Multi Agent Simulation for Decision Making in Warehouse Management

Massimo Cossentino, Carmelo Lodato, Salvatore Lopes, Patrizia Ribino
ICAR Institute
National Research Council of Italy
Palermo, Italy
Email: {cossentino, c.lodato, toty, ribino}@pa.icar.cnr.it

***Abstract*—The paper presents an agent-based simulation as a tool for decision making about automatic warehouses management. The proposed multi-agent system is going to be used in a real environment within a project developed with a company working on logistics. More in details, we have developed a simulation framework in order to study problems, constraints and performance issues of the truck unload operations. We aim to optimize the suitable number of Automated Guided Vehicles (AGVs) used for unloading containers arrived to the warehouse. This is a critical issue since an AGV is a costly resource and an augment in number does not necessarily correspond to an improved unloading speed. The experiment performed with our simulated environment allows us also to evaluate the impact of other elements to the performance.**

I. Introduction

Making effective and successful decisions about complex systems is a hard task, especially in business environments. This process usually exceeds human cognitive capabilities because of the huge amount of parameters influencing such systems. The human intuitive judgment and decision making become far from optimal in respect to the growing of the complexity. The quality of decisions is extremely important in many practical situations because a wrong or an ineffective decision could cause a great waste of resources. Overcoming the deficiencies of human judgment is one of the biggest challenges of the scientific community.

Nowadays, simulations are often used in scientific and research contexts in order to evaluate the behavior of several complex systems and especially the behavior of dynamical systems. The simulated system should have the capability of continuously reacting, with a re-organization process, to changes occurring in the environment. Because of their intrinsic nature, agents have been recognized to be a good way for solving complex problems [1][2].

Several studies are being carried out in the field agent-based simulations. Some interesting contributions are given by Franziska Klügl [3][4], Seth Tisue et al. [5], Sean Luke et al. [6] and Nick Collier [7].

In [4] F.Klügl et al. present an integrated framework, named SeSAm (Shell for Simulated Agent Systems), allowing the creation of simulated environments suitable to several kinds of context such as Logistics (coordination, storage layout optimization), Traffic (avoidance of traffic jams, traffic light control), Passenger Flow (market improvement, evacuation of buildings) etc...

Tisue et al. [5] have developed NetLogo, a modeling tool for simulating natural and social phenomena.

MASON [6], proposed by Sean Luke et al., is an extensible, discrete-event multi-agent simulation toolkit in Java. It was designed for a wide range of multi-agent simulation tasks ranging from swarm robotics to social complexity environments.

Finally, RePast [7] is a software framework for agent-based simulation created by Social Science Research Computing at the University of Chicago. It provides an integrated library of classes for creating, running, displaying, and collecting data from an agent-based simulation.

In the field of the logistics, several agent simulations are proposed for different purposes such as modeling and management of supply chains [8][9][10], optimization of production planning [11], traffic [12] etc...

The problems addressed in this paper concern the optimization of an automated logistic warehouse. In such kind of warehouse the handling of goods is performed by means of Automated Guided Vehicles (AGVs). Usually these vehicles move along optical guides drawn on the warehouse floor. These optical guides are defined at design time of warehouse and they are used during its entire life cycle imposing constraints about the traffic. A critical issue is the efficient employment of resources in order to avoid overcrowding of the guides.

In addition, another element constraining the performance of a logistic warehouse is the sorter, which task is directing toward a new destination the goods unloaded by AGVs. Generally, a sorter has a given capacity (sorting speed) to be considered in order to balance the elements whose a warehouse is composed of.

In this paper, we propose an agent-based simulation in order to solve a decisional problem about warehouse management generating performance measures. The simulation has been developed using Jason [13], a Java-based interpreter for an extended version of the AgentSpeak [14][15] language based on the BDI (Belief-Desire-Intentions) model [16].

The work presented in this paper was carried out under the IMPULSO [1] (*Integrated Multimodal Platform for Urban and*

[1] Further information available at http://www.vitrociset.it - Section Ricerca&Sviluppo

extra urban Logistic System Optimization) project funded by the Italian Ministry for Economic Development.

The remainder of the paper is organized as follow. The section II introduces the decision making problem addressed in this paper and defines the simulation objectives. Moreover it provides an overview of the AgentSpeak language and Jason interpreter. In section III, we then proceed to the presentation of the multi agents architecture for the proposed simulation by specifying the features of the agents and the environment in which they will perform their activities. The section IV shows the performance results obtained from the simulation which allow us making considerations. Finally some discussions and conclusions are drawn in sections V and VI respectively.

II. Theoretical background

The decision problem discussed in this paper concerns some aspects of the IMPULSO project. IMPULSO aims to develop new technologies and capabilities in order to improve the management and transport of products, based on cooperation models while ensuring highest levels of security. It offers an integrated system for goods management within the logistic districts, for their storage in special metropolitan distribution centers and finally, for distribution within the cities. One of the issues addressed by the IMPULSO project concerns the evaluation of efficient resources employment to be adopted for the handling of goods inside a logistic district. For clarity, a logistic district is a large area composed of several warehouses where the freight forwarders deliver their container.

In this paper, we see in detail a node of the supply chain. The specific case concerns the management of the automatic container unloading by means of the use of AGVs inside a warehouse of the logistic district.

In the presumed scenario, containers are carried by articulated lorries. Whenever a lorry arrives to the warehouse the container has to be unloaded. The container holds several kinds of goods grouped in boxes called *pallet*. Each pallet must be unloaded from the container and carried to a specific area dedicated to the sorting of goods. In this area each pallet will be opened and its contents (packages of goods) sent to a sorter. The sorter will cluster the packages according to their destination. Finally, smaller vehicles (e.g.: eco-friendly trucks) will take them to their new destination (usually in town).

The transport of pallets toward the sorting area is committed to automatic vehicles with optical guidance (AGVs). This means that each warehouse inside a logistic district must be equipped with appropriate optical signals which defines all permissible paths for an AGV.

Each defined layout of optical paths imposes limits on the use of the resources (AGVs). In accordance with the available paths only some AGVs can work effectively at the same time. Since an AGV is a costly resource, it is crucial to establish how many AGVs can work at the same time without getting in each others way thus delaying the unloading operations.

The choice of the maximum number of AGVs is also constrained by the capacity of the sorter. In other words, a semi-automatic sorter can process a maximum number of packages per time unit. Thus, within the limits imposed by the available paths, an augment in number of AGVs does not necessarily correspond to an improved unloading speed because of saturation of the sorter. The sorter could actually be a bottleneck of the warehouse and it can cause long waiting queues. A proper allocation of resources, which respects the constraints imposed by the warehouse layout and by the sorting capacity, can significantly reduce management costs.

In addition, it is useful to establish what are the critical paths inside a warehouse, that is those whose unavailability can cause a traffic block. For these reasons, we use a multi-agent simulation as a tool for decision making about warehouse management.

The simulation allows us to explore the variables constraining the problem. In this instance, we want to establish, for a given warehouse configuration, not only what is the maximum number of AGVs usable in order to maintain high performances but also what are the critical elements of the system.

The optimization of the warehouse layout is out of the scope of this paper because it is a task of another component of the IMPULSO project. However, our work provides some useful information for improving the design of automatic logistic warehouses.

The next subsection provides an overview of the tools used for the simulation.

A. Development environment

We decided to adopt a multi-agent based solution because it adequately fits the real scenario coming from the IMPULSO project. In fact, real AGVs are autonomous robots capable of executing the mission they received by a mission controller. Among the available platforms we decided to use Jason [13] that offers relevant utilities for the implementation of such system.

Jason is a Java-based interpreter for an extended version of AgentSpeak [14][15], a Prolog-like logic programming language. One of the most interesting aspects of AgentSpeak is that it based on a the belief-desire-intention (BDI) model[16].

In the BDI model, agents continually monitor their environments and act to change them, based on the three mental attitudes of belief, desire and intention.

Beliefs are information the agent has about the world (i.e. itself, others agents and the environment), which could also be out of date or inaccurate.

Desires represent all possible states of affairs that an agent would achieve. A desire is a potential influencer of the agents actions. So it is possible for a rational agent to have desires that are mutually incompatible each other. Desires can be represent possible option for an agent.

Intentions are the states of affairs that the agent has decided to work towards. An agent looks at its options and chooses between them. Options selected in this way become intentions.

The behavior of agents in Jason is defined by means of a set of plans created in AgentSpeak.

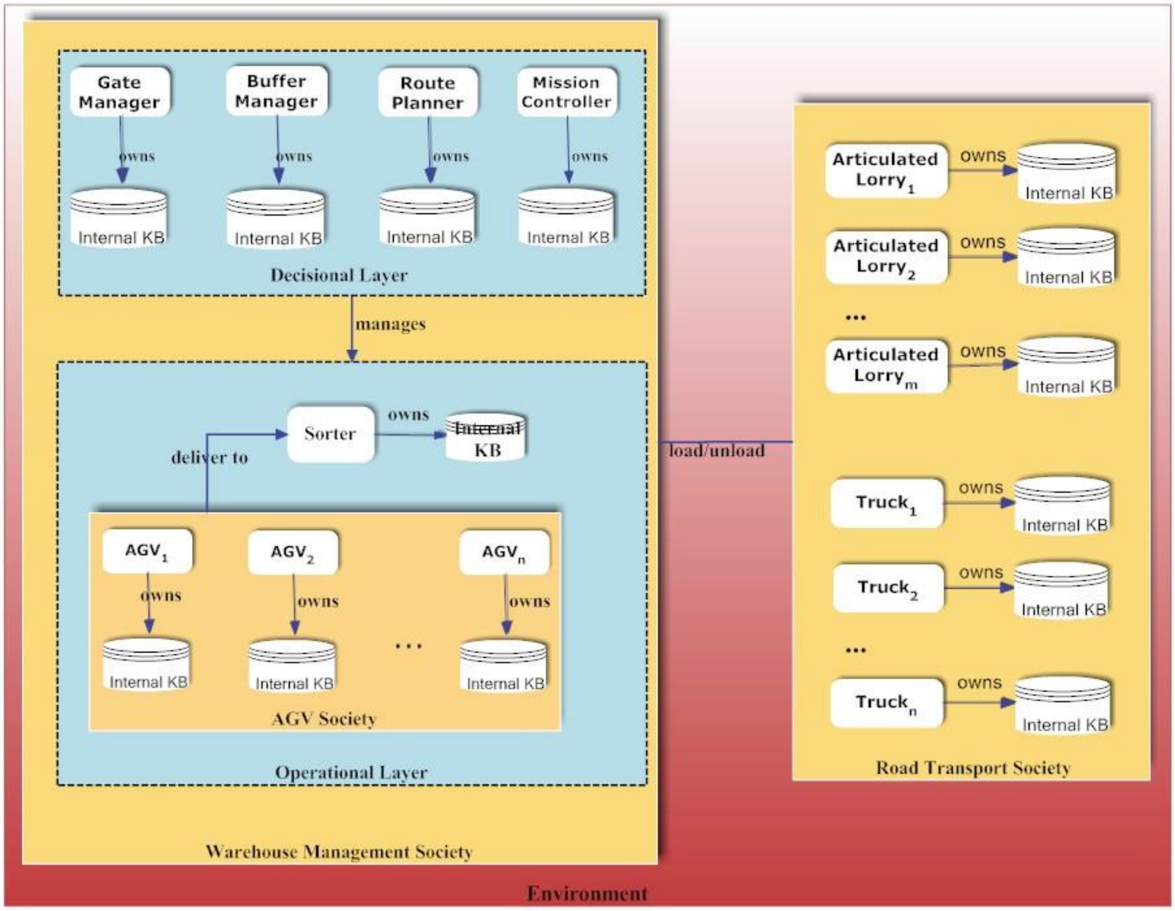


Fig. 1. The Multi-agent system architecture.

Practically, agents respond to the perceptions coming from environment changes. Such perceptions influence beliefs and commitment of agent goals. Agents respond to these changes by selecting plans from the plan repository for each change of beliefs and then by instantiating one of these plans as an intention. These intentions can be composed of actions, goals and plans to be achieved.

A plan in AgentSpeak is composed of three main elements organized in the following form:

$$+triggeringEvent : context < -body$$

The **triggeringEvent** describes the situations in which a plan may be applicable for the execution. The **context** can be used for specifying the condition to make the plan applicable even if an event has triggered that plan. The **body** can be considered the consequent of the event linked to the context. Within the body commonly are defined the actions that an agent must perform to fulfill its own goals.

In the next section we describe the design and the implementation of a multi-agent organization used for the proposed simulation.

III. The Proposed Simulation Framework

The proposed simulation framework is based on an agent organization situated in a specific environment. The multi agent organization is composed of (see fig.1):

- a *Warehouse Management Society* governing the activities inside a warehouse;
- a *Road Transport Society* for goods transportation from/toward a logistic district.

The agents of the Warehouse Management Society belong to different layers in accord with the role played in the society. We distinguish two layers: the *Decisional Layer* and the *Operational Layer*. Agents playing managerial roles belong to the former while the second is defined by the agents which perform operational activities. The decisional layer of the Warehouse Management Society is composed of four agents: the Gate Manager, the Buffer Manager, the Mission Controller and the Route Planner. The operational layer is formed by a Sorter agent and an AGVs Society.

The Road Transport Society consists of different means of transport such as articulated lorries and trucks.

In the subsection III-A we paid attention on constitutive elements of the environment. While the features of each agent will be defined in the subsection III-B.

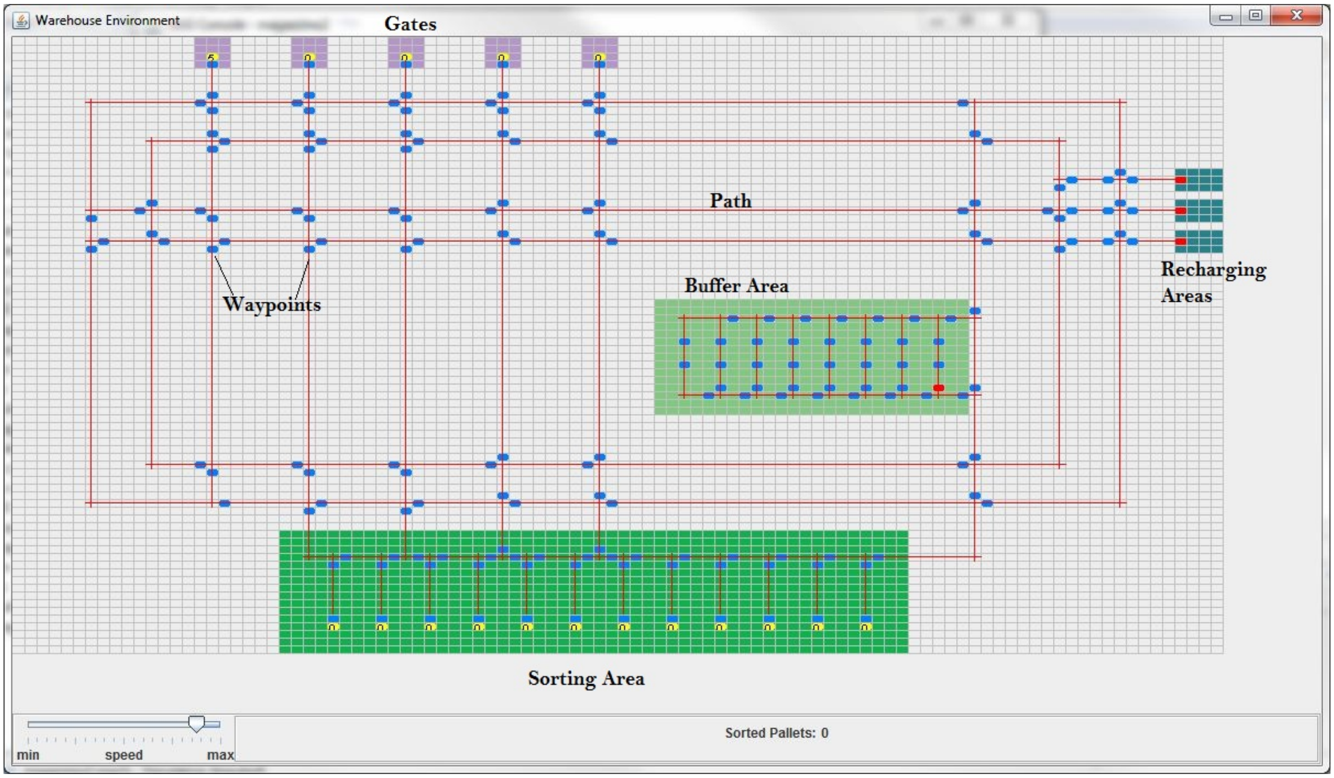


Fig. 2. The simulated environment.

A. Environment

As usual, we have defined not only the elements the environment is made of but also how agents can interact with the environment. Specifically we have defined what an agent perceives, when the agent is able to perceive and finally how the actions it performs influence the environment.

The studied environment represents a real warehouse situated inside a logistic district. It is a very dynamic environment because there are several agents performing unsynchronized actions but it is also an open environment due to the exchanges with the outside world.

The elements of such environment are (see fig. 2):

- a set of *Gates* in which articulated lorries can park waiting for unloading;
- a set of *Recharging Areas* where the AGVs can recharge their batteries;
- a *Sorting Area* where the pallets are forwarded toward a new destination trough several input points (called *Sorter Places*);
- a *Buffer Area* where it is possible to temporarily store pallets when the sorter is busy. This area is also used for parking AGVs that are waiting for a new mission;
- a set of possible *Paths* representing the optical guidance for AGVs. Each route section (i.e: path connecting only two waypoints) is usually one-way, but some of them can be two-way (e.g: the entrance of gates);
- a set of *Waypoints* near to the crossing points of paths.

B. Agents

For our purpose we have defined different kinds of agents (see fig.1):

- the *Gate Manager* manages the allocation of gates at the arrival of articulated lorries. It also takes into account the amount of pallets to unload;
- the *Buffer Manager* governs the parking areas and buffering. It can reserve a parking place for the agents that require it;
- the *Route Planner* allocates the paths for AGVs. Each path is computed by means of Dijkstra's shortest path algorithm [17];
- the *Mission Controller* implements the strategy of container unloading. It also assigns to each AGV the mission of carrying pallets from gates to the sorter in accord to a nearest neighbor policy;
- the *Sorter* manages the work inside the sorting area and communicates the free place where it is possible to deliver a pallet for an AGV. Moreover, it interacts with Truck agents for loading the ready boxes for the delivery toward new destinations;
- the *AGV* is the agent that simulates the behavior of real forklift that performs the pallet transport inside a warehouse from arrival gate to sorting area;
- finally the *Articulated Lorry* and the *Truck* are the agents that perform the transport of goods toward and from a logistic district respectively.

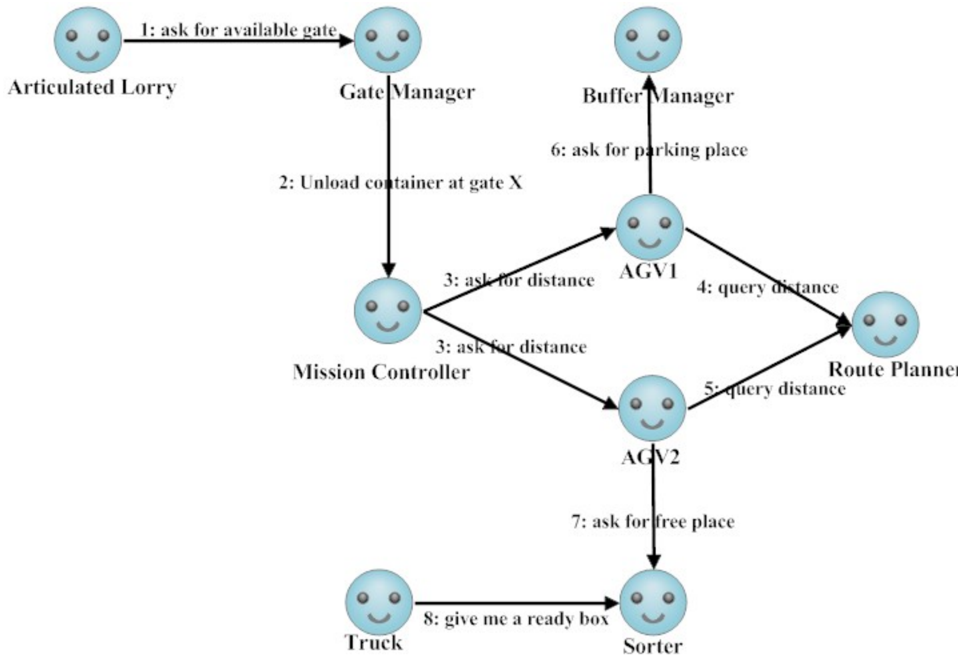


Fig. 3. A typical operative scenario.

Each agent owns an internal knowledge base. The agent knowledge base contains its initial beliefs, the beliefs resulting from perceptions about the environment, the goals and the plans to achieve them.

In the next section we show the experimental results obtained from simulation.

IV. Experimental setup

The tests have been conducted on a warehouse configuration coming from specifications of the IMPULSO project, but we want to underline that the multi agent model adopted for the simulation is independent of the specific warehouse configuration.

In the proposed instance, the simulated environment is a warehouse consisting of:

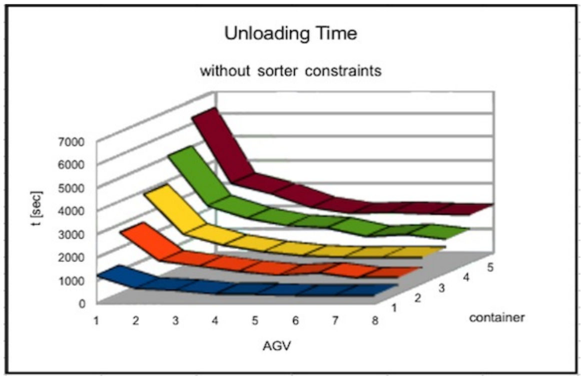
- n°5 Gates where the articulated lorries leave their containers waiting to be unloaded;
- n°1 Sorting Area with twelve Sorter Places (where the pallets are left in order to be addressed toward next destination);
- n°3 Recharge Areas where an AGV goes, whenever its battery is low;
- n°1 buffer area with 16 places;
- n°62 crossing with 142 waypoints;
- n°70 oriented route sections.

When the simulations start, the warehouse settings are the following:

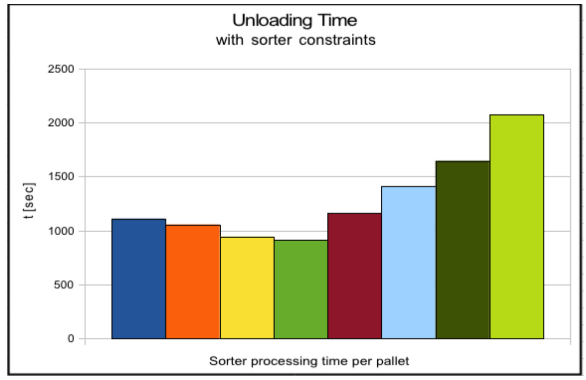
- the AGVs are located in different places of the warehouse (such as recharge area, buffer area, etc…). We assume they always are on a waypoint;
- all gates are free;
- the sorter is empty;
- all paths are available.

As previously said, we adopted the Dijkstra algorithm for finding the shortest path between gate and sorter and vice versa. The algorithm has been implemented in such a way to provide an alternative path if the shortest one is already busy. In fact, according with the specification of the IMPULSO project, we adopt a very conservative policy in order to avoid collisions between AGVs. This policy consists in reserving the entire path assigned to each single AGV if it is possible. Otherwise we reserve only an alternative intermediate path. We are conscious that the actual reservation strategy may probably cause a waste of time but we want to avoid any chance of a collision between AGVs because too costly and dangerous.

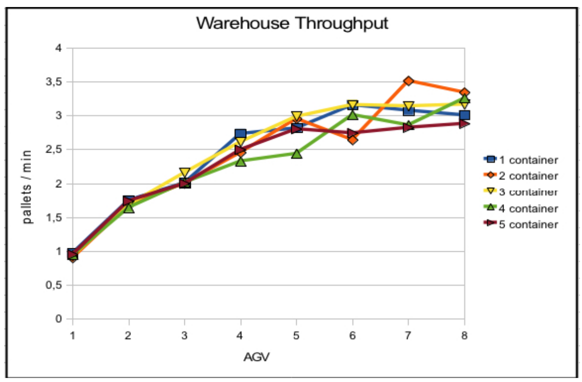
During our simulations, several articulated lorries may arrive. Each of them carries a container with a certain number of pallets. The receiving and unloading of different containers can be performed at any moment. When there is a container ready to be unloaded, some AGVs are assigned to take a pallet from the container and to transport that to the Sorter. The commitment is defined by the Mission Controller agent according to the nearest neighbor policy. The figure 3 shows a simplified diagram of the communications among agents during a typical simulation scenario. We prefer to show only the most meaningful messages exchanged among agents for the sake of clarity. In this scenario only an Articulated Lorry arrives. It asks to the Gate Manager for an available gate where to park. Then, the Gate Manager acquires information about the arrived load, and subsequently, it informs the Mission Controller that there is a container to unload at gate X. The Mission Controller asks to each AGV its distance from the gate



(a) The time spent by AGV (Automated Guided Vehicle) for simultaneously unloading containers using a sorter with infinite capacity.



(b) The time spent by 4 AGVs to unload 100 pallets from 5 containers versus the pallet processing time at the sorter.



(c) The number of pallets unloaded per minute versus the number of AGVs for different numbers of containers.

Fig. 4. Simulation results

where the lorry is parked. The closest AGV is committed to the unload of the first pallet. Then the committed AGV asks to the Sorter for an available place, while other not committed AGVs go to the parking zone. In accordance with our requirements, the Sorter always assigns places starting from the right side of the sorting area (see figure 2). It is worth to note that although in this scenario we suppose to employ only one AGV, usually all available ones (with a proper sequence) are committed to the unloading task.

The Route Planner agent determines, for each AGV, the shortest path among those available, also providing a measure of the distance. Pallets are left in the sorting area for a fixed amount of time simulating the time spent by human operators to transfer goods from the pallet to the boxes. When available, Truck agents release the places occupied by the already filled boxes by loading them.

We conducted several simulations with different parameter values in order to evaluate the behavior of the system. Particularly, we observed the relevance of two variables: the number of AGV agents versus the number of containers that are simultaneously unloaded. The charts shown in figure 4 highlight the results obtained from these simulations. These results are discussed in section V.

V. Discussion

Diagrams in Figure 4 show the results of several simulations for the warehouse configuration of the case study at issue.

All reported times are scaled according to the real time. We would like to underline that the reported results do not suffer of any random influence.

The diagram shown in the figure 4(a) displays the progress of the unloading time depending on the number of used AGVs and the number of unloaded containers during different simulations (20 pallets for each container). This diagram is based on the assumption that the time spent by the sorter to process one pallet is zero, this corresponds to have an infinite capacity sorter ($C_{sorter} = N^o SorterPlace/ProcessingTime$).

More in details, in the same figure we can see that the time necessary for unloading one container, decreases with the number of AGVs but the slope of the curve significantly decreases after about 4 or 5 AGVs.

Any decision about the acquisition of the suitable number of AGV should start from the estimation of the average number of containers that are simultaneously unloaded. In the following we will suppose this is 3. In this scenario, our simulations suggest the following decision guidelines: if the preferred criterion is optimizing the cost/benefit ration of the AGV employment, from figure 4(a) we can see that five AGVs is a reasonable choice. Buying more AGVs does not contribute significantly. For instance, with 5 AGVs we can unload 3 containers in 1205 seconds while with 6 AGVs we need 1139 seconds. The difference (5%) does maybe not justify the increase in cost.

Conversely, if no compromise may be accepted on the unloading time, at all costs, the suggested number of AGVs is 6. Of course we leave this strategical choice to the warehouse manager.

After that we have estimated the number of employed AGVs, we have conducted additional simulations in order to define the impact of the capacity of the sorter on the system performance.

The diagram shown in the figure 4(b) displays the warehouse performance using five AGVs and varying the processing time per pallet of the sorter. We can deduct that in this case the layout of the warehouse influences the performance of the system. In fact increasing the sorter processing time the first places (those on the right side of the sorting area in figure 2) are emptied more slowly forcing AGVs to deliver their pallet in the places positioned in the middle of the sorting area. Since these latter places are closer to gates than the previous ones, the unloading time decreases. This phenomenon may be observed in figure 4(b) for the first 4 experiments (with 0, 30, 60, 90 seconds of processing time per pallet). When the processing time of the sorter exceeds 90 seconds per pallet, the sorter begins to saturate thus causing longer waiting queues and consequently increasing the unloading time.

It is worth to note that 90 seconds is about the time spent by an AGV to carry a pallet from the gate to the sorter. This time is obviously a critical time for the whole system.

These allow us to highlight that the actual policy of allocation of sorter places (that starts always on the right of the sorting area) is far from optimum. As we previously said the optimization of the warehouse layout is out of the scope of the paper, nonetheless we can still use this simulation to suggest a better sorter allocation policy which prefers the middle sorter places when it is possible.

Moreover these simulations have highlighted that there are some critical elements in the given warehouse layout. As a matter of fact, there are some paths that are busier than some others (busier paths are located at the rightmost side in figure 2) and less used paths (those on the leftmost side). This is caused by the actual sorter allocation policy.

Finally, figure 4(c) shows the throughput of the warehouse. This is measured by computing how many pallets are unloaded per minute. This number depends on the number of AGVs and the number of containers to be simultaneously unloaded. In the cited figure, we can observe that the throughput for five AGVs is about 2,8 pallets per minute while using more than five AGVs we can obtain only little improvements. In fact, increasing the number of AGVs the throughput goes towards three (3,1 pallets per minute with 8 AGVs). This diagram highlights once again that adopting five AGVs is a reasonable choice.

VI. Conclusion

Multi-agent simulation is proposed as a tool for making decision about logistic problems. The multi-agent model adopted was tested for the simulation of real warehouse layouts. We have pointed out that the structural constraints of the given warehouse configuration limit the productivity. We have highlighted these limits and consequently we have made some considerations about the employment of resources.

We are currently exploring other resource allocation strategies for paths and sorter places in order to improve the performances of the system. Moreover at the moment we are working on the development of more effective strategies for a better exploitation of AGVs capabilities.

Moreover we are also prefiguring the application of an extension of our system to the study and optimization of the warehouse structural design including the number and position of gates as well as the internal layout.

Acknowledgment

This work has been partially funded by the IMPULSO *Integrated Multimodal Platform for Urban and extra urban Logistic System Optimization* project funded by the Italian Ministry for Economic Development and coordinated by Vitrociset S.p.A.

A special thanks to Dr. M. Bordin of Vitrociset S.p.A. for the extremely fruitful technical discussions about the details of the IMPULSO project and the related issues of logistics.

References

[1] M. Wooldridge and N. Jennings, "Intelligent Agents: Theory and Practice," *The Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152, 1995.

[2] M. J. Wooldridge, *Introduction to Multiagent Systems*. John Wiley & Sons, Inc. New York, NY, USA, 2001.

[3] F. Klügl, "Agent-based simulation engineering," *Habilitation Thesis, Würzburg University, Germany*, July 2009.

[4] F. Klügl, R. Herrler, and M. Fehler, "Sesam: implementation of agent-based simulation using visual programming," in *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*. ACM, 2006, pp. 1439–1440.

[5] S. Tisue and U. Wilensky, "Netlogo: A simple environment for modeling complexity," in *International Conference on Complex Systems*. Citeseer, 2004, pp. 16–21.

[6] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan, "Mason: A multiagent simulation environment," *Simulation*, vol. 81, no. 7, p. 517, 2005.

[7] N. Collier, "Repast: An extensible framework for agent simulation," *The University of Chicagos Social Science Research*, vol. 36, 2003.

[8] J. Swaminathan, S. Smith, and N. Sadeh, "Modeling supply chain dynamics: A multiagent approach*," *Decision Sciences*, vol. 29, no. 3, pp. 607–632, 1998.

[9] T. Lee, N. Park, and D. Lee, "A simulation study for the logistics planning of a container terminal in view of scm," *Maritime Policy & Management*, vol. 30, no. 3, pp. 243–254, 2003.

[10] M. Fox, M. Barbuceanu, and R. Teigen, "Agent-oriented supply-chain management," *International Journal of Flexible Manufacturing Systems*, vol. 12, no. 2, pp. 165–188, 2000.

[11] A. Karageorgos, N. Mehandjiev, G. Weichhart, and A. H
"ammerle, "Agent-based optimisation of logistics and production planning," *Engineering Applications of Artificial Intelligence*, vol. 16, no. 4, pp. 335–348, 2003.

[12] B. Burmeister, A. Haddadi, and G. Matylis, "Application of multi-agent systems in traffic and transportation," in *Software Engineering. IEE Proceedings*, vol. 144, no. 1. IET, 1997, pp. 51–60.

[13] R. H. Bordini, J. F. Hübner, and M. J. Wooldridge, *Programming multi-agent systems in AgentSpeak using Jason*. Wiley-Interscience, 2007.

[14] A. Rao, "AgentSpeak (L): BDI agents speak out in a logical computable language," *Agents Breaking Away*, pp. 42–55, 1996.

[15] M. d'Inverno and M. Luck, "Engineering agentspeak (l): A formal computational model," *Journal of Logic and Computation*, vol. 8, no. 3, p. 233, 1998.

[16] A. Rao and M. Georgeff, "Bdi agents: From theory to practice," in *Proceedings of the first international conference on multi-agent systems (ICMAS-95)*. San Francisco, 1995, pp. 312–319.

[17] E. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.