Using Simulation to Design an Automated Underground System for Transporting Freight Around Schiphol Airport

M. C. van der Heijden • A. van Harten • M. J. R. Ebben
University of Twente, Faculty of Technology and Management, PO Box 217, 7500 AE Enschede, The Netherlands
Y. A. Saanen • E. C. Valentin • A. Verbraeck
Delft University of Technology, Faculty of Technology, Policy and Management, PO Box 5, 2600 AA Delft, The Netherlands
m.c.vanderheijden@utwente.nl • a.vanharten@utwente.nl • m.j.r.ebben@utwente.nl
• yvos@tudelft.nl • e.valentin@tudelft.nl • alexandv@tudelft.nl

This paper was refereed.

To avoid road congestion, we are developing a highly automated underground transport system using automatic guided vehicles (AGVs) around Schiphol Airport. It is unique in its scale, incorporating 16 to 25 km tubes connecting five to 20 terminals, and it includes 200 to 400 AGVs to transport an estimated 3.5 million tons of cargo in 2020 with different ordering priorities. According to the current plans, the system will run from 2006 on. Since 1997, we have used object-oriented simulations to plan the dimensions of the system (number of AGVs, terminal sizes) and to design the layout (network, terminals). We showed that an investment reduction of plus or minus 20 percent is feasible using periodically switched one-way tube sections. We developed a variety of logistics optimization algorithms and heuristics, including allocating AGVs between terminals, scheduling terminals, and controlling traffic. We used simulation control structures to test prototype AGVs on a test site. Performing distributed simulations with a mixture of simulated and real objects, we could reduce the risks of the new technology.

(Simulation: applications. Transportation: models, networks.)

The increase in cargo shipments worldwide is causing congestion, especially around such traffic hubs as airports and harbors. Good facilities for moving cargo lead to a concentration of business activities and increasing volumes of inbound and outbound cargo. To accommodate increasing cargo, the main hubs must be accessible. New infrastructure should accommodate increasing flows of cargo. Efficient and speedy shipments are especially important for time-critical goods, such as air cargo (newspapers and perishables). Delays caused by congestion may decrease the value of such time-critical goods. The scarcity of space in densely populated areas hampers such classical solutions as simply enlarging the capacity of roads.

Near Amsterdam, the increasing congestion forced the government, in cooperation with such business partners as Schiphol Airport, the world’s largest flower auction at Aalsmeer, and logistics service providers, to consider underground construction. Using AGVs (automatic guided vehicles) in underground tube systems seemed an attractive possibility. Combined with automated transport, loading and unloading, and logistics planning and control, AGVs in underground tubes can form an efficient, fast, flexible, and reliable cargo transport system. Therefore, the stakeholders, being
the government and the major business partners, provided research grants to an independent institute (Connekt) to carry out an in-depth investigation of such a system’s technical feasibility, its merits for reliable logistics service, its environmental benefits, and its costs.

The system should be fully automated and compete with road transport. It should stimulate multimodal transport to reduce pollution. It will connect the modes air, rail, and road and is called the OLS (the Dutch abbreviation for underground logistics system). The investment required is between $250 million and $500 million, depending on the system design and the network layout. We show three options out of many that have been developed by a multidisciplinary expert team, based on technical, logistic, constructional, and urbanistic issues (Figure 1A–1C). Researchers from several disciplines cooperated to assess the OLS, to optimize the system design, and to explore its technological and economic viability. These disciplines included underground construction, AGV and mechatronics, automated docking and warehousing, information and communication technology, logistics, simulation, and economics.

In 1997, Connekt formed four multidisciplinary project teams, each focusing on a specific set of subproblems. A steering committee, consisting of the main stakeholders, coordinated the activities and decided upon issues that cross project-team boundaries. One of the project teams focused on transport technology and the design of a control system. Therefore, FROG Navigation Systems, a well-known Dutch producer of AGV control systems, was a member of the project team. Our simulation group belonged to this team as well. Now, in 2001, the project teams are working on some last design issues. Further, a project team is working on the foundation of a public-private partnership of government and major business partners for the funding and the realization of the OLS. This partnership will have the infrastructure built, and as its owner, it will be responsible for its future maintenance.

It will rent the system to an exploitation company that operates the AGVs. We expect that they will make a final decision on realization in 2002. The project team projects that the first phase of the OLS will be operational in 2006.

Innovative Aspects and System Characteristics

Compared with existing AGV systems for internal transport, the OLS is to be much larger. The planned tube length in the network is 16 to 25 km, connecting five to 20 terminals, and it should transport 3.5 million tons of cargo in 2020. The OLS Project Organization (2000) estimates that the revenues from logistics activities will be US $65 million in 2020. The AGVs must travel fast to ensure short throughput times (six meters per second (m/s) in tubes and two m/s at terminals). An AGV should handle 10-foot aircraft pallets (Figure 2). Thus these AGVs must be considerably faster and larger than the AGVs currently used in warehousing and production systems (with a length of 1.5 to 2.0 meters and a speed of 0.5–1.0 m/s). Also, the tube diameter should be wide (five meters). Such tubes cost about $20 million per km, not including facilities for energy, maintenance, and safety. Depending on the network layout, growth scenario, and throughput-time requirements, we expect it to require 200 to 400 AGVs.

We achieved flexibility by employing independently free-ranging AGVs, that is, AGVs that can travel any route guided by a magnetic grid. Especially at the underground terminals, these AGVs make efficient use of scarce space. Along with the magnetic grid, the AGVs use sensors to control their distance from each other. They conform to the brick-wall principle, that is, they should be able to stop in time to avoid hitting their predecessor, even if that AGV halts instantaneously. Allowing for an emergency deceleration rate of two meters per square second (normally one m/s²) and taking into account some safety margin, the AGVs, when traveling at full speed, should leave a minimum distance of 21.2 meters from the front of one to the front of the next. The various technical requirements for the OLS project mean that a new design of AGV will be needed. Innovations will also be required in the
Figure 1A: We considered several trajectories for the OLS that the project team on underground construction developed. Of these, we show three trajectories, the trajectory preferred in 1997 (A), in 1998/1999 (B), and in 2000 and later (C). Amsterdam Airport Schiphol (AAS) is at the upper right, and the Flower Auction Aalsmeer (VBA) is at the bottom right. During the project, the project team on underground construction proposed to move the rail terminal (RT) from Hoofddorp to Schiphol. In 1998, they introduced tube sections to be shared by traffic in two directions to reduce infrastructure investment. A public-private partnership must decide which layout will actually be implemented. At the moment of writing, they prefer option (C). As an indication of the scale of the maps, the length of the one-way tube section between AAS and VBA is 2,700 meters.

designs for terminals and automated loading docks. AGVs will have to be loaded and unloaded at docks very quickly to minimize terminal size and investment. A target for automated docking is to load or unload an AGV within one minute. At that speed, a terminal might need one to 16 loading docks, depending on the inbound and outbound flows of cargo.

To operate the new technology in a reliable and efficient way, the complex would need an innovative planning and control system for logistics. It would need to manage empty cars (prepositioning AGVs to meet predicted demand), schedule terminals, and control traffic. The new technology and the accompanying control systems would have to be tested. We tested the
Figure 1B.

Figure 1B.

A Central Role for Simulation

To plan the system, a multidisciplinary team took into account its technical, logistic, economical, and legal aspects and made trade-offs. We used simulation models to provide decision makers with a common structure for making trade-offs and to aid communication between project teams. Our group, the simulation group, was part of the project team on transport technology and control systems, which dealt with designing the infrastructure (terminals, docks, track system), assessing logistics performance, and testing the control technology. The project team designed terminals and docks using an iterative process with the aid of our simulation models. Given the functional requirements defined by the overall OLS steering committee, the project team made decisions on technical design issues subject to the approval of the steering committee. We supported issues involving several project teams with simulation as well. For example, we used simulation
to assess the impact of system layouts designed by the underground construction team on logistics performance (throughput times and the fraction of orders delivered on time). Also, we measured energy use in our simulations for use as an input for an environmental study, comparing the OLS and traditional road transport.

Research Questions

We worked on many research problems and practical issues using simulation, including the following research questions:

1. How many AGVs and docks will be needed for adequate logistics performance for different network layouts? In particular, can capital investment be reduced by inserting periodically switched one-way tube sections without damaging logistics performance?

2. Which terminal and dock design will lead to an optimal balance between terminal capacity, order throughput times, and space requirements?

3. Which AGV failure rate should we use as a target in designing vehicles?

4. What way of providing energy (batteries or electric wires in tubes) is best?

5. Which structure for planning and controlling logistics is appropriate for the OLS?

6. How must we refine the control system to make...
An Automated Underground System

Interfaces
Vol. 32, No. 4, July–August 2002

Figure 2: The project team considered three prototypes of AGVs. This figure shows one of them. This AGV travels on wheels with rubber tires. An electric motor provides the energy. All AGVs are able to transport aircraft pallets (length $\times$ width $\times$ height = 3.18 $\times$ 2.44 $\times$ 3.00 meters), they have a carrying capacity of 3,500 kilograms, and are six meters long. They can also handle six industrial pallets or four flower-auction-market cars.

it work with real technical systems instead of virtual ones?

Because of project deadlines, we had to answer these research questions in an iterative way. For example, question 1 about resource requirements had high priority in the project, but the answer depended on the capabilities of the system for planning and controlling logistics (question 6). Therefore we based our answer to question 1 on a simple temporary planning and control structure, thereby obtaining a reasonable estimate. During the project, we continuously refined the system for planning and controlling logistics.

Processes in the OLS

The physical processes in the OLS include transporting cargo between its origin and its destination (both terminals) and AGVs traveling between network nodes, enabling transportation of the cargo (Figure 3). The physical processes are simple, but the planning and control of the logistics are complex. Many decisions must be made to obtain high system performance with minimal resources (Table 1). A key challenge was to develop an appropriate planning and control structure with corresponding information and communication architecture, a joint activity of control-system experts and simulation experts.

The Object-Oriented Approach

Because this research-and-development project required close interaction among various disciplines and a user group, the model input and the precise content of the research questions varied over time. Participants in the project frequently proposed new layouts, demand patterns, and alternatives for handling AGVs and the loading docks. Modeling flexibility was a necessity. We need to be able to construct a variety of models quickly from basic building blocks, such as variants of terminals, docks, buffers, and tracks. We gained flexibility by using an object-oriented design approach (Adiga 1993, Randell, Holst, and Bolmsjö 1999, Booch 1994, Zeigler 1990). (Roberts and Dessouky (1998) review object-oriented simulation.)

Because logistics control is important, we needed the same flexibility in modeling control systems from the building blocks for the various control decisions, such as routing vehicles, controlling traffic, and scheduling orders. We wanted to compare alternative algorithms for key planning and control decisions. Therefore, we represented each algorithm by a control object. We are able to interchange and test algorithms in our simulation models by interchanging control objects. A prerequisite is that the various control objects (algorithms) for the same decision can exchange similar information with the physical objects. We adopted van der Zee’s (1997) framework for logistics and transport agents and their control and Evers and Lindeijer’s (2000) framework for traffic control.

Based on these frameworks, we constructed an object library, whose components can be classified as physical objects (for example, AGV, terminal, dock, parking, and track), control objects (for example, AGV dispatcher, order scheduler, and traffic controller), and information objects (for example, distance table and job list). We structured these objects hierarchically and provided a standardized interface. The information objects provide the control objects with the appropriate information for using the physical objects (resources)
Figure 3: The primary physical processes in an automated transport network consist of transporting cargo from origin to destination (both are terminals) and an AGV loop for transporting cargo and for repositioning empty AGVs.

efficiently. The reusability of objects in our library turned out to be a critical factor in the OLS project, because, during the orientation and design phases of the project, participants changed almost all the objects over time. Several times, we constructed new model variants quickly using objects from our library that we extended with new objects. For example, we added a new terminal layout to be tested, or we added control objects to extend the model's functionality and give us the ability to make an additional analysis (failure management, energy management).

A Local Concept for a Flexible and Robust Control System

For the sake of robustness and extendibility, the project team decided to focus on a local control concept within a hierarchical framework for planning and controlling logistics. For example, each terminal has a terminal manager that is responsible for releasing and scheduling orders, controlling the workload, planning cargo consolidation, managing parking, and managing the AGVs at the terminal. The terminal manager is software that bases its planning decisions and control activities only on local terminal information, which is embedded in the terminal information object. When it needs other information to optimize local decisions, it communicates with other information objects at a higher level, for example the global empty-car manager can make information about AGVs traveling to the terminal available to the terminal manager. For some scheduling problems, we analyzed the impact of using such coordinated local control versus using global scheduling. We found that local control does not necessarily impair the logistics performance of the system, provided that the local control structures and
information exchange between objects are appropriate (Appendix). Also local control reduces the communication needed for making decisions. Local control works well in a multiuser system, such as the OLS.

Two Statistically Linked Models

The questions to be covered in the OLS project range from questions at the resource level about AGV design and traffic control to questions at the network level about layout, system dimensions, and allocation of AGVs to terminals. In principle, it would be possible to construct one huge model, incorporating all details, but it is not practical to do so because we need to maintain the models and run them fairly quickly. Also, because system performance may be influenced by AGV characteristics, such as the safety distance between AGVs, we decided to construct two statistically linked simulation models: a network model and a traffic model. These two models meet at the terminal level (Figure 4).

In the network model, AGVs travel in a simplified manner, for example, they accelerate and decelerate instantaneously and require no extensive traffic control. We derived the effective driving speed at terminals, corrected to reflect the details of AGV behavior and interactions at crossings and junctions, from the traffic model as a statistical function of the number of AGVs. Also, we controlled the distance between AGVs at a few critical locations (such as terminal entrances and one-way tubes). On the other hand, the terminal in the traffic model uses the arrival patterns of AGVs

Table 1: By creating separate control objects with clear interfaces, we separated the logistics-planning-and-control structure from the physical system. In this way, we constructed a flexible library to test several alternatives for the main planning and control issues. We distinguished three levels in the planning hierarchy: network, node (terminal, central parking, one-way tube), and resource (AGV, dock, track).

<table>
<thead>
<tr>
<th>Level</th>
<th>Control Object</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>Demand forecasting</td>
<td>Predicting future orders in various priority classes</td>
</tr>
<tr>
<td></td>
<td>Global empty-car management</td>
<td>Distributing AGVs to terminals, based on predicted demand</td>
</tr>
<tr>
<td></td>
<td>System-failure manager</td>
<td>Initiating failure remediation, for example assigning a rescue vehicle to tow away a broken vehicle</td>
</tr>
<tr>
<td>Node: terminal, central parking, one-way tube</td>
<td>AGV routing</td>
<td>Assigning a route to an AGV, given its location and destination</td>
</tr>
<tr>
<td></td>
<td>Order release</td>
<td>Giving permission for an order to be scheduled</td>
</tr>
<tr>
<td></td>
<td>Order scheduling</td>
<td>Assigning an order to an AGV and a dock for loading or unloading</td>
</tr>
<tr>
<td></td>
<td>Terminal-workload control</td>
<td>Controlling the number of AGVs at the terminal to avoid congestion</td>
</tr>
<tr>
<td></td>
<td>One-way-tube control</td>
<td>Setting the traffic direction for tubes shared by traffic going in both directions</td>
</tr>
<tr>
<td></td>
<td>Cargo-consolidation planning</td>
<td>Consolidating orders for AGV loads, based on destination, weight, and release and due times</td>
</tr>
<tr>
<td></td>
<td>Parking manager</td>
<td>Assigning vehicles to specific parking places</td>
</tr>
<tr>
<td></td>
<td>Local empty-car management</td>
<td>Assigning empty AGVs to locations within a terminal (parkings or docks) or dispatching vehicles on request to other terminals</td>
</tr>
<tr>
<td>Resources</td>
<td>AGV energy management</td>
<td>Determining when and where AGVs should change batteries (optional, depending on energy system)</td>
</tr>
<tr>
<td></td>
<td>Resource-failure manager</td>
<td>Prohibiting the use of failed AGVs and docks and redirecting assigned cargo</td>
</tr>
<tr>
<td></td>
<td>Traffic control</td>
<td>Controlling intersections and prioritizing AGVs waiting for intersections</td>
</tr>
</tbody>
</table>

Figure 4: We developed two simulation models: a network model, focusing on logistics network control, layout choice, and performance measurement; and a traffic model, focusing on traffic control, terminal and dock design, and AGV behavior. The models are linked by information exchange: AGV and order information goes from the network model to the traffic model, and the effective driving speed of AGVs and the loading and unloading times go from the traffic model to the network model.
and the loads derived from the network model. In this way, we guaranteed consistency between the two models.

Calculating Required Resources for Various Network Layouts (Research Question 1)

To analyze and estimate the needed investment, we had to estimate the needed resources, such as the number of AGVs and the number of docks per terminal. We investigated the relationship between capacity and performance using simulation. As input to the simulations, we used estimates of transport demand for the OLS for the next 20 years, derived from Dutch national transport statistics (CTT 1997). As the reference point for determining the dimensions of the system, we chose the year 2020. We performed growth path analysis as well. We included variations in volume over the day and between days, because peaks affect system performance. We specified transport jobs by their due times and priority classes, defined by throughput-time requirements (between 30 minutes and two hours). The main indicator for logistics service was the percentage of cargo delivered on time. We generated many other performance indicators as well, such as throughput-time distribution, resource utilization, buffer occupation, AGV queue sizes, energy consumption, and failure statistics. We estimated the major resource requirements for various layouts (Table 2). For the layout shown in Figure 1A, we found that the system could meet the required throughput times in 2020 with 220 AGVs and five docks at terminal VBA, 12 at RT, and two at AAS. With these resources, 99 percent of orders would be delivered on time.

Using an alternative layout (Figure 1B), the project could save about $150 million on infrastructure (initially $500 million). This layout includes three periodically switched one-way tubes on the long-distance links, with lengths of 2.7 km (VBA to AAS), 2.3 km (AAS to RT), and 0.75 km (between the terminals at AAS). The travel time for a single AGV through these tubes is up to 7.5 minutes. This implies that AGVs at the other end of the track would incur long waits. The logistics consequences of using this layout are (1) that transport times would increase and would vary, (2) that convoys of up to 100 AGVs would be created during peak hours, causing heavily fluctuating work loads at the terminals, and (3) that the one-way tubes would be potential bottlenecks. Our simulations revealed that this layout would be feasible from a logistics perspective if we used somewhat larger terminals and 140 additional AGVs, but the average throughput time would increase by 15 minutes, reducing on-time service to 97.5 percent. Nevertheless, because docks and AGVs are cheaper than constructing underground tubes (an AGV costs about $75,000), this layout would guarantee profitability. Moreover, to use this layout, the system would need enough waiting space for AGVs at the entrances to terminals and one-way tubes (up to 700 m) because of congestion.

Later we investigated a third layout (Figure 1C). This layout includes 18 miniterminals at AAS to facilitate internal transport between the local warehouses of the cargo shippers and the providers of logistics service, and an alternative location near AAS for the rail terminal. Relocating the rail terminal reduces the distance between it and Schiphol Airport by about four kilometers and makes one one-way tube unnecessary. A disadvantage is that the new rail terminal would be more expensive, because it would have to be constructed underground. Therefore, the space for equipment and buffering would be limited. To analyze the logistics consequences of having only small buffers, we modeled an approximation of the train arrival-and-departure processes. We found that, compared to the layout in Figure 1B, this layout would reduce throughput times to and from the rail terminal by about 15 minutes, decrease the number of AGVs from 360 to 250, and increase the on-time-service percentage to 99 percent.
percent. Investment calculations showed that this option was preferable, especially because this system would be able to handle internal flows at Schiphol Airport better than the other two layouts.

Resolving the Conflicting Requirements for Terminal Design (Research Question 2)

Terminals and docks for an automated transport system as large as the OLS did not exist. In cooperation with experts on automated transport and transshipment technology, and using an iterative process, we designed two example layouts (Figure 5). We had to make a trade-off between space consumed and logistics performance. For instance, with spacious terminals, AGVs can maneuver without interfering with each other, which shortens throughput times. Available space is limited, however, because of existing structures and high prices for land in the region. A similar trade-off between space and throughput times must be made at the rail terminal, where high-speed trains should be loaded and unloaded quickly. Buffering loads at the platforms takes space. To be able to make such a trade-off, we needed a detailed analysis of AGV behavior. We used the traffic simulation model to support decisions about terminal and dock design. To examine the impact of design choices, we focused each of our analyses on one or two important criteria, such as small size or fast handling. We found that some terminal concepts that seemed attractive from a spatial perspective (such as concept B) perform poorly in terms of transshipment capacity, because AGVs cannot reach the docks quickly. Concept A with eight docks can handle 424 operations per hour, while concept B with 10 docks can handle only 130 operations per hour (Table 3). Based on our simulation experiments, we made some recommendations:

1. AGV buffering and parking locations should not interfere with passing traffic.
2. Performance improves when the number of AGVs allowed in the terminal at once is limited.
3. Instead of minimizing the lengths of tracks, as designers of rail systems do, we should install many
tracks in the terminal so that we can spread the vehicles over the available space, thus increasing AGV speeds.

As a result of our analysis, the project team chose terminal concept A as the basic design for the large terminals at VBA and RT.

**Setting Design Targets for Failure Behavior (Research Question 3)**

To be competitive with traditional road transport, the OLS should be reliable. Because AGVs and docks (for example, the conveyors) are subject to failure, we had to determine what failure rates would be acceptable. The project team decided to focus on the use of recovery vehicles that could drag away a failed AGV. The impact of AGV failures is unpredictable, because it depends on the location of the AGV when it fails. A failed AGV in the terminal can be removed quickly if a recovery vehicle is close by. Then traffic can usually pass a failed AGV, because AGVs are free ranging at terminals. On the other hand, a failure in a one-way tube blocks traffic from both directions, seriously delaying (up to 30 minutes) many AGVs.

<table>
<thead>
<tr>
<th>Terminal Concept</th>
<th>Dock Performance: AGVs per Hour</th>
<th>Dock Performance: Operations per Hour</th>
<th>Terminal Performance: Operations per Hour</th>
<th>Time in Terminal per AGV (min.)</th>
<th>Surface Occupied (m²)</th>
<th>Distance Travelled in Terminal (km)</th>
<th>Number of Accelerations &gt;0.5 m/s²</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (8 docks)</td>
<td>35</td>
<td>53</td>
<td>424</td>
<td>5.5</td>
<td>4,480</td>
<td>0.28</td>
<td>10</td>
</tr>
<tr>
<td>2 (10 docks)</td>
<td>8</td>
<td>13</td>
<td>130</td>
<td>9.6</td>
<td>5,236</td>
<td>0.63</td>
<td>11</td>
</tr>
</tbody>
</table>

Table 3: We compared several terminal layouts (Figure 5) and their performance. We show the performance of two examples in this table.
We found that dock failures would have little impact on terminals with multiple docks, whereas AGV failures have greater impact. The simulations showed that an AGV failure rate of once per 500 operating hours would be acceptable. Experts judged that this failure rate would be technically feasible, and they adopted it as a design target. Still, AGV failures would be a daily issue given the number of AGVs in the system (200 to 400), and failure management should be part of the standard operations of the overall OLS control system.

Supporting the Design of a System for Providing Energy for the AGVs (Research Question 4)

The AGVs (1) travel using an electric motor (linear induction), and (2) range freely at the terminals. To move freely at the terminals, the AGVs need to carry batteries. They could rely on charge rails in the tubes. The design options include (1) various types of batteries (lead-acid or nickel-cadmium), and (2) various ways of providing electricity: either swapping batteries or using charge rails in the tubes to recharge the batteries. Given that the charging time would be equal to the traveling time, recharging batteries on board would not be an option. For swapping batteries, we would have to determine the number, size, and location of swapping facilities and when and where AGVs should swap their batteries. We performed this analysis using the network model, because the various options would affect global performance. The energy required would depend on AGV behavior, in that acceleration and deceleration require more energy than traveling at a constant speed. The network model uses statistics about energy consumption from the traffic model.

Our experiments showed that the project would obtain the best logistics performance by using charge rails in the tube rather than battery swapping. We found that swapping times of five minutes for charging batteries would require very large battery stations if drastic effects on system performance were to be avoided. These stations would require large investments. If we could reduce swap time to one minute, we could swap batteries at the docks while the AGVs were being loaded or unloaded. We also found that lead-acid batteries were preferable to nickel-cadmium batteries from a financial point of view, even though they take longer to recharge and degrade faster.

Efficient Planning and Control (Research Question 5)

We developed several variants of control objects (Table 1) to analyze their impact on logistics performance. We discuss three control objects, one on each level in the modeling hierarchy: empty-car management, one-way-tube control, and traffic control (Appendix).

Managing empty cars (prepositioning AGVs to meet predicted demand) is important, as are managing truck and rail-wagon fleets (Powell et al. 1988, Powell and Carvalho 1998). Depending on known and expected orders and their priorities, the manager would relocate empty AGVs from terminals with excess AGVs to terminals with shortages, taking into account the scheduled arrival times of loaded AGVs. We considered several methods, including (1) a simple first-come-first-serve (FCFS) method, (2) a coordinated look-ahead heuristic with information exchange among terminals, (3) a method based on logistics queuing networks (Powell and Carvalho 1998), and (4) a serial scheduling method. The serial scheduling method integrates empty-car management with order scheduling. Therefore we could also compare local logistics control and integral logistics control.

Comparing on-time-service percentages for all methods (Appendix), we found that the coordinated look-ahead rule (method 2) and the serial scheduling method (method 4) work best. This shows that local control is not necessarily worse in a practical sense than integral control. Serial scheduling performs better than coordinated look-ahead for some special cases and requires five to 10 percent fewer kilometers in most instances. Because coordinated look-ahead fits better with local control, we recommended it. We also investigated the impact of incorporating early information about orders (van der Heijden et al. 2002, Ebben 2001). We found that knowing about orders 30
minutes in advance is all that is required in that the maximum travel time between terminals is 30 minutes. Our results provide input for decisions about the information and control systems that are to be implemented. More information leads to better logistics performance and fewer required resources, but it may also require more expensive information systems.

For one-way-tube control, a simple solution is a “traffic-light” solution, in which the permitted direction for travel in the one-way tube is switched periodically. However, we found this simple solution impractical, because the logistics performance was sensitive to the switching frequency and because the number of AGVs per direction changes over the day AGV failures would be a daily issue.

and may be asymmetric. Therefore we developed several adaptive rules for one-way-tube control, including simple rules based on the number of AGVs waiting at each side of the tube, look-ahead rules, and dynamic-programming-based algorithms, all with the goal of minimizing waiting times (Appendix). We found that adaptive control rules perform well and are robust in that they deal properly with changing workloads over the day. Dynamic programming yields the best results but also requires the most information. In fact, the trade-off is between more advanced information systems versus 10 additional AGVs. We provided our results as input for the investment calculations.

Because of traffic interactions at crossings, access lanes, and one-way tube entrances, the OLS will require a new type of traffic-control system. The existing AGV traffic control technology is considered to be reliable but inefficient in its use of space, and space is a scarce resource in our case (Evers and Lindeijer 1999, Evers et al. 2000). Therefore, we adapted the TRACES (Traffic Control Engineering System) concept, a new, intelligent AGV traffic-control framework developed by Evers and Lindeijer (1999). This traffic-control system is based on semaphore, that is, dynamic controllers for the workload (number of AGVs) on each infrastructure section using a ticket system. Thus, the system avoids conflicts over section use. AGVs act in a decentralized way in deciding which route to take when they arrive at a decision point (for example, a terminal entrance), so that the traffic-control system prevents local congestion (Appendix).

As a step towards implementing TRACES, we used simulation in cooperation with FROG Navigation Systems. The issues we had to solve to implement the TRACES principles in the simulation environment include the following:

—We had to define claim areas to be guarded by semaphores in such a way that we avoided AGVs becoming deadlocked. Because AGVs make their claims independently, simultaneous claims by several AGVs may lead to deadlocks. Therefore, we tested the layouts and their semaphores thoroughly until the layouts were free of deadlocks.

—Because of the time required for an AGV to communicate its claim to use a section and for the traffic-control system to grant the claim, a second AGV may try to claim the same area during this time. Therefore, we had to lock up the claim process with a so-called communication semaphore.

—TRACES did not take into account the AGV braking time. Because an AGV must be able to stop before entering the next area, the time for requesting entrance to that area must be scheduled to take the actual AGV speed into account.

We found that TRACES provides a safe mechanism for routing AGVs through a complex infrastructure. We used semaphores for small pieces of infrastructure, such as crossings and branches, and for large areas, such as terminals. To prevent congestion in the terminal, we restrict the number of AGVs in a terminal with a semaphore. Such a terminal semaphore can reduce AGV throughput times by 25 percent by preventing them from hindering each other.

Validating and Refining the Traffic-Control System at the Test Site (Research Question 6)

Because the technology was new, we wanted to test the equipment and control systems under laboratory conditions. To this end, the steering committee ordered the construction of a 40 m. × 40 m. test site in Delft, consisting of scale models and life-size prototypes of AGVs and docks. The test site focuses on automated transport and loading, automated control systems, and
automated information exchange without human intervention. We used the traffic-simulation model as a real-time control system to test and refine TRACES. We did not use real-time simulations for our first five research questions, because we needed to get reliable indicators of system performance quickly. Since we could implement only a small part of the OLS-system at the test site, we had to simulate the other part of the system, including the physical equipment. Therefore, we linked the simulation to the test-site system for controlling AGVs and docks.

At the test site, our simulation libraries acted as prototypes for the real-time control system. We thus avoided spending the time needed to develop the real control system, estimated to be between nine and 18 months. However, replacing simulated components with real objects in our models implied a number of consequences:

—We found that in the simulation environment the control object used information residing with the physical objects without proper requests. This problem would have to be resolved when the simulation model used real objects.
—In contrast to simulated AGVs, real AGVs’ failures cause deviations in the process, and the control system must be informed of such deviations. Therefore, we had to define additional failure functions.
—In reality, AGVs skid, turn differently, and position themselves less exactly than software AGVs. These deviations caused deadlocks, so we had to define extra margins for the semaphores.

Lessons Learned

The object-oriented approach we used appeared to be fruitful for a complex design project like the OLS with its many uncertainties. We developed a basic object library to answer questions in the first phase of the project. Later on, we refined the objects as the project required. We could include new aspects, such as providing energy, by adding the appropriate physical objects with corresponding information and control objects. Our experience is in line with others’ experiences in object-oriented simulation (Roberts and Dessouky 1998). Also our traffic-object library gave the terminal-and-dock-design subproject a real boost, facilitating flexible experiments. Further, the models gave clear direction to the design process and fostered designers’ creativity. During the terminal-design experiments, we simulation experts spent about 40 percent of our time on communication and on exchanging results with other members of the multidisciplinary design teams, which is worthwhile for building consensus.

Although we were very satisfied with the flexibility offered by the object-oriented approach, we also realized that a true object-oriented library is not as easy to construct as theory suggests. For example, we could not add failure behavior to our models by adding new objects for failure management; we had to add functionality to many objects that should react to failures, such as the empty-car manager (send a new vehicle to handle an order when an AGV fails) and the one-way-tube controller (do not switch the driving direction in a one-way tube section if a failed AGV is present).

Once we finished the object-oriented library, we shifted our attention to tuning control objects. We found that a new model constructed from our library ran but showed poor logistics performance. For example, in the project phase, when we added physical, information, and control objects for one-way tube sections to our library and models, our model ran, but our empty-car manager handled the strong fluctuations in throughput times that arose from waiting at the one-way tubes badly. We must design independent control objects but must also design control objects that are robust. Even with an object-oriented approach, one must adapt the existing control objects to cope with changed requirements. In the end, logistics performance is determined by the contents and interaction of objects and not by the library structure.

The experiments at the test site proved to be a necessary step before building the real OLS system. As far as we know, no one has described linking virtual and

Cooperation in a multidisciplinary team provides great benefits.
real objects in the literature before. Although the software within the physical equipment was almost a copy of the software with which we ran the simulations, the real equipment showed all kinds of deviations from the equipment in our model; AGVs skid and overshoot or undershoot on curves, and sensors may fail to function. Right now, tests with physical equipment remain necessary when developers transfer new technology from the research stage to the development stage. Further, in moving from the simulation to reality, we encountered deviations from the object-oriented fundamentals that we had to correct. An example is a piece of information on parking places that an AGV picked up in our models as a hidden assumption. In reality, the AGV cannot “read” anything from the physical parking place. We solved the problem by alerting the parking manager but only after an AGV was damaged at the test site. Clearly the test site was essential in designing a robust and intelligent control system. The testing phase will probably greatly reduce the number of problems during the final implementation.

Last but not least, we found that cooperation in a multidisciplinary project team provides great benefits. Still the multidisciplinary approach is difficult, because experts and scientists tend to focus on their own areas of interest. Cooperation requires constant attention. Using flexible simulation models as a common reference for the members of the multidisciplinary project team as they faced uncertainties had striking positive effects.

Acknowledgments
The authors thank Connekt for its funding of their simulation study. Also, they are grateful for the valuable suggestions of two anonymous referees and the managing editor that improved the structure and presentation of the paper.

APPENDIX

Empty Car Management
For repositioning AGVs to meet expected demand, we considered five options: a simple first-come-first-serve (FCFS) rule, FCFS with prior information on future orders, a coordinated look-ahead heuristic with information exchange between terminals, a planning algorithm inspired by Powell and Carvalho’s (1998) logistics-queuing-network approach, and a serial scheduling method. Using a serial scheduling method, one integrates management of empty cars with scheduling orders. Therefore we could also compare local logistics control and integral logistics control. We estimated the percentages of on-time service for the five methods (Table 4) and found that integral planning shows only slightly better performance than a well-chosen coordinated look-ahead rule. Below we describe the latter method in more detail. Ebben (2001) and van der Heijden et al. (2002) describe the five methods in detail.

We distinguish between global management of empty cars and local scheduling of empty cars (Table 1). The global empty-car manager has full control of AGVs outside the terminals. Within terminals, the global empty-car manager controls only the number of AGVs, and the local empty-car manager assigns AGVs to docks and orders. A global empty-car manager may ask the local empty-car manager to send empty cars to another terminal, but the local authority decides which AGVs to send and when. The global empty-car manager prioritizes requests for empty cars based on latest dispatch times, denoted by $s_{Lm}$. That is, the manager uses the latest time that an empty car can be dispatched from a specific terminal or parking area to pick up a transport job $l$ at terminal $j$ and still depart by the latest departure time $t_{Ll}$.

The local and global managers coordinate their activities as follows. The global empty-car manager periodically plans empty-car redistribution between terminals and parking areas (for example, every 10 minutes). It has two lists for planning purposes:

<table>
<thead>
<tr>
<th>Methods of Managing Empty Cars</th>
<th>Monday</th>
<th>Tuesday</th>
<th>Friday</th>
</tr>
</thead>
<tbody>
<tr>
<td>First-come-first-serve (FCFS)</td>
<td>94.7%</td>
<td>80.3%</td>
<td>85.5%</td>
</tr>
<tr>
<td>FCFS, order known 30 minutes in advance</td>
<td>97.1%</td>
<td>89.9%</td>
<td>93.2%</td>
</tr>
<tr>
<td>Coordinated look-ahead heuristic</td>
<td>99.8%</td>
<td>99.1%</td>
<td>99.7%</td>
</tr>
<tr>
<td>Logistics queuing network approach</td>
<td>99.5%</td>
<td>97.7%</td>
<td>98.8%</td>
</tr>
<tr>
<td>Serial scheduling approach</td>
<td>99.9%</td>
<td>98.9%</td>
<td>99.6%</td>
</tr>
</tbody>
</table>

Table 4: We estimated percentages of on-time service for five methods of managing empty cars over three order patterns, representing Monday, Tuesday, and Friday, respectively. Each simulation run covered 30 days. We chose the coordinated look-ahead heuristic that fits in with the local control framework.
An Automated Underground System

Table 5: Our analysis of four methods of controlling travel directions in one-way tubes showed that dynamic programming yields the highest percentage of on-time service. For both the look-ahead heuristic and dynamic programming, we assumed that AGV arrivals were known 25 minutes in advance.

<table>
<thead>
<tr>
<th>Methods for Controlling One-Way Tubes</th>
<th>On-Time Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Periodic control</td>
<td>90.2%</td>
</tr>
<tr>
<td>Adaptive control, local information</td>
<td>92.4%</td>
</tr>
<tr>
<td>Adaptive control, look-ahead</td>
<td>93.5%</td>
</tr>
<tr>
<td>Dynamic programming (DP)</td>
<td>99.1%</td>
</tr>
</tbody>
</table>

One-Way Tube Control

To control the traveling direction in a one-way tube, we considered periodic control, adaptive control, and optimization using dynamic programming (DP). For periodic control, we developed a heuristic to estimate the mean waiting time as a function of the traffic intensity for (compound) Poisson arrivals (van der Heijden et al. 2001). Our analysis of the logistics performance of these four methods showed that adaptive control and dynamic programming performed best (Table 5) (Ebben et al. 2000).

Adaptive control can be based on local information on queue sizes or, for the look-ahead variant, information on future AGV arrivals as well (Ebben et al. 2000). We use the following notation:

- $T =$ the time a single AGV takes to travel through the one-way tube.
- $\delta =$ the minimum time between two successive AGVs entering the one-way tube.
- $t_0 =$ the decision moment.
- $C(t_0) =$ the time allowed to clear the one-way tube at $t_0$, that is, the time needed for the last AGV traveling in a particular direction to leave the one-way tube.
- $t_i^r, t_i^l =$ the $i$th arrival at the right (left) side, where $t_i^r (t_i^l) \leq t_0$ if an AGV has already arrived.
- $q_i(t); q_i^r(t) =$ the number of AGVs in queue at the right (left) at time $t$.

If the current direction of travel is from left to right, the manager has two options: (1) to allow the queue at the left to pass first (no change in direction), or (2) to allow the queue at the right to pass first (a change). The mean additional waiting times for the options, denoted by $W_{\text{nochange}}$ and $W_{\text{change}}$ respectively, can be expressed as follows:

- $W_{\text{change}} = q(t_0)(C(t_0) + T + \delta \cdot \max(q_i^l(t_0) - 1, 0))$
- $W_{\text{nochange}} = q(t_0)(T - C(t_0) + \delta \cdot \max(q_i^l(t_0) - 1, 0))$

The manager changes the direction immediately when $W_{\text{change}} < W_{\text{nochange}}$. When the next AGV arrives, the
managers makes a new decision. The mean additional waiting time does not include waiting time already incurred.

Using dynamic programming, the manager searches for the sequence of convoys that minimizes total waiting time. We denote by $N_L$ ($N_R$) the number of AGVs from the left (right) within the information horizon at $t_0$ and make the objective function

$$\text{Min } \{ f_n(N_L, N_R, t^*) \}.$$ 

Here we define $f_n(i, j, t)$ as the minimum total waiting time for all AGVs present at the one-way tube or arriving within the horizon $H_i$ if at time $t$, $i$ AGVs that traveled through the one-way tube have already passed from the left and $j$ AGVs have passed from the right ($i = 0, 1, ..., N_L; j = 0, 1, ..., N_R$) and the last convoy came from direction $n$ ($n = L, R$). Hence $t^*$ is a point in time at which all known AGVs have been processed ($t^*$ is just an auxiliary variable for the recursion). $f_n(0, 0, t_0) = 0$, the cost functions $f_n(\cdot)$ can be formulated as follows:

$$f_L(i, j - k_2, t') = \min_{k_1 = 1, \ldots, j} \left\{ \sum_{y = i - k_1 + 1}^{y = j} \max[t - t'^y] + \delta \cdot (y - i + k_1 - 2, 0) + f_L(i - k_1, j - k_2, t') \right\},$$

$$f_R(i, j, t') = \min_{k_2 = 1, \ldots, j} \left\{ \sum_{y = j - k_2 + 1}^{y = j} \max[t' - t'^y] + \delta \cdot (y - j + k_2 - 2, 0) + f_R(i, j - k_2, t') \right\},$$

with

$$t' = T + \max[t + \delta \cdot (q'(t) - 1, t')]$$

and

$$t'' = T + \max[t' + \delta \cdot (q'(t') - 1, t'')]_y.$$ 

Here we neglect the fact that AGVs might arrive while the queue is entering the one-way tubes.

**TRACES: An Advanced Traffic-Control System**

To succeed, the project team must provide a safe transport system based on a safe and simple traffic-control system. One of the mechanisms that shows promise is the TRACES (Traffic Control Engineering System) concept, a new, generic, and layered AGV traffic-control framework developed by Evers and Lindeijer (1999) and Evers et al. (2000). Local autonomy, scalability, and hierarchy are built into the TRACES framework.

The basic idea behind TRACES is that the vehicles use a formal language (gathered in a script) to determine their movement through the infrastructure. Some of the statements in the language are responsible for executing actions, such as driving from one location to another. Other statements help an AGV claim critical pieces of infrastructure without safety mechanisms. They make the claims for critical pieces of infrastructure using semaphores. These semaphores are an extension of the semaphores used in concurrent programming (Dijkstra 1968, Tanenbaum 1990, Ben-Ari 1990) in that they can be multivalued. Each semaphore has a number of ‘tickets.’ Simple semaphores contain only one ticket; more advanced semaphores contain several tickets.

The AGV gets its script from a script-dispatcher control object, which has a virtual map of the terminal. When executing its script, the AGV requests access to conflict locations, such as joins or crossings, at local semaphores. If successful, the AGV receives a ticket,

\[ \text{Script for AGV1 to go from A to D:} \]

Exec AB
Inspect SemBCF, 1
Exec BC, SemBCF, 1
Exec CD

\[ \text{Script for AGV2 to go from E to D:} \]

Exec EF
Inspect SemBCF, 1
Exec FC, SemBCF, 1
Exec OD

**Figure 6: Basic principle of TRACES (Evers and Lindeijer 1999).** The AGVs execute small individual programs (“scripts”) when driving. The scripts contain commands for requesting one ticket to enter an intersection (“Inspect”), which are sent to a local intersection controller at the latest moment. This local controller grants access if no other vehicle has already claimed the intersection. Next, the AGV passes the intersection (“Exec”) and returns the ticket.
which it returns after leaving the conflict location (Figure 6).

An AGV can see in its script that, before accessing the conflict location ABC, it has to send a request to the semaphore in that area for a ticket with a capacity of one “INSIST SemBCF,1.” The semaphore SemBCF guards the conflict location. After receiving the ticket, the AGV may access the conflict location for which it requested a ticket in its previous statement. When the AGV has completely left the conflict location, the AGV returns the ticket to the semaphore.

The script dispatcher control object can assign scripts based on a wide range of conditions, such as the traffic density in different areas, the destination of the AGV, information about failures, and the actual status of the AGVs battery. Furthermore, it can add intelligence to the scripts, so that the AGV can select the least-congested route dynamically.

References


system reliability as function of AGV and dock failures.

“In this way, the simulation group played an important role in our multidisciplinary project team. The results of the simulation group are basic for the AGV test site, opened quite recently at Delft. Technologically, we consider this test site as significant progress compared with previous AGV test programs in the Rotterdam area, an addition to the Dutch economy with considerable added value in its own respect. Commercially, the test site is a great facility to interest our future customers for this innovative initiative by Schiphol Airport and Flower Auction Aalsmeer.”

Rudi de Vos-Burchart, R&D director, Frog Navigation Systems B.V., Cartesiusweg 120, 3534 BD Utrecht, The Netherlands, writes: “One of overall goals in this project was to show the feasibility of an AGV based solution for the large-scale transport problem arising in the Schiphol area. The contribution of the simulation group was essential to analyze the logistical performance of the system.

“However, for our company the interaction with the simulation group went far beyond this. In this project, we developed a new and innovative generation of AGV control software. Simulation was extremely helpful in developing and testing our new control system. A prototype of the control system has been evaluated in a flexible object oriented simulation environment, revealing many opportunities for refinement and optimization. Also, we got insight in the performance of our control system in terms of AGV throughput times at various types of tracks (intersections, access, and exit roads), docking behavior, radio communication traffic on terminals, etc. Because we could connect our prototype AGV’s as physical objects in the simulation model at a test site, we could evaluate the control system under real life circumstances in an early development stage. This also gave us the confidence that we had chosen the right concept for our new control system.

“As a consequence, our development time has been reduced considerably. We are very satisfied to be able to use this sort of simulation now at the AGV test-site as an overall control tool to analyze realistic terminals. It is really an advantage for the costs of testing that virtual simulation objects and our real prototype AGV’s can be intermixed. Further, based on previous experiences we consider this as an important achievement to avoid future problems in the real system.”

P. Luitjens, Project Leader, Schiphol OLS, Stichting Initiatiefgroep Het Ondergronds Logistiek Systeem (OLS), Postbus 7501, 1118 ZG Luchthaven Schiphol, The Netherlands, writes: “Simulation was very helpful both to analyze the logistical performance and to provide input for our exploitation figures. Due to this work we feel confident that reliable service can be achieved with a system layout partly with single (one way) underground tubes instead of double (two way) ones. This reduces the investment costs in the order of magnitude of 40%. Also, the simulation results provided us input for a proper time-phased construction planning of the OLS, thereby saving considerable interest expenditures by postponing investments. Moreover, we got well-founded data for our exploitation analysis, such as number of AGVs depending on their speed, the size of the terminals and the parking, etc. Due to the flexibility of the simulations we were able to optimize the system design, which altogether leads to savings in the order of another 10% compared with our first guesses. Finally, it was important that the simulation group showed that the sensitivity of throughput times for AGV and dock failures is not too large. This gave us confidence in the reliability of the OLS. Another benefit of the simulation we derived by using the imagination. In convincing stakeholders and future users of the feasibility of the OLS concept our arguments were strongly supported by the images.

“Altogether, it is my opinion that the simulation results were very valuable as input in the decision process and, referring to the DSS for exploitation that are currently being developed, they will remain so in the near future.”