EFFICIENT TASK PRIORITISATION FOR AUTONOMOUS TRANSPORT SYSTEMS

Maximilian Selmair
Vincent Pankratz
BMW Group
80788 Munich, Germany
Email: maximilian.selmair@bmw.de

Klaus-Jürgen Meier
University of Applied Sciences Munich
80335 Munich, Germany

KEYWORDS

Automated Guided Vehicle; autonomous Automated Guided Vehicle, Self-Driving Vehicle; Task Allocation; Pick-up and Delivery; Prioritisation

ABSTRACT

The efficient distribution of scarce resources has been a challenge in many different fields of research. This paper focuses on the area of operations research, more specifically, Automated Guided Vehicles intended for pick-up and delivery tasks. In time delivery in general and flexibility in particular are important KPIs for such systems. In order to meet in time requirements and maximising flexibility, three prioritisation methods embedded in a task allocation system for autonomous transport vehicles are introduced. A case study within the BMW Group aims to evaluate all three methods by means of simulation. The simulation results have revealed differences between the three methods regarding the quality of their solutions as well as their calculation performance. Here, the *Flexible Prioritisation Window* was found to be superior.

LIST OF ABBREVIATIONS

AGV Automated Guided Vehicle
FMS Flexible Manufacturing System
GAP Generalised Assignment Problem
HM Hungarian Method
ILP Integer Linear Programming
KPI Key Performance Indicator
NVA-share non-value-adding share
POI Point of Interest
SDV Self-driving Vehicle
STR Smart Transport Robot
VAM Vogel's Approximation Method
VAM-nq Vogel's Approximation Method for non-quadratic Matrices
aAGV Autonomous Automated Guided Vehicle

Communications of the ECMS, Volume 34, Issue 1,
Proceedings, ©ECMS Mike Steglich, Christian Mueller,
Gaby Neumann, Mathias Walther (Editors)
ISBN: 978-3-937436-68-5/978-3-937436-69-2(CD) ISSN 2522-2414

INTRODUCTION

Constantly changing external conditions associated with global competition, individualised customer demands and more complex production structures force companies to reorganise their production systems to become more flexible (Braunisch 2015). Amongst others, Flexible Manufacturing Systems (FMSs) rely on so-called Autonomous Automated Guided Vehicles (aAGVs), which are able to find the fastest way from a source to a sink by themselves and drive around obstacles, something that Automated Guided Vehicle (AGV) are unable to achieve. The function of the superordinate fleet management system is to allocate the transportation tasks to agents in form of vehicles (Wagner 2018). According to Le-Anh and Koster (2006), there are three different main criteria that ought to be considered when allocating tasks: time, utilisation and distance. The dispatching can be either static or dynamic. When operating with static dispatching, a task allocation decision is made and carried out without considering system changes that occur during the execution of said task (Gudehus 2012). These system changes (for instance task urgency, deadline changes, machine malfunctions) can make the planned sequence inoperative (Ouelhadj and Petrovic 2009). Dynamic dispatching, in contrast, adapts the planned sequence of the transportation tasks to changes either in certain time intervals or when a certain event happens. This increases the flexibility of the task allocation process (Gudehus 2012). After a task is allocated to a vehicle, the vehicle drives to the source of the task and collects the material. Subsequently, it moves to the source and delivers the container at the designated place (sink). During this process, delays can occur due to pedestrians, other vehicles or obstacles that have to be circumvented (Clausen et al. 2013).

RELATED LITERATURE

The issue of transportation is an extensively studied topic in operational research (Díaz-Parra et al. 2014).

The methods for solving the assignment problem aim to minimise the total transportation costs while bringing goods from several supply points (e. g. warehouses) to demand locations (e. g. customers). In general, each point of departure features a prearranged amount of goods that can be distributed. Correspondingly, every destination requires a certain amount of units (Shore 1970). The underlying use case, where tasks have to be assigned to Self-driving Vehicles (SDVs), differs in some regards from the classical transportation problem. In our case, each vehicle has a capacity restriction of one, i. e. a maximum of one load carrier can be transported at a time. Furthermore, each task corresponds to a demand of one. This means that every task can only be allocated to one single vehicle. In the research domain, the just mentioned case is known as the Generalised Assignment Problem (GAP). In an effort to minimise it, researchers aim for minimal costs Z between n tasks and m agents while each task is assigned to one agent (Kundakcioglu and Alizamir 2009). According to Srinivasan and Thompson (1973) the formulation of the GAP is:

$$minimize\ Z = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} \quad (1)$$

In subject to the constraints:

$$\sum_{j \in J} r_{ij} x_{ij} \leq b_i, \text{ for } i \in I \quad (2)$$

$$\sum_{i \in I} x_{ij} = 1, \text{ for } j \in J \quad (3)$$

$$x_{ij} \in \{0, 1\}, \text{ for } i \in I \text{ and } j \in J \quad (4)$$

A set of agents $I = \{1, 2, \dots, i, \dots\}$ has to be assigned to a set of tasks $J = \{1, 2, \dots, j, \dots\}$. Variable c is representing the costs accrued when the task j is allocated to the agent i. r_{ij} represents the capacity which is needed when the task j is allocated to the agent i. Variable b represents the available capacity of agent i in total. The binary variable x_{ij} equals 1 if task j is assigned to agent i, otherwise it remains 0.

There are several methods that solve the GAP. Beside optimisation methods like Integer Linear Programming (ILP), there are, for example, algorithms like the Hungarian Method (HM) proposed by Kuhn (1955), Vogel's Approximation Method (VAM) proposed by Reinfeld and Vogel (1958) or Vogel's Approximation Method for non-quadratic Matrices (VAM-nq) proposed by Selmair et al. (2019). The last method was used in the simulation study of this paper due to its superior calculation performance and satisfactory results in terms of non-quadratic matrices.

DEVELOPED PRIORITISATION METHODS

Solving the GAP minimises the transport costs, in this case operationalised as meters driven, between tasks and agents. For the presented use case, transport costs can be equated to the transport distances or – for prioritised orders – the transportation duration. Minimising transport distances is an important goal as it leads to faster task processing and less traffic on the routes. As the transport route between source and sink is the same for every vehicle, in this scenario only the empty drive distance from the current location of each vehicle to the source of a task is minimised.

If there are consistently more tasks than free vehicles in a system, waiting times can occur for tasks that could not be allocated in former allocation cycles due to high transport costs or long pick-up distances. These waiting times can cause the performance of the overall system. In automotive intralogistics, which oversees the supply of parts to assembly lines, delays can cause high costs and must be reduced as much as possible. Consequently, the following prioritisation methods were developed to unite efficiency and in time delivery in the task allocation of autonomous transport systems.

Fix Prioritisation Window

The first prioritisation method presented here is the *Fix Prioritisation Window*. This method was inspired by the backwards calculation in dynamic task dispatching, which can be found in the article contributed by Gudehus (2012). The idea behind this method is to force the allocation of tasks that are at risk of becoming delayed. When a task exceeds the point of time where a in time delivery is critical, it is marked "prioritised" and is thus allocated in the next allocation cycle. In this manner, any delays should be avoided. The prioritisation window is the period between the deadline of a task and time of prioritisation. During this time the task must be allocated to a vehicle, the vehicle must drive to the source and collect the material, drive to the sink and deposit the material. This method requires that the prioritisation window has the same length for each task and is determined by analysing the data of prior simulations.

The prioritisation window must allow enough time for the transport vehicles to fulfil tasks with long distances between the source and the sink as well as include a buffer for unexpected delays. However, a too generously planned prioritisation window will reduce the system's flexibility and thus impact negatively on its efficiency. Furthermore, if too many tasks were to become prioritised and had to compete for vehicles, delays might become inevitable.

Flexible Prioritisation Window

The *Flexible Prioritisation Window* is quite similar to the *Fix Prioritisation Window*, yet they differ in that the *Flexible Prioritisation Window* assigns an individual time window for each task, depending on the distance between source and sink and the time required to cover this distance. In order to determine the duration of this time window, an analysis of statistical data is performed. In this case, the time between the allocation of the task and the delivery is measured,

Parameter	*Characteristics*			
Prioritisation Methods	No Prioritisation	Fix Prioritisation Window	Flexible Prioritisation Window	Bidding Approach
Number of Vehicles	10	25	50	
Ratio Vehicles / Tasks	1 / 0.5	1 / 1	1 / 1.5	1 / 2

TABLE I: Experimental Plan illustrated as Morphological Box

Combination (vehicles / mission pool)	Fix Time Window	Flexible Time Window
10 / 5	468 s	309 s
10 / 10	468 s	305 s
10 / 15	519 s	407 s
10 / 20	475 s	298 s
25 / 13	485 s	309 s
25 / 25	437 s	279 s
25 / 38	437 s	346 s
25 / 50	485 s	344 s
50 / 25	452 s	280 s
50 / 50	439 s	244 s
50 / 75	442 s	279 s
50 / 100	444 s	291 s

TABLE II: Fix and Flex Time Window Values for the Scenarios of the Simulation-Study

but without the theoretical travel time between source and sink (distance divided by the average speed of the vehicle). This theoretical travel time is different for each task and is added to the determined time window value. This total duration represents the *Flexible Prioritisation Window*, which is calculated individually by backward-scheduling from the deadline of each task.

This approach can be considered an extension of the *Fix Prioritisation Window* method. It is expected that by means of the *Flexible Prioritisation Window*, fewer tasks are going to be prioritised, which will result in advantages in terms of efficiency and in time delivery.

Bidding Approach

The bidding approach adds a further strategy for prioritising tasks increased in complexity. Its operation method is different from the former presented approaches. The main difference being that this approach does not allocate prioritised tasks ahead of non-prioritised tasks, but adjusts the transportation costs when tasks become critical, ensuring these tasks are allocated and are dispatched on time. This approach was inspired by multi-attribute dispatching rules like Lampe and Clausen (2006) or Klein and Kim (1996) which take into account multiple criteria to determine the priority / costs of a task. These multi-attribute costs are processed by the GAP method which is solved

Vehicle delivers in time?	Bidding Factor
Yes and more than two others	1
Yes and two others	0.7
Yes and one other	0.2
Yes – the only one	0.1
No	3

TABLE III: Bidding Factors for different Situations

by the VAM-nq heuristic developed by Selmair et al. (2019).

The bidding factors that lower or raise the costs are calculated by including the distance of pick-up and transport from source to sink in a formula, to establish whether a vehicle is able to carry out the task in time. The prioritisation window w is compared with the period between the deadline of the task and the current time. If w is smaller than the period between the deadline and current time the task can be executed on time.

$$w = \frac{d_P + d_S}{v} * (1 + u) + t_P + t_D \qquad (5)$$

Where d_P is the distance from the current location of the vehicle to the source, d_S is the distance between the source and sink of the task, v is the average speed of the vehicle, u is the uncertainty factor that increases the travel time to compensate for unforeseen events, t_P is the time for collecting the material at the source, t_D is the time to deposit the material at the sink.

After ascertaining if the vehicle is able to fulfil the task in time, the vehicle checks if it is the only one to do so and recalculates the costs for the task with the bidding factor in Table III. These factors were determined by means of a parameter simulation.

SIMULATION STUDY

The following study built in the software *AnyLogic* aims to examine whether the presented methods of prioritisation achieve results that are similar in terms of efficiency to those obtained when solving the GAP without prioritisation. The main KPI used to proof the efficiency of the system will be the non-value-adding share of vehicles. Vehicles are not adding value when driving without any load – accordingly they are adding value when delivering a load from a source to its sink. The non-value-adding share of movement per task will be referred simply to NVA-share in the further course. Additionally, it is proposed that, by means of these measures, better results in terms of deliveries in time will be yielded. Therefore, the different methods were tested in a simulation study with different scenarios (see Table I).

The scenarios are simulated in a homogeneous grid-structured production environment with a total length of 4,072 meters in which a number of 5,000 tasks are processed for each scenario. The ratio between vehicles and tasks remains constant during every simulation run. That is, every time a task is completed, a new one appears – until the limit of 5,000 tasks is reached. The vehicles move with a maximum speed of 1.5 $\frac{m}{s}$. Their acceleration is parametrised to 1 $\frac{m}{s^2}$, the deceleration is set to 5 $\frac{m}{s^2}$. The tasks must be performed at an randomly selected point in time that lies between 9 to 15 minutes after a task has been generated. Additionally, tasks are more likely to be allocated at the beginning of said time interval. A solution is calculated in a cycle-time of 20 seconds to react to changes in the system, such as the appearance of new tasks or changes of the vehicles' status. While picking up or depositing material, the vehicles lower their speed so that the corresponding path is blocked for 25 seconds. In actual industrial applications, during this time period, the vehicles calculate the exact angle in which to drive to the Point of Interest (POI), by means of information gathered by 3D cameras.

The parameters of the *Fix* and *Flexible Prioritisation Windows* are determined by analysing the times between allocation and delivery of a task in a simulation run performed without prioritisation by using the GAP for allocating the tasks. The final parameter was set to the longest time that was measured for each vehicle / utilisation (v/u) combination (equals the 100 %-percentile) plus a 20-second-cycle time. Table II shows the parameters for the *Fix* and *Flexible Prioritisation Windows.*

For the uncertainty variable u of the bidding approach, a value of 30 % was used. This was based on the average travel uncertainty of the combinations, which lies between 20 % and 25 %. The set value includes a small buffer for accidental uncertainties.

RESULTS

The following section provides a summary of the results with particular focus on the two main Key Performance Indicators (KPIs): in time delivery and NVA-share of the vehicles. Moreover, it was decided to only include the results of simulation runs with 50 vehicles, as all other scenarios yielded very similar behaviour.

In order to explore the lower range of values associated with the NVA-share of vehicles, a scenario free of any prioritisation rules was utilised. This means that for the task allocation determined by the GAP, all tasks were assigned by following the fewest possible NVA-share (from the overall system perspective) – without imposing any deadlines and thus time pressure. In Figure 1, this value is represented by the first bar, from left to right, of every group, where each group represents one scenario of utilisation. The different levels of utilisation are achieved by keeping the simultaneous task pool sizes constant (25, 50, 75 and 100 tasks).

The second bar in Figure 1 represents the originally applied task allocation strategy: first-in-first-out in combination with the nearest-agent-first policy. For

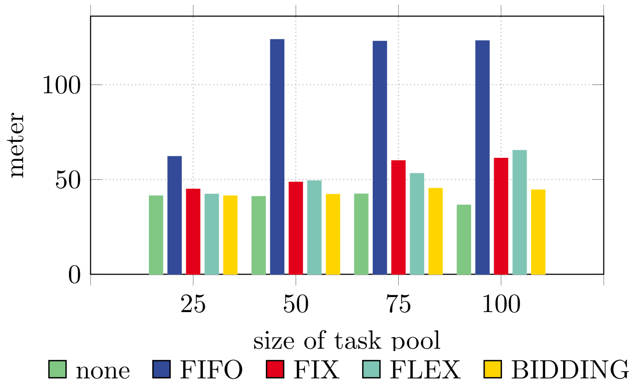

Fig. 1: Mean NVA-share per task for a scenario with 50 vehicles (5,000 Tasks)

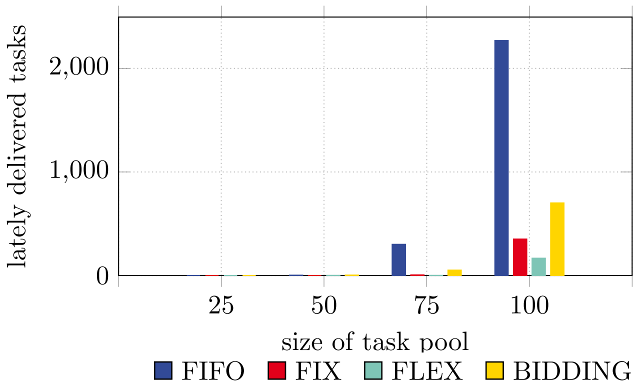

Fig. 2: Late Tasks of the original FIFO method and the three developed prioritisation approaches (5,000 Tasks)

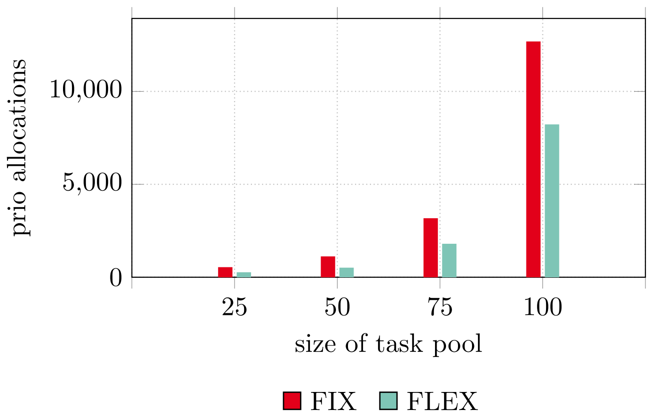

Fig. 3: Total Allocations with Priority for the *fix* and *flex* approach (5,000 Tasks) – Each task can be assigned more than once if decisions are frequently changed

the first scenario with a pool of 25 tasks, the system can freely select the closest of two vehicles, as the task to vehicle ratio is favourable at 1 : 2. As the number of tasks exceeds 50, substantially changing the allocation ratio, the system's choice is restricted to only one idle vehicle, which may not be the closest. It follows that the pick-up distances, measured in meters, are thus more likely to exceed 120 m.

For low utilisation scenarios (25 and 50 tasks), the *fix* and the *flex* approaches yield results close to those of the *bidding* approach with a maximum difference of 18 %. For a high utilisation (100 tasks), the NVA-share per task is 40 % higher than for the *bidding* approach. This will be elucidated further in the discussion.

The next three bars in the same Figure, bars 3 to 5, represent the pick-up meters for the three developed prioritisation methods in combination with the assignment by means of the GAP. For low utilisation levels, e. g. 25 and 50 tasks in the pool, all three prioritisation methods are quite similar to the "optimal" first bar which is not involving any prioritisation. The differences are only between 1 % and 8 %.

To evaluate all three methods, a closer look at the respective late tasks is deemed essential. Figure 2 illustrates the number of late orders for each scenario of utilisation. For the first scenarios with task pools of 25 and 50, all methods resulted in late orders below 0.3 %. However, the initially applied FIFO method resulted in 6 % late orders for the scenario with 75 tasks, that is, as soon as the content of the task pool exceeded the number of available vehicles. The high-pressure scenario with a task pool of 100, yielded late orders for every applied method. 55 % were on time when applying the FIFO rule, the *fix* approach delivered 93 % of all orders on time, while the *flex* approach resulted in 97 % of tasks being delivered on schedule. The results of the *bidding* approach in combination with the high utilisation scenario showed that only 86 % of all orders were able to be delivered on time.

DISCUSSION

The previously presented simulation results support the notion that the three methods, developed within this scope of research, are quite similar in terms of efficiency (see Figure 1). Furthermore, in high utilisation scenarios (75 and 100 tasks), all three methods yielded superior results in comparison to the currently applied FIFO method (see also Figure 1). Based on the concept that the *fix* and *flex* approach will prioritise more tasks in a high utilisation than in a low utilisation scenario (see Figure 3), it was expected that for high utilisation scenarios, the pick-up meters would exceed those of the scenarios without prioritisation. That is, all three approaches reduce flexibility and force the system to allocate urgent tasks immediately and these priority-tasks are delivered as fast as possible instead of way-efficient.

That means, that the *bidding* approach is able to prioritise each scenario with only little impact on NVA-share of vehicles in terms of meters. Nonetheless, it is presumed that the late tasks contribute to this outcome: starting with the scenario of 75 tasks, the *bidding* approach begins to deliver undesirable results. In order to ascertain the reason behind this, a close look at the calculation method of the *bidding* approach was necessary. In this approach, the calculation effort for urgent tasks is scaled down to support an urgent allocation. When handling more tasks with fewer available vehicles, the calculation efforts are scaled down in the calculation matrix for formulating the GAP. This leads to a reduction in the relative differences between the already scaled down efforts. Subsequently, tasks which are located peripherally are more likely to be processed than others. Especially these tasks contribute to the large number of delayed deliveries. Furthermore, the *bidding* approach is the approach with the highest calculation effort of all three introduced methods due to its complexity.

Fig. 4: The Smart Transport Robot (STR) of the BMW Group in its natural habitat

The *fix* and the *flex* approach yielded comparatively good results, both in terms of the distance of NVA-share as well as number of late tasks. The disadvantage of the *fix* and *flex* approach is associated with their independence regarding their possible vehicle choice. This independence is assessed through a time buffer that leads to lower flexibility, which in turn ensures that tasks are processed in time. The advantage of both methods is the low calculation effort: only one statistical value has to be determined in advance and applied continuously to each task. As described in the Section *Simulation Study*, this value is determined by using the 100 %-percentile. Due to this fact, most of the values are ascertained with a long buffer (see Table II). Nevertheless, this technique is necessary to ensure a minimal number of delayed tasks.

CONCLUSION

This paper has introduced three prioritisation approaches for a task allocation system of a transport vehicle system. All approaches differ in their complexity and calculation efforts. To evaluate their suitability for industrial application, all approaches were compared by means of simulation. Three different fleet sizes and four levels of utilisation were combined in order to assess performance under a variety of conditions. For the evaluation, two KPIs were focused on: the number of late tasks and the NVA-share in the system, mapped as average distance per task. While the significance of the first KPI is presumed to be self-evident, the importance of the second KPI is associated with resource conservation. That is, by minimising the NVA-share, fewer traffic conflicts occur, which allows traffic to flow more smoothly.

On the whole, all three approaches have qualified for regular utilised transport vehicle systems due to their ability to ensure that tasks are delivered in time. For performance and robustness reasons, the *fix* and *flex* approach are suggested to be more economically attractive within actual industrial use-cases. At the BMW Group, the *flex* approach has been implemented in a self-developed and self-built transport vehicle system (see Figure 4). The critical factor for this decision were the readily comprehensible calculation effort and the consistently positive results.

Further research might evolve the methods in order to provide a higher flexibility to the system whenever possible. Currently, the statistical determined values of Table II are overstated for the most tasks. This issue might be tackled in the next level of development.

REFERENCES

Le-Anh, Tuan and M. B. M. De Koster (2006). "A review of design and control of automated guided vehicle systems". In: *European Journal of Operational Research* 171.1, pp. 1–23.

Braunisch, Dirk (2015). *Multiagentensysteme in der rückführenden Logistik: Entwurf einer Systemarchitektur zur Steigerung der Prozesseffizienz durch dynamische Disposition der Sekundärrohstofflogistik.* Berlin.

Clausen, Uwe, Peiman Dabidian, Daniel Diekmann, Ina Goedicke, and Moritz Pöting (2013). "Entwicklung und Analyse einer multikriteriellen Einsatzsteuerung von Staplern in einem manuell bedienten Distributionslager." In: *Simulation in Produktion und Logistik*, pp. 207–216.

Díaz-Parra, Ocotlán, Jorge A. Ruiz-Vanoye, Beatriz Bernábe Loranca, Alejandro Fuentes-Penna, and Ricardo A. Barrera-Cámara (2014). "A Survey of Transportation Problems". In: *Journal of Applied Mathematics* 2014.3, pp. 1–17.

Gudehus, Timm (2012). *Dynamische Disposition: Strategien, Algorithmen und Werkzeuge zur optimalen Auftrags-, Bestands- und Fertigungsdisposition.* Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg.

Klein, C. M. and J. Kim (1996). "AGV dispatching". In: *International Journal of Production Research* 34.1, pp. 95–110.

Kuhn, H. W. (1955). "The Hungarian method for the assignment problem". In: *Naval Research Logistics Quarterly* 2.1-2, pp. 83–97.

Kundakcioglu, O. Erhun and Saed Alizamir (2009). "Generalized Assignment Problem". In: *Encyclopedia of Optimization.* Ed. by Christodoulos A. Floudas and Panos M. Pardalos. Boston, MA: Springer US, pp. 1153–1162.

Lampe, Heiko and Uwe Clausen, eds. (2006). *Untersuchung von Dispositionsentscheidungen in Umschlagterminals des kombinierten Verkehrs Schiene-Straße.* Logistik, Verkehr und Umwelt. Dortmund: Verl. Praxiswissen.

Ouelhadj, Djamila and Sanja Petrovic (2009). "A survey of dynamic scheduling in manufacturing systems". In: *Journal of Scheduling* 12.4, pp. 417–431.

Reinfeld, N. V. and W. R. Vogel (1958). *Mathematical Programming.* Prentice-Hall, Englewood Cliffs.

Selmair, Maximilian, Klaus-Jürgen Meier, and Yi Wang (2019). "Solving non-quadratic Matrices in assignment Problems with an improved Version of Vogel's Approximation Method". In: *European Conference of Modelling and Simulation.*

Shore, Harvey H. (1970). "The Transportation problem and the Vogel Approximation Method". In: *Decision Sciences* 1.3-4, pp. 441–457.

Srinivasan, V. and G. L. Thompson (1973). "An Algorithm for Assigning Uses to Sources in a Special Class of Transportation Problems". In: *Operations Research* 21.1, pp. 284–295.

Wagner, Rainer Maria (2018). *Industrie 4.0 für die Praxis: Mit realen Fallbeispielen aus mittelständischen Unternehmen und vielen umsetzbaren Tipps.* Wiesbaden: Springer Fachmedien Wiesbaden.

MAXIMILIAN SELMAIR is doctoral student at the University of Plymouth. Recently employed at the SimPlan AG, he was in charge of projects in the area of material flow simulation. Currently, he is working on his doctoral thesis with a fellowship of the BMW Group. His email address is: maximilian.selmair@bmw.de and his website can be found at maximilian.selmair.de.

VINCENT PANKRATZ contributed to the paper during his master thesis which was written in collaboration with the BMW Group. With this thesis, he finished his logistics studies at the Regensburg University of Applied Sciences and works now for a renowned automotive company. His email address is: vincent.pankratz@gmx.de.

KLAUS-JÜRGEN MEIER holds the professorship for production planning and logistic systems in the Department of Engineering and Management at the University of Applied Sciences Munich and he is the head of the Institute for Production Management and Logistics (IPL). His email address is: klaus-juergen.meier@hm.edu.