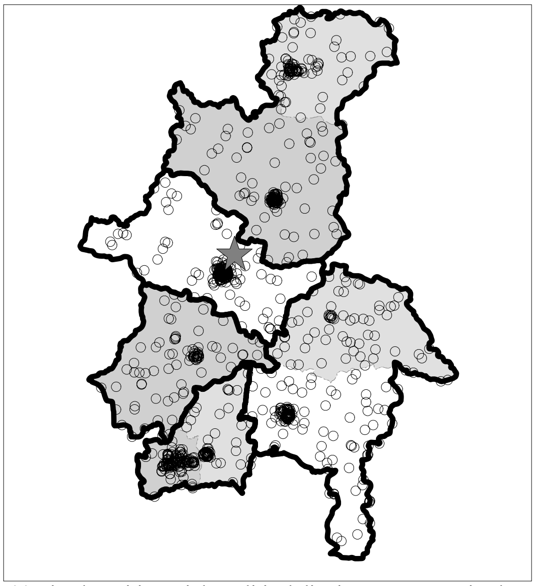


Figure 3. Co-evolutionary process of the partition set SET_P

TABLE 3. PARETO OPTIMAL PARTITIONS

No.	Partition (D)				Districts (d_i)			
	f_1	f_2	f_3		f_4	f_5	f_6	f_7
	R(D)	L(D)	T(D)		R[i]	L[i]	T[i]	Q[i]
1	115	11911	198.5		0	244	5.96	718
2	115	11897	198.2		0	268	6.25	754
3	115	11911	198.5		0	244	5.88	899
4	115	11911	198.5		0	244	5.88	899
5	116	12219	203.6		0.4472	162	3.61	3714
6	115	11897	198.2		0	167	3.76	1524
7	115	11897	198.2		0	190	4.26	1403
8	117	12366	206.1		0.5477	98	2.41	2661
9	117	12366	206.1		0.5477	102	2.63	1507
10	115	12012	200.2		0	178	4.13	1379
11	115	11911	198.5		0	244	5.88	899
12	115	11911	198.5		0	244	5.88	899
13	115	11911	198.5		0	244	5.88	899
14	116	12234	203.9		0.4472	60	1.76	3354
15	118	12224	203.7		0.5477	64	1.91	6149
16	118	12224	203.7		0.5477	82	2.06	6067
17	118	12224	203.7		0.5477	76	2.2	5660
18	118	12224	203.7		0.5477	76	2.2	5660
19	118	12224	203.7		0.5477	96	2.73	4386
20	117	12655	210.9		0.5477	103	2.13	2891
21	117	11948	199.1		0.5477	95	2.13	2008

Note: $A[i] = Std\left(A\left(d_i\right)\right), A \in \{S, R, T, Q\}$



(a) Fixed partition: eight political districts are reorganized as five districts.

(b) Balanced partition: the balanced districts break the fixed partition; every district looks like fan sector with the depot as the center.

Figure 4. The fixed partition and balanced partition: the asterisk is the depot, little circle presents detailer, bold border surrounded region is district.

In Table 4, three distribution methods are compared: (1) “Fixed”: fixed districts and routes; (2) “The DSS”: periodic balanced districts and routes; (3) “Pure VRP”: dynamic routes by VRP scheduling method as a whole. Another measure “adjust time” is introduced to represent the time induced by the new or closed detailers. The data in Table 4 are the summary of real runtime data. Although the “Pure VRP” method achieves the best performance in total routes, load ratio and total time, however the adjust time is five times of the “Fixed” method for the dynamic routes, which make the maximal tour time exceeds the regular working time greatly. “The DSS” balanced partition method improves the “Fixed” method and can satisfy the other constraints for its adaptive ability to the minor changes. In the managerial and administrative view, the periodic balanced partition does not increase the managerial and distribution cost, and it accelerates the optimization of other managerial goals. In this view, the proposed DSS for partition balance optimization is valid and effective.

TABLE 4. COMPARISONS OF PLANNING METHODS

Planning methods	Fixed	Periodic	Dynamic
Total routes	150	117	110
Load ratio	70%	90%	95%
Time on road (h)	310	221	190
Adjust time (h)	54	107	300
Tour time (h)	3~5.12~10	3~7.1~8	3~9.03~11.5

ACKNOWLEDGMENT

This work was supported in part by the high-tech research and development program of China (No. 2007AA04Z105), National Natural Science Foundation of China (No. 70871075), Shanghai Education Committee Research Project (No. 2008079), Shanghai Science Commission Capability Construction Project (No. 071705107), Shanghai Municipal Economic Commission Absorption of Introduced Technologies and Innovative Annual Planning Project (No. 07XI-058) and Shanghai Municipal Science Commission Local University Capability Project (No. 08170511300).

V. CONCLUSION

In this paper, we propose a DSS for TDPBP. The architecture of the DSS with six modules is studied in details. The partition balancing of Linfen city in Shanxi province for tobacco distribution as a real-world case is studied. The comparison studies show the effect of the proposed DSS.

As for future suggestions, the study generates two important points. First, the simplification and approximation of the problem may help to achieve better performance and decrease the decision time. Second, the stochastic simulation method can be introduced for the next generation of the DSS.

REFERENCES

[1] M. Hojati, "Optimal political districting," *Computers and Operations Research*, vol. 23, pp. 1147-1161, 1996.

[2] A. Mehrotra, E. L. Johnson, and G. L. Nemhauser, "An optimization based heuristic for political districting," *Management Science*, vol. 44, pp. 1100-1114, 1998.

[3] B. Bozkaya, E. Erkut, and G. Laporte, "A Tabu search heuristic and adaptive memory procedure for political districting," *European Journal of Operational Research*, vol. 144, pp. 12-26, 2003.

[4] C.-I. Chou and S.-P. Li, "Spin systems and political districting problem," *Journal of Magnetism and Magnetic Materials*, vol. 310, pp. 2889-2891, 2007.

[5] T. Tominaga and Y. Sadahiro, "Evaluation of school family system using GIS," *Geographical Review of Japan, Series A*, vol. 76, pp. 743-758, 2003.

[6] F. Caro, T. Shirabe, M. Guignard, and A. Weintraub, "School redistricting: embedding GIS tools with integer programming," *Journal of the Operational Research Society*, vol. 55, pp. 836-849, 2004.
[7] P. K. Bergey, C. T. Ragsdale, and M. Hoskote, "A decision support system for the electrical power districting problem," *Decision Support Systems*, vol. 36, pp. 1-17, 2003.
[8] V. Marianov and F. Fresard, "A procedure for the strategic planning of locations, capacities and districting of jails: Application to Chile," *Journal of the Operational Research Society*, vol. 56, pp. 244-251, 2005.
[9] J. W. Male and J. C. Liebman, "Districting and routing for waste collection," *Journal of the Environmental Engineering Division ASCE*, vol. 104, pp. 1-14, 1978.
[10] K. Park, K. Lee, S. Park, and H. Lee, "Telecommunication node clustering with node compatibility and network survivability requirements," *Management Science*, vol. 46, pp. 363-374, 2000.
[11] B. Fleischmann and J. N. Paraschis, "Solving a large scale districting problem: a case report," *Computers and Operations Research*, vol. 15, pp. 521-533, 1988.
[12] L. Muyldermans, D. Cattrysse, D. V. Oudheusden, and T. Lotan, "Districting for salt spreading operations," *European Journal of Operational Research*, vol. 139, pp. 521-532, 2002.
[13] M. Blais, S. D. Lapierre, and G. Laporte, "Solving a home-care districting problem in an urban setting," *Journal of the Operational Research Society*, vol. 54, pp. 1141-1147, 2003.
[14] P. Bertolazzi, L. Bianco, and S. Ricciardelli, "Method for determining the optimal districting in urban emergency services," *Computers and Operations Research*, vol. 4, pp. 1-12, 1977.
[15] G. W. DePuy, R. J. Moraga, and G. E. Whitehouse, "Meta-RaPS: a simple and effective approach for solving the traveling salesman problem," *Transportation Research Part E: Logistics and Transportation Review*, vol. 41, pp. 115-130, 2005.
[16] T. Du, F. K. Wang, and P.-Y. Lu, "A real-time vehicle-dispatching system for consolidating milk runs," *Transportation Research Part E: Logistics and Transportation Review*, vol. 43, pp. 565-577, 2007.
[17] M. A. Haughton, "Assigning delivery routes to drivers under variable customer demands," *Transportation Research Part E: Logistics and Transportation Review*, vol. 43, pp. 157-172, 2007.
[18] H.-K. Chen, C.-F. Hsueh, and M.-S. Chang, "The real-time time-dependent vehicle routing problem," *Transportation Research Part E: Logistics and Transportation Review*, vol. 42, pp. 383-408, 2006.
[19] T. Gao and Z. Michalewicz, "Evolutionary algorithms for the TSP," presented at Proceedings of the 5th Parallel Problem Solving from Nature Conference, Berlin, 1998.
[20] B. E. Gillett and L. R. Miller, "A heuristic algorithm for the vehicle-dispatch problem," *Operations Research*, vol. 22, pp. 340-349, 1974.
[21] X.-B. Cao, H. Qiao, and Y.-W. Xu, "Negative selection based immune optimization," *Advances in Engineering Software*, vol. 38, pp. 649-656, 2007.
[22] Z.-H. Zhang, "Immune optimization algorithm for constrained nonlinear multiobjective optimization problems," *Applied Soft Computing Journal*, vol. 7, pp. 840-857, 2007.
[23] K. Polat and S. Güneş, "Principles component analysis, fuzzy weighting pre-processing and artificial immune recognition system based diagnostic system for diagnosis of lung cancer," *Expert Systems with Applications*, vol. 34, pp. 214-221, 2008.
[24] T.-H. Hou, C.-H. Su, and H.-Z. Chang, "Using neural networks and immune algorithms to find the optimal parameters for an IC wire bonding process," *Expert Systems with Applications*, vol. 34, pp. 427-436, 2008.
[25] Y.-S. Ding, Z.-H. Hu, and H.-B. Sun, "An antibody network inspired evolutionary framework for distributed object computing," *Information Sciences, revised*.
[26] W. Dong, G. Shi, and L. Zhang, "Immune memory clonal selection algorithms for designing stack filters," *Neurocomputing*, vol. 70, pp. 777-784, 2007.
[27] F. Freschi and M. Repetto, "VIS: An artificial immune network for multi-objective optimization," *Engineering Optimization*, vol. 38, pp. 975-996, 2006.
[28] N. Toma, S. Endo, and K. Yamada, "A study of a parallelized immune coevolutionary algorithm for division-of-labor problems," *Artificial Life and Robotics*, vol. 9, pp. 76-80, 2005.
[29] R. P. Wiegand, "An Analysis of Cooperative Coevolutionary Algorithms, Ph.D. Thesis," vol. Ph.D: George Mason University, 2004.
[30] M. J. Aitkenhead, "A co-evolving decision tree classification method," *Expert Systems with Applications*, vol. 34, pp. 18-25, 2008.
[31] T. H. Roh, K. J. Oh, and I. Han, "The collaborative filtering recommendation based on SOM cluster-indexing CBR," *Expert Systems with Applications*, vol. 25, pp. 413-423, 2003.

Zhi-Hua Hu was born in 1977. He is a researcher at Logistics Research Center of Shanghai Maritime University. Hu's present research interests are computation intelligence, immune computation, co-evolutionary computation, distributed object computing, engineering optimization, logistics and supply chain optimization.