International Journal of Management Science
Vol 10, No 2, November 2004

Optimal Routing and Uncertainty Processing using Geographical Information for e-Logistics Chain Execution

Jin Suk Kim*
Postal Technology Research Center,
ETRI, Daejeon, 305-350, Korea

Keun Ho Ryu**
School of Electrical and Computer Engineering,
Chungbuk National University, Cheongju, 361-763, Korea

(Received Jul. 2003 ; Revised Nov. 2003 ; Accepted Sep. 2004)

ABSTRACT

The integrated supply chain of business partners for e-Commerce in cyber space is defined as Logistics Chain if the cooperative activities are logistics-related. Logistics Chain could be managed effectively and efficiently by cooperative technologies of logistics chain execution.

In this paper, we propose a routing and scheduling algorithm based on the Tabu search by adding geographical information into existing constraint for pick-up and delivery process to minimize service time and cost in logistics chain. And, we also consider an uncertainty processing for the tracing of moving object to control pick-up and delivery vehicles based on GPS/GIS/ITS. Uncertainty processing is required to minimize amount of telecommunication and database on vehicles tracing.

Finally, we describe the Logistics Chain Execution (LCE) system to perform plan and control activities for postal logistics chain. To evaluate practical effects of the routing and scheduling system, we perform a pretest for the performance of the tabu search algorithm. And then we compare our result with the result of the pick-up and delivery routing plan generated manually by postmen.

Keywords: e-Logistics, Supply chain, Logistics chain, Routes and scheduling, Tabu search, Moving object, GPS/GIS/ITS, Vehicles tracing, Uncertainty processing, Spatiotemporal database

* Email: kimjs@etri.re.kr
** Email: khryu@dblab.chungbuk.ac.kr

1. INTRODUCTION

Internet changes lots of things in our life. Especially, the explosive increase of e-Commerce enables us to buy products without time and space limitation, and to pay a charge using secure electronic methods through network. e-Marketplace like shopping malls needs cooperative works among several business partners (enterprises) in such business processes as procurement of raw materials, production, sales and order management, transportation, delivery, and customer service for cost-effective and efficient services in cyber-space. The cooperative activities among business partners can be achieved more effectively and productively based on Supply Chain Management (SCM) concept. The individual activities of each business partner for e-Commerce in cyber-space are some parts of its independent supply chain activities, while the cooperative activities among business partners are those of integrated supply chain of business partners involved in e-Commerce. Figure 1.1 shows supply chain in the logistics business.

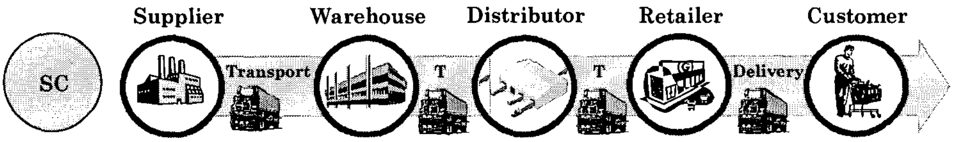


Figure 1.1. Supply chain

Particularly, the integrated supply chain of business partners is defined as "*Logistics Chain*" if the cooperative activities are logistics-related as shown in Figure 1.2. Accordingly, Logistics Chain Management (LCM) can be defined as management principle and enabling technologies for managing logistics chain effectively and efficiently.

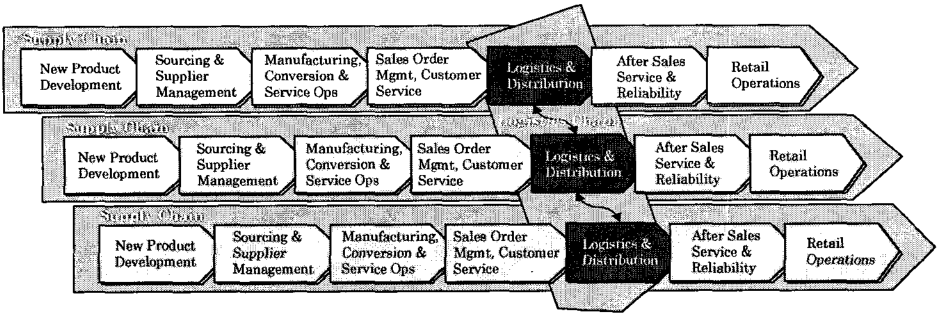


Figure 1.2. Logistics chain on supply chain

For example, if we consider the cooperation among business partners of e-Marketplace, there are series of cooperation processes to manage customer order (Order Management) of shopping mall (Seller), to manage stock of supplier (Warehouse Management), to distribute products to the destination by logistics service provider (Transport Management), and to deliver the products to the final customer (Buyer) through internet as shown in Figure 1.3. These cooperative processes can be called Logistics Chain, and we define the activities performed through internet as e-Logistics.

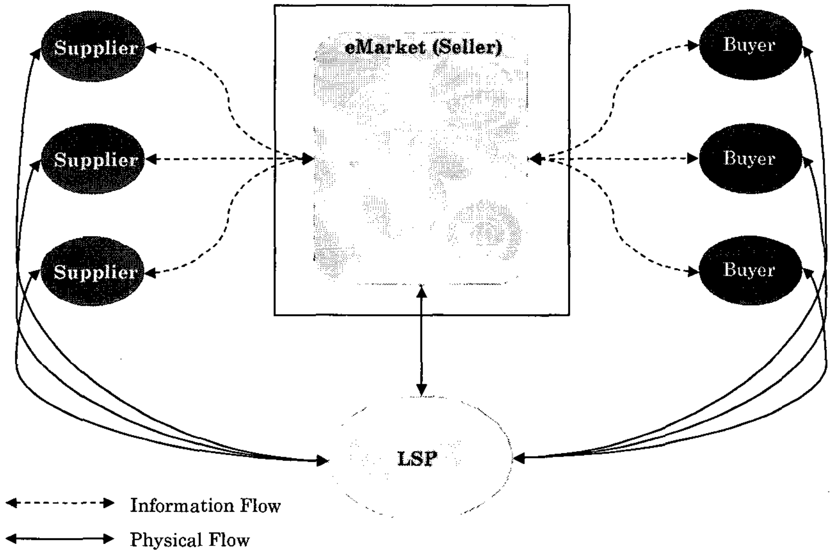


Figure 1.3. e-Logistics environment

The technical issues on e-Logistics are integration, visibility, and optimization as shown in Figure 1.4. The integration requires ebXML and workflow technology for data integration and process collaboration. The visibility is put on tracking and spatiotemporal database technology. The optimization is optimal design and routing on the logistics network.

The objective of this paper advances the Routing and Scheduling technology and Uncertainty Processing technology for e-Logistics Chain Execution to apply real world applications.

In order to develop technologies that are required to manage logistics chain of Korea Post, we develop the e-Logistics Chain Execution (e-LCE) system to plan, to execute, and to control activities in logistics chain intelligently. To perform plan and control activities for logistics chain, the e-LCE system has the pick-up and delivery planning system (as a subsystem) that uses meta heuristic algorithm (tabu search) and the moving object tracking system of delivery vehicles based on GPS/GIS/ITS information (another subsystem of e-LCE).

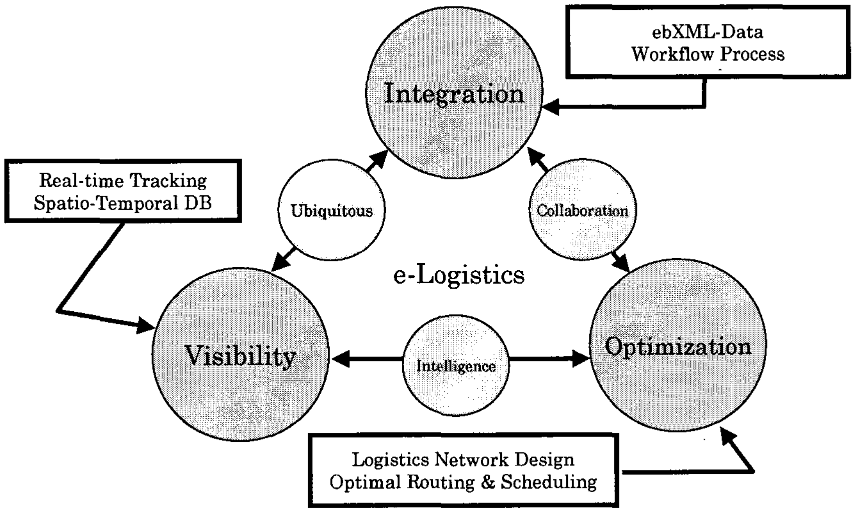


Figure 1.4. Issue on e–Logistics

In this paper, we consider a routing and scheduling problem for pick-up and delivery processes in postal logistics chain. The pick-up and delivery routes planning is to establish routes and time schedules to complete pick-up and delivery requests using several types of vehicles owned by a local post office with the objective of minimizing service time and cost. We also consider an uncertainty processing for the tracing of moving object to control pick-up and delivery vehicles.

The remainder of this paper is organized as follows. In section 2, the routing and scheduling problem is defined and the algorithm based on the tabu search for the problem is proposed. The routing and scheduling problem considered in this work includes complications such as multiple types of vehicles, multiple types of customer demands with time window, and geographical information – travel time, distance, traffic, velocity, and road type – beyond the basic vehicle routing problem. We develop a tabu search algorithm for a solution. Also, to obtain the optimal pick-up and delivery routes plan, we raise the suitability of possible solution by adding GIS/ITS information into existing constrained information.

In section 3, the uncertainty processing algorithm to manage efficiently moving object information in the database is provided to minimize amount of telecommunication and database in vehicles tracing. In the application area of the real world, the pick-up and delivery is performed with the optimal route and schedule obtained by the suggested algorithm. In this case, vehicle position is traced for the efficient operation of moving vehicle through Mobile/GPS, and the control for the actual activity is conducted by comparing with the schedule. However, the position data obtained from GPS require an amount of telecommunication and database which are difficult to be handled by information system. To solve this kind of problems, we implement the uncertainty processing algorithm on spatiotemporal database for minimizing the requirement volume of database and communication by extending time interval to acquire position data of GPS, and presuming position data between the extended time intervals.

In section 4, the architecture and implementation details of the e-LCE system are given. We describe the routes planning and tracing system of pick-up and delivery vehicles implemented as e-LCE system for postal logistics chain.

Finally, to evaluate practical effects of the routing and scheduling system, we conduct a pretest for the performance of the tabu search algorithm. The test results show that good solutions can be found within short simulation time. We also apply the system to real operation processes of a local post office, and then compare these test results with the results from the pick-up and delivery routing plan generated manually by postmen.

2. ROUTING AND SCHEDULING PLAN FOR PICK-UP AND DELIVERY

Pick-up and delivery transportation refers to the movement of product from one location to another as it makes its way from a local post office to the customer's hands(or otherwise). Pick-up and delivery plays a key role in every logistics chain because products are rarely produced and consumed in the same location [1]. Moreover, pick-up and delivery costs typically range between one-third and half of total logistics costs. Thus, improving efficiency through the maximum utilization of pick-up and delivery equipment and personnel is a major concern. To reduce pick-up and delivery costs and also to improve customer service, finding the best paths of vehicles that will minimize time or distance is a frequent decision problem [2].

In postal logistics chain, the efficient pick-up and delivery is a very important to reduce operating and investment cost and to provide the high level customer service. In this section, we describe a routing and scheduling problem related to the pick-up and delivery in postal logistics chain and propose an algorithm to solve the problem as fellows.

- Other routing and scheduling algorithms cannot be used as the engine of e-LCE system because of the algorithms focused on specific routing problems. We must generalize the VRP algorithm with the various problem types as pick-up and delivery, time windows, multiple depot and time dependent travel time.
- Since other algorithms does not consider realistic constraints for the pick-up and delivery, our algorithm must get realistic constraints as compatibility, precedence, rest time, vehicle type, customer's request, travel speed, and etc.
- The existing algorithm cannot solve a problem within reasonable time, but application in real world requires fast algorithm for every day planning.
- Above all, data generation for algorithm is difficult to be considered in real road information which is distance, traffic, crossroads, and road type.

2.1 Routing and Scheduling Problem Description

In postal logistics chain, there are various routing problems. Here, we consider a routing problem which is related to the customers directly. The problem is to find the routes for picking-up or delivering the customer's parcel.

When a depot (a local post office has the role of the scheduling to visit a customer and the pick-up or delivery of given the parcel in every day), customer's demands (pick-up or delivery) and available vehicles (identical or not) are given, we decide the assignment of the customers demands to vehicles, the visiting sequences and time schedule. Each customer may have a specified time window, defined by the earliest service time and the latest service time. We call the problem routing and scheduling problem.

Then the routing and scheduling problem can be considered as a variation of a well-known and difficult combinatorial optimization problem which is found in many practical applications connected with transportation. The VRP (Vehicle Routing Problem) is concerned with the determination of the optimal routes used by a fleet of vehicles stationed at a central depot to serve a set of customers with known demands. In the basic version of the problem all the vehicles are identical, only the capacity restrictions for the vehicles are imposed, and the objective is to minimize the total routing cost (or length) of the routes.

The routing and scheduling problem considered in this paper includes more realistic restrictions such as heterogeneous vehicles, precedence constraints between customer, combined pick-up and deliveries, time window constraints, time dependent travel times, etc. The details of major constraints are as follows.

- Multiple types of vehicles: The vehicle can be heterogeneous. Each vehicle has a distinct capacity.
- Pick-up and deliveries with time windows: Vehicles can deliver and pick-up product within a designated time window.
- Time dependent travel times: The travel time between two locations can depend on not only the distance (static view) but also the traffic volume (dynamic view) of the day.
- Compatibility and precedence constraints: Customer-vehicle compatibility constraints can restrict the set of customers that a vehicle can service. Depot-customer compatibility constraints can restrict the set of customers that a depot can service. Customer-customer precedence constraints can impose a partial ordering on the customer sequence.
- Route constraints: The total time duration or the total distance of a route or the number of customers serviced by a route can be constrained.

2.2 Algorithm for Routing and Scheduling Problem

To solve the routing and scheduling problem, we propose the tabu search based algorithm. The algorithm has two phases, route construction phase and route improvement phase. In route construction phase, we find initial feasible solution by using simple heuristic methods. In route improvement phase, we find better solution than the obtained initial solution. In the phase, we use tabu search. The tabu search moves from the current solution to its best neighbor at each iteration, the cost of the neighborhood is not necessarily less than the cost (travel time or length of tour in our problem) of the current solution, until a stopping criterion is satisfied. In order to prevent cycling, solutions that have been examined before are forbidden and inserted in a tabu list. The tabu list records the iteration number at which a customer is removed from a route [3].

As shown in section 5, the proposed routing and scheduling algorithm gives the better solution than other research within a reasonable time. And, our algorithm can solve all of the various problems with the VRP with Time Windows (VRPTW), the VRP with Backhauls (VRPB), the Time Dependent VRP (TDVRP), and the Multiple Depot VRP (MDVRP).

In this work, the saving heuristic algorithm has a complexity of $O(n^3 \log n)$, and the neighborhood heuristic algorithm has a complexity of $O(n^3)$.

A. Route Construction Phase

The route construction procedure is composed of two steps: customer assignment and saving heuristics. The Algorithm 2.1 explains the route construction phase.

(1) In case of multi-depot, initially each customer is assigned to its nearest depot. If there is a depot-customer compatibility constraint between a customer and a depot, the customer is assigned to one of the other depots. Two customers who must be serviced by the same vehicle on the same route are assigned to the same depot. If there is only one depot, this procedure is omitted.
(2) The savings heuristics by Clarke and Wright [4] is applied to the customer set. In case of multi-depot, the heuristics is applied to the customer set of each depot. The method starts with vehicle routes containing the depot and one customer. At each step, two routes are merged according to the largest saving that can be generated [5]. Two routes containing customers which must be serviced by the same vehicle on the same route are merged first of all.

[Algorithm 2.1] Route Construction Phase

Step 1 If there are multiple depots, each customer is assigned to its nearest depot. Otherwise go to Step 2.

Step 2 Repeat the following steps to the customer set of each depot.

Step 2.1 Create vehicle routes containing the depot and one customer.

Step 2.2 Compute the savings for all pairs of nodes as follows: $S_{ij} = d_{i0} + d_{0j} - d_{ij}$, where S_{ij} = savings obtained from linking together any two nodes i and j, d_{i0} = distance (cost) between node i and depot 0,
d_{0j} = distance (cost) between depot 0 and node j,
d_{ij} = distance (cost) between two nodes i and j.

Step 2.3 Sort the savings in a non-increasing order.

Step 2.4 If feasible maximum saving $S_{ij} > 0$, merge these routes by adding arc (i, j) and by deleting arcs $(i, 0)$, $(0, j)$, and go to Step 2.2.
If maximum saving $S_{ij} < 0$, stop.

In step 1, if there are multiple depots, we assign all customers to each customer's nearest depot.

Step 2 illustrates a saving heuristic algorithm. After creation of initial route sets in step 1, we check each route which is composed depot-customer-depot chain with all customers on feasibility, and generate initial solution. Here, initial solution reflects the customer-customer compatibility constraints. We compute a saving value with distance (time or cost) and then sort saving by decrease. If maximum saving value is grater than 0, then merge two routes.

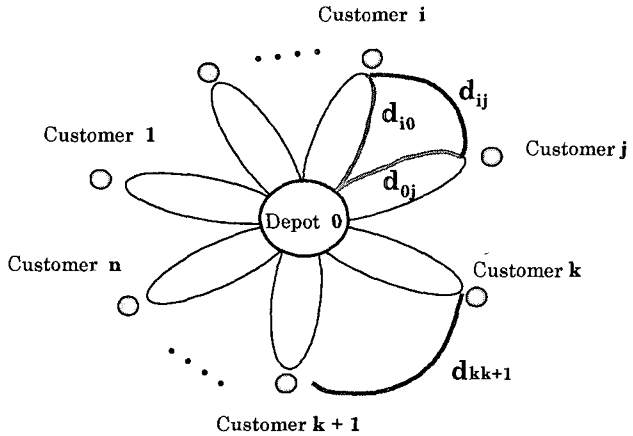


Figure 2.1. Compute the savings ($S_{ij} = d_{i0} + d_{0j} - d_{ij}$)

As Figure 2.1 shows the saving heuristics in step 2, we compute the savings for all pairs of nodes by $S_{ij} = d_{i0} + d_{0j} - d_{ij}$, and if the maximum saving s_{ij} is greater than 0, then merge these customer i and j. The processes in route construction phase construct an initial solution composed to many routes. This algorithm implemented has a complexity of $O(n^3 \log n)$.

B. Route Improvement Phase

In the route improvement procedure, a tabu search is used. The tabu search moves from the current solution to its best neighborhood at each iteration, the cost of the neighborhood is not necessarily less than the cost of the current solution, until a stopping criterion is satisfied. To avoid cycling, a move is considered as tabu if it tries to reinsert customers removed in one of the previous moves in the tabu list. To check tabu, a tabu list is used. The algorithm can be summarized as follow:

[Algorithm 2.2] Route Improvement Phase

Step 1. Create an initial solution x by the route construction phase. Set the tabu list $T = \phi$.

Step 2. Generate a neighborhood set $N(x)$ of x.
For all node pair in initial solution x, apply operators, i.e. two-opt, inser-

tion, or-exchange, swap-exchange, chosen by random at each iteration.

Step 3. If, $N(x) \setminus T = \phi$, i.e. if all move in neighborhood set $N(x)$ are in tabu list, then go to Step 4.
Otherwise, select a least cost solution y in $N(x) \setminus T$, and set $x = y$.
If x is best solution in all iteration, then update the best known solution with x.

Step 4. If the stopping criterion is satisfied, then stop. i.e. all iteration is executed or there are no more the best solution.
Otherwise, update T and go to Step 2.

In step 1, an initial solution is created by the route construction phase as shown in Figure 2.2. The tabu list records the iteration number at which a customer is removed from a route.

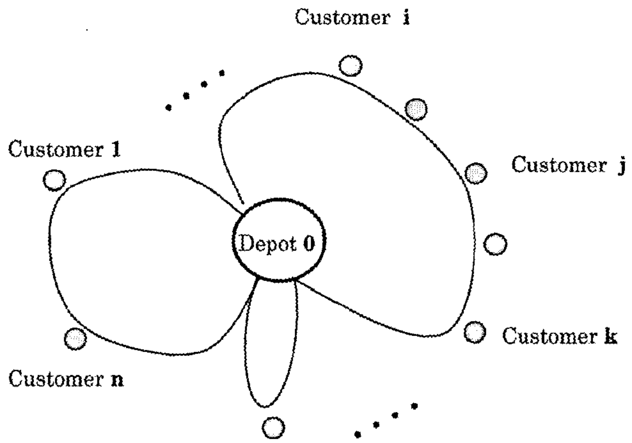


Figure 2.2. Initial solution x by the route construction phase

In Step 2, to generate a neighborhood set, we use simple neighborhoods based on arc exchanges or customer movements, such as two-opt, insertion, or-exchange, and swap-exchange in this tabu search [6]. Figure 2.3 shows the generation of neighborhood set.

In order to reduce the computing time of neighborhood exploration, a sparse graph that includes short arcs as well as important arcs is used. One of the four neighborhood generation method is selected randomly each iteration. By adding the penalty terms about the route constraints, such as the total time duration, the total distance of a route, and the number of customers which can be serviced by a route, the visit of infeasible solutions is allowed.

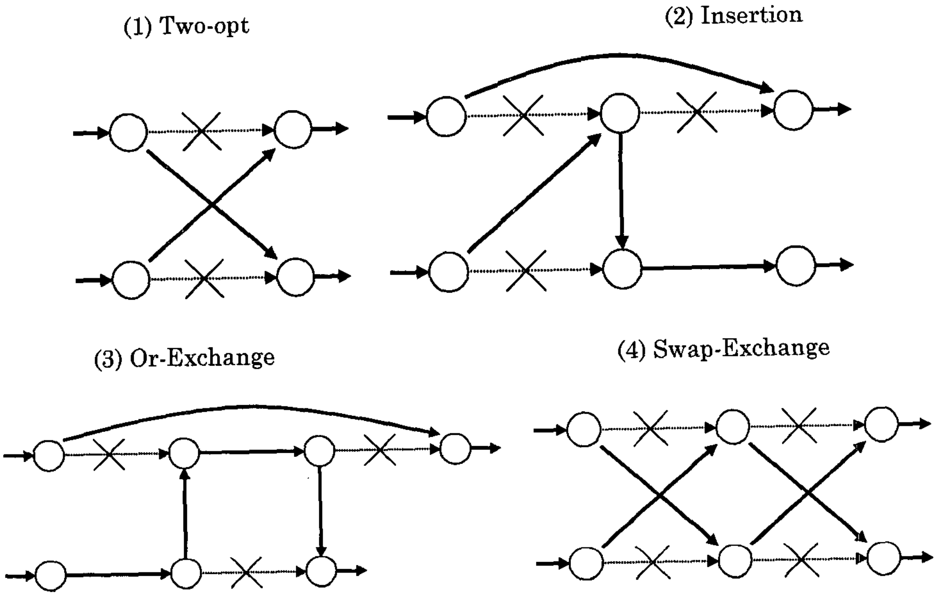


Figure 2.3. Generate neighborhood by heuristics

In step 3, if all neighborhoods moved from x are in tabu list, this means that all neighborhoods in current iteration were checked for the past iterations. Otherwise, we select a least cost solution y among the neighborhoods which are not tabu list, and change x with y. Here, if x is the best solution in all iteration, then update the best known solution to x.

In step 4, if the maximum number of iterations since the last update reaches a predetermined criterion, the algorithm is stopped. Otherwise update the tabu list and go on searching.

This neighborhood heuristic algorithm implemented in this work has a complexity of $O(n^3)$.

C. Generate the Input Data of Algorithm using GIS Information

In most of all other researches, the routing and scheduling algorithm utilizes the simple time and distance. Here, the simple time and distance means the value is calculated in the table, not considers realistic constraints in real world application. The time is considered that the logistics vehicle consumes the time on the complex constraint, i.e., the road with traffic, a crossway, and the road type. The distance is calculated on real digital map.

To get more realistic routing and scheduling solution, we use geographical information by means of the generation of input data as shown in Figure 2.4.

We get the shortest distance based on real road data among depot and customers from GIS engine. To consider the traffic information of the road at certain time zone such as 07:00~08:00, we can use routing execution data obtained from moving object engine. We update the traveling time of each time zone in database, and then we use the updated data to solve the routing and scheduling problem. By repeating this procedure, we can get the better routing and scheduling solutions.

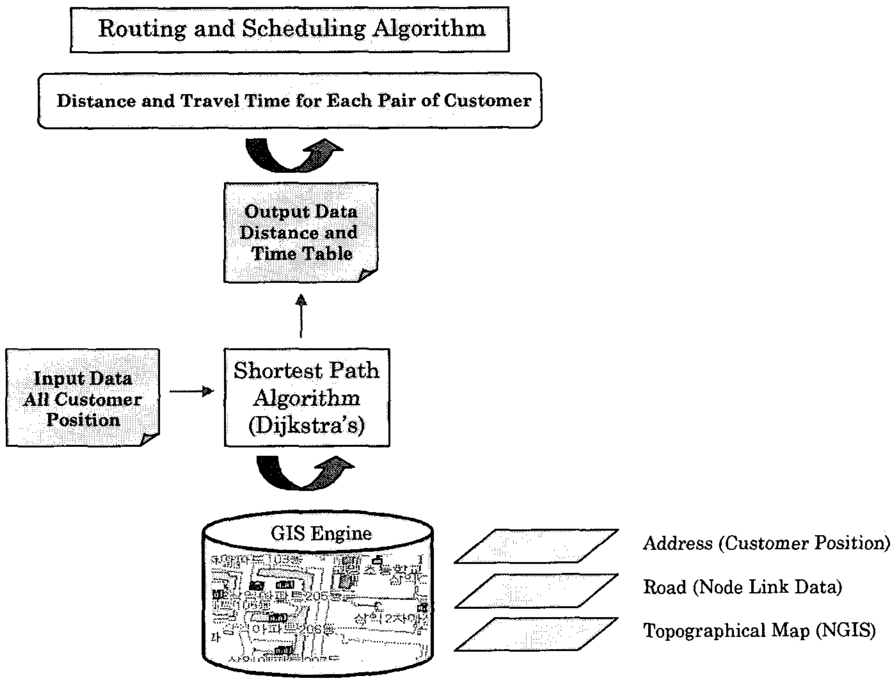


Figure 2.4. Generate distance and travel time by GIS

3. UNCERTAINTY PROCESSING IN MOVING OBJECT TRACKING

3.1 Moving Object Model

A. Conceptual Structure

A field of ongoing research in the area of spatial databases and Geographic Information Systems (GIS) involves the accurate modeling of real geographical applications, i.e., applications that involve objects whose position, shape and size change over time [7]. Spatiotemporal applications may be distinguished based on the data they manage, which may pertain to the past, the present, and the future, or a combination of these [8].

The moving object can be defined as a special kind of spatiotemporal data which the location and shape changes continuously over time. For modeling these moving objects, the continuous and discrete model has been so far suggested in the previous works. The continuous model allows us to represent the moving object in terms of infinite sets of points, and to view the moving point as a continuous curve in the 3D space. This model can accurately describe the motion information, but it is inadequate to be implemented since we cannot store or manipulate infinite points in computer. On the other hand, the discrete model allows us to describe the moving object in terms of finite sets of points and to view the moving point as a polyline in the 3D space. This can be implemented by showing the motion information of the moving object in terms of approximate values. Although the changing positions of the moving object are a continuous concept, it is necessary to use the discrete concept for modeling a system that can express these changes of, since computer systems we use have limited resources.

Also there are two kinds of moving objects such as the moving point related to the change of location and moving region related to the change of shape. In this paper, we are dealing only the moving point because the vehicles in postal logistics can be classified to the moving point [9-11].

[Definition 1] (Moving Point Object: MP) It stands for the special type of spatio-temporal data that changes its location over time flows. It consists of the temporal attributes (T_A), spatial attributes (S_A) and the general attributes (G_A). The conceptual structure can be described as $MP = < T_A, S_A, G_A >$. □

[Definition 2] (Temporal Attributes) It is one of elements of the moving point object (MP), and is composed of the both beginning (VT_s) and ending (VT_e) of the valid time when the object were valid at the location. Its conceptual structure is $T_A = < VT_s, VT_e >$. The VT_s and VT_e is an element of valid time domain (D_{VT}). Here, the D_{VT} means the set of timestamps that is used in the real world and represented as $D_{VT} = \{t_0, t_1, t_2, \cdots, t_k, \cdots, t_{now}\}$. Each of elements in D_{VT}, there are some special characteristics as follows: $t_0 < t_1 < t_2 < \cdots < t_k < \cdots < t_{now}$, $t_k = t_{k-1} + 1$, $t_k = t_0 + k$. □

[Definition 3] (Spatial Attributes) It is also one of components of moving point object. Its conceptual structure is described as $S_A = < x, y >$. Here, x and y is a coordinate value. □

B. Database Operators and Schema

A route (ς) in a delivery zone is a set of sub-routes. A sub-route (R_i) is a sequence of pair of a location and a timestamp, and (R'_i) is a planed sub-route. In expression (2), Here, (P_i) is place defined in 2D spatial domain with a pair coordinates, i.e. (x, y) and (T_i) is an event-time timestamp defined in 1D temporal domain. An Event (ϑ) is a set of sub-event (E_i) that happens by mainly customer's call and consists of a location, event-time timestamp, and due-time timestamp in expression (3) and (4).

$$\varsigma = \{R_1, R_2, \cdots, R_n\} \quad (1)$$

$$R_i = \{(P_i^1, T_i^1), (P_i^2, T_i^2), \cdots, (P_i^n, T_i^n)\} \quad (2)$$

In expression (2), (P_i^1, T_i^1) means 1st position value of ith trajectory (R_i), and 1st timestamp value of ith trajectory (R_i), respectively.

$$\vartheta = \{E_1, E_2, \cdots, E_n\} \quad (3)$$

$$E_i = (P_i, T_i, D_i) \quad (4)$$

New Operators for the moving objects such as *AtTime*, *Direction*, *NearestTrajectory*, and *STSync* can be defined like below. The *AtTime* operator, in expression (5), returns a place (P_k) included in route (R_i) located at (T_i). The *Direction* operator, in expression (7), returns one or two characters that mean where the route (R_i) is headed from the place at time (T_i), i.e., East, West, South, and North. In expression (8), the *NearestTrajectory* operator returns a sub-route (R_k) from a route (ς) that is nearer than any other sub-route from a place (P_i) at time (T_i). The Euclidean distance (δ) returns a float value which stands for the straight line between two points in expression (9). Finally, the *STSync* operator, in expression (10), returns two positions which is assigned to given two sub-route (i.e., (R_i) and (R_j)) at time (T_k), respectively. An important innate constraint is the two positions have no same geometric coordinates.

$$AtTime(R_i, T_i) = \{P_k \mid \exists k((P_k, T_k) \in R_i \wedge T_i \equiv T_k)\} \quad (5)$$

$$\exists\, j, k(\forall T_k((P_i, T_i), (P_j, T_j), (P_k, T_k) \in R_i \wedge T_i < T_j < T_k) \quad (6)$$

$$\begin{aligned} Direction(R_i,T_i) = & \{E \mid (6) \wedge P_{i.x} < P_{j.x} \wedge P_{i.y} = P_{j.y})\} \vee \\ & \{W \mid (6) \wedge P_{i.x} > P_{j.x} \wedge P_{i.y} = P_{j.y})\} \vee \\ & \{S \mid (6) \wedge P_{i.y} > P_{j.y} \wedge P_{i.x} = P_{j.x})\} \vee \\ & \{N \mid (6) \wedge P_{i.y} < P_{j.y} \wedge P_{i.x} = P_{j.x})\} \vee \\ & \{ES \mid (6) \wedge P_{i.x} < P_{j.x} \wedge P_{i.y} > P_{j.y})\} \vee \\ & \{WS \mid (6) \wedge P_{i.x} > P_{j.x} \wedge P_{i.y} > P_{j.y})\} \vee \\ & \{EN \mid (6) \wedge P_{i.x} < P_{j.x} \wedge P_{i.y} < P_{j.y})\} \vee \\ & \{WN \mid (6) \wedge P_{i.x} > P_{j.x} \wedge P_{i.y} < P_{j.y})\} \end{aligned} \tag{7}$$

In expression (7), ($P_{i,x}$) and ($P_{i,y}$) means the x and y coordinate value of i[th] position, respectively.

$$\begin{aligned} NearestTrajectory(P_i,T_i) = & \{R_k \mid \exists i,j,k,l,m,n(R_i,R_j,R_k \in \varsigma \wedge \\ & (P_l,T_l) \in R_i \wedge (P_m,T_m) \in R_j \wedge (P_n,T_n) \in R_k \wedge \\ & \delta(P_i,P_n) < \delta(P_i,P_l) \wedge \delta(P_i,P_n) < \delta(P_i,P_m) \wedge \\ & T_i \equiv T_l \wedge T_i \equiv T_m \wedge T_i \equiv T_n)\} \end{aligned} \tag{8}$$

$$\delta(P_i,P_j) = \sqrt{(P_{j.x} - P_{i.x})^2 + (P_{j.y} - P_{i.y})^2} \tag{9}$$

$$STSync(R_i,R_j,T_k) = \{(P_i,P_j) \mid (P_i,T_k) \in R_i \wedge (P_j,T_k) \in R_j\} \tag{10}$$

The moving objects database based on the spatiotemporal databases technique is composed of two major tables to store sampled location data of moving objects. The detailed scheme structure for each of them has the following schema:

Table 3.1. Moving Object Databases Schema

(a) VehicleInfo_T

<u>VehicleID</u>	Type	<u>DriverName</u>	CellularPhone	Depot	Etc
VarChar2(10)	VarChar2(10)	VarChar2(8)	VarChar2(13)	VarChar2(32)	VarChar2(64)

(b) VehicleOperation_T

<u>VehicleID</u>	XS	YS	XE	YE	Velocity	Direction	<u>VF</u>	<u>VT</u>	Etc
VarChar 2(10)	Number (18,9)	Number (18,9)	Number (18,9)	Number (18,9)	VarChar 2(3)	VarChar 2(3)	VarChar 2(12)	VarChar 2(12)	VarChar 2(16)

In VehicleInfo_T table, the *VehicelID* and *DriverName* is the key value as an identifier of the moving object. The *Type*, *CellularPhone*, *Depot*, *Etc* are for the properties of each object including non-spatial information. Movement information table, i.e. VehicleOperation_T, stores historical information regarding sampled time and location of the object. In VehicleOperation_T table, the set of *VehicleID*, *VF*(Valid From), and *VT*(Valid To) is the key value. And, the *XS*, *YS*, *XE*, *XY* specifies sampled time point and location coordinate values. Finally, the *Velocity* and *Direction* attributes are the additional information calculated directly from the location detecting devices.

3.2 Uncertainty Processing

A. Basic Concepts

There are generally four sorts of reasons that give rise to happen uncertainty of moving object such as (1) the inadequate time interval which gathers location data, (2) the errors in telecommunication and databases (physical layer), (3) the illegal road and traffic information (logical layer), (4) the unexpected events in applications.

In case of the first type of them is related to the cost for gathering the location data of moving objects. Also, the overall performance can be determined directly by the size of packets among the units and databases in the center. The efficient strategy to estimate the locations that were not collected by the detecting devices is one of major issues in the uncertainty domain.

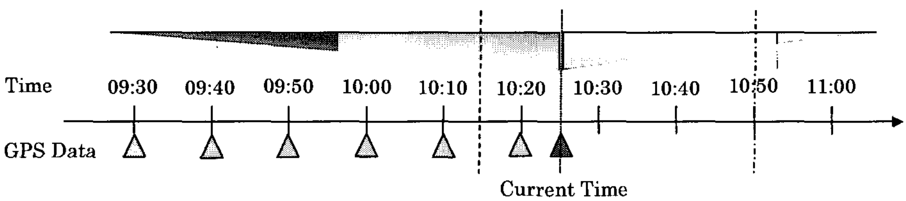


Figure 3.1. Moving object uncertainty query types

The uncertainty concepts are bringing into relief when users request the locations at some time for the moving objects. Usually, the temporal management is always included into the moving object techniques. Intuitively, there are two kinds of query types in uncertainty processing which can be handed in the users to the system as shown in Figure 3.1. On the basis of current time, (1) the retroactive uncertainty queries request the locations that can be estimated among the past locations. On the other hand, (2) the pro-active uncertainty queries demand the future location which means the location where should be calculated in terms of time, i.e., immediately after of current time. According to the application domain, there are no means to get the pro-active uncertainty locations because they are generally determined by the human so that no one can decide the exact location of moving objects, for examples, the exact location of vehicles after 1 hour in the metropolitan city.

B. Operator and Methodology

There is a representative operator, *AtTime*, which is one of typical moving object operators and is related directly to the uncertainty processing. The *AtTime* operator searches the point included in the trajectory, and can be rewritten to support the uncertainty of location like below expression (11) :

- Whenever the location requested be the users query is not found in the moving object databases, it should at first retrieve the locations which is just prior to the input time and immediately after of the time.
- And then special operation such as interpolation would be applied to the locations to estimate the point that means the location at given time.

$$AtTime(R_i, T_i, \mathrm{Pr}) = \begin{cases} P_k \mid \exists k((P_k, T_k) \in R_i \wedge T_i \equiv T_k),\ if(P_k) \neq null \\ P_k \mid P_k := UcR_{\mathrm{Pr}}(R_i, T_i),\ otherwise \end{cases} \quad (11)$$

In expression (11):= means the value of RHS assigns to the value of LHS.

$$(DUcR_{\mathrm{Pr}}(R_i, T_i) = \begin{cases} P_k \mid \exists j,k((P_j, T_j) \in R_i' \wedge P_j \in UcB(P_k, \mathrm{Pr}) \rightarrow P_k := P_j),\ if(P_j) \neq null \\ P_k \mid \exists k(P_k := LI(R_i, T_i)),\ otherwise \end{cases} \quad (12)$$

In expression (12), ($R'i$) is a planed sub-route by the routing and scheduling algorithm.

$$\begin{aligned} LI(R_i, T_i) = \{P_k \mid \exists m,n((P_m, T_{i-1}) \in R_i \wedge (P_n, T_{i+1}) \in R_i \rightarrow \\ P_k := (\frac{P_n - P_m}{T_{i+1} - T_{i-1}}(T_i - T_{i-1}) + P_m))\} \end{aligned} \quad (13)$$

$$\begin{aligned} UcB(P_k, \mathrm{Pr}) = \{C_l \mid \exists m,n((P_m, T_{i-1}) \in R_i \wedge (P_n, T_{i+1}) \in R_i \rightarrow \\ C_l := \delta(P_m, P_n) * \mathrm{Pr}/100)\} \end{aligned} \quad (14)$$

There are also several kinds of method to estimate the location of uncertain time. (1) *Nearest value* method determine the nearest location from the two or three locations which were stored in the database and is nearest to the estimated location, (2) *Linear interpolation* method estimates the uncertain location where is one location exist in the line which stems from two locations with simple equation, (3) *Spline interpolation* method differs from the linear interpolation, and uses two locations and additional location with spline equation, (4) *Stochastic interpolation* method is a probabilistic scheme which is based on the quasi-random generator, dimension-split line, and several locations.

C. Uncertainty Processing Algorithm for Extended Operator *AtTime*

The uncertainty processing concepts are bringing into relief data when users request a location not in database at some time for tracing moving objects. We propose the uncertainty processing algorithm to solve the problem as fellows.

- Vehicle tracking applications based on existing paradigm require a large amount of telecommunication and database to apply the e-Logistics Chain Execution system for the reason of gathering and managing of GPS data.
- Previous works have problems in terms of computation speed and degree of accuracy of estimation for vehicle location owing to the complex conditions such as human and environmental status in metropolitan area.

To estimate the location of moving object at uncertain time, this paper adopts a linear interpolation method for uncertainty processing. The detailed processing algorithm is can be described as given in Algorithm 3.1:

[Algorithm 3.1] Uncertainty Processing for Extended Operator *AtTime*

Step 1 Basic Processing Step

Step 1.1 Get the query time (t) and probability (p_r), and search location (p_s); If found return with p_s;

Step 1.2 Retrieve two locations (p1, p2) which are corresponding the timestamps (t1, t2) contains user's query time (t);

Step 1.3 Calculates the location (pu) using the linear interpolation method. Here the location (pu) can be defined with the ratio of query time (t) from the locations (p1, p2);

Step 1.4 Computes the uncertainty boundary of pu, based on the input probability (pr).

Step 2 Additional Processing Step

Step 2.1 If there is any planning route data (p') corresponding to pu at query time (t), pu should be revised to p'. p' also should be exist within the uncertain boundary;

Step 2.1 In order to set pu to the real world value, there is also additional processing such as map matching process;

Figure 3.2 shows uncertainty boundary computed by algorithm, and vehicle position corresponding estimated position in planed route.

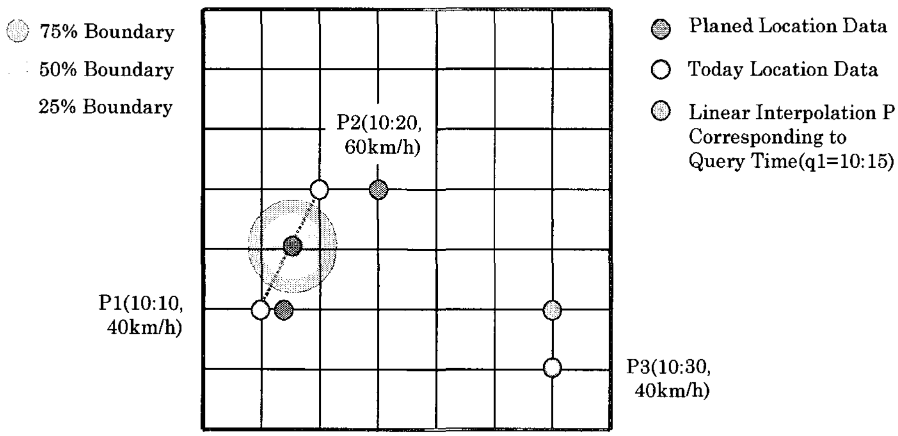
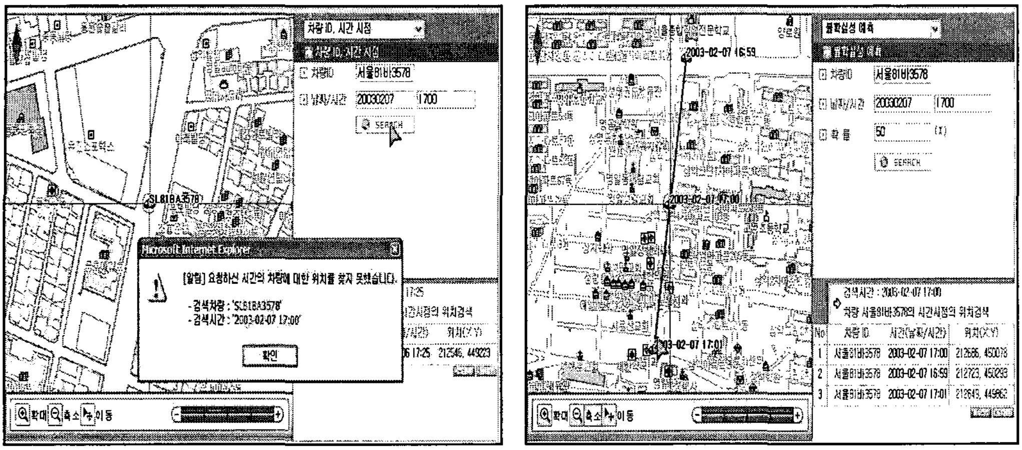


Figure 3.2. Example of uncertainty boundary

D. Example of Uncertainty Query in Postal Logistics

As discuss above, there are representatively two kinds of query type in the moving object uncertainty. The detailed query type and their respective processing sequence are described below :

- **Query Example** : *Display the location of vehicle (DJ80Ga1234) at 2002-9-4 15:15 As of 2002-9-5 10:10 with probability 25%.*
- **Processing Example** : Estimate the location of vehicle as below :
  - Search location for the vehicle with exact matching of vehicle number and time, i.e. SL81Ba3578 at 2003-2-7 17:00
  - If there is no match location data but given probability is not 0%, retrieve two locations which timestamp contains the input timestamp, i.e., 2003-2-7 16:59 and 17:01
  - Calculate the uncertainty boundary from the location where is expected to be from the input timestamp using the input probability (50%)
  - Check the planned route within uncertainty boundary
  - None → Return the location calculated with linear interpolation method
  - Otherwise → Return the location retrieved from the routing and scheduling system
- **Output** : The result of uncertainty query is as shown in Figure 3.3 :



(a) Before uncertainty processing (b) After uncertainty processing

Figure 3.3. Result of uncertainty processing

4. IMPLEMENTATION OF E-LOGISTICS CHAIN EXECUTION SYSTEM FOR POSTAL LOGISTICS

The e-Logistics Chain Execution (e-LCE) system shows the application example on postal logistics area to apply the routing and scheduling algorithm and the uncertainty processing algorithm proposed in this paper. Through the application example, we show our researches are efficient to e-Logistics chain. Especially, these results make the good solution for the pick-up and delivery in post office.

The local post office must devise a plan using information of high level organization to conduct customer's order where call center has to grasp the resources and schedule of low level organization which consists of logistics chain to reserve visiting time about customer's order. Especially, when the new request occurs during pick-up and delivery execution, it is necessary to determines using dynamic routing and scheduling simulator whether customer's requirement can be satisfied under the restricted conditions or not as shown in Figure 4.1.

The system to grasp operating situation about pick-up and delivery of post offices is the vehicles tracking subsystem of e-LCE system, and the system to devise pick-up and delivery plan and to respond to real-time order of customer is the routing and scheduling subsystem. Figure 4.2 explains e-LCE system for parcel pick-up and delivery service of Korea Post, and, Figure 4.3 and Figure 4.4 shows the result of e-LCE system running.

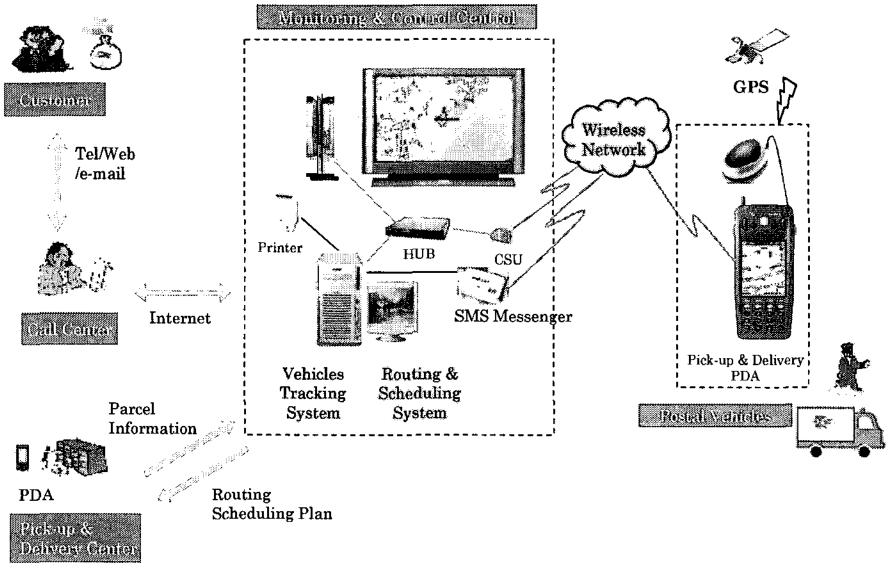


Figure 4.1. e-Logistics Chain Execution System for parcel service of Korea Post

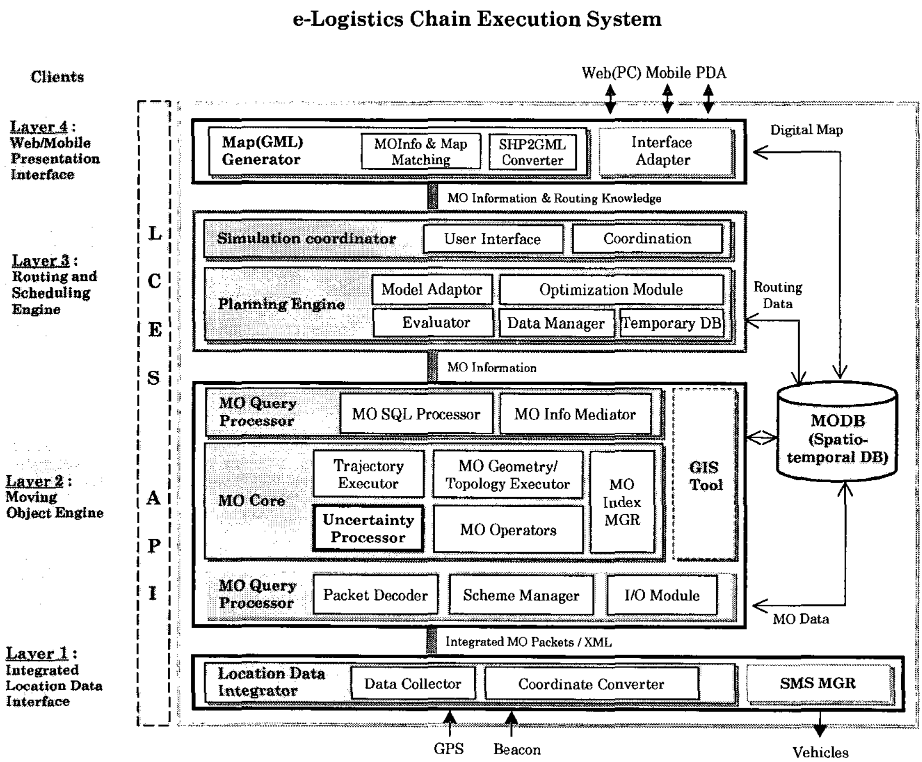


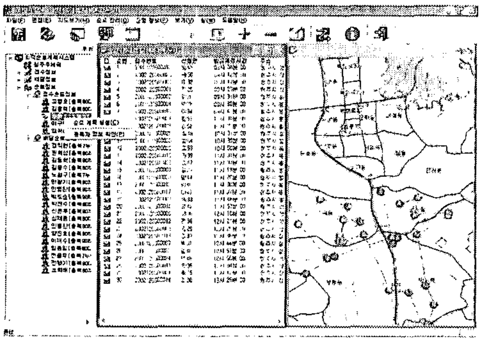
Figure 4.2. System architecture for routing simulation and vehicles tracking

Location Data Integrator of layer 1 is the subsystem to acquire current vehicle position after interfacing with terminal installed in postal vehicles. Location data integrator includes the function to convert both the GPS position coordinates and the beacon position coordinates into standard coordinates to provide the service about all position coordinates. GPS position coordinates are collected through satellite and Beacon position coordinates are collected through base station.

Moving Object Engine of layer 2 is the subsystem to manage the historical position data of vehicles using database, and to handle query related to trace, phase, geometry using historical position data of vehicles. It consists of core, query processor, and data loader about the moving object. Core has (1) main database to manage the moving historical data of vehicles in spatio-temporal database which can well express time and space changes; (2) index manager to support TB-tree algorithm for storing mass data of moving object efficiently and searching them quickly; (3) MO operators to have the operation function about the moving object data; (4) Trajectory/Geometry/Topology executor to support the operation function about position extended; (5) uncertainty processor to have the guess function for vehicle position data about time not stored from position data stored in database. Query Processor conducts SQL related to moving object query required in application system, and supports transmission/reception about the query request and the result. Data Loader has the function to convert packet transmitted from location data interface into schema data stored in database.

Routing and Scheduling Engine of layer 3 are used to generate the optimized route for the pick-up and delivery of parcel service. Temporary DB includes information about sender and receiver of parcel such as postal address, requested time, and name and phone number, while GIS DB in GIS tool includes geographical information of whole pick-up and delivery area such as postal addresses and its coordinates values, and road and building information. In GIS DB, each intersection on physical road is represented as one node. Also, all postal addresses in GIS DB are matched to corresponding nodes. Corresponding node for a postal address is determined as the nearest intersection on physical road to the address. In addition, shortest paths, its travel time, and lengths between all of node pairs are in GIS DB. The shortest path between any two nodes is obtained using well-known Dijkstra's algorithm. Results of routing and scheduling engine, the visiting schedule for pick-up and delivery, are stored in DB.

Web/Mobile Presentation Interface of layer 4 provides presentation about information related to the moving object exchangeable with client terminal such as PC, mobile phone, and PDA. At this time, digital map like trace data of the moving object is converged into GML (Geographical XML) data, and it is then offered by the standard data presentation method to be used without any restriction about user environment.



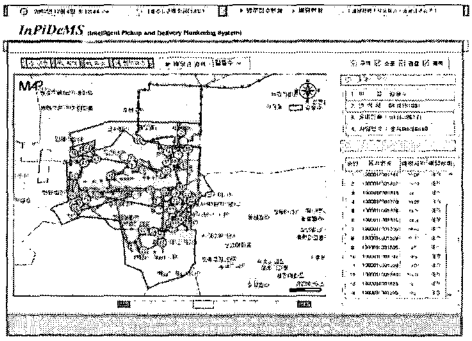


Figure 4.3. Result on the routing and scheduling Figure 4.4. Result on the vehicles tracking

5. EVALUATION

5.1 Pretest results of routing and scheduling algorithm

The algorithm for the routing and scheduling problem described in section 2 is implemented in Java. To evaluate the performance of the algorithm, we test the algorithm with several data from the literature announced.

- Objective of the pretest - the pretest is executed to evaluate the performance of the routing and scheduling algorithm, and to show the efficiency of a solving ability in various application problems.
- Test data and process - we choose the several instances from the papers which were well announced in several problem areas to obtain test data of the routing and scheduling algorithm. The problem areas are the VRP with Time Windows (VRPTW), the VRP with Backhauls (VRPB), the Time Dependent VRP (TDVRP), and the Multi-Depot VRP (MDVRP). We utilize the test data and result in each problem by the GTS algorithm of P. Toth [12] for basic VRP, the AKRed algorithm of R. Cordone [13] for VRPTW, the EHP algorithm of A. Mingozzi [14] for VRPB, the TS algorithm of D. J. Jeon [14] for TDVRP, and the CGL algorithm of J. F. Cordeau [15] for MDVRP. After the algorithm is run 10 times with the selected test data for each known problem set, we determine the best solution value in our algorithm. We also compare our best solution value with the best known solution from

literature to search the gap. The gap means the efficiency for our algorithm to solve the various problems in general.

- Test environments - the test is executed on Pentium 1.7 GHz PC, and the routing and scheduling algorithm is coded in Java. The iteration parameter is N_MAX = max(1500, 30×N), i.e., the number of maximum iteration is less than 1,500 or 30×N times. The tabu list size is selected by random number [5, 10].

The pretest results are summarized in Table 5.1. Each instance was run ten times. The 'Best sol.' column indicates the best solution value among ten runs. The 'GAP (%)' column means the percentage ratio of the difference between the best solution value and the best known solution value with respect to the best known solution value (means the optimal solution value). We used an integer uniformly distributed random value in the interval as a tabu list size. If the objective function value is not decreased for the last 1500 iterations, the tabu search algorithm is terminated.

Test results show that our algorithm does not give better solution than other researches. Note that the previous researches focused on specific routing problems. But, our algorithm can solve all of the variations. Moreover, our algorithm gives very good solutions within a reasonable time.

Table 5.1. Pretest results of the algorithm

Problem Set**	Instance	Num of Customers	Best known Sol.	Tabu search based algorithm		
				Best sol.	Time (s)	Gap (%)
VRP	Vrpncl	50	524.61	524.81	27.23	0.00
	Vrpnc2	75	835.26	841.83	99.04	0.32
VRPTW	c101	100	828.94	828.94	25.82	0.00
	c201	100	591.56	591.56	44.81	0.00
VRPB	A1	25	229886	229885.65	8.29	0.00
	B1	30	239080	239080.16	15.79	0.00
MDVRP	p01	50	576.87	576.87	17.39	0.00
	p03	75	641.19	643.57	15.07	0.37
TEVRP	155_1	15	1722.351	1749.732	3.93	1.59
	155_2	15	1756.965	1759.309	4.24	0.13
Average						0.24

In Table 5.1, the result of our algorithm for the basic VRP is considered 50~100 customers, and the GTS algorithm [12] of Toth and Vigo(**) is generally better than our algorithm in basic VRP problem. The algorithm of GTS has tested on a Pentium 200 MHz PC, and coded in FORTRAN 77. The comparison with gap of GTS is ranged from 0.0 % to 0.32 %.

The result of our routing and scheduling algorithm for VRPTW is considers customer's service time requested to VRP problem. The number of customer 100 is considered. In VRPTW problem, the result of test is the same to our algorithm as AKRed algorithm [13]. The AKRed algorithm of Cordone and Calvo has tested on a Pentium 166 MHz PC, and coded in FORTRAN 77.

In the our algorithm for VRP problem with pick-up and delivery (VRPB), the number of customer 25~60 is considered. The EHP algorithm [14] of Mingozzi and Giorgi is the same to our algorithm in VRPB problem. The algorithm of EMP has tested on a Silicon Graphics Indy (MIPS R4400/200 MHz processor), and coded in FORTRAN 77.

The VRP problem with multiple depot (MDVRP) produces the inaccurate result than the CGL algorithm [15]. The number of customer 50~100 is considered. The algorithm of CGL has tested on a Sun Sparc-station 10. The comparison with gap of CGL is ranged from 0.0 % to 0.37 %.

And, Table 5.1 shows the result of our algorithm in VRP problem with time dependent travel time (TDVRP). Our algorithm is considered the number of customer 15. The TS algorithm [16] of D. J. Jeon is generally better than our algorithm in TDVRP problem. The algorithm of Jeon has tested on a Pentium 800 MHz PC, and coded in C. The comparison with gap of TS is ranged from 0.13 % to 1.59 %.

5.2 Practice results of pick-up and delivery routing in postal Logistics

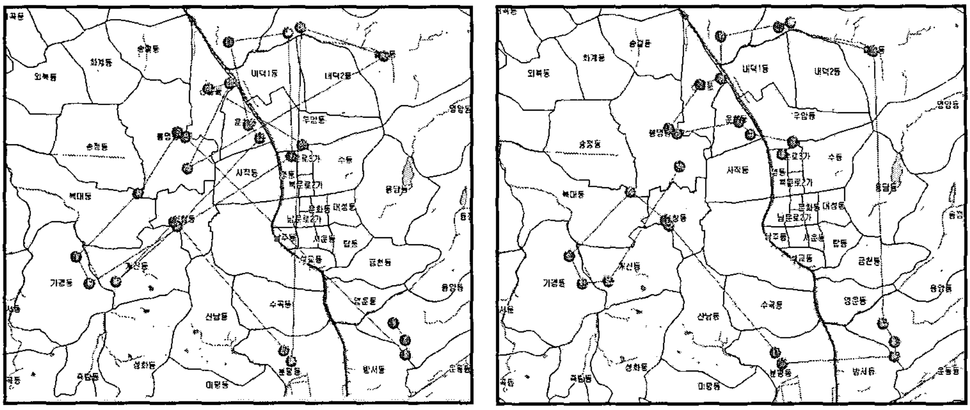
Through the test of the prototype system, we found some possibilities of the system for practical use in real situations of local post office.

We use prototype system on practical test with the e-Logistics Chain Execution system proposed in section 4. The postal logistics prototype system is implemented on 1.7 GHz PC server and PDA with GPS function. The system is coded C++ and Java language. To evaluate the test result, we compare the pick-up and delivery plan by the routing and scheduling algorithm with the manual visiting plan by postman.

- Routing and scheduling plan on pick-up and delivery service

In order to perform the pick-up request of 24 customers which is managed by a postman in one day, the routing path is drawn up manually based on the postman's knowledge as shown in Figure 5.1(a). Its total operated distance and disturbance time is 35.21 km and the 336 minutes, respectively.

On the other hand, Figure 5.1(b) represents the routing path which is generated from the routing and the scheduling engine of the e-LCE system. It is easy to recognize the improvement of complexity of the routing route. Also, the total operated distance is 20.85 km and, the disturbance time is 245 minutes.



(a) Route by postmen's knowledge (b) Route by algorithm

Figure 5.1. Practical results of pick-up and delivery routing

With the result of practical experiment, we can dare insist that in terms of operation distance and disturbance time, the degree of improvement of the proposed system lead 42% and 27%, individually. Because the result presumes the best solution, the actual situation such as the precedence constraints, the hour reservation of customer, and the postman's restricted conditions is not considered.

- Uncertainty processing on vehicles tracking

For the evaluation of uncertainty processing algorithm proposed, we test the vehicle tracking prototype system on the changing of time interval to gain location data.

The vehicle tracking system acquire a location data from PDA with GPS function on postal vehicles in every time interval, manage the vehicle trajectory in database, and response to user query.

In addition, through the uncertainty processing, we regulate the collection of GPS location data without failure of vehicles tracking from 10 second units to 60 second interval. It will be able to attain the reducing of frequency in telecommunication and DB updating as one to six.

6. CONCLUSIONS

In this paper, we proposed the intelligent e-LCE system that applies GIS/GPS, mobile, and optimization technologies for the routing and scheduling of the door-to-door parcel service in Korea Post. To perform the plan and the control activities for pick-up and delivery of postal logistics chain, the proposed e-LCE system was composed of the route and schedule planning system based on tabu search algorithm and the real-time vehicle tracking system based on the GPS/GIS/ITS information.

The test results of the tabu search algorithm proposed showed that the algorithm gives very nice solution within a reasonable execution time for the vehicle routing and scheduling problem. Though the other algorithms in previous researches were focused on only specific routing problems, our algorithm can solve the various problems as the VRP with time windows, the VRP with Backhauls, the Time Dependent VRP, and the Multiple Depot VRP. Moreover, our algorithm considered the realistic complex constraints using digital map. The proposed algorithm can be used as the engine of the routing and scheduling module in e-LCE system.

From the practical experiment, it appeared that e-LCE system reduced 27 % in the pick-up and delivery operation time, and 42 % in the vehicle operation distance. In addition, the uncertainty processing algorithm for vehicles tracking reduced the size of the spatio-temporal database which was used with up to 1/6, and lowered the telecommunication cost to transmit GPS data.

The e-LCE system suggested in this work is currently in the state of the field test for the door-to-door parcel service of Korea Post. The system we have proposed may provide the basis for numerous further research areas. For example, the algorithm can be extended to the multi-typed business model for e-logistics companies.

REFERENCES

[1] S. Chopra and P. Meindl, *Supply Chain Management: strategy, planning, and operation,* Prentice Hall, New Jersey, 2001.

[2] R. H. Ballou, *Business Logistics Management,* Prentice Hall, New Jersey, 1999.

[3] I. H. Osman, "Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem," *Annals of Operations Research* 41 (1993), 421-451.

[4] G. Clarke and J. W. Wright, "Scheduling of vehicles from a central depot to a number of delivery points," *Oper. Res.* 12 (1964), 568-581.

[5] G. Laporte, "The Traveling Salesman Problem: An overview of exact and approximate algorithms," *European Journal of Operational Research* 59 (1992), 231-247.

[6] P. Toth and D. Vigo, "A heuristic algorithm for the symmetric and asymmetric vehicle routing problems with backhauls", *European Journal of Operational Research* 113 (1999), 528-543.

[7] Y. Theodoridis, J. Silva, and M. Nascimento, "On the Generation of Spatiotemporal Datasets", *Proceedings of the 6th International Symposium on Large Spatial Databases* (*SSD*-99), Jul. 1999.

[8] D. Pfoser, et al., "Capturing the Uncertainty of Moving-Object Representations", *Proceedings of the International Symposium on Spatial Databases* (*SSD*), 1999.

[9] R. Guting, M. Bohlen, M. Erwig, C. Jensen, N. Lorentzos, M. Schneider, and M. Vazirgiannis, "A Foundation for Representinf and Querying Moving Objects", *ACM Transactions on Database Systems* (*TODS*) 25, 1 (Mar. 2000).

[10] M. Erwig, R. H. Güting, M. Schneider and M. Vazirgiannis, "Spatio-Temporal Data Types: An Approach to Modeling and Querying Moving Objects in Databases", *GeoInformatica* 3, 3 (1999), 269-296.

[11] D. Kim, K. Ryu and H. Kim, "A Spatiotemporal Database Model and Query Language", *The Journal of Systems & Software*, 55, 2 (Feb. 2001).

[12] P. Toth and D. Vigo., *The Granular Tabu Search* (*and its Application to the Vehicle Routing Problem*), Working paper, 2001.

[13] R. Cordone and R. W. Calvo., "A Heuristic for the Vehicle Routing Problem with Time Windows", *Journal of Heuristics* 7 (2001), 107-129.

[14] A. Mingozzi, S. Giorgi and R. Baldacci, "An exact method for the vehicle routing problem with backhauls", *Transportation Science*, 33, 3 (1999), 315-329.

[15] J. F. Cordeau, M. Gendreau and G. Laporte, *A tabu search heuristic for periodic and multi-depot vehicle routing problems*, John Wiley & Sons, Inc., (1997), 105-119.

[16] Duck-Joong Jeon, *A heuristic algorithm for the vehicle routing problem with time-dependent travel time*, MS thesis, KAIST, Daejeon, 2002.