Concept of a Multi-agent Based Decentralized Production System for the Automotive Industry

Christian Blesing, Dennis Luensch, Jonas Stenzel[(✉)], and Benjamin Korth

Fraunhofer Institute for Material Flow and Logistics, 44227 Dortmund, Germany
{christian.blesing,jonas.stenzel}@iml.fraunhofer.de

Abstract. To face the challenges of today's market requirements, a huge effort is made to plan continuous flow manufacturing systems used today. Simultaneously disturbances during the production have decisive negative effects on the effectiveness. To mitigate this problem, current research programs try to use flexible production systems with a high degree of self-organization. In this paper a novel concept for a flexible decentralized production system is described which combines the planning method of a precedence graph and a multi-agent-system that forms a modular control system. Furthermore first results are presented that have been achieved by a pilot demonstrator and simulation experiments.

Keywords: Cyber-physical systems · AGV routing · Path planning · Task assignment · Production planning · Decentralized production systems · Agent ontology · Industry 4.0 · Intra-logistic simulation

1 Introduction

Nowadays, companies are faced with the increasing need to operate effectively and efficiently. Within automotive industry, mass-market vehicles are commonly produced with a continuous flow manufacturing system that consists of a combination of highly efficient production and assembly lines. Therefore, the automotive industry planning processes needs a high effort to keep assembly lines fully utilized. This is because of the high complexity in the product structure with a high diversity in product variants and the huge number of parts that must be delivered just in time or even just in sequence at the right place at the assembly line. Through the high degree of dependencies in the assembly line the effects of disturbances are critical [4]. Delays in part supply or assembly steps directly influence other orders as uncompleted steps or missing parts can lead to a higher amount of rework. *Yu et al.* [13] present a reconfigurable manufacturing execution system (RMES) to face this problem. Within this paper, a novel concept how to replace the assembly line by a flexible production system based on workstations which is controlled by an autonomous acting planning and transport system is described. Each workstation can have multiple capabilities to execute assembly steps and therefore can provide redundancy. Through the loosely coupling of the assembly stations alternative sequences in the process of assembly

steps become also feasible. With the new obtained advantages, delays in a single step do not lead to delays for other orders and may be compensated by other stations. Audi, a German automobile manufacturer, has announced that they are working on modular assembly strategies using a similar concept [7]. The project SMART FACE focused on the development of a simulation and a proto-type demonstrator using this approach. Therefore, as a partial result, this paper describes the use of an agent based production planning and control system to move car bodies and parts along a flexible assembly process.
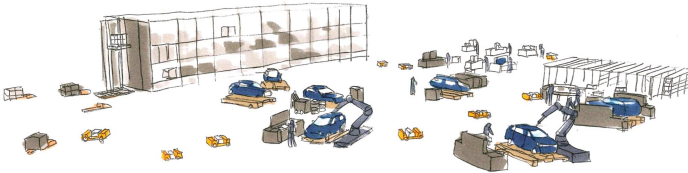

Fig. 1. The vision of SMART FACE: decentrally and autonomously acting units (e.g., assembly stations and automated guided vehicles) are responsible for transporting and producing goods on a self-organizing shop floor. Human workers are still an important part since they exhibit larger flexibility than most of their technical counterparts.

2   Algorithmic Background

Initially, an overview of some promising task assignment and route planning algorithms for **A**utomated **G**uided **V**ehicle (AGV) and mobile robot systems within intralogistics is given as these are essential to create highly automated and flexible production systems.

2.1   Overview of Dynamic Task Assignment Algorithms

In order to maximize the throughput of manufacturing systems, warehouses and outbound logistics, the use of AGVs for transporting goods has increased steadily since the 1950s. The main objective to achieve this goal is to find an optimal solution regarding the assignment of transportation tasks to the AGVs. One approach to solve this problem is to optimize both, the loading point-AGV assignment and the scheduling of tasks to different AGVs. *Giglio* therefore sep-arates both tasks and formulates each problem as an optimization problem [5].

Another approach which is described by *Branisso et al.* is the definition of an optimality criterion by defining a fuzzy inference system in which the transport agents that represent the different AGVs, decide which transport order they accept next (cf. [3]). In general, optimality criteria have to match the constraints of the production planning system (e.g., the latest possible finishing time) and they have to be known by the other software modules).

Schwarz et al. [11] presented a decentralized approach based on a **m**ulti **a**gent **s**ystem (MAS) for handling transport orders with AGV's in a beverage bottling

line. Hence their approach focuses primarily on procurement, routing and conflict resolution. Contracts are assigned through FIPA auctions, with a main focus on the order calculation. AGVs appear as bidders, stations as vendors within the auctions. Bids were made based on the occupancy and position of a vehicle. Within an evaluation the decentralized approach was tested against a central procedure. It turned out that the decentralized approach leads to advantages with regard to a higher utilization of the vehicles as well as a shorter throughput time of the orders.

In this paper, the task assignment problem is solved by applying auctions similar to *Branisso et al.* and *Schwarz et al.* and by defining three decision points (for production order-, partial order- and transport order assignment respectively) whereas each decision point has its own fuzzy sets. The algorithm is further described in Sect. 6.

2.2 Overview of Dynamic Route Planning Algorithms

Path planning and navigation for a fleet of mobile robots or AGVs is a challenge that has been addressed by lots of researchers. The following sections focuses some popular path planning approaches. One approach to the coordination of multi-robot path planning is prioritized planning, where robots plan their trajectories sequentially one after another. *Čáp et al.* adopted variants of classical prioritized planning algorithm to decentralized ones [12]. They proved that the decentralized implementation is guaranteed to provide a solution under the same conditions as its centralized counterpart. In addition, their revised prioritized path planning approaches tend to find path planning solutions for scenarios with a high number of robots where classical prioritized planning approaches or reactive collision-avoidance algorithms like ORCA [2] fail.

Another decentralized path-planning approach, the "Cooperative Dynamic algorithm" (*CoDy*), is described in [9]. It is based on a broadcast exchange of messages between the participants which is used for a dynamic and coordinated path planning of each vehicle. Dynamic conflicts between the robots are solved by the heuristic adjustment of priority values. The advantages of this algorithm are that it can cope with the coordination of large robot teams and that it is able to avoid and solve dead-lock situations.

Our routing approach for the AGV-based production system is based on an implementation of a context aware route planner from *ter Mors* that uses time slots for calculating conflict free route plans with respect to the existing route plans of other vehicles [8]. For a basic concept description of this algorithm please refer to Sect. 5. The advantages of this approach are that it directly computes the expected arrival time and path dynamics of the vehicles which then are used for the task assignment algorithms.

3 System Overview

This section gives an overview of the decentralized production system. At first, the order structure and secondly the concept of the *MAS* that is used to

control the production system are described. In the automobile production, so called precedence graphs are used to schedule the assembling steps during the whole production process. This graphs further describes the predecessor-successor relationships between the specific steps. Therefore a certain step can only be executed if all its successor processes where completed before. In the project SMART FACE a simplified precedence graph consisting of ten assembling steps was used (see Fig. 2). Hereinafter, assembling steps are also referred as *partial orders*. Within this project, the shown assembling steps can be considered as workstation capabilities. Therefore each workstation is able to carry out a certain number of steps for e.g. *headlights*, *rims* and *seats*. It is possible to distribute the capabilities redundantly over the workstations. This ensures reliability and increases the throughput.
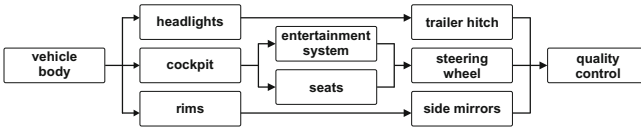

Fig. 2. Precedence graph: the predecessor-successor relationships between the production steps are shown.

3.1   Agent Interaction Concept

In this section, an overview about the system structure and the realized agents is given. All agents are implemented by using the **J**ava **A**gent **D**evelopment Framework (*JADE*). JADE comes along with several interaction protocols, which all correspond to the FIPA standard [1]. All of our agent interactions are based on these protocols. At runtime the agents can be distributed over several independent computers. For better cooperation and coordination between the agent, a layer based approach, which is also presented by *Yu et al.* [13], is chosen. The developed MAS can be divided into three hierarchical layers (see Fig. 3). The top layer is called *Production Planning* and deals with order management and the subordinated planning processes. Furthermore this layer contains a decision point that is part of the **P**roduction **O**rder **M**anagement Agent (POM-Agent). It is used to decide which production order from the order portal will be loaded into the system next. The second layer is called *Production and Warehousing*. This level contains agents that deal with the physical production process in combination with the component supply and delivery management. Furthermore this layer contains two other decision points. One is located in the **P**roduction **O**rder Agent (PO-Agent) and deals with the decision which partial order from the precedence graph is manufactured next. The other decision point is located at the warehouse agent and is responsible for the transport order assignment. The lowest hierarchical layer is called *Material flow*. Basically this layer is responsible for planning, managing and executing the material flow between the warehouse and the workstations. The major element of this layer is the routing agent.
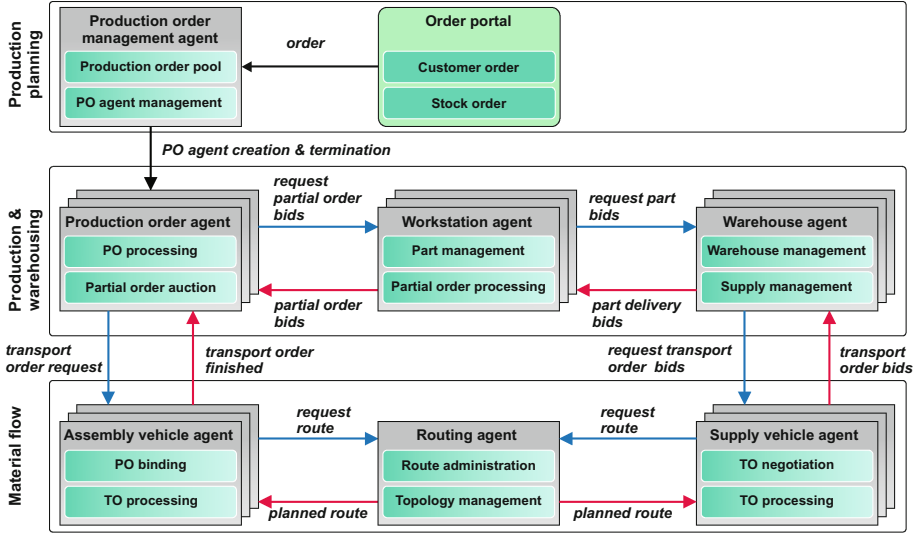
Fig. 3. Schematic system overview: three hierarchical layers with their corresponding tasks and the included agents are shown. Additionally the agent communication within (horizontal) and between the hierarchical layers (vertical) is mapped.

The upcoming conversations during the production process can basically be divided into two interaction protocol specifications. The implemented auctions correspond to the FIPA **C**ontract-**Net** (CNET) specification. All other conversations are conform to the *FIPA request* or *request when* protocol specifications.

4   Agent Model

The following section describes the applied agent model, the individual agents, their tasks and behaviors. Agents are divided into two agent representations, a physical and a logical agent representation. Additionally the agents can be grouped into two different agents classes which follows the definitions of *Russell and Norvig* [10]. The vehicle, warehouse and POM agent corresponds to the *Simple reflex agents*, whereas all other agents corresponds to the *Model-based reflex agents*. All subsequently described agents refer to Fig. 3.

4.1   Logical Agents

A logical agent represents a software component, which reacts to its environmental influences. Based on the perceived influences the agent acts autonomously within its own behaviors and tries to reach its predefined goals. Subsequently the POM-Agent, PO-Agent, warehouse agent and routing agent will be presented.

Production Order Management Agent. The POM-Agent is able to access the set of particular orders from a production order pool. One key task of this agent is to determine which production order has to be started next. In order to make this decision inside the included decision point, the agent is always informed about the current utilization of the whole system. This means that the agent knows the capabilities associated with the occupancy rate of the different workstation agents. Beside the decision-making a further task of this agent is to start specific PO-Agents that represents the chosen production orders and terminate them after finishing assembly.

Production Order Agent. The initial task after a production order agent was started, is to search for an assembly vehicle. According to this, the PO-Agent starts a binding auction. For binding, the production order agent requests bids from all available assembly vehicle agents. The bids contains information such as: battery life, driving distance and travel time to the mounting location of the car body. Based on this information, the production order agent can choose the best assembly vehicle to reach a global target, such as: reducing driving distances or travel times. The main task of a production order agent is to complete all partial orders of a given precedence graph (see Fig. 2). Therefore, all open and feasible partial orders will be selected and then auctioned to the workstations. After receiving the bids for all auctioned partial orders, an evaluation of them will be proceeded within the decision point of this agent. Within the evaluation process, the best bid for a partial order will be evaluated to reach a global assigned target. Global goals are e.g., maximum utilization for special workstations, uniform utilization of all workstations or shortest throughput times of the production orders.

Warehouse Agent. The responsibilities of the warehouse agent includes the management of part supplies, assignment of supply vehicles and management of stock. For stock management the warehouse agent implements an interface to an external warehouse management system, which persists bin and part movements. For the delivery of goods, transport orders are auctioned to supply vehicles (see Sec. 4.2). Based on the bids of the supply vehicles, the selection of an optimal vehicle can be carried out in a local decision point (shortest route for the supply vehicle, earliest arrival time, etc.).

Routing Agent. The major task of this agent is to compute routes that are requested by both supply and assembly vehicle agents. The incoming route requests are processed and answered in a serial way. Therefore this agent offers a routing service which is based on the centralized routing algorithm described in more detail in Sect. 5. Beside the route calculation another key task is the route administration, whereby previous planned routes are saved and retrievable for new route calculations. The basis of the route calculation is a topology which is managed by the routing agent.

4.2 Physical Agents

A physical agent extends a logical agent by a technical component or a connection to an external control unit. Following the vehicle agents and the workstation agent will be introduced. The physical agents can be executed directly on the specific hardware (e.g. a vehicle controller) or on a dedicated PC. In the latter case, communication is realized via WiFi connection.

Vehicle Agent. A vehicle agent is a physical agent which is directly connected to a real robot, in our case a **C**ellular **T**ransport **V**ehicle (CTV) [6]. A CTV is capable of changing its location, carry bins, provide status information for e.g. position and battery information and receive external transport orders. All these functions are bundled in a basic vehicle agent. The basic vehicle agent also contains behaviors to translate calculated routes from a routing agent into driving commands, receive travel destinations from other agents and provide vehicle information to other agents. In our experimental setup two variants of the basic vehicle agent exists, whereby every derived agent supplemented additional behaviors to fulfill the roles of a supply or assembly vehicle.

In our scenario a supply vehicle delivers ordered parts from a highbay warehouse to distributed workstations on the shop floor. Therefor the basic vehicle agent was extended by a behavior to process transport orders. To obtain orders, a supply vehicle sends bids on auctioned transport orders from the warehouse agent. For this purpose, the vehicle agent has an additional behavior to conduct contract negotiations. Auction for orders are given in the form of a direct inquiry of a supply vehicle agent and a request to submit a bid. The bid of a supply vehicle includes information such as: driving distance, travel time, battery life and workload.

An assembly vehicle is used to transport a car body between the workstations on the shop floor. For this purpose a car body is permanently assigned to an assembly vehicle during the entire assembly process, from mounting the car body till completion. Hence, a binding between an assembly vehicle and a production order exist during the overall assembly. After binding the assembly vehicle can directly be instructed by the production order agent, for e.g. sending a transport order request to a workstation. In case of finishing the assembly of the car, the binding will be canceled and the assembly vehicle agent starts bidding on new binding auctions.

Workstation Agent. The workstation agent can be represented by two different physical types of workstations. On the one hand, there are workstations where a robot carries out the assembly of the components. On the other hand, there are workstations where a human worker takes over the assembly (see Fig. 1). To involve workers, an interaction with the system is given via a mobile device, e.g. a tablet, where open tasks are displayed and interaction opportunities are given. An important task of a workstation agent is to manage its own local inventory system. Each station may have buffers for parts whose filling level

is monitored by the specific workstation agent. If the filling level falls below a defined level, a re-order is automatically triggered. To carry out partial orders a workstation agent must be able to create bids for these orders which are auctioned by a PO-Agent. In case that the required component is not available locally it will be ordered via the warehouse agent. All the necessary information like availability or travel time are collected and returned to the specific workstation agent.

5   Routing Algorithm

We consider a set V of vehicles, where each vehicle has to find the shortest route from a start to a destination location, without colliding with any of the other agents, or ending up in a deadlock situation. For this purpose a graph routing algorithm, based on the **C**ontext **A**ware **R**oute **P**lanning *(CARP)* approach of *ter Mors* [8] was implemented. The calculated routes can be defined as $\pi_n = ([r_1, \tau_1], \dots, [r_n, \tau_n])$, $\tau_i = [t_i, t'_i)$ where r_n is a node of the resource graph and τ_n is a time window that indicates that resource r_n is available from time t_i to t'_i. The basic idea of CARP can be defined as follows. Within routing, individual agents plan their routes successively one after the other. If an agent n plans its route, the already planned $(n-1)$ routes are included in the current calculation. Every node of the resource graph has its own time windows that defines when the node is visitable. For planning conflict-free routes, overlapping time windows between the individual nodes of a route are required, in a way that $\tau \cap \tau' \neq \emptyset$. A calculation of the needed time windows takes place on the basis of real time values of the deployed vehicles. These values can be used to estimate the vehicle positions at a given time point. Based on this information the time windows of each node in the graph can be created. The algorithm also offers the possibility to integrate current events, such as delays, into the planning of future routes through a delay propagation approach [8].

6   Decision Making

To make decisions based on different input variables various concepts such as genetic algorithms, fuzzy logic or neuronal networks can be applied. A disadvantage of genetic algorithms and neuronal networks is the high complexity. Decisions are made in a black box and the process of decision-making is not comprehensible. The big advantage by using fuzzy logic is that the solution process can be converted to human understandable operations. *Branisso et al.* evaluates different techniques to increase the performance of an AGV fleet in a warehouse. The best results were obtained by applying fuzzy logic to the decision process [3]. By applying fuzzy logic it is possible to extensively influence the decision making process by changing the fuzzy rules and membership functions which changes the behavior of the whole shop floor. Each of our decision points has corresponding fuzzy input and output sets and specific optimization goals. Subsequently the decision point of the warehouse agent, which is used for the

assignment of transport orders, is examined in more detail. For the included auction in that decision point the FIPA CNET protocol was applied. In [3] it is pointed out that CNET produces a larger task throughput as the First Come First Serve approach. All of the following input values are part of a bid that was sent from a supply vehicle agent to the warehouse agent during a transport order auction. Three physical input values are used at this decision point. More precisely, these are the travel distance between the source and destination location, the travel time and the battery charging state of the vehicle. Based on this three input values, the corresponding fuzzy sets (U_n, μ_n) were formed where U_n is a particular set and $\mu_n : U_n \to [0,1]$ one of the associated membership functions for set U_n. In this particular case the defined output set determines a fitness value for each of the CTV's which has submitted a bid. Subsequently, the warehouse agent notifies the winner CTV (Fig. 4).
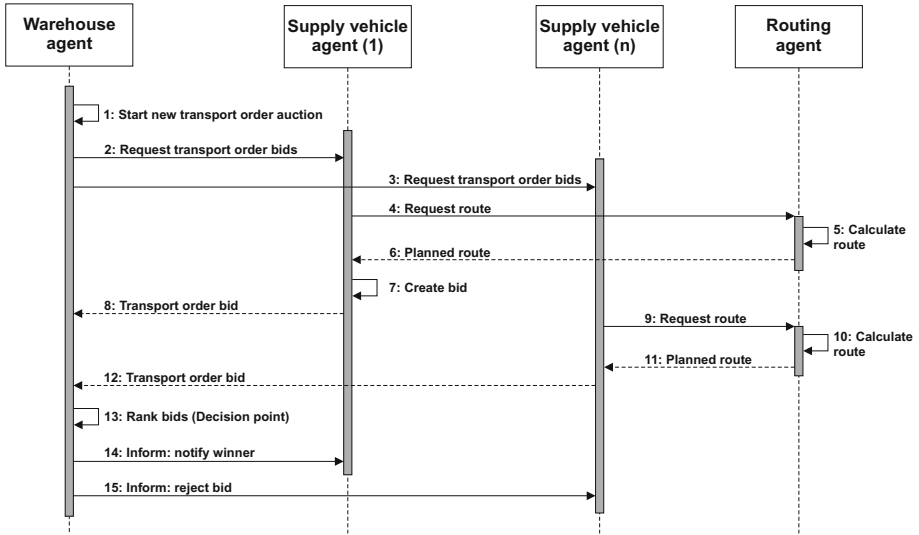

Fig. 4. Sequence diagram of a transport order auction: the participating agents as well as the communication process between them are shown.

7 Evaluation

The evaluation of the concept of the decentralized production system architecture is done by two different approaches. First, a multi-agent system was developed based on the agent model (cf. Sect. 4) that controls the a real-world testbed, called SMART FACE shop floor demonstrator. While running this demonstrator, the behavior of the vehicles can be observed by a visitor. For example, the operability of the decentralized production system can be examined and the

lead times of a real-world implementation can be analyzed. The second evaluation step was to create a simulation model that depict a larger order amount that utilized all work stations for a longer period. In the first version of the simulation only the transportation of car bodies and the assembly steps were taken into account. The objectives of the experiments were to determine the makespan, station utilization and the waiting times of the assembly vehicles in dependence of the number of available assembly vehicles.

7.1   Live System

In order to show that the developed agent model is applicable to a real-world scenario, the SMART FACE shop floor demonstrator has been created at Fraunhofer IML. This testbed consists of 4 workstations (3 are human-operated and one is an automated workstation operated by an industrial robot), 4 assembly vehicles and 6 supply vehicles and a highbay warehouse that is accessible by the supply vehicles. The system is capable of producing more than 5000 variants of car models that consist of a car body and different 3D-printed car components. These components are then assembled to the car body at the 4 workstations sequentially whereas the assembly order is defined by the precedence graph (cf. Fig. 2). This cyber-physical production system has shown to be capable of producing up to 6 car models per hour, depending on the speed of the human operators working at the human operated workstations. See following URL for shop floor demonstrator video: https://download.iml.fraunhofer.de/paams17/.

7.2   Simulation Model

For the simulation model an event driven simulation was created. The basic concept was to model the process for products to be produced that can be subdivided in subprocesses and rules in which order they can go through. To execute a process, resources like workstations or assembly vehicles can be requested at resource pools. Workstations are stationary resources with different skills that are needed in processes. Assembly vehicles are mobile resources that can move car bodies. The topology of the shop floor is modeled by discrete locations for workstations and a distance matrix. In addition, every workstation can have a buffer place for assembly vehicles with a adjustable size. With the building blocks described above a simulation model was builded as a reproduction of the live system. This model enables to carry out experiments with larger quantity of orders and with a flexible amount of assembly vehicles in shorter times.

7.3   Simulation Results

In first simulation runs, the simulation results show that the production processes are always executed in a valid order regarding the precedence graph. Because the selection algorithm for the next process to execute takes the distances to

possible execution locations into account, the number of necessary transports was reduced. In addition to the proof of concept of the decentralized control system, some experiments to balance the assembly system were carried out. For example in a series of experiments the number of assembly vehicles was considered. Figure 5 shows a summary of these experiments. On the x-Axis, the number of assembly vehicles in an experiment is shown. For each experiment, the blue line shows the average station utilization while the orange bars represent the sum waiting time for the assembly vehicles. The green bar shows the total make span of all orders. In the experiment with four assembly vehicles, the utilization of the workstations is very low (about 60%) while the makespan is high (green bar). As the red bar shows, the assembly vehicles have a low waiting time when they are approaching a workstation. If more vehicles are in use, the station utilization increase fast to over 90% while the makespan decreases. But if the number of vehicles is increased further, the waiting times of them increases disproportionately to the decrease of the makespan time.
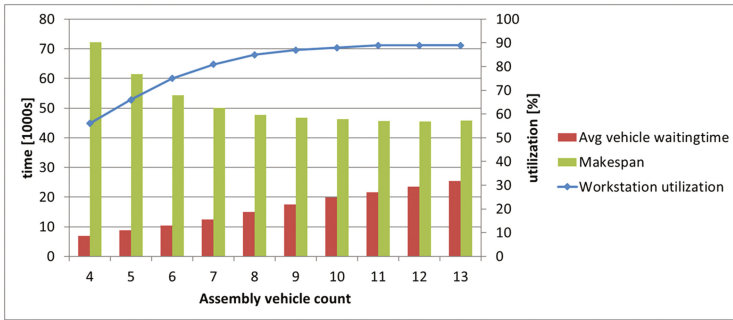

Fig. 5. Simulation results: The station utilization and the relative change of makespan and waiting time at simulations with different numbers of assembly vehicles are shown. (Color figure online)

8   Conclusion and Outlook

Within this paper a new decentralized, agent-based concept for production planning and control has been introduced. The system architecture is defined by an agent model which contains the elements of a modern production system (namely an ERP system, a Warehouse Management System and a routing and task assignment software for AGVs). A first evaluation within a simplified simulation model and results from a testbed with a simplified production process and real assembly vehicles have been proven to be successful. In the future, the time window principle of the routing algorithm might be used to extend it with respect to the integration and prediction of human behavior within intralogistics environments. Additionally, the temporal influence of the human-operated system parts on the decision making and on makespan and waiting times of the vehicles within the live system will be investigated. Another future task is

to extend the simulation model. Especially the part supply is a very important part for the assembly that must be integrated.

References

1. Bellifemine, F.L., Caire, G., Greenwood, D.: Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology). Wiley, New York (2007)
2. van den Berg, J., Lin, M., Manocha, D.: Reciprocal velocity obstacles for real-time multi-agent navigation. In: IEEE International Conference on Robotics and Automation, ICRA 2008, pp. 1928–1935, May 2008
3. Branisso, L.B., Kato, E.R.R., Pedrino, E.C., Morandin, O., Tsunaki, R.H.: A multi-agent system using fuzzy logic to increase agv fleet performance in warehouses. In: 2013 III Brazilian Symposium on Computing Systems Engineering, pp. 137–142 (2013)
4. Frey, D., Woelk, P.O., Stockheim, T., Zimmermann, R.: Integrated multi-agent-based supply chain management. In: WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp. 24–29, June 2003
5. Giglio, D.: Task scheduling for multiple forklift AGVs in distribution warehouses. In: Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA), pp. 1–6, September 2014
6. Kamagaew, A., Stenzel, J., Nettstraeter, A., ten Hompel, M.: Concept of cellular transport systems in facility logistics. In: 2011 5th International Conference on Automation, Robotics and Applications (ICARA), pp. 40–45, December 2011
7. Koebler, J.: Paula on tour (2017). https://audi-illustrated.com/en/audi-encounter-01-2017/paula-on-tour
8. ter Mors, A.: Conflict-free route planning in dynamic environments. In: International Conference on Intelligent Robots and Systems 2011, pp. 2166–2171 (2011)
9. Regele, R., Levi, P.: Cooperative multi-robot path planning by heuristic priority adjustment. In: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5954–5959, October 2006
10. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach, Global Edition. Always Learning. Pearson Education, Limited, Harlow (2003)
11. Schwarz, C., Schachmanow, J., Sauer, J., Overmeyer, L., Ullmann, G.: Self guided vehicle systems. Logistics J. **2013**(12) (2013)
12. Čáp, M., Novák, P., Kleiner, A., Seleck, M.: Prioritized planning algorithms for trajectory coordination of multiple mobile robots. IEEE Trans. Autom. Sci. Eng. **12**(3), 835–849 (2015)
13. Yu, M., Zhang, W., Klemm, P.: Multi-agent based reconfigurable manufacturing execution system. In: 2007 IEEE International Conference on Industrial Engineering and Engineering Management, pp. 718–722, December 2007