



Multi-agent path planning: comparison of different behaviors in the case of collisions

Michaela Krä¹, Ludwig Vogt¹, Valentin Spannagl¹ and Johannes Schilp^{1,2}

¹ Chair of Digital Manufacturing, Faculty of Applied Computer Science, Augsburg University, Eichleitnerstr. 30, 86159 Augsburg, Germany

² Fraunhofer Research Institution for Casting, Composite and Processing Technology – IGCV, Provinostr. 52, 86153 Augsburg, Germany

Abstract.

In the context of Industry 4.0 and cyber-physical production systems, the role of production logistics is perceived as more and more important in order to reach the overall manufacturing targets. One central aspect in organizing the flow of material consists in task allocation and path planning for transport resources disposing of growing autonomy. There are various approaches for multi-agent path planning as well as the way of dealing with collisions. Collisions are possible due to traffic volume and can either be treated on planning level or in a short-term way on control level.

The paper presents existing strategies for path finding before giving an overview of methods to deal with autonomous transport resources that meet in a manufacturing environment. Then, different existing behaviors and reactions in the case of collision detection based on several criteria are compared. This step allows classifying the strategies depending on the manufacturing environment and its organization.

Keywords: production logistics, autonomous transport resources, collision avoidance.

1 Introduction

With the introduction of the concepts of Industry 4.0 like digitization and decentralization, various challenges arise in production environments. In addition, customer demand regarding increasing number of product variants as well as shorter product life cycles result in smaller lot sizes. [1, 2] Production logistics play an important role in the manufacturing process and companies understand more and more its influence on the throughput time [3].

The utilization of autonomous resources is supposed to turn logistics more efficient due to their possibility to organize themselves on a decentral level. This means that the operational level obtains decision-making autonomy within defined constraints. A challenge in this context is the balance between taking advantage of the autonomy and reaching central planning targets. One central aspect is the interaction of various resources with possibly different skill sets. [4–6]

Mobile autonomous resources, i.e. self-directed vehicles also called autonomous guided vehicles (AGV), take center stage in current research. There exist different concepts for multi-agent path finding (MAPF) presenting each advantages and disadvantages. [7, 8] Considering the range of concepts for MAPF and especially collision avoidance an evaluation of these approaches comparing their applicability in a specific manufacturing environment is missing. An approach is needed in order to identify a suitable strategy or set of strategies for avoiding collisions in production environments.

The scope of this paper is to give an overview of existing approaches for dealing with and avoiding collisions of autonomous transport resources. In addition, these ways are compared based on a choice of valuable criteria.

2 State of the art

Relevant literature concerning production logistics, autonomous resources and MAPF is summed up in this section to present applicable concepts. Before taking a deeper look on path planning and collision avoidance, a short introduction to logistics and mobile resources in general is given.

2.1 Logistics and mobile resources

As described in the introduction, AGVs are used for a variety of single and multi-agent applications but the main sector is logistic systems. The most common logistic scenarios are the transportation of dock containers in a terminal [9–11] and transportation services in warehouses as well as production lines [12]. While in the past, AGVs were only used for transportation services recent projects have also focused on expanding their skill set. Hoff and Sarker [13] have distinguished between six different major types of AGVs that include unit load, towing, parallel truck, forklift, light load and assembly line vehicles. The newest addition to these classic AGV systems have been drones which are tested in warehouses [14]. The vehicle design and fleet combination is just one of many central research topics in AGV Systems. Others involve the sensors, navigation, localization, vehicle control and the communication between the agents [15].

2.2 Path planning for multiple resources

The classical MAPF problem for a set of n agents receives an input tuple (G, s, t) where $G = (V, E)$ is an undirected graph with V vertices denoting locations and E edges. $s : [1, \dots, n] - V$ maps an agent to a start vertex, and $t : [1, \dots, k] - V$ maps an agent to a target vertex. Often the graph is given with its corresponding adjacency matrix that displays the connected vertices and the weight of the edges (c.f. figure 1). For the solution of this problem, one can find methods, which take different constraints into consideration.

The Dijkstra's algorithm is one of the most commonly known algorithms for the shortest path search [16]. It calculates the shortest path between two vertices in an undirected grid.

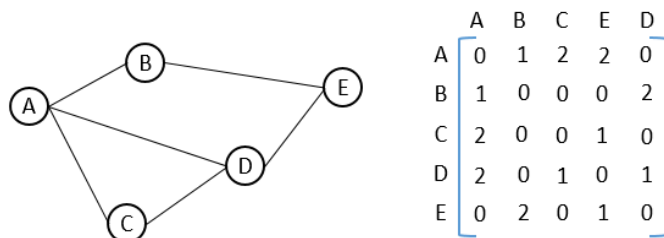


Fig. 1. Representation of an undirected graph with its adjacency matrix.

An evolution of Dijkstra's algorithm is the A*Algorithm, which estimates the distance, i.e. costs, from the start vertex to the target vertex to preselect appropriate vertices with a heuristic [17]. Warren [18] proposes a modified A*Algorithm which is faster than the normal A*Algorithm due to a further reduced solution quantity.. Both the Dijkstra and A*Algorithm are not flexible and cannot react to a changing environment [19]. Therefore the D*Algorithm (Dynamic A*) was developed by Stentz [19] as an extension of the A*Algorithm. Koenig and Likhachev [20] propose the D*Lite Algorithm which combines the heuristic approach of the A*Algorithm with the incremental one of the D*Algorithm. Other extensions of the D*Algorithm are the Focussed D* and Field D*Algorithm. The Focussed D*Algorithm focusses on the repair of the cost map, thus reducing the computational costs [21]. While all of these approaches are limited by a discrete set of possible transitions in the grid the Field D*Algorithm is interpolation based and calculates smooth paths for the agents [22].

These algorithms were developed for single agent path planning but can also be used for MAPF problems. The M*Algorithm is specified for MAPF problems and minimizes the global cumulative cost function of all agents [23]. An extension of the M*Algorithm is the M*UM*Algorithm which considers uncertainty in the position and pose of the agent [24, 25]. Nebel et al. [26] propose an M*Algorithm for uncertainty in the destination of the agents path while Ma et al. [27] developed a MAPF algorithm with delay probabilities.

2.3 Collision detection and avoidance

One of the main topics within the MAPF problem is the collision detection and avoidance between the agents. There are two main approaches for the collision detection: one is a centralized and the other one a decentralized. For the centralized approach the information about the actual position, path and speed must be collected in a central location [28]. Felner et al. [29] are adding a heuristic to the Conflict-Based Search (CBS) by aggregating collisions among the agents. While this approach uses discretized time steps and assumes that all actions have the same time duration Andreychuk et al. [30] propose a Continuous Time Conflict-Based Search (CCBS).

The decentralized approach enables the agents to detect collisions by themselves and react accordingly. Wei et al. [31] present an approach where only the affected agents need to communicate with each other in case of a collision. Trodden and Richards [32] use a Model Predictive Control (MPC) where only one agent replans, while all others

continue with their path. The agent that replans its path is identified via a bidding system, where the highest bidder receives the replanning token in each iteration. Desaraju and How [33] use a two-step approach where the first step is the individual component and the second step is the interaction component. While in the first step each agent minimizes the cost of its own paths, the interaction component minimizes the total cost across all agents.

If a collision is detected, researchers propose a variety of strategies for the behavior of the involved resources. Liu et al. [34] plan paths for a multi-agent pickup and delivery problem where no paths cross each other during the duration of the exercised task. A similar approach is used by Kiarostami et al. [35] where a Monte-Carlo Tree Search is used to find non-overlapping paths.

Sun et al. [36] developed a decoupled approach where every agent's path is planned with an A*Algorithm. Every agent continuously checks its front area and if another agent is detected in a critical region, a behavior-based collision avoidance strategy is executed. The behaviors are specialized for an intralogistic problem description and selected by specified traffic rules. Another decentralized approach where possible collisions are detected by the agent itself is proposed by Gochev et al [37]. Instead of a rule based collision avoidance the agents switch in a collision avoidance mode. In this mode the agents create a roundabout along the original path which is faster than stepping back and changing directions. Chang et al. [38] are using a detection shell around each agent and according to the nearest object inside this detection shell breaking and gyroscopic forces are calculated.

A five layer neural network with three hidden layers which is able to generate crossing paths and the associated velocity vectors of the agents so that no collision occurs is presented by Chen et al [39]. This approach is based on the optimal reciprocal collision avoidance (ORCA) which is often used for the collision avoidance in free navigating agents [40].

3 Concept

Based on the approaches presented in chapter 2, criteria to be considered in manufacturing systems are identified before giving a comparison of the concepts for collision avoidance. For the use of autonomous transport resources in real manufacturing environments, this is a necessary step in order to choose appropriate strategies depending on individual production constraints.

3.1 Criteria to compare collision avoidance strategies

To evaluate the different collision avoidance strategies it is necessary to define the environmental conditions. The criteria to compare the different collision avoidance strategies are not to be confused with the environmental settings. Since there exists no past work on the ranking or comparison of these influences for the different algorithms we have identified the following environmental influences which were the most common research topics in chapter 2.2:

- **Order Duration:** One of the most researched and discussed problem for the MAPF sector is the uncertainty in the order duration. It is not feasible to assume that the supposed order duration is equal to the real executed order duration because of delays that can occur. Obstructions can occur because of obstacles in the path or delays at the charging and discharging processes, which lead to different execution times. The uncertainty is not only limited to the duration of a delay but also on the occurrence of a delay during the order.
- **Time Constraints:** After the planning of each order, every task has a starting time and a deadline when it should be finished. For the AGVs, this means that it must reach its target vertex prior its planned deadline. Since there exists uncertainty in the delay of orders, the planned time for the task of the AGV must have a buffer.
- **Prioritization:** Most Companies prioritize their orders because of customer relationships, demands or order volume. For this, they use different principles like First in First out, Least Production or scalar as weights to classify each order.

All of these aspects have to be considered not in an isolated test area, but in real manufacturing conditions, the previously mentioned environmental settings. These comprise the next parameters based on Adam [41]:

- Product information including volume and lot sizes
- Production organization including layout, workflow and in-house production depth
- Organization of planning and control processes
- Organization of logistic processes
- Presence of workers including their skills and schedules

3.2 Comparison of different collision avoidance strategies

There exist many different approaches for AGVs to behave in the case a collision is detected. We have identified five basic strategies from the research done in chapter 2.3 and the collision avoidance behavior presented by Sun et al. [36]. Though they explain eight different behaviors, all of them can be reduced to two basic strategies. That is why we adapted the avoiding and waiting strategies of Sun et al. plus the variation of the start time, no crossing paths and a change of velocities for our evaluation. All strategies are further described in the following segment. To avoid repetitions it is assumed that the paths of the agents cross at vertex n , the velocities of the agents are v and the distance from the start vertex to vertex n is d .

1. **Path Crossing:** Probably the oldest strategy in case a collision is detected consists in finding another path, which traverses no previously planned path. This approach seems only suitable if there is a grid big enough to reach the target through different sets of vertices and edges.
2. **Priority:** The agents take over the different priorities from the orders and if they collide at vertex n the higher prioritized agent passes the vertex n first while the other agent waits at the previous vertex $n-1$. In this scenario, it is possible that the waiting agent produces a deadlock or has to wait for a long time because of higher prioritized agents. The approach is shown in figure 2.

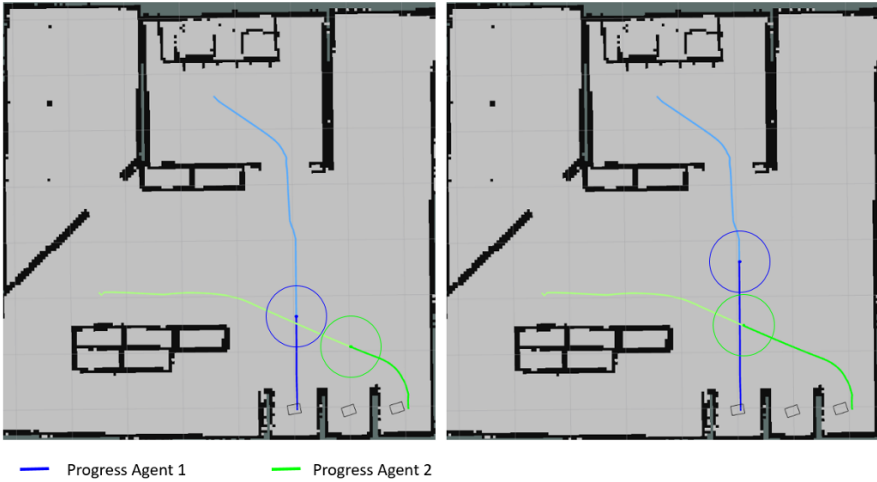


Fig. 2. Two crossing paths where the agents have different priorities. agent 1 passes vertex n before agent 2 because of a higher priority.

3. Start time: Another possible strategy is to vary the start time of the task to avoid a detected collision. For each agent's path the speed v and distance d is known, so it is possible to calculate the time when the agent reaches vertex n . With this information, the starting time of the specific order can be varied so the agent passes vertex n before or after the other agent reaches vertex n .
4. Velocities: Based on the computed time to reach vertex n the velocity v can be varied, so that the AGV passes the vertex n earlier or later to avoid a collision. This approach can be applied in two versions. The velocity of the AGV can be set at a fixed value at the start of each order or the velocity can be changed temporarily during travelling the path.
5. Collision avoidance mode: If one agent detects another agent or obstacle on its path, one or both agents can switch into the collision avoidance mode. In this mode, the agents pass the obstacle on their own without the need to replan. For this, all agents must be equipped with the necessary sensors to do such a maneuver.

4 Application and evaluation settings

The environment for the test cases is a ROS based path planner and its simulation environment. A test scenario is set up to evaluate different criteria for the collision avoidance strategies presented in chapter 3.2. The parameters mentioned in chapter 3.1 are meant to be analyzed and structured regarding their influence on the choice of suitable collision avoidance strategies. They make the simulation better comparable to real life applications but also computational expensive.

Workers are seen as obstacles and these will cross and obstruct the planned paths at a certain percentage. The duration of the obstruction is determined with a Gaussian normal distribution. In addition, we assume that the workstations have different states,

so the vertices of the grid are changing. Vertices will have the capacity one, so only one agent can occupy each vertex at a specific time step. The edges also have a capacity of one. Despite this, it is possible that multiple agents are in the same cross or pass each other in a lane because the ROS path planner builds its own grid on the prescribed map.

To evaluate and compare the different strategies we analyze a set of key performance indicators in the simulation. Since most researchers try to minimize the sum of costs of all agents this is one of the identified parameters. By analyzing additional parameters, the influences of the different collision avoidance strategies can be shown. Furthermore, we analyze the traffic which is generated from the different strategies and also the grid capacity utilization. With these parameters, we can make a first assessment of how the applied strategies influence workers or how easily additional orders can be implemented in the plan:

- Sum of all costs: $\sum_{i=1}^I q_i$ where q_i is the path cost of agent i and I is the quantity of all planned agents. The cost of each path is equal to its length.
- Grid capacity utilization: Ratio between all the occupied vertices and edges to all available vertices and edges in the grid. This value shows how much space of the grid is occupied by agents or the agents' path with the different strategies.
- Traffic: Ratio between the actual duration of an order and the planned time of an order. With a higher ratio, more interruptions occur which equals more traffic.

The first tests in a simulation environment compare the average throughput time of four AGVs with each executing orders of two or three jobs for the strategies priority and path crossing (cf. chapter 3.2). The results for the mean value of the order duration show that avoiding collisions by circumnavigating already planned paths leads, as expected, to 10 up to 30 % longer throughput times. It is evident that these strategies depend highly on the considered manufacturing layout and the existing traffic. This needs to be regarded in a more detailed way.

5 Conclusion

Current challenges linked to the concepts of Industry 4.0 affect in particular production logistics. This comprises among others the need of more flexible solutions for material transport in manufacturing environments. Thus, the use of autonomous resources increases as these means of transport present advantages compared to standard logistic solutions and as they allow relocating decisions on resource level.

This contribution presents an approach to compare different ways of dealing with collisions between AGVs based on valuable criteria. The treated types of collision avoidance were path crossing, priorities, start time, velocities and internal collision avoidance mode. Relevant criteria to evaluate these strategies are order duration, time constraints and prioritization.

The next step will be a more extensive evaluation, validation and verification of the concept presented in chapter 3 within a simulation scenario. These analyses will aim at

verifying the applicability of the stated criteria and complete them if necessary. Additional varying speeds of the vehicles on their way through the production environment have to be implemented to make the application more realistic.

Acknowledgements.

The presented results have been developed within the research project FORobotics (www.forobotics.de). The Bavarian Research Foundation funds this project.

References

1. Kolberg D, Berger C, Pirvu B-C et al. (2016) CyProF - Insights from a Framework for Designing Cyber-Physical Systems in Production Environments. *Procedia CIRP* 57: 32–37
2. Reinhart G (ed) (2017) *Handbuch Industrie 4.0: Geschäftsmodelle, Prozesse, Technik*. Hanser, München
3. Riechmann C (2013) *Reorganisation im Logistikbereich zur Verringerung der Durchlaufzeiten und Lagerbestände: Am Beispiel des Logistikzentrums des Automobilzulieferers HÜCO electronic GmbH*. Wirtschaft. Diplomarbeiten Agentur, Hamburg
4. Delfmann W, ten Hompel M, Kersten W et al. (2018) *Logistics as a science: Central research questions in the era of the fourth industrial revolution: Logistics Research*, Vol. 11, Bremen
5. Wiendahl H-P, Reichardt J, Nyhuis P (2014) *Handbuch Fabrikplanung: Konzept, Gestaltung und Umsetzung wandlungsfähiger Produktionsstätten*, 2., überarb. und erw. Aufl., [elektronische Ressource]. Hanser, München
6. Kersten W, Seiter M, See Bv et al. (2017) *Chancen der digitalen Transformation: Trends und Strategien in Logistik und Supply Chain Management*. DVV Media Group GmbH, Hamburg
7. Günther WA (2012) *Algorithmen und Kommunikationssysteme für die Zelluläre Fördertechnik*. Forschungsbericht / Fml, Lehrstuhl für Fördertechnik Materialfluß Logistik, Technische Universität München. Lehrstuhl für Fördertechnik Materialfluß Logistik (fml) TU München, Garching b. München
8. Pfeifer H (1989) *Grundlagen der Fördertechnik*, 5., verbesserte Auflage. Vieweg+Teubner Verlag, Wiesbaden
9. Li Q, Adriaansen AC, Udding JT et al. (2011) *Design and Control of Automated Guided Vehicle Systems: A Case Study*. In: *Proceedings of the 18th World Congress*, vol 44, pp 13852–13857
10. Henesey L, Aslam K, Khurum M (2015) *Task Coordination of Automated Guided Vehicles in a Container Terminal*
11. Liu C-I, Jula H, Vukadinovic K et al. (2004) *Automated guided vehicle system for two container yard layouts*. *Transportation Research Part C: Emerging Technologies* 12(5): 349–368. doi: 10.1016/j.trc.2004.07.014
12. Draganjac I, Miklic D, Kovacic Z et al. (2016) *Decentralized Control of Multi-AGV Systems in Autonomous Warehousing Applications*. *IEEE Transactions*

- on Automation Science and Engineering 13(4): 1433–1447. doi: 10.1109/TASE.2016.2603781
13. Bordelon Hoff E, Sarker BR (1998) An overview of path design and dispatching methods for automated guided vehicles. *Integrated Manufacturing Systems*: 296–307
 14. (2019) Drone Automation for Warehouse 4.0: White Paper
 15. Schulze L, Behling S, Buhrs S (2008) Automated Guided Vehicle Systems: a Driver for Increased Business Performance. In: *International MultiConference of Engineers and Computer Scientists, IMECS 2008*: 19 - 21 March, 2008, Regal Knowloon Hotel, Knowloon, Hong Kong. IAENG, Hong Kong
 16. Dijkstra EW (1959) A note on two problems in connexion with graphs. *Numerische Mathematik* 1(1): 269–271. doi: 10.1007/BF01386390
 17. Hart PE, Nilsson NJ, Raphael B (1968) A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions of Systems Science and Cybernetics* 2
 18. Warren CW (1993) Fast Path Planning Using Modified A* Method. *IEEE*: 662–667
 19. Stentz A(T) (1994) *The D* Algorithm for Real-Time Planning of Optimal Traverses*, Pittsburgh, PA
 20. Koenig S, Likhachev M (2002) *D* Lite*. AAAI
 21. Stentz A(T) (1995) *The Focussed D* Algorithm for Real-Time Replanning*. IJCAI proceedings
 22. Fergusin D, Stentz A(T) (2005) *The Field D* Algorithm for Improved Path Planning and Replanning in Uniform and Non-Uniform Cost Environments*, Pittsburgh, PA
 23. Wagner G, Choset H (2011) *M*: A Complete Multirobot Path Planning Algorithm with Performance Bounds*. In: Amato NM (ed) *RSJ International Conference on Intelligent Robots and Systems*: 25 - 30 Sept. 2011, San Francisco, USA. IEEE, Piscataway, NJ, pp 3260–3267
 24. Gonzalez JP, Stentz A(T) (2005) *Planning with uncertainty in position: an optimal and efficient planner*. In: *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp 2435–2442
 25. Wagner G, Choset H (2017) *Path Planning for Multiple Agents under Uncertainty*. In: *Proceedings of the 27th International Conference on Automated Planning and Scheduling*, pp 577–585
 26. Nebel B, Bolander T, Engesser T et al. (2019) *Implicitly Coordinated Multi-Agent Path Finding under Destination Uncertainty*. *jair* 64: 497–527. doi: 10.1613/jair.1.11376
 27. Ma H, Kumar TKS, Koenig S (2017) *Multi-Agent Path Finding with Delay Probabilities*. In: *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, pp 3605–3612
 28. Wang K-HC (2009) *Bridging the Gap Between Centralised and Decentralised Multi-Agent Pathfinding*. AAAI

29. Felner A, Li J, Boyarski E et al. (2018) Adding Heuristics to Conflict-Based Search for Multi-Agent Path Finding. In: Proceedings of the 28th International Conference on Automated Planning and Scheduling, pp 83–87
30. Andreychuk A, Yakovlev K, Atzmon D et al. (2019) Multi-Agent Pathfinding with Continuous Time. IJCAI proceedings
31. Wei C, Hindriks KV, Jonker CM (2014) Multi-robot Cooperative Pathfinding: A Decentralized Approach. In: Ali M, Pan J-S, Chen S-M et al. (eds) Modern Advances in Applied Intelligence: Proceedings, Part I. Springer International Publishing, Cham Heidelberg New York Dordrecht London, pp 21–31
32. Trodden P, Richards A (2006) Robust Distributed Model Predictive Control using Tubes. In: Proceedings of the 2006 American Control Conference. IEEE, pp 2034–2039
33. Desaraju VR, How JP (2011) Decentralized path planning for multi-agent teams in complex environments using rapidly-exploring random trees. In: 2011 IEEE International Conference on Robotics and Automation. IEEE, pp 4956–4961
34. Liu M, Ma H, Li J et al. (2019) Task and Path Planning for Multi-Agent Pickup and Delivery. In: Proceedings of the 18th International Conference on Autonomous and Multiagent Systems 2019, pp 1152–1160
35. Kiarostami MS, Daneshvaramoli MR, Monfared SK et al. (2019) Multi-Agent non-Overlapping Pathfinding with Monte-Carlo Tree Search. In: IEEE Conference on Games 2019: London, United Kingdom, 20-23 August 2019. IEEE], [Piscataway, New Jersey]
36. Sun D, Kleiner A, Nebel B (2014) Behavior-based Multi-Robot Collision Avoidance. In: 2014 IEEE International Conference on Robotics & Automation. IEEE, pp 1668–1673
37. Gochev I, Nadzinski G, Stankovski M (2017) Path Planning and Collision Avoidance Regime for a Multi-Agent System in Industrial Robotics. In: Scientific technical Union of Mechanical Engineering "Industry 4.0" (ed) Machines. Technologies. Materials.: International journal for science, technics and innovations for the industry, pp 519–522
38. Chang DE, Shadden SC, Marsden JE et al. (2003) Collision avoidance for multiple agent systems. In: 42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475). IEEE, pp 539–543
39. Chen YF, Liu M, Everett M et al. (2016) Decentralized Non-communicating Multiagent Collision Avoidance with Deep Reinforcement Learning
40. van den Berg J, Lin M, Manocha D (2008) Reciprocal Velocity Obstacles for Real-Time Multi-Agent Navigation. In: IEEE International Conference on Robotics and Automation, 2008: ICRA 2008 ; 19 - 23 May 2008, [Pasadena, CA]. IEEE, Piscataway, NJ, pp 1928–1935
41. Adam D (1998) Produktions-Management. Gabler Verlag, Wiesbaden

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

