Locating Idle Vehicles in Tandem-Loop Automated Guided Vehicle Systems to Minimize the Maximum Response Time

Shiwoo Lee [†]

School of Mechanical, Industrial and Manufacturing Engineering
Oregon State University, Corvallis, OR, USA
+1-541-737-4505, Email: Shiwoo.Lee@oregonstate.edu

Abstract. An automated guided vehicle (AGV) system is a group of collaborating unmanned vehicles which is commonly used for transporting materials within manufacturing, warehousing, or distribution systems. The performance of an AGV system depends on the dispatching rules used to assign vehicles to pickup requests, the vehicle routing protocols, and the home location of idle vehicles, which are called dwell points. In manufacturing and distribution environments which emphasize just-in-time principles, performance measures for material handling are based on response times for pickup requests and equipment utilization. In an AGV system, the response time for a pickup request is the time that it takes for the vehicle to travel from its dwell point to the pickup station. In this article, an exact dynamic programming algorithm for selecting dwell points in a tandem-loop multiple-vehicle AGV system is presented. The objective of the model is to minimize the maximum response time for all pickup requests in a given shift. The recursive algorithm considers time restrictions on the availability of vehicles during the shift.

Keywords: Facility Design, Logistics, Automated Guided Vehicle (AGV)

1. INTRODUCTION

A flexible manufacturing system consists of a group of material processing cells connected by an automated material handling system to manufacture a wide variety of different products with low-to-medium volume. Among various material handling systems that are employed in flexible manufacturing environments today, automated guided vehicle (AGV) systems have acquired greater importance and attention. An AGV system features battery-powered and driver-less vehicles moving on a layout of guide paths. Each vehicle has programming capability for path and location selection, and can be reconfigured easily to accommodate changes in pro-duction volume, product mix, product routing, and equipment interfacing requirements (Rajotia *et al.* 1998).

Achievements of high flexibility and high performance in an AGV system are related to several design and control issues. Many studies have been done to address some of these issues. These problems include (i) guide path layout design and location selection of pickup/dropoff (P/D) stations (Egbelu 1993), (ii) vehicle scheduling and dispatching, (iii) traffic control and ve-

hicle routing, (iv) determination of the number of necessary vehicles, and (v) home location selection for idle vehicles (Lee and Ventura 2001, Ventura and Lee 2001, Ventura and Rieksts 2006).
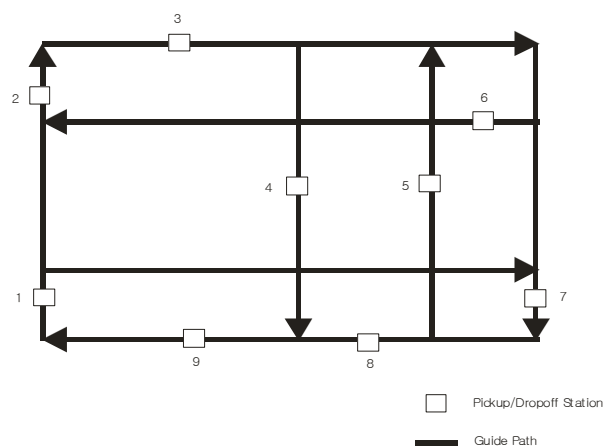

Figure 1. Example of a Conventional network layout with 8 pickup stations

---

† : Corresponding Author

There are several AGV guide path layouts which have been proposed in the literature. The conventional network layout usually employs a general or grid network layout with unidirectional guide paths. A typical conventional network layout is shown in Figure 1. Because of intersections in the layout, a complex control scheme is required to avoid deadlocks and congestion. Egbelu and Tanchoco (1986) proposed the single-loop layout for an AGV system. As shown in Figure 2, a typical single-loop layout contains a unidirectional guide path with one or more AGVs. An important advantage for using the single-loop layout is the simplicity of traffic control, where neither deadlock nor collision exists.
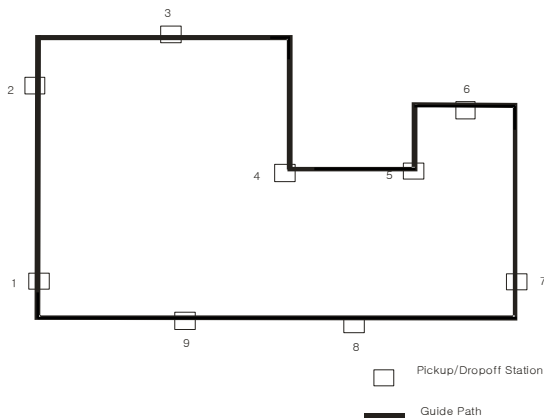

Figure 2. Example of a single-loop layout with 8 pickup stations

The single-loop layout also needs a shorter length of guide path than other layouts, which may reduce the significant capital cost of installation. The disadvantages are long vehicle travel times and difficulties in reconfiguration upon system changes (Ross 1996, Sinriech 1995). To overcome the disadvantages of the single loop layout, Bozer and Srinivasan (1992) proposed the tandem loop layout, consisting of non-overlapping single-vehicle bidirectional loops (see Figure 3). Material transfer between loops is completed at pre-determined locations (or *transfer ports*) through transfer devices such as conveyors, gravity rails, AGVs, etc. There have been many studies on partitioning P/D stations into multiple tandem loops. By having one vehicle per loop, this layout is also free of congestion and deadlocks. An advantage over the single loop layout is the shorter travel distance of vehicles due to smaller loops and bidirectional guide path. However, a significant potential of system failure exists because of the condition that each loop must have only one vehicle. It makes the failure of a single vehicle directly to risk the integrity of the system operation. Partitioning of a large loop into smaller loops must be carefully designed, so that each vehicle can cope with the workload associated with its loop. In addition, interloop traffic tends to rapidly increase with a growing number of loops in the

system, which may jeopardize the system performance. The disadvantages of the tandem loop layout indicate that there is an optimal partitioning level for every system (Sinriech and Tanchoco 1992).
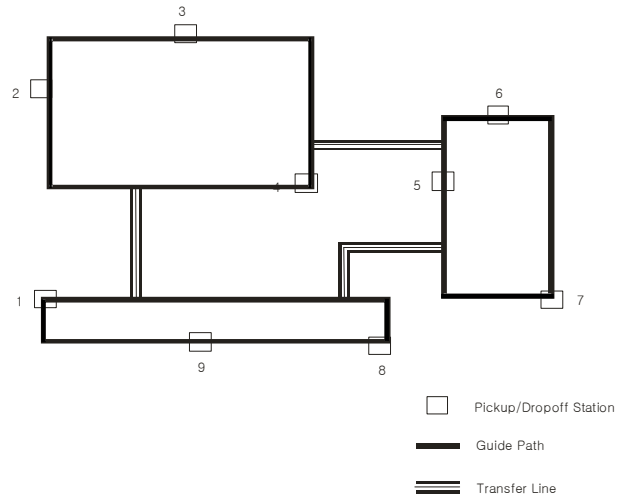

Figure 3. Example of a tandem-loop layout with 8 pickup stations

Ventura and Lee (2001) proposed a system of tandem-loops with multiple-vehicles (TLMV) to avoid system interruptions that could be caused by a single vehicle failure in the tandem-loop layout. Like the tandem-loop layout, the TLMV system consists of non-overlapping loops and transfers between adjacent loops occur at transfer ports. In this research, a transfer port is either placed at an existing P/D station or converted to an additional P/D station acting as a pickup station for incoming traffics and a dropoff station for outgoing traffics. The TLMV system may contain more than one AGV in each loop and a unidirectional guide path is usually employed to avoid collisions. However, with some buffering space (Egbelu 1993), a bidirectional guide path can be implemented as well. The TLMV system can evenly distribute the workload among AGVs by assigning an adequate number of vehicles to each loop.

Unless an AGV system is overloaded, the occurrence of vehicle idleness is an inevitable event (Egbelu 1993). Vehicle idleness occurs whenever a vehicle completes a P/D task and there is neither immediate nor pending pickup request. Under the *dwell point positioning rule*, when a vehicle becomes idle, it moves to its home location (dwell point), and waits there until a new P/D task is assigned to the AGV. Optimal dwell points are determined by assigning P/D stations to vehicles so that the maximum vehicle response time or the mean vehicle response time is minimized. The response time is defined as the time elapsed from the moment the vehicle is assigned to a P/D task until the time instance in which the vehicle reaches the pickup station. Ventura and Lee (2001) performed a simulation and a cost analy-

sis on the four basic types of layouts: conventional network layout, single loop layout, tandem loop layout, and TLMV layout. They concluded the unidirectional TLMV layout gave the best performance with reasonable installation and operating costs. Ventura and Lee performed a second simulation study comparing three idle vehicle positioning rules in an AGV system with TLMV layout. They concluded that the dwell point positioning rule outperformed the *circulatory loop positioning* rule and the *point-of-release positioning rule*. In the circulatory loop positioning rule, idle vehicles circulate around their pre-assigned loop until they are reassigned to a new task. Upon the completion of a P/D task, under the point-of-release positioning rule, the vehicle remains at the dropoff station until it is reassigned to another task. Ventura and Rieksts (2006) studied optimal locations of dwell points in a single-loop AGV system with time restrictions on vehicle availability.

In this article, a polynomial-time dynamic programming (DP) algorithm for determining dwell points for an AGV system with a TLMV layout is presented. The algorithm considers time constraints concerning the vehicle availability per shift. The recursive algorithm provides an exact solution to the problem of minimizing the maximum response for all requests in a shift.

Section 2 introduces the necessary notation and assumptions to formulate the problem as an integer nonlinear program. The proposed DP algorithm and an illustrative example are provided in Sections 3 and 4, respectively. The conclusions of this research are summarized in Section 5.

2. PROBLEM DESCRIPTION

In the TLMV layout under consideration, a station will send a pickup request to the controller and the controller will dispatch an AGV, which has been pre-assigned to a set of consecutive stations, if it is available. Otherwise, the controller will delay the dispatching order until the AGV becomes available. The dispatched vehicle will pick up the job (a unit load of parts) and deliver it to the dropoff station. Upon completion of the task, the vehicle will return to its dwell point and wait until it is reassigned to another request. When the pickup and dropoff stations belong to the same loop, the task is completed by a single AGV without transfers between tandem loops. Such a trip is defined as *intra-loop travel*. When pickup and dropoff stations belong to different loops, an AGV first travels to the pickup station and moves the job to an appropriate transfer port. A transfer device then moves the job to another loop and an AGV in the second loop needs to pickup the job at the transfer port, and so on. The process continues until the job reaches its destination loop. Such a trip is defined as *interloop travel*. For this interloop travel, a control issue needs to be resolved concerning the routing of the job from pickup stations to dropoff stations. In interloop travel, the incoming traffic from the previous loops through transfer devices will become additional pickup requests in the successive loops. This additional traffic can create congestion problems if routing protocols are not well-planned. A similar situation occurs in Internet routing. There have been extensive studies on how to get data packets to destinations in the Internet, which is similar to getting loaded vehicles to dropoff stations in the TLMV system. This process is a great burden for the central controller that needs to figure out the shortest routing between each pair of P/D stations, especially in a dynamic situation like the Internet. In the TLMV system considered in this research, the layout is static over the time and it is feasible for the controller to keep the routing information between any pair of stations. Therefore, each dispatched vehicle knows the intermediate loops to get to the destination station in advance, which is called *source routing* in communication networks. The initial routing information can be obtained by a shortest-path algorithm for the best efficiency. The controller in the TLMV system should update the routing information to detour upon system changes due to transfer device failures or traffic congestion.

By having a side track at each P/D station and transfer port in the TLMV system, no vehicle inter-ference will take place. For control simplicity, it is assumed that a job can be picked up and another job delivered simultaneously at any station. When a tandem loop layout is given, there are $n^{(k)}$ P/D stations and $p^{(k)}$ transfer ports in loop k. As noted above, $p^{(k)}$ transfer ports are converted to additional P/D stations in loop k by being pickup stations for the incoming traffics and being dropoff stations for the outgoing traffics, so that there exist $m^{(k)} = n^{(k)} + p^{(k)}$ stations in loop k. These P/D stations (including transfer ports) are denoted as $v_i^{(k)}$ for the location of station i in loop k and $x_j^{(k)}$ as the location of dwell point j in loop k. In addition, let $f_{ij}^{(k)}$, $i \neq j$, denote the number of pick-up requests from $v_i^{(k)}$ to $v_j^{(k)}$ per shift,

$$f_i^{(k)} = \sum_{j \in M^{(k)} - \{i\}} f_{ij}^{(k)}$$ the number of pickup requests from

$v_i^{(k)}$ per shift, and $$f^{(k)} = \sum_{i \in M^{(k)}} f_i^{(k)}$$ the total number of

pick-up requests per shift in loop k, where $M^{(k)} = \{1, 2, 3, \ldots, m^{(k)}\}$. A unidirectional single loop layout can be modeled as a digraph $G^{(k)} = (V^{(k)}, A^{(k)})$, where $V^{(k)} = \{v_1^{(k)}, v_2^{(k)}, \ldots, v_{m^{(k)}}^{(k)}\}$ is the set of nodes and $A^{(k)} = \{(v_1^{(k)}, v_2^{(k)}), (v_2^{(k)}, v_3^{(k)}), \ldots, (v_{m^{(k)}-1}^{(k)}, v_{m^{(k)}}^{(k)}), (v_{m^{(k)}}^{(k)}, v_1^{(k)})\}$ is the set of arcs. Let $X^{(k)} = \{x_1^{(k)}, x_2^{(k)}, \ldots, x_{h^{(k)}}^{(k)}\}$ be a set of $h^{(k)}$ dwell points of AGVs in $G^{(k)}$. For simplicity, all stations and dwell points can be labeled sequentially in the direction of movement without loss of generality.

The distance between locations $v_i^{(k)}$ and $v_j^{(k)}$ in loop k is denoted as $d(v_i^{(k)}, v_j^{(k)})$ and is unique because of the

unidirectional movement. Ventura and Rieksts (2006) assumed constant AGV speeds between two locations. This assumption is extended in this study, so that a vehicle is accelerated to a certain maximum speed s_{max} at a rate of a^+ and is decelerated at a rate of a^- to stop at a specified location.

Depending on the distance between two locations, an AGV may reach the maximum speed (see (2)), or it may decelerate immediately after acceleration before reaching the maximum speed (see (1)). The travel time between two locations is given as follows. The derivation is left to the reader.

$$T(d(v_i^{(k)}, v_j^{(k)}), a^+, a^-, s_{max}) =$$

$$\begin{cases} \sqrt{\dfrac{2d(v_i^{(k)}, v_j^{(k)})}{a^+}\left(1 + \dfrac{a^+}{a^-}\right)}, \\ \quad\quad \text{if } d(v_i^{(k)}, v_j^{(k)}) \leq \dfrac{s_{max}^2}{2a^+} + \dfrac{s_{max}^2}{2a^-}, \quad (1) \\ \dfrac{d(v_i^{(k)}, v_j^{(k)})}{S} + \dfrac{s_{max}}{2}\left(\dfrac{a^+ + a^-}{a^+ \cdot a^-}\right), \\ \quad\quad \text{otherwise.} \quad\quad\quad\quad\quad\quad\quad (2) \end{cases}$$

Considering varying speeds of a vehicle during the travel, let $t(x_r^{(k)}, v_i^{(k)}, v_j^{(k)})$ be the time that it takes for an AGV at $x_r^{(k)}$ to pick up a unit load from $v_i^{(k)}$, to deliver it to $v_j^{(k)}$ and to return to $x_k^{(k)}$, including the times to load and unload. It includes unloaded travel between $x_r^{(k)}$ and $v_i^{(k)}$, loading time T_L, loaded travel between $v_i^{(k)}$ and $v_j^{(k)}$, unloading time T_{UL} and unloaded travel between $v_j^{(k)}$ and $x_r^{(k)}$. An AGV can have different acceleration and deceleration when loaded and unloaded.

$$t(x_r^{(k)}, v_i^{(k)}, v_j^{(k)}) = T(d(x_r^{(k)}, v_i^{(k)}), a_{UL}^+, a_{UL}^-, s_{max}^{UL})$$
$$+ T_L$$
$$+ T(d(v_i^{(k)}, v_j^{(k)}), a_L^+, a_L^-, s_{max}^L)$$
$$+ T_{UL}$$
$$+ T(d(v_j^{(k)}, x_r^{(k)}), a_{UL}^+, a_L^-, s_{max}^{UL}),$$

where a_{UL}^+ : acceleration rate when unloaded,
a_{UL}^- : deceleration rate when unloaded,
s_{max}^{UL} : maximum speed when unloaded,
a_L^+ : acceleration rate when loaded,
a_L^- : deceleration rate when loaded,
s_{max}^L : maximum speed when loaded.

Note that each request requires one rotation around loop k, if $d(x_r^{(k)}, v_i^{(k)}) < d(x_r^{(k)}, v_j^{(k)})$ because the vehicle at $x_r^{(k)}$ is routed in the order of $v_i^{(k)}$ for pickup and $v_j^{(k)}$ for dropoff and then returns to $x_r^{(k)}$. Otherwise, the request requires two rotations because the vehicle at $x_r^{(k)}$ is routed in the order of $v_i^{(k)}$, $v_j^{(k)}$ for pickup, $x_r^{(k)}$, $v_i^{(k)}$ for dropoff, $v_j^{(k)}$ and $x_r^{(k)}$. The total time, required by an

AGV located at $x_r^{(k)}$ to complete all requests to be picked up from $v_i^{(k)}$, is denoted as $t(x_r^{(k)}, v_i^{(k)}) = \displaystyle\sum_{j \in M^{(k)} - \{i\}} f_{ij}^{(k)}$

$t(x_r^{(k)}, v_i^{(k)}, v_j^{(k)})$.

Let A be the vehicle availability per shift (measured in time). If $t(v_i^{(k)}, v_i^{(k)}) > A$, i.e., station $v_i^{(k)}$ has too many requests for an AGV to complete, an AGV can be assigned to $v_i^{(k)}$ to serve part of requests from only $v_i^{(k)}$. Those served requests can be excluded from further consideration. This process can be repeated until $t(v_i^{(k)}, v_i^{(k)}) \leq A$ for all $i \in M^{(k)}$.

It is assumed that all requests from a station should be served by a single AGV and a series of consecutive stations should be assigned to an AGV. This assumption can reduce the complexity of vehicle control by dedicating requests from consecutive stations to a single vehicle. Let $S^{(k)} = \{S_1^{(k)}, S_2^{(k)}, \ldots, S_{h^{(k)}}^{(k)}\}$, where $S_j^{(k)} \subseteq M^{(k)}$ denotes the set of P/D stations whose requests are served by the AGV pre-positioned at location $x_j^{(k)} \in X^{(k)}$

The solution $(X^{(k)}, S^{(k)})$, where $X^{(k)} = \{x_1^{(k)}, x_2^{(k)}, \ldots, x_{h^{(k)}}^{(k)}\}$ and $S^{(k)} = \{S_1^{(k)}, S_2^{(k)}, \ldots, S_{h^{(k)}}^{(k)}\}$, is feasible if and only if the following conditions are satisfied.

1. $X^{(k)} \subset G^{(k)}$.
2. Each set $S_r^{(k)}$ contains consecutive pickup/dropoff stations which are served by $x_r^{(k)}$.
3. $\displaystyle\sum_{u \in S_r^{(k)}} t(x_r^{(k)}, u) \leq A$, for all $S_r^{(k)} \in S^{(k)}$.
4. $S^{(k)}$ forms a complete partition of $V^{(k)}$, i.e., $\displaystyle\bigcup_{r \in \{1,2,\ldots,h^{(k)}\}} S_r^{(k)} = V^{(k)}$ and $S_r^{(k)} \cap S_p^{(k)} = \varnothing$, for all $r \neq p$.

The *response time* $r(x_r^{(k)}, v_i^{(k)})$ for a request from $v_i^{(k)}$, processed by an AGV at $x_r^{(k)}$, is the unloaded travel time from $x_r^{(k)}$ to $v_i^{(k)}$, i.e., $r(x_r^{(k)}, v_i^{(k)}) = T(d(x_r^{(k)}, v_i^{(k)}), a_{UL}^+, a_L^-, s_{max}^{UL})$. The maximum response time for requests from all stations in $S_r^{(k)}$ served by an AGV at $x_r^{(k)}$ can be determined as follows.

$$r(x_r^{(k)}, S_r^{(k)}) = \begin{cases} \max_{u \in S_r^{(k)}} T(d(x_r^{(k)}, u), a_{UL}^+, a_{UL}^-, s_{max}^{UL}), \\ \quad\quad \text{if } \displaystyle\sum_{u \in S_r^{(k)}} t(x_r^{(k)}, u) \leq A, \\ \infty, \quad\quad \text{otherwise.} \end{cases}$$

The *maximum response time of a loop* k, $r(X^{(k)}, S^{(k)})$ with respect to a feasible solution $(X^{(k)}, S^{(k)})$ is given by $r(X^{(k)}, S^{(k)}) = \max_{r \in \{1,2,\ldots,h^{(k)}\}} r(x_r^{(k)}, S_r^{(k)})$. Finally, a feasible solution $(X^{(k)*}, S^{(k)*})$ is optimal if and only if $r(X^{(k)*}, S^{(k)*}) \leq r(X^{(k)}, S^{(k)})$, for all feasible $(X^{(k)}, S^{(k)})$.

Each dwell point $x_j^{(k)}$ in a single loop coincides with a station in $S_j^{(k)}$ and it is not necessarily the first station in $S_j^{(k)}$, $j \in \{1, 2, \ldots, h^{(k)}\}$ (Ventura and Rieksts

2006). Let $r(S_j^{(k)})$ be the optimal response time with respect to any request initiated by stations in $S_j^{(k)}$. Because the optimal dwell point for the vehicle assigned to $S_j^{(k)}$ coincides with a station in $S_j^{(k)}$, $r(S_j^{(k)}) = \min\limits_{u \in S_j^{(k)}} r(u, S_j^{(k)})$.

Ventura and Rieksts (2006) developed an exact polynomial-time algorithm to solve the idle AGV positioning problem in a unidirectional single loop system for the objective of minimizing the maximum response time. This section presents an extension of their algorithm to the TLMV system with existence of vehicle's acceleration, deceleration and maximum speed when loaded and unloaded, respectively.

In loop k, let $I(j)$ be the index of the first P/D station in $S_j^{(k)}$ in the direction of movement and $F_w^{(k)}(i, j)$ be the optimal maximum response time for the first j pickup stations, starting from the first pickup station w, when they are served by i AGVs in loop k. The proposed algorithm is based on a dynamic programming (DP) model. The DP model divides the idle vehicle positional problem into n stages, so that the initial station $v_i^{(k)}$ in $S_j^{(k)}$ and an AGV dwell point $x_j^{(k)}$ assigned to $S_j^{(k)}$ is the only decision to be made in stage $j \in \{1, 2, \dots, h^{(k)}\}$. Since $\bigcup\limits_{r=1}^{i} S_r^{(k)} = \{v_{I(1)}^{(k)}, v_{I(1)+1}^{(k)}, \dots, v_{I(1)+j-1}^{(k)}\}$,

$$F_{v_{I(1)}^{(k)}}^{(k)}(i, j) = \min \max\limits_{r=1,\dots,i} r(S_r^{(k)}) = \min \max\limits_{r=1,\dots,i} r(\{v_{I(r)}^{(k)}, v_{I(r)+1}^{(k)}, \dots, v_{I(r+1)-1}^{(k)}\}),$$ where $v_{I(r)}^{(k)} = v_{I(r-h^{(k)})}^{(k)}$ when $r > h^{(k)}$ and $v_i^{(k)} = v_{i-m^{(k)}}^{(k)}$ when $i > m^{(k)}$.

Suppose that i-1 AGVs (in stage i-1) are assigned to loop k in order to serve the first p stations from station $v_{I(1)}^{(k)}$: $\bigcup\limits_{r=1}^{i-1} S_r^{(k)} = \{v_{I(1)}^{(k)}, v_{I(1)+1}^{(k)}, \dots, v_{I(1)+p-1}^{(k)}\}$. Therefore, $S_i^{(k)} = \{v_{I(1)+p}^{(k)}, \dots, v_{I(1)+j-1}^{(k)}\}$. There are two maximum response times involved. The one is $F_{v_{I(1)}^{(k)}}^{(k)}(i-1, p)$ for $\bigcup\limits_{r=1}^{i-1} S_r^{(k)}$ and the other is $r(\{v_{I(1)+p}^{(k)}, \dots, v_{I(1)+j-1}^{(k)}\})$ for $S_i^{(k)}$. Because $F_{v_{I(1)}^{(k)}}^{(k)}(i-1, p)$ is previously calculated, $F_{v_{I(1)}^{(k)}}^{(k)}(i, j) = \min\limits_{p=i-1,\dots,j-1} \max \{F_{v_{I(1)}^{(k)}}^{(k)}(i-1, p), r(\{v_{I(1)+p}^{(k)}, \dots, v_{I(1)+j-1}^{(k)}\})\}$, where the optimal policy p minimizes $F_{v_{I(1)}^{(k)}}^{(k)}(i, j)$. The optimal policy p establishes the initial station of $S_i^{(k)}$, $v_{I(i)}^{(k)} = v_{I(1)+p}^{(k)}$, and its dwell point $x_i^{(k)}$, such that $r(S_i^{(k)}) = r(x_i^{(k)}, S_i^{(k)})$. Note that p = i-1, …, j-1 are the only policies that need to be considered as $r(\{v_{I(1)+j-1}^{(k)}, \dots, v_{I(1)+j-1}^{(k)}\}) = 0$ and $F_{v_{I(1)}^{(k)}}^{(k)}(i-1, p) = 0$ for $1 \le p \le i-1$.

Forward Recursive Relation :

$$F_{v_{I(1)}^{(k)}}^{(k)}(i, j) = \min\limits_{p=i-1,\dots j-1} \max\{F_{v_{I(1)}^{(k)}}^{(k)}(i-1, p),$$
$$r(\{v_{I(1)+p}^{(k)}, \dots, v_{I(1)+j-1}^{(k)}\})\}$$
for i = 2, …, $h^{(k)}$, j = 1, …, $m^{(k)}$-$h^{(k)}$+i.

Boundary Condition :

$$F_{v_{I(1)}^{(k)}}^{(k)}(1, j) = r(\{v_{I(1)}^{(k)}, \dots, v_{I(1)+j-1}^{(k)}\})$$
for j = 1, 2, …, $m^{(k)}$-$h^{(k)}$+1.

If $h^{(k)} = 1$, the definition of $r(S_j^{(k)})$ leads to an intuitive solution to the dwell point problem in a unidirectional single loop layout with a single vehicle. When $h^{(k)} \ge m^{(k)}$, all response times become 0's by locating at least one vehicle at each station in loop k. When $1 < h^{(k)} < m^{(k)}$, the problem of finding the smallest maximum response time can be formulated and optimally solved via the proposed algorithm in the following.

3. PROPOSED ALGORITHM

A recursive DP algorithm to select the optimal dwell points for AGVs considering time restrictions on vehicle availability in a given TLMV system is proposed, so that the maximum response time of the given AGV system is minimized. In order to do so, an extended algorithm in Section 2 can be repeated $c \cdot c^{n-c} = c^{n-c+1}$ times, by an exhaustive enumeration method in n-vehicle, c-loop TLMV system. However, it is computationally inefficient. Therefore, this paper focuses on developing an optimal polynomial-time algorithm that determines the number of AGVs and the corresponding dwell points in each loop. The following additional notation and assumptions are necessary to develop a proposed algorithm.

C : number of loops in the TLMV system
m_k : number of P/D stations including transfer ports in loop k,
t_k : number of AGVs assigned to loop k,
$R^{(k)}(t_k)$: optimal response time in loop k when t_k AGVs are assigned to it,
A : vehicle availability time per shift.

The problem under consideration is formulated as follows :

Minimize $\quad$ Res(n) = $\max\limits_{k=1,\dots,c} R^{(k)}(t_k)$,

subject to $\quad\quad \sum\limits_{k=1}^{c} t_k = n,$

$t_k \ge 1$, integer, k = 1, …, c

The above problem is two-fold; (1) to choose ap-

propriate numbers of AGVs for all given loops and (2) to determine the best dwell points of AGV(s) for minimizing the maximum response time. Note that the objective function $R^{(k)}(t_k)$ is not simply a function of the number of stations in a loop, which is one of arguments in Ventura and Lee (2001), because it is a function of the topology of TLMV layout, which are described by configurations of stations in each loop. Response time in a loop is determined by a combination of the number of stations, the number of transfer ports, the topological location of stations, the numbers of AGV(s), traffic volume, traffic pattern and their dwell points. The constraints ensure that each loop has at least one AGV to satisfy requests and all available AGVs should be deployed because Res(n) is monotonously decreasing with the number of AGVs (i.e., n). It is natural to see that more AGVs lead to the smaller response time in a loop. Therefore, the optimal maximum response time in loop k, $R^{(k)}(t_k)$ decreases with respect to t_k, k = 1, …, c. Because $\underset{k=1,\,…,\,c}{\text{Max}}\, R^{(k)}(t_k)$ determines the maximum response time for the system, if an additional AGV is provided, it must be assigned to the loop, of which the maximum response time is the largest. An additional AGV will contribute to the reduction of the maximum response time in the assigned loop, which automatically leads to the reduction of the maximum response time of a whole TLMV system. Otherwise, the additional AGV cannot make any contribution to the minimization of the objective function.

The minimum number of AGVs, required for each loop in a given TLMV system, is related to the volume and pattern of pickup/dropoff requests (traffic) issued by stations. Though numbers of requests between pairs of stations are given, it is difficult to find an analytic solution on the minimum number of AGVs for each loop. Therefore, the proposed algorithm uses the incremental approach. This approach is extended from the following theorem. The proof of Theorem 1 can be easily extended from Theorem 4 of Ventura and Rieksts (2006).

Theorem 1: In loop k, for any station $v_i^{(k)} \in M^{(k)}$, let

$$E_i^{(k)} = \{ v_i^{(k)}, v_{i+1}^{(k)}, …, v_{i+p}^{(k)} \}, \text{ where p } =$$

$$\underset{j=0,…,m^{(k)}-1}{\max} \{ j \mid r(\{v_i^{(k)}, v_{i+1}^{(k)}, …, v_{i+j}^{(k)}\}) < \infty \}, \text{ and}$$

$$F_i^{(k)} = \{ F_{i,1}^{(k)}, F_{i,2}^{(k)}, …, F_{i,q^{(k)}(i)}^{(k)} \}, \text{ where } F_{i,1}^{(k)} =$$

$E_i^{(k)}$, $F_{i,j}^{(k)} = E_{I(j)}^{(k)}$, j = 2, …, $q^{(k)}(i)$-1, $v_{I(j)-1}^{(k)}$ is

the last station in $F_{i,j-1}^{(k)}$, $F_{i,q^{(k)}(i)}^{(k)} = E_{I(q^{(k)}(i))}^{(k)}$ -$E_i^{(k)}$

with $v_{i-1}^{(k)} \in E_{I(q^{(k)}(i))}^{(k)}$, and $q^{(k)}(i)$ is the minimum number of AGVs required for a feasible solution starting at station i. Then $n^{(k)*} = \min \{q^{(k)}(i) \mid v_i^{(k)} \in M^{(k)}\}$ is the number of AGVs required to get a feasible solution in loop k.

The proposed algorithm starts with assigning to each loop the minimum number of AGVs, which is obtained by Theorem 1. The best dwell points can be obtained using the recursive relation described in Section 2. Then, assign one AGV to the loop which has the largest contribution to the objective function by calculate the contribution of an additional AGV when it is assigned to each loop, until all n AGVs are in use. In this research, the objective is to minimize the maximum response time so that loop p, where $R^{(p)}(t_p) = \underset{k=1,\,…,\,c}{\text{Max}}\, R^{(k)}(t_k)$, will be chosen in each iteration. The proposed algorithm is illustrated in Figure 4 and summarized in the following. The proposed algorithm consists of two stages: feasibility stage and optimization stage.

Feasibility stage :

Step 0: For n-vehicle, c-loop TLMV system, convert transfer ports to pickup stations. For each port, set the number of pickup requests to the number of incoming traffics from other loops into the corresponding loop via the corresponding port. Label all P/D stations including transfer ports in all loops as $v_i^{(k)}$ (I = 1, 2, …, $m^{(k)}$, and k = 1, 2, …, c) in the clockwise or counter-clockwise direction as appropriate.

Step 1: If c > n, then stop and this system is not feasible.

Step 2: Calculate the minimum number t_k of AGVs for each loop k, k = 1, …, c (using Theorem 1). Locate t_k vehicles to minimize the maximum response time in loop k (using the algorithm in Section 2) for k = 1, …, c.

Step 3: if $\sum_{k=1}^{c} t_k > n$, then stop and this system is not feasible.

Optimization stage:

Step 4: If $\sum_{k=1}^{c} t_k = n$, then stop and the current set of dwell points is optimal. The maximum response time can be easily calculated, using the current set of dwell points. Otherwise, find loop p that satisfies the condition $R^{(p)}(t_p) = \underset{k=1,…,c}{Max}\, R^{(k)}(t_k)$ and set $t_p = t_p +1$. Tie can be broken arbitrarily.

Step 5: Locate t_p vehicles to minimize the maximum response time in loop p (using the algorithm in Section 2). Update the new set of dwell points and the maximum response time $R^{(p)}(t_p)$ in loop p. Note that exactly one additional AGV only causes the $m^{(p)} - t_p$ calculations of the maximum response time, because $R^{(p)}(t_p-1)$ is previously evaluated in the prior iteration. Go to step 4.
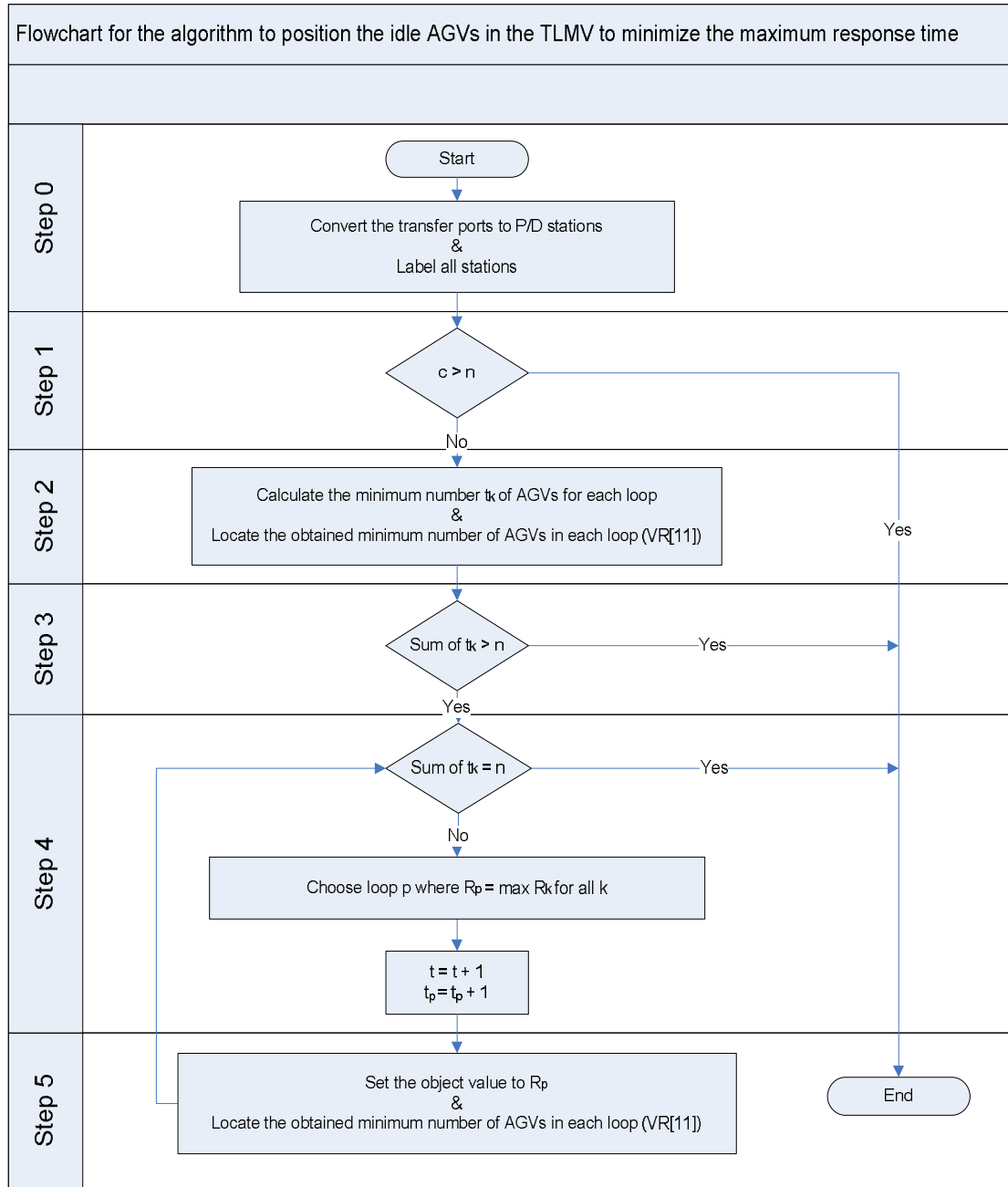
Flowchart for the algorithm to position the idle AGVs in the TLMV to minimize the maximum response time

Step 0

Start

Convert the transfer ports to P/D stations
&
Label all stations

Step 1

c > n

No

Step 2

Calculate the minimum number t_k of AGVs for each loop
&
Locate the obtained minimum number of AGVs in each loop (VR[11])

Yes

Step 3

Sum of t_k > n   Yes

Yes

Step 4

Sum of t_k = n   Yes

No

Choose loop p where R_p = max R_k for all k

t = t + 1
t_p = t_p + 1

Step 5

Set the object value to R_p
&
Locate the obtained minimum number of AGVs in each loop (VR[11])

End

Figure 4. Flowchart for the algorithm to position the idle AGVs in the TLMV to minimize the maximum response time

Theorem 2: Algorithm in Figure 4 provides a set of optimal dwell points for n-vehicle, c-loop TLMV system.

Proof. Let $y(t) = \underset{k=1,...,c}{Max}\, R^{(k)}(t_k)$, where $t = \sum_{k=1}^{c} t_k$. Note that because $R^{(k)}(t_k)$ is a monotonically decreasing function of t_k for I = 1, …, c, y(t) is monotonically decreasing with the respect to t, too. By induction, the proposed algorithm can be shown to generate an optimal set of

dwell points. Let γ be the smallest number of AGVs to make the TLMV system feasible. Suppose that the proposed algorithm provides a set of optimal dwell points when t = q, q ≥ γ and that $R^{(p)}(t_p) = \underset{k=1,...,c}{Max}\, R^{(k)}(t_k)$. Now, t is increased by 1, i.e., t = q +1. If any t_k, k ≠ p, is increased, the maximum response time of the TLMV system will become equal to $R^{(p)}(t_p)$, while an additional AGV in loops other than loop p will not decrease the maximum response time. Therefore, the best policy at

the current stage t = q+1, is to assign one more AGV only to loop p, i.e., $t_p = t_p + 1$, with the number of AGVs in other loops being kept same. $y(q) = \text{Max} \{ \underset{k \neq p}{Max\, R^{(k)}}(t_k)$,

$R^{(p)}(t_p\text{-}1)\}\} \geq \text{Max} \{ \underset{k \neq p}{Max\, R^{(k)}}(t_k)$, $R^{(p)}(t_p)\} = y(q+1)$,

where y(q+1) is the response time obtained by the proposed algorithm when t = q + 1 vehicles are available. This proves the Theorem.

The computational complexity of the algorithm in Figure 4 is polynomial. The complexity of the algorithm in Section 2 is $O(m^2(m\text{-}n)^2)$ when there are m P/D stations and n vehicles in a single loop. This procedure is used only once in Step 1 and at most n-c Step 5. Every loop should have at least one P/D stations and the minimum number of AGVs which is at least one. Therefore, the worst case of computational efforts in Step 5 is O $((m\text{-}c+1)^2(m\text{-}n\text{-}c+1)^2)$. This needs to be repeated at most n-c times. Therefore, the complexity of the proposed algorithm is $O((n\text{-}c)((m\text{-}c+1)^2(m\text{-}n\text{-}c+1)^2))$.

4. ILLUSTRATIVE EXAMPLE

To illustrate the proposed algorithm in this article, an 8-vehicle 3-loop unidirectional TLMV system shown in Figure 5 is exemplified. The vehicle availability per shift is 1000 units of time under consideration, i.e., A = 1000. Numbers on the arc represent the travel distance between adjacent stations. Because there are incoming and outgoing traffics between tandem loops through transfer ports, they are converted to additional P/D stations as Step 0 in the algorithm. Thus, P/D stations and transfer ports in the tandem loop layout in Figure 5 are relabeled in Figure 6. In all three loops, vehicles are assumed to move in the clockwise direction.

The traffic, i.e., the requests between P/D stations, is given in Table 1. This traffic is divided into intraloop and interloop travels. The frequency of interloop travels between each pair of loops is used as the number of requests at the transfer ports. After Step 0, the traffic frequency $(f_{ij}^{(k)})$ of all stations including transfer ports in each loop are tabulated in Table 2-4.
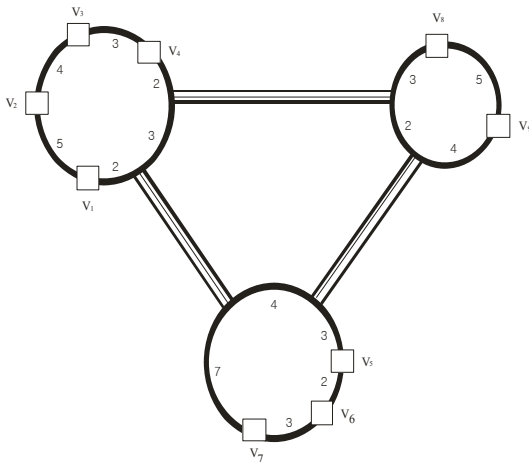

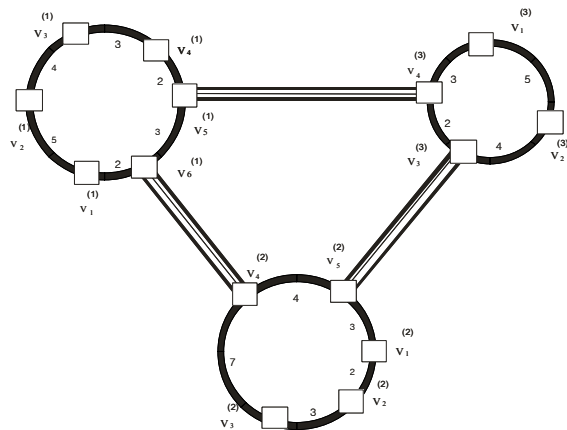**Figure 5.** 8-vehicle 3-loop unidirectional TLMV system with 9 stations

Figure 6. 8-vehicle 3-loop unidirectional TLMV system with 9 stations after Step 0 in the proposed algorithm

Table 1. Pickup requests/shift between pickup/dropoff stations in Figure 5

		dropoff								
		v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9
pickup	v_1	0	0	5	8	0	2	0	0	0
	v_2	7	0	0	9	0	0	0	0	3
	v_3	0	2	0	0	4	2	2	0	2
	v_4	0	3	4	0	0	0	0	4	2
	v_5	2	0	0	0	0	4	6	0	3
	v_6	0	0	2	0	4	0	10	0	1
	v_7	0	0	0	2	8	2	0	2	0
	v_8	0	2	1	0	0	2	0	0	8
	v_9	2	3	0	0	3	0	0	4	0

Table 2. Pickup requests/shift ($f_{ij}^{(1)}$) between pickup/dropoff stations of loop 1 in Figure 6

		dropoff					
		$v_1^{(1)}$	$v_2^{(1)}$	$v_3^{(1)}$	$v_4^{(1)}$	$v_5^{(1)}$	$v_6^{(1)}$
Pickup	$v_1^{(1)}$	0	0	5	8	0	2
	$v_2^{(1)}$	7	0	0	9	3	0
	$v_3^{(1)}$	0	2	0	0	2	8
	$v_4^{(1)}$	0	3	4	0	6	0
	$v_5^{(1)}$	2	5	1	0	0	0
	$v_6^{(1)}$	2	0	2	2	0	0

Table 3. Pickup requests/shift ($f_{ij}^{(2)}$) between pickup/dropoff stations of loop 2 in Figure 6

		dropoff				
		$v_1^{(2)}$	$v_2^{(2)}$	$v_3^{(2)}$	$v_4^{(2)}$	$v_5^{(2)}$
pickup	$v_1^{(2)}$	0	4	6	2	3
	$v_2^{(2)}$	4	0	10	2	1
	$v_3^{(2)}$	8	2	0	2	2
	$v_4^{(2)}$	4	4	2	0	0
	$v_5^{(2)}$	3	2	0	0	0

Table 4. Pickup requests/shift ($f_{ij}^{(3)}$) between pickup/dropoff stations of loop 3 in Figure 6

		Dropoff			
		$v_1^{(3)}$	$v_2^{(3)}$	$v_3^{(3)}$	$v_4^{(3)}$
Pickup	$v_1^{(3)}$	0	8	2	3
	$v_2^{(3)}$	4	0	3	5
	$v_3^{(3)}$	2	4	0	0
	$v_4^{(3)}$	4	7	0	0

Step 0: c = 3 and n = 8. Convert all transfer ports in all loops to P/D stations and calculate $f_{ij}^{(1)}$, $f_{ij}^{(2)}$, and $f_{ij}^{(3)}$.

Step 1: c < n. Thus go to Step 2.

Step 2: Check the minimum numbers of AGVs for loops 1, 2 and 3 $t_1 = 3$, $t_2 = 2$, and $t_3 = 1$ are obtained based on Theorem 1. The current feasible set of dwell points are $v_2^{(1)}$, $v_3^{(1)}$, $v_5^{(1)}$, $v_2^{(2)}$, $v_4^{(2)}$, and $v_2^{(3)}$, using the algorithm in Section 2.

Step 3: $\sum_{k=1}^{3} t_k = 6 \le n = 8$. Thus continue to Step 4.

Step 4: $\sum_{k=1}^{3} t_k = 6 \le n = 8$. The maximum response time in loops 1, 2 and 3 are $R^{(1)}(3) = 5.25$, $R^{(2)}(2) = 8.25$, and $R^{(3)}(1) = 10.25$, respectively. The corresponding optimal dwell points are $v_2^{(1)}$, $v_3^{(1)}$, $v_5^{(1)}$, $v_2^{(2)}$, $v_4^{(2)}$, and $v_2^{(3)}$. Therefore, the loop with the maximum response time in this iteration is loop 3. Set increase t_3 by 1.

Step 5: The algorithm in Section 2 has been applied to loop 3. The maximum response time in loop 3 is reduced to $R^{(3)}(2) = 5.25$. The new corresponding dwell points in loop 3 are $v_2^{(3)}$ and $v_4^{(3)}$.

Step 6: $\sum_{k=1}^{3} t_k = 7 \le n = 8$. The maximum response time in loops 1, 2 and 3 are $R^{(1)}(3) = 5.25$, $R^{(2)}(2) = 8.25$, and $R^{(3)}(2) = 5.25$, respectively. The corresponding optimal dwell points are $v_2^{(1)}$, $v_3^{(1)}$, $v_5^{(1)}$, $v_2^{(2)}$, $v_4^{(2)}$, $v_2^{(3)}$, and $v_4^{(3)}$. Therefore, the loop with the maximum response time in this iteration is loop 2. Increase t_2 by 1.

Step 7: The algorithm in Section 2 has been applied to loop 2. The maximum response time in loop 2 is reduced to $R^{(2)}(3) = 4.25$. The new corresponding dwell points in loop 2 are $v_2^{(2)}$, $v_4^{(2)}$, and $v_5^{(2)}$.

Step 8: $\sum_{k=1}^{3} t_k = 8 = n$. Stop. The optimal dwelling points are $v_2^{(1)}$, $v_3^{(1)}$, $v_5^{(1)}$, $v_2^{(2)}$, $v_4^{(2)}$, $v_5^{(2)}$, $v_2^{(3)}$, and $v_4^{(3)}$, where the numbers of AGVs in loops 1, 2 and 3 are $t_1 = 3$, $t_2 = 3$, and $t_3 = 2$. The optimal maximum response time is $\underset{k=1,...,c}{\text{Max}} R^{(k)}(t_k) = \{5.25, 4.25, 5.25\} = 5.25$.

The dwell points of AGVs are shown as filled boxes in Figure 7. Note that transfer ports serve as dwell points, depending on the volume of interloop travels.

The summary of the solution is provided in Table 5. The locations of dwell points, vehicle utilization and maximum response time in each loop when given numbers of vehicles are presented.

If an additional AGV is available, the tie of the maximum response time occurs in the next iteration. Either of loops 1 or 2 can be chosen arbitrary. The effectiveness of the proposed algorithm has been shown in

Table 5. Summary of the proposed algorithm for the example in Figures 5 and 6.

Loop	# of AGVs	Dwell Points	Dedicated Stations	Vehicle Usage	Total Usage	Maximum Response Time /AGV	Maximum Response Time
1	1	Infeasible				Infinite	Infinite
	2	Infeasible				Infinite	Infinite
	3	$v_5^{(1)}$	$v_5^{(1)}, v_6^{(1)}, v_1^{(1)}$	784.96	1871.09	5.25	5.25
		$v_2^{(1)}$	$v_2^{(1)}$	469.92		0	
		$v_3^{(1)}$	$v_3^{(1)}, v_4^{(1)}$	616.21		4.25	
	4	$v_2^{(1)}$	$v_2^{(1)}$	469.92	1823.63	0	3.25
		$v_3^{(1)}$	$v_3^{(1)}$	291.51		0	
		$v_4^{(1)}$	$v_4^{(1)}, v_5^{(1)}$	516.94		3.25	
		$P_6^{(1)}$	$p_6^{(1)}, v_1^{(1)}$	545.26		3.25	
2	1	Infeasible				Infinite	Infinite
	2	$v_4^{(2)}$	$v_4^{(2)}, v_5^{(2)}, v_1^{(2)}$	822.00	1641.86	8.25	8.25
		$v_2^{(2)}$	$v_2^{(2)}, v_3^{(2)}$	819.86		4.25	
	3	$v_2^{(2)}$	$v_2^{(2)}, v_3^{(2)}$	819.86	1577.36	4.25	4.25
		$v_4^{(2)}$	$v_4^{(2)}$	254.21		0	
		$v_5^{(2)}$	$v_5^{(2)}, v_1^{(2)}$	503.29		4.25	
	4	$v_1^{(2)}$	$v_1^{(2)}, v_2^{(2)}$	812.76	1563.61	3.25	3.25
		$v_3^{(2)}$	$v_3^{(2)}$	380.24		0	
		$v_4^{(2)}$	$v_4^{(2)}$	254.21		0	
		$v_5^{(2)}$	$v_5^{(2)}$	116.39		0	
3	1	$v_2^{(3)}$	$v_2^{(3)}, v_3^{(3)}, v_4^{(3)}, v_1^{(3)}$	906.05	906.05	10.25	10.25
	2	$v_2^{(3)}$	$v_2^{(3)}, v_3^{(3)}$	357.06	837.30	5.25	5.25
		$v_4^{(3)}$	$v_4^{(3)}, v_1^{(3)}$	480.24		4.25	
	3	$v_2^{(3)}$	$v_2^{(3)}$	233.23	836.05	0	3.25
		$v_3^{(3)}$	$v_3^{(3)}, v_4^{(3)}$	347.37		3.25	
		$v_1^{(3)}$	$v_1^{(3)}$	255.45		0	

the illustrative example above. The complexity of algorithm cannot be easily obtained in terms of m and n, because the complexity directly relies on the topology of tandem loop layout. However, it is conjecture that the proposed algorithm will be completed in polynomial time.
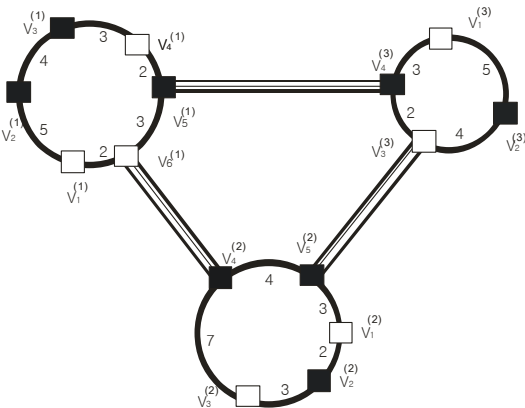

Figure 7. Optimal dwell points of 8 AGVs

5. CONCLUSIONS

In this article, a polynomial-time DP algorithm has been proposed to determine an optimal set of dwell points that minimizes the maximum vehicle response time in AGV systems with a TLMV layout. The proposed algorithm generalizes the approach developed by Ventura and Rieksts (2006) by considering the acceleration and deceleration of vehicles in the calculation of the response time and by taking into account time constraints on the availability of vehicles. The proposed algorithm does not consider traffic congestion at the P/D stations and transfer ports due to random arrivals of pickup requests. In future research, we plan to consider stochastic delays in travel times due to congestion at the P/D stations and transfer ports.

back on this work.

REFERENCES

Bozer, Y. A. and Srinivasan, M. M. (1992), Tandem AGV Systems: A partitioning algorithm and performance comparison with conventional AGV systems, *European Journal of Operational Research*, **63**(2), 173-192.

Egbelu, P. J. (1993), Positioning of automated guided vehicles in a loop layout to improve response time, *European Journal of Operational Research*, **71**(1), 32-44.

Egbelu, P. J. and Tanchoco, J. M. A. (1986), Potentials for bidirectional guide-path for automated guided vehicle based systems, *International Journal of Production Research*, **24**, 1075-1097.

Heragu, S. (1997), *Facilities Design*, PWS Publishing, Boston, MA.

Lee, C. and Ventura, J. A. (2001), Optimal dwell point location of automated guided vehicles to minimize mean response time in a loop layout, *International Journal of Production Research*, **39**(17), 4013-4031.

Rajotia, S., Shanker, K., and Batra, J. L. (1998), De-termination of optimal AGV fleet size for FMS, International Journal of Production Research, **36**(5), 1177-1198.

Ross, E. A., Mahmoodi, F., and Mosier, C. T. (1996), Tandem configuration in automated guided vehicle systems: A comparative study, *Decision Sciences*, **27**(1), 81-102.

Sinriech, D. (1995), Network design models for discrete material flow systems: A literature review, *International Journal of Advanced Manufacturing Technology*, **10**, 277-291.

Sinriech, D. and Tanchoco, J. M. A. (1992), Impact of empty vehicle flow on performance of single-loop AGV systems, *International Journal of Production Research*, **30**(10), 22237-2252.

Ventura, J. A. and Lee, C. (2001), A study of the tandem loop with multiple vehicles configuration for automated guided vehicle systems, *Journal of Manufacturing Systems*, **20**(3), 153-165.

Ventura, J. A. and Rieksts, B. Q. (2006), Optimal location of dwell points in a single loop AGV system with vehicle capacity constraints, Working Paper, Department of Industrial and Manufacturing Engineering, The Pennsylvania State University, University Park, PA.