Journal of the Korean Institute of Industrial Engineers
Vol.24, NO.1, March, 1998

단방향 유도경로AGVS상의 유휴차량 대기위치 결정

Positioning idle vehicles of AGVS on unidirectional guide paths

김재연* · 김갑환**

Jae Yeon Kim* · Kap Hwan Kim**

Abstract

It is discussed how to locate idle vehicles of AGVS on unidirectional guide paths. The problem is modeled as a median location problem on a cyclic network. The characteristics of the optimal location are analyzed and some useful properties are suggested to reduce the solution space. Based on the properties found, several special types of networks - a pure single-loop type, the loop type with a single branching intersection and the loop type with local branching node - are also analyzed as well as the general cyclic network. In each case of the networks, a solution procedure is suggested. Some numerical examples are provided to illustrate the algorithms developed.

Key words : Automated Guided Vehicle System, Network location problem, Parking location

1. Introduction

Automated Guided Vehicle System(AGVS) becomes a popular material handling equipment in such application areas as flexible manufacturing systems, warehouse, assembly lines, etc.. Flexibility is considered as a main advantage for these applications.

The performance of the material handling system is significantly influenced by several operation policies such as routing, scheduling, dispatching and management of idle vehicles [Sinriech et al. 1991, 1992, Tanchoco et al. 1992]. The positioning strategy of idle vehicles on the guide path is one of the important operational issues which have been neglected by academic researchers until recently [Egbelu 1993, Kim 1995, Chang 1996].

The positioning problem of idle handling equipment has been studied frequently for several automated material handling systems including AS/RS [Egbelu 1991] and AGVS. It is considered to be an important factor to the performance of the corresponding material handling system when the load of the handling activity is relatively low.

When an AGV completes a delivery task, it remains idle until the next pickup request is issued. At this

* 동양대학교 산업공학과
** 부산대학교 산업공학과

moment, the supervisory computer directs the idle vehicle to move to a specific positioning location and wait for a future pickup call there. The positioning location usually affects the response time of the vehicle significantly when a pickup call is issued actually.

Studies on AGV positioning strategies have been performed by Egbelu(1993), Kim(1995) and Chang(1996). Egbelu(1993) proposed the following three criteria when determining the parking location of idle vehicles :

1) *minimization of maximum vehicle response time*,

2) *minimization of mean vehicle response time*,

3) *even distribution of idle vehicles in the network*.

He also provided a mathematical model (linear programming) based on the objective of minimizing the maximum vehicle travel time to reach a pickup location from the parking location. Both bidirectional and unidirectional loops with single and multiple vehicles were studied. Kim (1995) also suggested a methodology to determine the parking location of idle vehicles on a loop guide path. In his paper, the expected response time for a pickup call is minimized. Static and dynamic central zone positioning strategies were analyzed. And both the bidirectional and unidirectional guide paths were considered. Chang et al. (1996) suggested a dynamic relative positioning strategy. They assumed a situation that the possibility of pickup calls by each workstation changes dynamically over time.

According to the literature on AGVS [Co and Tanchoco 1991, Egbelu 1993], the following rules are commonly used when positioning idle vehicles :

1) *central zone positioning rule*,

2) *circulatory loop positioning rule*,

3) *point-of-release positioning rule*.

And the positioning strategies can also be classified as follows according to the type of the information being used in decision making :

1) *static positioning*,

2) *dynamic positioning*,

3) *look-ahead positioning*.

When the static positioning strategy is used, the parking location is not changed once it is determined in the designing stage and a specific path segment is designated (provided) as the parking location. In the dynamic positioning strategy, the parking location for the next idle vehicle is changed from time to time according to the parking locations of other vehicles. Finally, in look-ahead positioning, the expected time for the next pickup call by each workstation is forecasted in advance and utilized in determining the parking location for the next idle vehicle.

In this study, we assume that the central zone positioning rule is used to control idle vehicles. And, the algorithms in this paper can be used not only for the static positioning strategy but also for the dynamic positioning strategy when the routing probabilities among workstations are changed as the time goes by.

In most actual applications of manufacturing systems, the average service time of AGVS is a more popular criteria to evaluate operation strategies of AGVS than the maximum service time that is often validated in the cases of emergency services. The service time consists of the assigned but empty travel (response) time, the loaded travel time and the load transfer time. Since the latter two ones can be considered to be fixed for a given route and given delivery requirements, the expected response time is a reasonable criterion for determining the parking location of vehicles. Thus, we adopt the expected response time as the objective function to determine the positioning location.

Although several researchers dealt with the positioning problem, they all assumed the guide path of the loop type [Egbelu 1993, Kim 1995, Chang 1996]. In this paper, the problem of positioning idle vehicles is addressed for the case of a general unidirectional guide path. We investigate some special characteristics of the unidirectional guide paths and develop an efficient algorithm to find the optimal parking location.

In the next section, the problem is defined mathematically. In section 3, some useful characteristics of the optimal solution are analyzed. In section 4, the procedures to find the optimal solution on a general cyclic guide path network and several guide paths of special types are provided and analyzed. And some numerical examples are illustrated on each case of guide paths. In the final section, concluding remarks are suggested.

2. Problem definition and formulation

For example, consider a guide path for AGVS as shown in Figure 1. We assume that the travel speed of vehicles is 0.9m/sec [Chang 1996]. Suppose that vehicles are responsible to transport materials with processing routes as in Table 1. We can transform flow requirements in Table 1 to From/To matrix of Table 2.

Table 1. Flow requirements for the example

Material	Route(department number)	Travel freq. per shift
Product 1	1→4→3→6→1	50
Product 2	1→5→3→2→1	40
Product 3	1→4→3→2→1	30
Product 4	1→5→7→2→6→1	20
Product 5	1→3→2→6→1	10

each other, the probability that the next pickup call is issued by a specific pickup station can be estimated by the columnwise summation of entries in the corresponding row divided by the total of entries in From/To matrix. The probabilities are illustrated in the last column of Table 2.

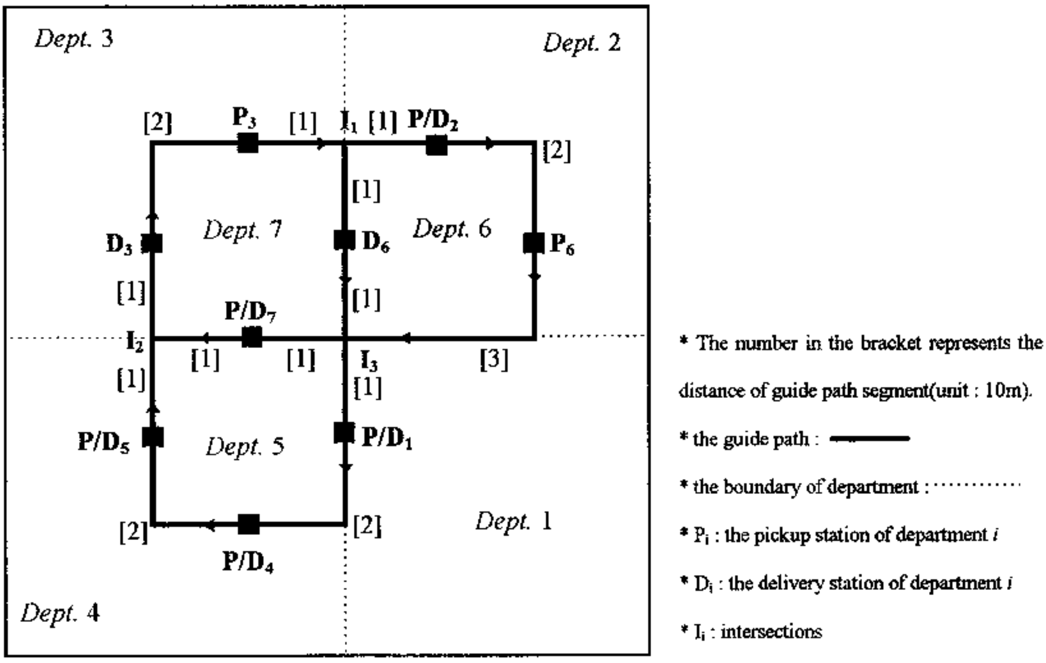

Figure 1. Example of the guide path layout for AGVS

To make the expected response time as short as possible, it is reasonable to locate idle vehicles close to the station with a high probability of the next pickup call. If we assume that pickup requests occur independently

The guide path of AGVS can be represented as a network in which a pickup station, a delivery station and an intersection are replaced by nodes and guide path segments between two of stations and intersections by

Table 2. From/To matrix for flow requirements of Table 1.

From \ To	P/D_1	P/D_2	D_3	P/D_4	P/D_5	D_6	P/D_7	Total	Pickup probability
P/D_1	-	0	10	80	60	0	0	150	0.242
P/D_2	70	-	0	0	0	30	0	100	0.161
P_3	0	80	-	0	0	50	0	130	0.210
P/D_4	0	0	80	-	0	0	0	80	0.129
P/D_5	0	0	40	0	-	0	20	60	0.097
P_6	80	0	0	0	0	-	0	80	0.129
P/D_7	0	20	0	0	0	0	-	20	0.032

directed arcs. Thus, the guide path of Figure 1 may be converted to the corresponding network of Figure 2. At that time, delivery-only stations are excluded from the network, since there is no pickup request from these stations. But, the intersections are included in the network.

The following assumptions are introduced for the problem definition :

① The central zone positioning rule is used to control idle vehicles.

② A single parking area for idle vehicles is provided. We mean by "single" that all the vehicles are dispatched to the parking area when they become idle.

③ Unidirectional guide path is used.

The following notations will be used :

n: the number of pickup nodes on the network,

P_i: the probability that an arbitrary delivery order is issued by node i,

$d(i,j)$: the shortest travel distance of the vehicle from node i to node j,

x: the location of the parking site in the layout which is the decision variable,

v: the travel velocity of vehicle.

The objective function of the problem, the expected response time of an idle vehicle, may be expressed as

$$\underset{x}{\text{Min}} \quad (1/v)\sum_{i=1}^{n} P_i d(x,i). \tag{1}$$

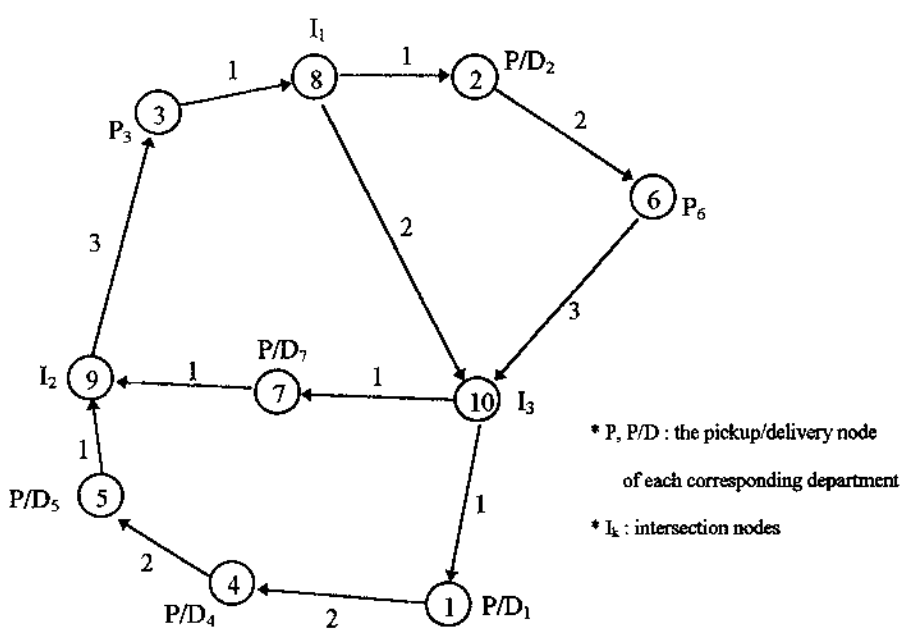

Figure 2. The converted network

If we delete the constant term, (1/v), we finally get the following objective function:

$$\underset{x}{\mathrm{Min}}\ f(x) = \sum_{i=1}^{n} P_i d(x,i). \qquad (2)$$

In the following section, some properties of the problem will be investigated in order to reduce the solution space.

3. Analysis of unidirectional cyclic guide path networks

3.1 Vertex optimality

The formulation (2) in the last section can be interpreted as minimizing the weighted average distance from the parking site to all the other pickup stations.

The following property will reduce the solution spaces significantly:

Property 1. In an AGVS with an unidirectional guide path, the optimal parking site with the least expected response time coincides with a position of the nodes on the converted network.

Proof: If the vehicle is located at an arbitrary node j, then the expected travel distance from node j to all the other nodes becomes

$$f(j) = \sum_{i=1}^{n} P_i d(j,i).$$

Suppose that there is an arc directed from node i to node j. Let j' be a position on the arc apart from node j to the direction of node i by the distance of δ. Then,

$$f(j') = \sum_{i=1}^{n} P_i d(j',i) = \sum_{i=1}^{n} P_i\{d(j,i)+\delta\} = f(j)+\delta.$$

Since δ is a positive constant, it is obvious that $f(j') > f(j)$. Thus, the conclusion holds. Q.E.D.

Property 2. Any merging node (intersection) with no pickup call possibility can not be the optimal location for positioning of idle vehicles.

Proof: See Figure 2. Node 9 is a merging node. If $P_9 = 0$, the expected travel distance decreases by the amount of $d(9,3)$ which is one in this case when the parking location is moved from node 9 to node 3. This statement can be generalized easily. Thus, the conclusion holds. Q.E.D.

3.2 Simplification of the cyclic guide path network by a decomposition

We can reduce the degree of the complexity of the problem by decomposing the converted network into multiple tree networks as follows:

① When there is one or more than one branching nodes, cut each branching node from its succeeding arcs.

② When there is no branching node, the network is a single-loop network. Cut an arbitrary node from its succeeding arc.

For the example problem in Figure 2, we can cut node 8 and node 10 from their succeeding arcs, arc(8,2), arc(8,10) and arc(10,1), arc(10,7), respectively. Then, we get multiple tree networks as shown in Figure 3. Note that a triangular node is attached to the end of each cut arc instead of the original cut node.

And then, we construct a shortest distance matrix from each of root nodes to each of leaf ends (we call "reduced distance matrix"). The reduced distance matrix can be constructed in the following way:

1) Let the root node and the lth leaf end node of the kth decomposed tree be R_k and L_{kl}, respectively. We can construct a reduced network which consists of only the leaf ends and the root nodes as in Figure 4. Note that the distance from every leaf end to the root node of the same decomposed tree can be evaluated in a straightforward way from the property of the tree network and the distance between nodes corresponding to each cut point is zero.

2) Using the reduced network, we can get the shortest distances from each root node to all the leaf ends by applying Dijkstra's method for each root node one by one

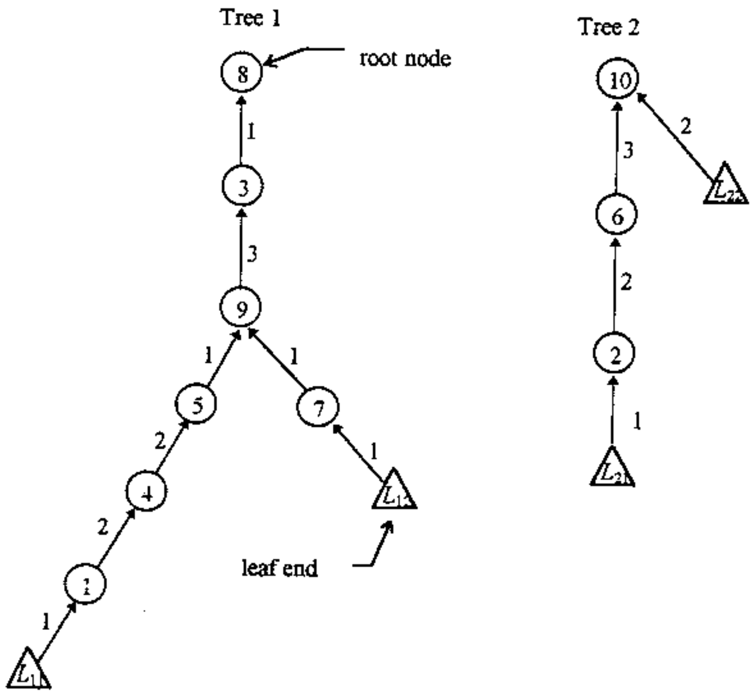

Figure 3. Two decomposed networks of the tree type

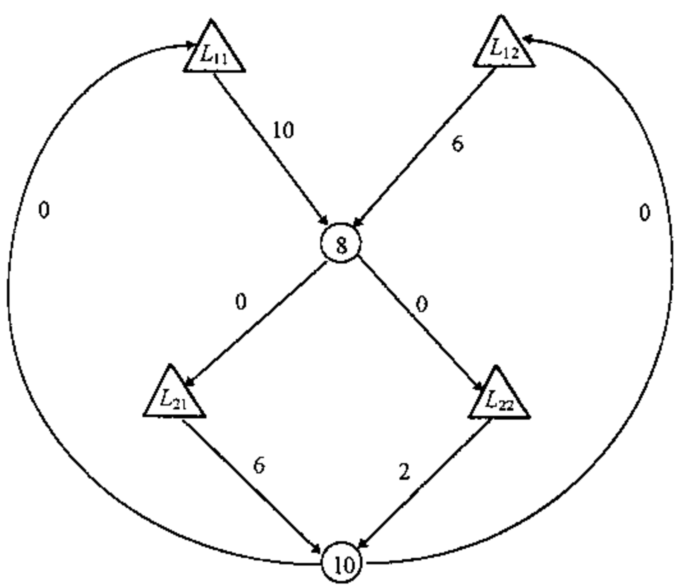

Figure 4. The reduced network

[Francis et al. 1992, Handler et al. 1979]. These shortest distances become values of entries of the reduced distance matrix. For the example, the reduced distance matrix is illustrated in Table 3. Instead of calculating the shortest distance between every pair of nodes which is needed to find the optimal parking location, we only have to calculate this reduced distance matrix and use the distance between nodes in a same tree which can be evaluated much easily. Note that once we get the reduced distance matrix, the distance between arbitrary nodes i and j may be evaluated easily as follows:

① In case that node i is on the sth tree and node j is on the tth tree ($s \neq t$),

$$d(i,j) = d(i,R_s) + \min_l [d(R_s,L_{tl}) + d(L_{tl},j)]. \qquad (3)$$

② In case that node i and node j are on the same tree (s) and node j is not on the path from node i to the root node of tree s,

$$d(i,j) = d(i,R_s) + \min_l [d(R_s,L_{sl}) + d(L_{sl},j)]. \qquad (4)$$

③ In case that node i and node j are on the same tree (s) and node j is on the path from node i to the root node of tree s,

$$d(i,j) = \sum_{(u,v) \in T(i,j)} d(u,v) \qquad (5)$$

where $T(i,j)$ is the set of all the arcs on the unique path from node i to node j.

Note that $d(R_s,L_{tl})$ and $d(R_s,L_{sl})$ in (3) and (4) are obtained from the reduced distance matrix and the value $d(L_{tl},j)$ and $d(L_{sl},j)$ in (3) and (4) may be evaluated in the same way as (5).

Table 3. The reduced distance matrix of the example

From \ To	L_{11}	L_{12}	L_{21}	L_{22}
8(R_1)	2	2	0	0
10(R_2)	0	0	6	6

3.3 Relative differences in the expected response time among nodes on a tree

In this section, we provide a methodology to evaluate the relative differences in the expected response time among nodes in each decomposed tree. Following a proper sequence of a tree search (for example, see the sequence in Figure 5 for the case of the depth first search), we examine each node as follows:

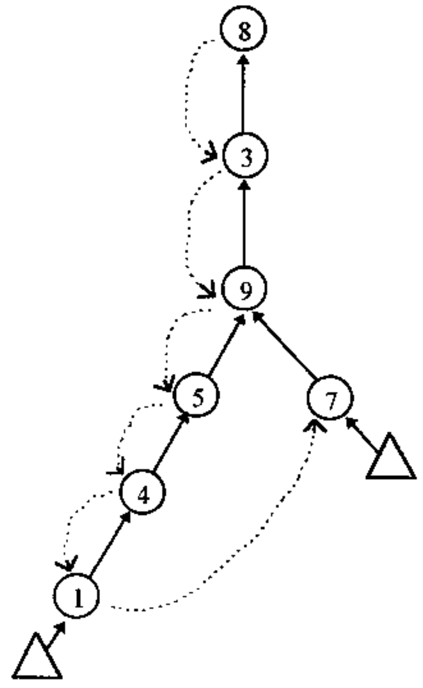

Figure 5. The evaluation sequence based on the depth-first search

We evaluate the relative difference in the value of the expected travel distance under the condition that the parking area is located at the corresponding node. At that time, the root node becomes the reference point.

In Figure 6, suppose that the location of the parking site, x, moves from node j to node i. Then, the travel distance from the parking site to node i decreases by the amount of $d(j,i)$, while the travel distances to the other nodes increase by the amount of $d(i,j)$.

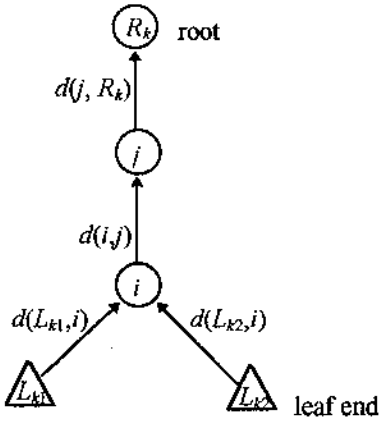

Figure 6. A decomposed tree network

Therefore, the amount of change in the expected travel distance of (2) becomes

$$
\begin{aligned}
& d(i,j)(1-P_i) - d(j,i)P_i \\
&= d(i,j) - [d(i,j) + d(j,i)] \times P_i. \\
&= d(i,j) - \min_{l} [d(L_{kl},R_k) + d(R_k,L_{kl})] \times P_i. \qquad (6)
\end{aligned}
$$

Thus, the relative difference between any two adjacent nodes in a decomposed tree can be evaluated by equation (6) easily. Note that $d(R_k, L_{kl})$ can be obtained directly from the reduced distance matrix and $d(i,j)$ and $d(L_{kl}, R_k)$ are distances between nodes on the same tree which can be evaluated easily.

For example, when the parking site moves from node 3 to node 9 in Figure 3, the expected travel distance changes by the amount of $d(9,3)$ - *min* [$d(L_{11},8) + d(8, L_{11})$, $d(L_{12},8) + d(8,L_{12})$] P_9. Therefore, it becomes 3 - (6 + 2) × 0 = 3, which is the relative difference of the expected travel distance between node 3 to 9.

Let W_x be the cumulative sum of the change ((6))

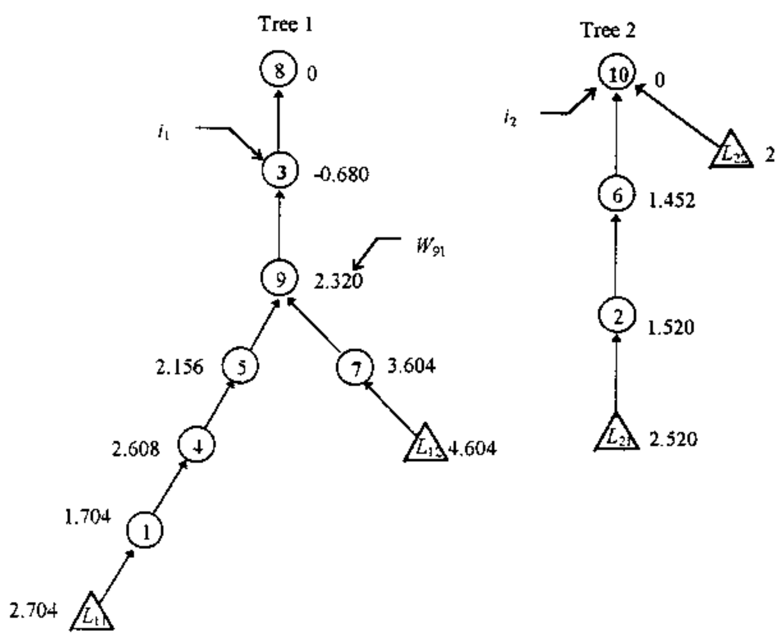

Figure 7. W_x and candidate nodes for the example

from the root node to node i on the kth tree network. Then, W_{ik} represents the relative difference of the expected response time between the case of the parking area located at node i and the one located at the root node.

Now, we can choose a node which has the lowest value of W_{ik} in tree k, which we call the candidate node (i_k) of tree k. Figure 7 illustrates W_{ik} and candidate nodes for the example.

Property 3. All the nodes on tree k except the one with the lowest value of W_{ik} can be eliminated from the further consideration as candidates for the optimal parking location.

Proof: The conclusion directly follows from the definition of W_{ik}. Q.E.D.

Property 3 eliminates a significant number of nodes from the solution space for the optimal parking location.

3.4 Indifference of the relative preference among candidate nodes to the locations of non-candidate nodes

We can decompose each tree network further into multiple segments by cutting the network at merging nodes as shown in Figure 8. Note that we become to have two additional nodes which corresponds to the locations of cuts at a merging node. Then the following property holds:

Property 4. Any movement of a non-candidate node does not change the relative preference among candidate nodes as the parking location only if the non-candidate node stays on the same segment and does not pass the candidate node on the same segment ahead.

Proof: Suppose that the location of node j is changed by the amount of d to the direction of the root. Let the new location be j'. Then, its expected response time from $x = x_i$, which is the ith candidate node, becomes

$$\sum_{k=1}^{j-1} P_k \times d(x_i,k) + P_j \times d(x_i,j') + \sum_{k=j-1}^{n} P_k d(x_i,k).$$

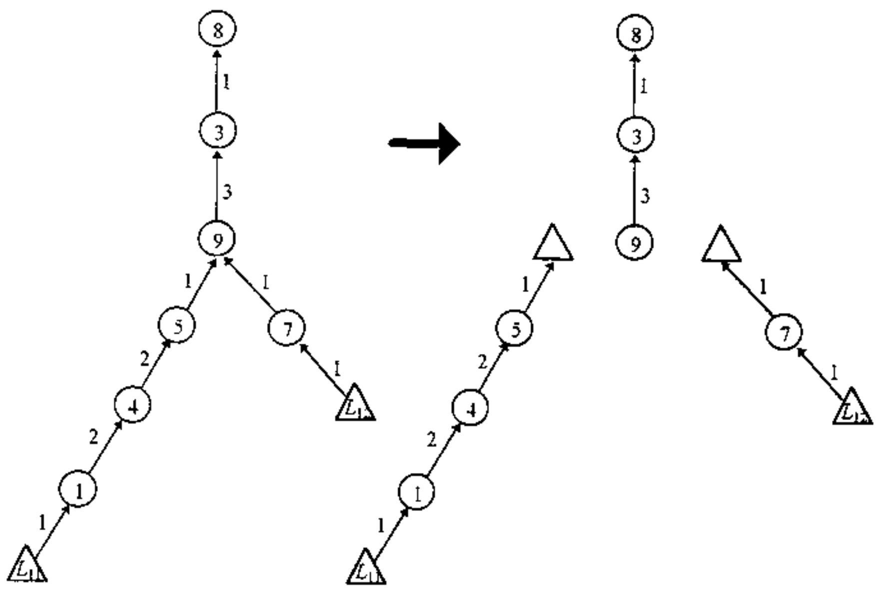

Figure 8. Illustration for further decomposing of a tree network

Note that $d(x_i,j')$ is evaluated by the same formula as $d(x_i,j)$ among (3), (4) and (5) from the condition of the possible movement and $d(x_i,j') = d(x_i,j) + \alpha$. Thus, the expected response time from $x = x_i$ becomes

$$\sum_{k=1}^{n} P_k \times d(x_i,k) + \delta. \tag{7}$$

The relationship of (7) holds for all the candidate node x, which implies that the expected response time from every candidate node changes by the same amount. Thus, the conclusion holds. Q.E.D.

4. Finding the optimal parking location and the evaluation of the expected response time

4.1 Finding the optimal parking location on a general guide path network

In the following, we suggest a procedure to find the optimal parking location when the guide path network is of the general cyclic network type utilizing the properties of the previous section:

Step 1: *Decompose the network into multiple tree networks by cutting off each branching node from its succeeding arcs.*

The guide path network is transformed into the converted network as in Figure 2 and the converted network is decomposed into multiple tree networks as in Figure 3.

Step 2: *Construct the reduced distance matrix.*

Utilizing the root nodes and the leaf nodes in the decomposed tree networks, the reduced network is constructed as in Figure 4. And the reduced distance matrix is constructed based on the reduced network as in Table 3.

Step 3: *Choose a candidate node on each decomposed tree network by evaluating the relative differences in the expected response time among nodes and choosing the node with the lowest value on the tree network.*

For each decomposed tree network, the cumulative sum of change in the expected response time from the root node (W_x) is evaluated and the node with the lowest value of W_x is selected as a candidate node for the optimal parking location. For the example problem, they are illustrated in Figure 7.

Step 4: *Condense multiple nodes on a segment into one or two, which results in the significant reduction in the number of nodes.*

The network can be further simplified by condensing multiple nodes on each segment into one or two nodes using property 4 as follows:

For a segment which contains the candidate node, (1) condense all the nodes preceding the candidate node into one by summing all P_i's and locate the resulting node at the location of an arbitrary condensed node, (2) and condense all the nodes succeeding the candidate node (including the candidate node itself) into the candidate node.

For a segment without a candidate node, condense all the nodes on the segment into one arbitrary node on the segment.

For the example problem, we can condense node 4 and node 5 into node 1, node 6 into node 2 and node 8 into a candidate node 3. Now, we become to have six condensed nodes of node 1, 2, 3, 7, 9 and 10 with pickup probabilities of 0.468, 0.290, 0.210, 0.032, 0.0 and 0.0, respectively. Among these nodes, node 3 and 10 are also candidate nodes.

Step 5: *Compare the objective function values of the candidate nodes each other by evaluating the expected travel distance from each candidate node to the condensed nodes. Select one candidate node with the lowest objective value.*

We find the optimal parking area by evaluating

$$\operatorname*{Min}_{i} \sum_{\substack{i \in C \\ j \in D}} \tilde{P}_j d(i,j). \tag{8}$$

where C is the set of candidate nodes and D is the set of condensed nodes, and $\tilde{P}_j$ is the revised pickup probability.

Table 4 illustrates the travel distance and the expected travel distance from each candidate node for the example. Node 3, P_3, is the optimal parking location.

Table 4. The expected travel distance for the example

Candidates \ Condensed nodes	1	2	3	7	The expected travel distance (unit:10m)
3	4	2	0	4	2.580
10	1	7	5	1	3.580
$\tilde{P}_j$	0.468	0.290	0.210	0.032	

In order to get the objective value of the original function (1) or (2), we have to modify the value of (8) by the amount corresponding to the changed distance.

The expected travel distance of the example problem becomes 34.84m(= 25.80 + 9.04) and the expected vehicle response time is 38.71sec.

Discussions on the computational complexity

If we use the Dijkstra's method to find the shortest path between every pair of nodes on the network, $(3n(n-1)/2)n$ operations are required when n is the number of nodes [Evans et al. 1992]. And in order to find the optimal median solution, $n(n-1)$ multiplication, $n(n-1)$ additions and $(n-1)$ comparisons are required additionally. In total, $3n^2(n-1)/2 + 2n(n-1) + (n-1)$ operations are required.

But, $\sum_{k=1}^{m} 3n_k(n_k+1)/2$ operations, where m is the number of the decomposed tree networks and n_k is the number of the leaf nodes of the kth decomposed tree, are required to find the shortest path from root nodes to every leaf node in the suggested algorithm. To find the optimal solution, $\sum_{k=1}^{m}(n_k+1)$ multiplications, $\sum_{k=1}^{m}(n_k+1)$ additions, $(m-1)$ comparisons are needed additionally. In total, $\sum_{k=1}^{m}(3n_k(n_k+1))/2 + 2\sum_{k=1}^{m}(n_k+1) + (m-1)$ operations are required approximately.

For the example problem, which has 10 nodes and 2 decomposed trees, 1539 operations in the full enumeration and 167 operations in our algorithm are required. Therefore, the suggested algorithm can reduce the calculation time significantly.

4.2 Finding the optimal parking location on a guide path of a pure single-loop type

A pure single-loop guide path for AGVS is an effective approach to reduce the complexity of the control logic for vehicles and is illustrated in Figure 9. This guide path layout can be converted to the corresponding single-loop network as in Figure 10.

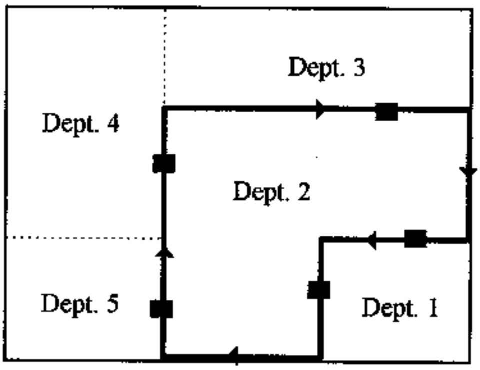

Figure 9. A single-loop guide path for AGVS

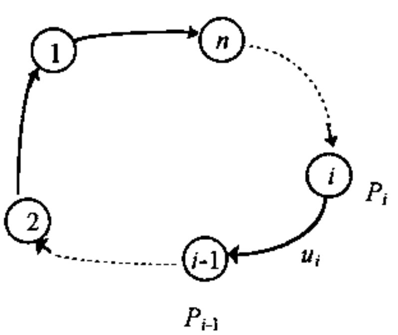

Figure 10. An example of a single-loop network

This single-loop network is converted into a tree network with a single root and a single leaf end by cutting an arbitrary node from its succeeding arc. We can apply the suggested algorithm to get the candidate node which is the very optimal solution from property 3.

Let T be the total length of the loop and u_i be the length from a node i to the adjacent node in the direction of path as shown in Figure 10. Then the following property holds :

Property 5. If $T \times P_i < u_i$ in case of a loop type guide path, node i can not be the optimal location.

Proof: From equation (6), we can get

$$f(i\text{-}1) - f(i) = T \times P_i - u_i. \quad (9)$$

If equation (9) is less than zero, then $f(i\text{-}1) < f(i)$. Therefore, node i can not be the optimal location. Thus, the conclusion holds. Q.E.D.

Algorithm for a pure single-loop type network

Step 1: Cut an arbitrary node.

Without loss of generality, we will assume that the cut node is node 1.

$s = 1$ and $W_s = 0$.

$W^* = 0$, $s^* = 1$.

Step 2: Choose the optimal parking location.

$s = s + 1$.

If $s > n$, stop.

Otherwise, $W_s = W_{s-1} + d(s,s\text{-}1) - T \times P_s$.

If $W_s < W^*$, $W^* = W_s$ and $s^* = s$. Go to the beginning of step 2.

4.3 Finding the optimal parking location on a guide path with a single branching intersection

A guide path layout of AGVS for a part delivery from a storage location to several different assembly lines is illustrated in Figure 11-(a). This guide path layout is converted to a loop type network with a single branching node.

This network has only one branching node. Thus, we can convert it into a single tree network by cutting the branching node from its succeeding arcs. The converted tree has a single root and multiple leaf ends and so it is of the fork form as in Figure 11-(b). We can apply the suggested algorithm to find the candidate node on a tree network. Then, the candidate node is the very optimal solution.

4.4 Finding the optimal parking location on a guide path of the loop type with local branching nodes

The guide path layout of AGVS for assembly lines is

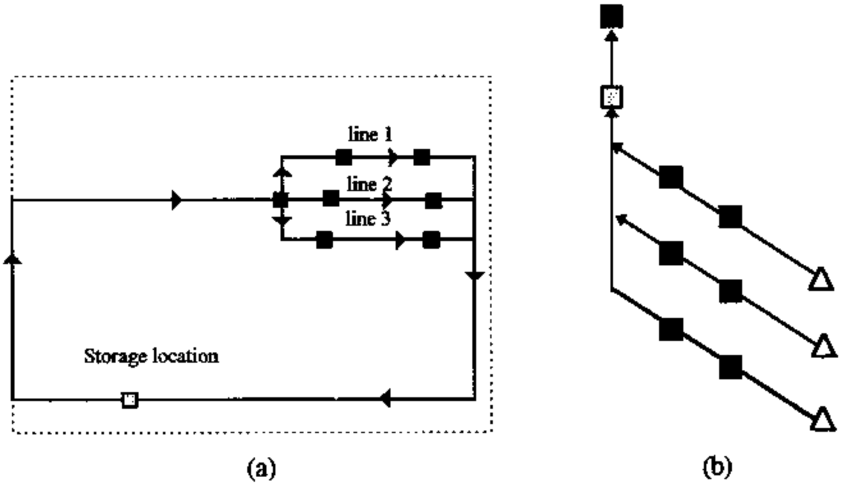

Figure 11. The guide path of AGVS for part delivery for assembly stations (a) and its converted tree network (b)

illustrated in Figure 12. This layout can be converted to the loop type network with multiple branching nodes as shown in Figure 13.

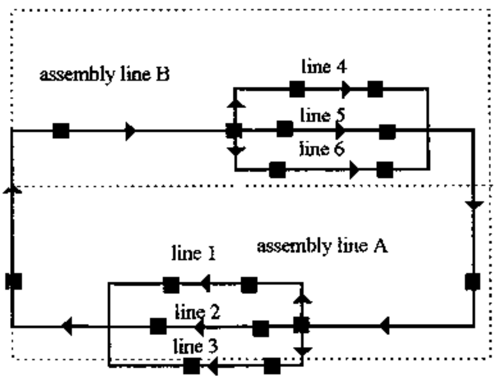

Figure 12. The guide path of AGVS for assembly lines

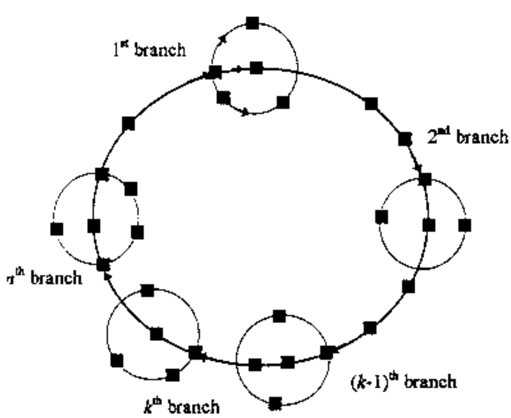

Figure 13. A loop type network with local branching nodes

Suppose that there are n local branches as in Figure 13. When we cut off the branching nodes, n decomposed tree networks are obtained.

To find out a candidate node on each decomposed tree in the procedure of section 3, the reference value W_{ik} of the root node was set to be zero. But, utilizing a special structure of the network, we can easily evaluate the absolute value of the expected travel distance for each root node without evaluating the objective function value at each candidate node by (8) as follow:

Suppose that the absolute value of the objective function at root node and the ith node on the kth tree be Y_{ok} and Y_{ik}, respectively. Then, $Y_{ik} = Y_{ok} + W_{ik}$ for all nodes on the kth tree network.

Let the node L_{sk} be the leaf node which is the nearest to the merging node on the kth tree. And let

T = the shortest turn-around travel time starting from a merging node and returning to the same node,

S_k = the sum of weights of all the nodes on the branches of the kth tree network excluding the branch with the leaf node L_{sk},

P_{ok} = the probability that an arbitrary delivery order is issued by the root node of the kth tree network. Then,

$$Y_{o(k-1)} + TP_{o(k-1)} + TS_k = Y_{L_sk} = Y_{ok} + W_{L_sk},$$
$$k=1,2,...,n \text{ and } Y_{oo} = Y_{on} \text{ and } P_{oo} = P_{on}. \quad (10)$$

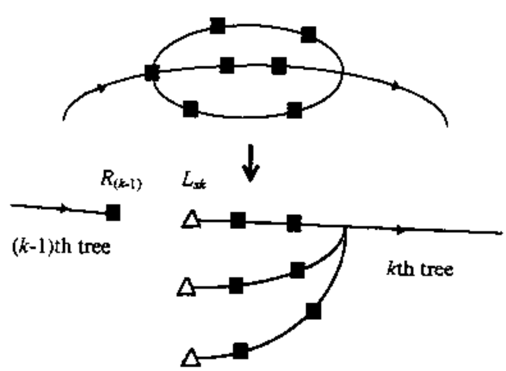

Figure 14. Decomposed tree networks with local branching nodes

The left hand side of equation (10) evaluates the expected travel distance at the leaf node L_{sk} by adding the change resulting from the move of the parking location from $R_{(k-1)}$ to L_{sk} to the expected travel distance at $R_{(k-1)}$. The right hand side of equation (10) represents the expected travel distance at L_{sk} which is obtained by adding the relative difference of L_{si} from the one of R_k to the expected travel distance at R_k.

Since we have the same number of equations as the one of the unknown variables Y_{ok}'s, we can solve the simultaneous equations to get the value of the absolute values of the expected travel distance at each root node. Once we get the values for root nodes, it is straight forward to get those for all the candidate nodes. Thus, we become to get the optimal solution without using a shortest path finding algorithm.

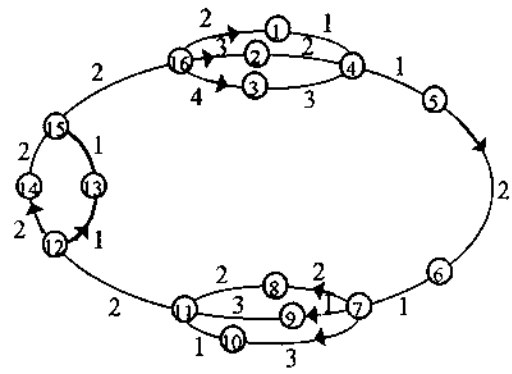

Figure 15. Example of a loop type network with 3 local branching nodes

A numerical example

In this section, we illustrate the solution procedure for the example in Figure 15.

We assumed that P_i's are 0.05, 0.1, 0.08, 0, 0, 0, 0.2, 0.05, 0.01, 0.1, 0.02, 0.1, 0.15, 0.02, 0.02, and 0.1 in ascending order of node numbers. Then, the network is decomposed into 3 tree networks as in Figure 16. From

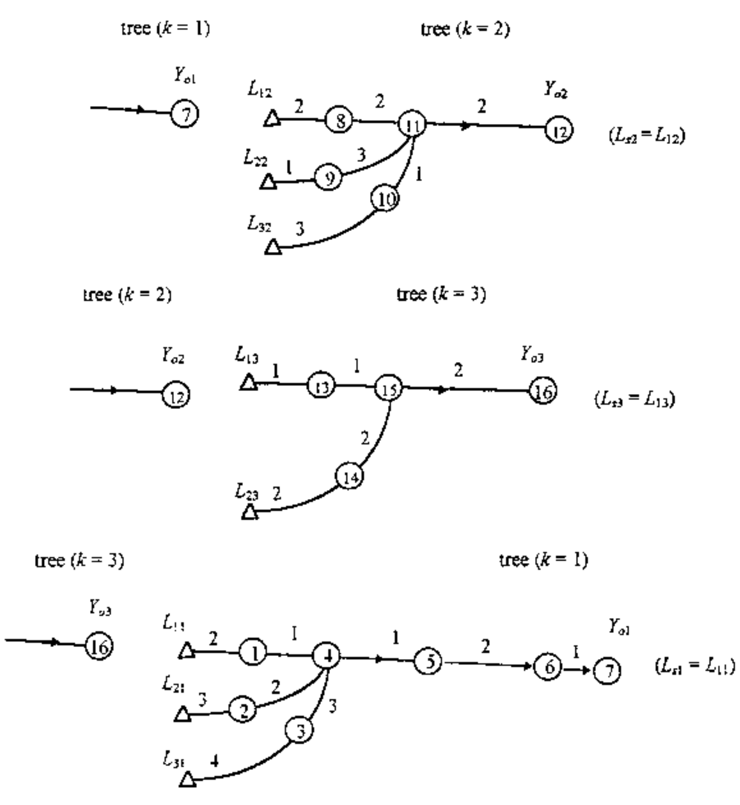

Figure 16. Decomposed tree networks of the example

equation (6), we can find the candidate nodes for three tree networks and know W_{jk} for all nodes on three tree networks. Node 7 on tree 1, node 12 on tree 2, and node 16 on tree 3 are candidate nodes. In the next, we can construct three simultaneous equations as follows:

$$Y_{o1} - Y_{o2} = -0.46 \qquad (11)$$

$$Y_{o2} - Y_{o3} = -0.93 \qquad (12)$$

$$Y_{o3} - Y_{o1} = 1.39 \qquad (13)$$

If we set $Y_{o1} = 0$, we can get $Y_{o2} = 0.46$ and $Y_{o3} = 1.39$. Note that the equations (11)~(13) have multiple solutions. Now, we can determine that the optimal parking location is node 7.

5. Evaluating the expected response time at an arbitrary location on a general guide path network

The theoretical optimal parking location may not be realizable on an actual guide path for AGVS because of various physical limitations. At that time, we should be able to evaluate the expected response time quickly for any next promising location on the guide path network. In the following, we explain how to evaluate the expected response time at an arbitrary position.

By (8), we already know the value of $f(i_k)$ for every candidate node i_k. And W_{jk}'s are known for all the nodes on the guide path network. Then, we can get easily the value of $f(j)$ of node j on the kth tree network by evaluating

$$f(j) = W_{jk} + f(i_k) - W_{i_k k} \qquad (14)$$

Remember that W_{jk} represents the relative difference of $f(j)$ from a reference point (root node). Since the absolute value of $f(i_k)$ is known now, it can be used to evaluate $f(j)$ of all the nodes on the tree network.

Then, we can plot the curve of the expected travel distance for each segment on the network as follows:

(1) Let x_i be the distance of node i from the node(node 9) on the same segment which is located nearest to the root node(reference point, node 8 in the example problem) of the corresponding tree network. Then, we can mark the coordinates $(x_i, f(i))$ for all the nodes on the segment.

(2) We draw a straight line to the right whose slope is +1, starting from each marked point $(x_j, f(j))$ until it meets a vertical line that passes the next coordinate $(x_i, f(i))$.

For the example problem, the curve for a segment of Figure 8 is plotted in Figure 17.

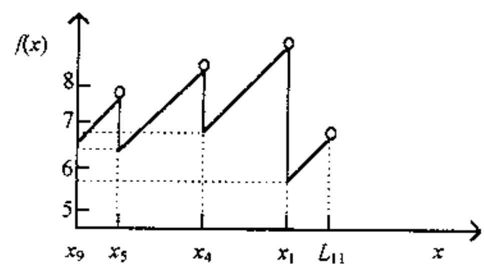

Figure 17. The curve of the expected travel distance on a segment

When we want to evaluate the expected travel distance for a parking area located at an arbitrary position between

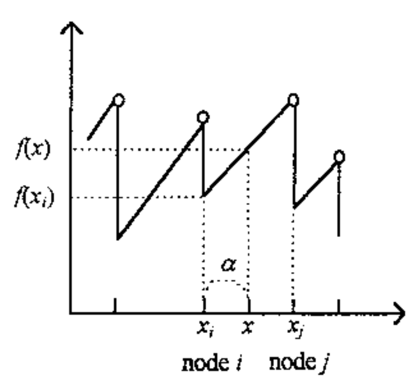

Figure 18. The expected travel distance from an arbitrary position on a segment

nodes, it can be evaluated as follows:

Let x be a position on the arc apart from node i to the direction of node j by the distance a as shown in Figure 18.

Then,

$$f(x) = f(x_i) + a \quad (15)$$

6. Conclusions

One of the important operation policies for AGVS is the positioning strategy of idle vehicles on the guide path, since it significantly affects the service time of AGVS for a load pickup request. Although several researchers dealt with the positioning problem, they all assumed that the guide path is a simple loop type.

In this paper, we discussed how to locate idle vehicles of AGVS on a general unidirectional guide path network. We analyze the characteristics of the optimal location and suggest some useful properties to reduce the solution space. A simplified process is suggested to find the optimal parking location which consists of decomposing the original guide path network into multiple tree networks, constructing the reduced distance matrix, choosing a candidate node on each tree network, condensing multiple nodes on a segment into one or two, and evaluating the candidate nodes. Each step of the procedure is illustrated using a numerical example. And, the computational complexity of the algorithm developed is analyzed. Several special types of networks are analyzed in addition to the general cyclic network. Those includes pure single-loop type, loop type with a single branching node, and loop type with local branching nodes. It is shown that much simpler algorithm may be applied to each type of special guide paths. In case of each type of special guide paths, it is illustrated using some example problems. Methodology of evaluating the expected response time at an arbitrary location on a general guide path network is provided.

In this research, we considered only unidirectional cyclic guide path network. The result of this study may be extended to the case of bidirectional guide paths in the future study. And we did not consider the problem with the multiple parking sites and the dynamic positioning problem. Thus, these problems may be promising topics for the further extensions.

References

[1] Chang, S.H. and Egbelu, P.J., Dynamic relative positioning of AGVs in a loop layout to minimize mean system response time, *International Journal of Production Research*, Vol.34, No.6, pp.1655-1673, 1996.

[2] Co, G. Christine and Tanchoco, J. M. A., A review of research on AGVS vehicle management, *Engineering Costs and Production Economics*, Vol.21, pp. 35-42, 1991.

[3] Egbelu, P. J., Framework for dynamic positioning of storage/retrieval machines in an automated storage/retrieval system, *International Journal of Production Research*, Vol.29, No.1, pp.17-37, 1991.

[4] Egbelu, P. J., Positioning of automated guided vehicles in a loop layout to improve response time, *European Journal of Operational Research*, Vol.71, pp.32-44, 1993.

[5] Evans, James R. and Minieka, Edward, *Optimization Algorithms for Networks and Graphs*, 2nd Ed., Marcel Dekker, Inc., New York, USA, 1992.

[6] Francis, R.L., McGinnis, Jr., Leon F. and John A. White, *Facility Layout and Location: An Analytical Approach*, 2nd Ed., Prentice Hall, New Jersey, USA, 1992.

[7] Handler, Gabriel Y. and Mirchandani, Pitu B., *Location on Networks Theory and Algorithm*, The MIT Press, Cambridge, Massachusettes, USA, 1979.

[8] Kim, K. H., Positioning of automated guided vehicles in a loop layout to minimize the mean vehicle response time, *International Journal of Production*

Economics, Vol.39, pp.201-214, 1995.

[9] Sinriech, D. and Tanchoco, J.M.A., Intersection graph method for AGV flow path design, *International Journal of Production Research*, Vol.29, No.9, pp. 1725-1732, 1991.

[10] Sinriech, D. and Tanchoco, J. M. A., The Centroid Projection Method for Locating Pick-Up and Delivery Stations in Single-Loop AGV Systems, *Journal of Manufacturing Systems*, Vol.11, No.4, pp. 297-307, 1992.

[11] Tanchoco, J. M. A. and Sinriech, D., OSL-optimal single loop guide paths for AGVS, *International Journal of Production Research*, Vol.30, No.3, pp. 665-681, 1992.

97년 6월 최초 접수, 98년 2월 최종 수정