



**HAL**  
open science

# Mobile-agents based hybrid control architecture - implementation of consensus algorithm in hierarchical control mode

Guillaume Demesure, Hind Bril El Haouzi, Benoît Iung

► **To cite this version:**

Guillaume Demesure, Hind Bril El Haouzi, Benoît Iung. Mobile-agents based hybrid control architecture - implementation of consensus algorithm in hierarchical control mode. *CIRP Annals - Manufacturing Technology*, 2021, 70 (1), pp.385-388. 10.1016/j.cirp.2021.04.065 . hal-03238595

**HAL Id: hal-03238595**

**<https://hal.science/hal-03238595>**

Submitted on 2 Aug 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License



## Mobile-Agents based Hybrid Control Architecture—Implementation of consensus algorithm in hierarchical control mode

Guillaume Demesure, Hind Bril El-Haouzi, Benoit Iung (1)

*Université de Lorraine, CRAN, UMR CNRS 7039, Campus Sciences, BP 70239, Vandœuvre-lès-Nancy, 54506, France*

The concept of autonomous mobile robots has already been implemented in some manufacturing fields; however, it is not yet effective in the field of shop floor logistics because issues linked to decision-making control remain. A contribution to this challenge is proposed in this study through an innovative hybrid control architecture in which mobile agents adapt their degree of autonomy by switching between hierarchical and heterarchical operating modes to dynamically face disturbances and absorb them. The focus is on operational manufacturing control and the navigation layer in hierarchical mode, where a consensus control algorithm is elaborated to reduce the instability with respect to the detailed schedule. Simulation results are provided to demonstrate the effectiveness of the proposed consensus algorithm.

Flexible manufacturing system, Decision making, Consensus.

### 1. Introduction

Industry 4.0, with its technological advances, is revolutionising industrial practices, while providing improvements in agility [1], particularly through the use of digital tools or methods such as simulation, digital twins, distributed artificial intelligence (e.g., multi-agent systems), and robotic arms or vehicles. To contribute more strongly to this agility in the context of robotics, autonomous mobile robots (AMRs) have been introduced in an increasing number of industrial applications, such as in warehouses, but with few or no applications in the transportation of products within the manufacturing process [2]. AMRs are the evolution of automated guided vehicles (AGVs), in which decisions related to scheduling, routing, and dispatching are controlled through a central entity [3, 4]. Compared to these AGVs, AMRs have embedded hardware and software that are sufficiently advanced to perform different types of tasks autonomously and dynamically [5]. These capabilities lead to a decentralised decision-making process that covers multiple industrial requirements, such as system reactivity or adaptability. However, the implementation of these AMRs within the manufacturing process requires reconsidering traditional control and planning methods, especially in terms of mobility. Indeed, the mobility of AMRs allows for higher transport flexibility, but requires more complex navigation tools (e.g., motion planning). Despite the possibility of coupling complex navigation tools with methods of manufacturing control [6], a high degree of flexibility (induced by mobility) leads to complex production planning and control due to the variability of transportation times between machines because they depend on conflicts that cannot be predicted. Moreover, this flexibility leads to nervousness that is the result of changes in production planning, causing instability at the lower level of product control. Nervousness is induced by the flexibility of lower-level entities because flexibility provides a higher scope of actions and lower-level entities decide which action to perform locally [7]. Therefore, the scientific challenge is to balance the flexibility of agent navigation according to the situation, while minimising the impact of nervousness.

Research efforts in the field of manufacturing control have led to hybrid control in the intelligent manufacturing systems (IMS) and

holonic manufacturing systems (HMS) communities [8, 9]. In addition, some results proved that switching between hierarchical and heterarchical modes is an effective way to deal with internal or external changes, while maintaining global performance. Indeed, in a hierarchical mode, production decisions are made by a central entity to globally optimise manufacturing performance, whereas in a heterarchical mode, local entities (e.g. products and/or machines) autonomously decide how the system can be processed for reactivity/anticipation purposes. This hybrid aspect can be used as a solution to balance flexibility because the navigation methods may change while switching between modes.

Consequently, two objectives are proposed in this study: (a) to propose a novel hybrid control architecture with mobile agents and (b) to focus on the hierarchical operating mode of this architecture as the beginning of a wider framework. This innovative architecture is called the “Mobile agent-based Hybrid Control Architecture (MoHyCA)” and has the originality to couple navigation with decision-making process. In the proposed approach, the AMRs are considered as mobile agents in a multi-agent systems control paradigm [10], which independently make decisions, communicate or negotiate with other agents (resources, machines), and move freely on the production floor (i.e., without necessarily following any imposed routes). The architecture has been designed using a human-inspired approach to ease understanding and human system integration [11], both in hierarchical and heterarchical operating modes. The main originality of this architecture is that it offers advanced capacities to support the navigation of mobile agents by considering potential evolutions, and facing internal (e.g., machine breakdown) and external disturbances (e.g., rush order). The heterarchical mode is proposed to evaluate the ability of mobile agents to autonomously ensure the completion of products according to a given plan, despite conflicts between these agents during navigation. A human-inspired algorithm adapted from previous work on consensus-like negotiation [14] is proposed, which is consistent with existing methods and approaches [10, 12, 13]. This algorithm has the advantages of fast computation with few communication exchanges and reduces the impact of nervousness by ensuring the stability requirements because consensus allows the convergence of individual performances. To

defend these contributions, the remainder of this paper is organised as follows. The MoHyCA architecture is introduced in Section 2. In Section 3, the consensus algorithm used in the hierarchical operating mode is presented. Next, case studies and simulation results are proposed in Sections 4 and 5, respectively. Finally, conclusions and future work are presented in Section 6.

## 2. MoHyCA architecture

To introduce the MoHyCA architecture, let us consider a manufacturing system that processes  $n$  products, transported by mobile agents, a set of machines, and paths to navigate between these machines. Figure 1 shows the different decision layers of the MoHyCA architecture.

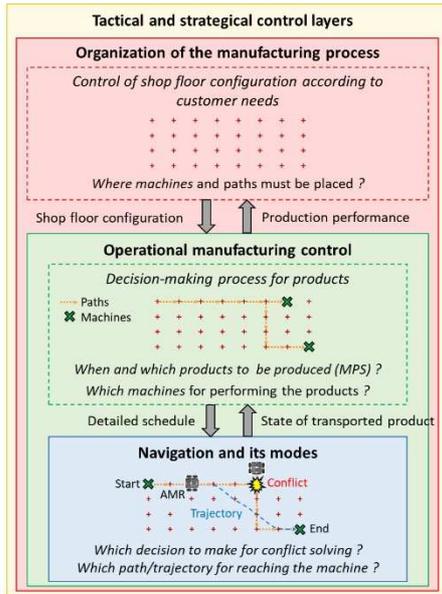


Figure 1. MoHyCA architecture layers

The organisation of the manufacturing process layer (red box) focuses on shop floor configuration or reconfiguration. It involves determining where machines and paths must be placed to optimise production according to information from strategic and tactical control layers (product portfolio, volume target, inventories, etc.). The organisation of a manufacturing system with transport flexibility remains a complex issue [15] and requires more studies that will not be addressed in this study. According to the configuration provided by the organisation layer, the operational manufacturing control layer (green box) has the role of deciding when and which product to be processed, and on which machine these products will be processed [corresponding to master production scheduling (MPS) and production activity control (PAC)] based on the needs of the final product. This layer computes a detailed schedule for each product based on the current states of transported products and shop floor, which are then provided to the mobile agent that will transport this product. The navigation layer (blue box) is classically decentralised and processed using mobile agents. Its role is to ensure the achievement of the product while facing the different problems (conflicts and disturbances) that occur during navigation.

The objective of the MoHyCA architecture in manufacturing control and navigation layers is the management of mobile agents to address such problems. This architecture maintains the advantages of switching between hierarchical and heterarchical modes [8] while adapting the transport flexibility of products according to the operating mode. The hierarchical mode is

dedicated to the fulfilment of the production schedule (i.e., imposed machines and dates) through the mobile agents in a normal situation (i.e., without disturbance). To reduce the complexity of production scheduling, the agents in this mode must follow imposed paths between machines, leading to low variability in transportation times. However, all agents in this mode continue to decide how to solve conflicts between themselves on the paths based on their own individual performance. Conversely, the heterarchical mode allows facing different disturbances and absorbing them for reactivity and resilience purposes. To better absorb the disturbances, the agent becomes capable of moving freely on the production floor and modifying the detailed schedule, as mentioned in [6]. These autonomous capabilities enhance the flexibility that allows AMRs, in a dynamic way, to face different disturbances such as machine breakdown, rush order etc. The switching decisions between hierarchical and heterarchical modes are based on the performance evaluation of the transported product. For example, each disturbance will cause some delays that will impact the product performance, causing an AMR to switch to a heterarchical mode. Hereafter, the scope of this study is narrowed to the hierarchical mode in the operational manufacturing control (OMC) layer to introduce the first contribution to the MoHyCA architecture.

## 3. Hierarchical mode in OMC layer

To clarify the hierarchical mode, let us consider a predefined configuration of the production floor where machines are motionless and mobile agents transport products from machine to machine following navigation paths. The objective of each mobile agent is to ensure the completion of the transported product, assigned at the input of the production system through a detailed schedule with finite capacity, according to product planned orders (from MPS). This schedule assumes AMR availability at the input of the system, and unavailability is considered as a disturbance. The computed planning, composed of starting and ending dates (different from the due date), and the product routing procedure, are provided to agents. Through this routing procedure and the configuration of the production floor, the mobile agents are responsible for controlling the product activity until its completion. The decision-making algorithm is divided into two parts: the inter-agent conflicts solving at crossed paths (sink node) and moving to waiting area-decisions at splitting paths (burst node). In a hierarchical mode, to enhance flexibility while ensuring the expected level of production performance and stability, a consensus algorithm is proposed: the mobile agents enter negotiations to obtain a trade-off between their individual performance.

The consensus algorithm is a procedure applied in control theory, which is based on the interactions between agents that collaborate for their goals through means of the convergence of a common state [14]. The interacting aspect of this algorithm allows adaptation to flexible manufacturing systems, where the collaboration is performed in a performance-based state. The state is designed according to the performance objectives that depend on the desired strategy. In this study, the proof of concept is achieved through a scenario of a manufacturing system, managed using a just-in-time strategy where lateness must be minimised and customer satisfaction is ensured. Therefore, state  $x_i$  is adapted to ensure a just-in-time production by relating the due date  $d_i$  with the estimation of the completion date  $c_i$  and its lower limit  $c_i^*$ , as follows:

$$x_i(t) = \frac{d_i - c_i(t)}{d_i - c_i^*(t)} \quad (1)$$

, where  $t$  is the instant at which state  $x_i$  is computed. The completion date  $c_i$  can be estimated using different means, such as online simulation (or digital twin), and its lower bound  $c_i^*$  is computed when the product is assigned to an AMR, according to the minimum travel and processing times as if the agent is alone in the process (i.e., without waiting time). In the hierarchical mode, it is assumed that  $c_i^* < d_i$  because an AMR will switch to the heterarchical mode when this condition is not fulfilled. The state  $x_i$  has the advantage of comparing the margin of product completion with the best possibility of completion. Reaching a state equal to 0 means that the product will be completed *in time*, whereas positive or negative values indicate *earliness* or *tardiness*, respectively.

To obtain a trade-off when deciding which action to perform, each agent compares its state to those of other interacting agents, called neighbours, to compute the disagreement (i.e., the difference between agent states). The minimisation of this disagreement is obtained when the individual states of all collaborating agents converge to the same value at the end of the process. This convergence prevents instability, which reduces the effect of nervousness because the disagreements are less than the given threshold. The agent and its neighbours compute the criteria  $u_i$ , as expressed in Eq. 2, which corresponds to the sum of the disagreements between the agents.

$$u_i(t) = \sum_{j \in N_i} (x_i(t) - x_j(t)) \quad (2)$$

The agent with the lowest value of criteria among the group of neighbours has the poorest completion performance, compelling others to adapt their plan to facilitate the completion of product transported by this agent. In Eq. 2, the set  $N_i$  corresponds to the neighbours of agent  $i$ , which are the other agents involved in the decision-making process. According to the node (burst or sink) reached by an agent, the set of neighbours considers all other agents that will be impacted by this agent, leading to waiting times of these neighbours according to the decision made. On the one hand, the decision to move to the waiting area at burst nodes is made when the agent has the lowest value of criteria among its neighbours, that is,  $u_i < \min(u_j), \forall j \in N_i$ . On the other hand, to solve conflicts at sink nodes, the neighbours are grouped according to the input path, and the criteria  $u_i$  are computed for all neighbours in each group. The decision of an agent  $i$  to wait at a burst node means that the lowest value of criteria among its neighbours  $j \in N_i$  is not the minimum among the lowest values of criteria of all groups, i.e., there exists an agent  $k$  such that  $u_k < \min(u_i, u_j), \forall j \in N_i$ . This consideration of neighbours, which are not in direct conflict at the node, expands the scope of the negotiation. Indeed, the agent at the node should not wait if another agent arriving at the sink node through the same path has the worst performance among all agents affected by the decision.

#### 4. Case study to validate consensus algorithm

The proposed consensus algorithm, used for the hierarchical operating mode of the MoHyCA architecture, was applied to a case study extracted from the TRACILOGIS test-bed platform (Figure 2) that serves as a research experimentation frame [14]. This platform is close to a real-world industrial situation (i.e., with machines, robots, and PLCs) of product manufacturing and is composed of four zones, where products transported by AMRs are processed at Zone A, can wait at Zone B, in the case might be assembled in Zone D, with parts sorted in Zone C. Points (drilling) or lines (grooving) represent the operations performed at Zone A using machines  $M_1$  and  $M_2$ . The buffer area in Zone B has the capability to welcome the six AMRs. Zone D is the bottleneck of

the platform where products are assembled at “machines”  $M_3$  and  $M_4$ . As shown by the arrows in Figure 2, the routes are unidirectional and without narrow passages. The AMRs, in hierarchical control mode, have the possibility to loop at zone A, to reach machine  $M_2$ , and zone B, and then to wait and let lately other products to be assembled earlier in zone D. The AMRs were identical, with speed set to 20 cm/s for the simulations, and the maximum capacity of the platform was set to 10 AMRs (i.e., 10 products simultaneously).

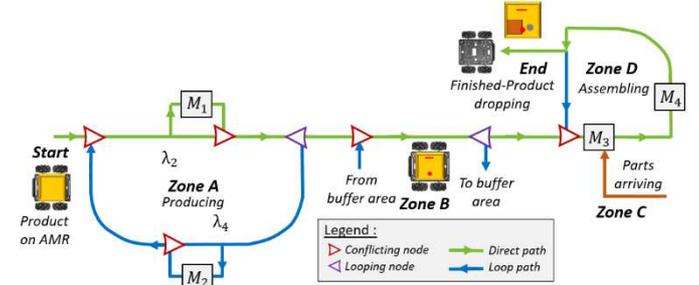


Figure 2. Configuration of TRACILOGIS-based simulation environment

For the proposed experiments, let us consider a planning (given by a detailed schedule computed using an optimisation algorithm for example). The objective is to highlight the efficiency of AMRs to follow a given schedule, despite the imperfections, to ensure the completion of the transported product in time, enabled by the proposed consensus control algorithm. However, the given schedule is constrained to prevent the completion of products on the same date. To test this efficiency of completion, the processing times at machines  $M_1$  and  $M_2$  are variable, allowing the emulation of some small disturbances that may occur when AMRs navigate through paths.

#### 5. Simulation results

To test the consensus algorithm, five detailed schedules (S1–S5) based on a set of ten products (P1...P10) to complete, with six types of product (t1...t6), have been proposed. The sequence of operations and the respective processing times of each type of product are listed in Table 1. One can notice that each schedule does not have the same sequence of products and has variable processing times (i.e., normal  $N(\mu, \sigma)$  or exponential  $e(\lambda)$ ).

Table 1 Sequence and processing times at machines of types of product

Type	Sequence of machines				
	$M_1$	$M_2$	$M_1$	$M_3$	$M_4$
t1	$N(2,0.2)$	$N(1,0.1)$	-	$28.5+e(2)$	$23.1+e(3)$
t2	$N(2,0.2)$	$N(1,0.1)$	-	$39+e(2)$	$30.8+e(4)$
t3	$N(2,0.2)$	-	-	$18+e(2)$	$15.4+e(2)$
t4	$N(3,0.3)$	$N(1,0.1)$	$N(4,0.4)$	$9.5+e(3)$	$7.7+e(2)$
t5	$N(8,0.8)$	-	-	$9.5+e(3)$	$7.7+e(2)$
t6	$N(11,0.1)$	$N(3,0.3)$	-	$9.5+e(3)$	$7.7+e(2)$

The consensus algorithm was compared to the current competitive algorithm implemented on the TRACILOGIS platform. This algorithm is generic for various application cases and consists of ensuring the completion of the product on time, regardless of other product completion, as if the AMRs have short-range communication capabilities and compute their production range at each event, similar to FCFS (first come first served) [14]. Three performance indicators will be evaluated for the comparison: (a) the service rate  $S_r$  corresponding to the percentage of products arriving at or before the due date, (b) the disagreement  $d_s = \sum_{i,j=0}^n (x_i - x_j), n = 10$  between products, which evaluate the consensus achievement and consequently the

stability, and (c) the average lateness  $l_a = \overline{|d_i - c_i|}$  among the products for one run, where  $c_i$  refers to the final completion time of product  $i$ . The average lateness is often used for just-in-time decision-making processes because it considers both earliness and tardiness.

For the comparative studies, the simulations were performed in Python, and each schedule was run 100 times. Figure 3 shows the comparative results of each performance indicator. Consequently, compared to the competitive algorithm, the production performance of the consensus algorithm shows its effectiveness on the hierarchical control problem. Moreover, the obtained results highlight the impact of consensus on system stability because convergence reduces the spread of average lateness and service rate. Indeed, the lack of collaboration to solve conflicts leads to worse disagreement and prevents some agents from improving their own performance at the expense of others.

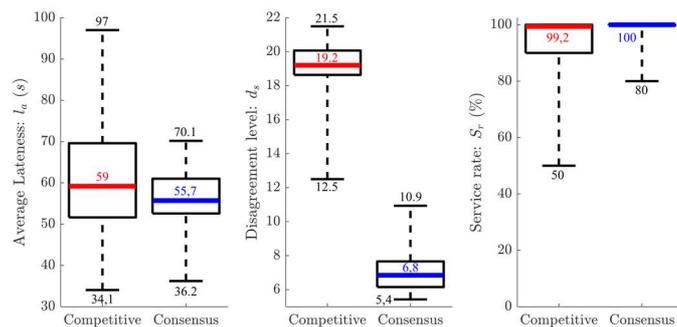


Figure 3. Comparative results of production performance

Conversely, the negotiation proposed by the consensus algorithm forces AMRs to evaluate the impact of their actions and decide according to the requirements of their neighbours. Figure 4 enhances the need for collaboration to improve the performance of a fleet while reducing nervousness. For example, the AMRs that transport products  $P_1$  and  $P_8$  allow others to complete their product in time and reduce lateness (because a positive or negative state means early or tardy completion, respectively). Moreover, the agents are less impacted by others during navigation, reducing the nervousness and instability, as shown by the variations in state until 300 s in Figure 4.

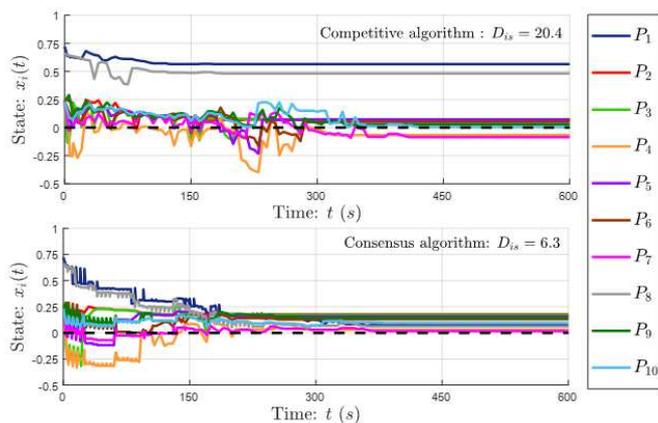


Figure 4. Comparison of product states

## 6. Conclusion

This study introduces a mobile agent-based hybrid control architecture, in which mobile entities are used in both transportation of products and the process as a machine. Each mobile agent has the responsibility to complete the transported product despite the disturbances that may occur. The originality

of this architecture is the adaptation of the transportation flexibility that leads to the complete autonomy of mobile agents in terms of controlling the product through the process and allowing free navigation in the process (no imposed routes) to ensure the completion of products. The first focus is on the hierarchical operating mode, in which a consensus algorithm is proposed. This human-inspired algorithm provides agents with capabilities for adaptation according to their own performance and neighbours. Therefore, all agents negotiate to obtain a trade-off and minimise disagreement in the group. Simulation results are provided to demonstrate the effectiveness of the proposed consensus algorithm. Moreover, these results highlight the advantages of collaborating because the convergence of agent states leads to improvement in other performances (ex: lateness and service rate).

From the first focus on hierarchical mode, several prospects can be investigated. First, other decisions must be considered in this mode, such as the decision to decrease or increase the AMR speed, which may prevent useless switching to the heterarchical mode. Second, a control algorithm must be designed for the heterarchical mode. This algorithm needs to be adaptable to the navigation method that allows AMRs to navigate without following any imposed routes, as mentioned in [9]. Next, the switching mechanisms between the hierarchical and heterarchical modes must be addressed to define the moment at which the mobile agents switch between the modes, regardless of the disturbance that leads to this switch. Subsequently, additional perspectives related to the organisation of the production floor must be managed. Indeed, the configuration of the production floor, that is, the position of the machines and paths between them, impacts the environment in which the AMRs navigate. Therefore, the optimisation of a configuration according to market demand, as well as the decision to reconfigure the production floor, are other aspects to be considered. The final requirements before real implementation in the industry of this architecture are the experimental parts that may lead to the reconsideration of some of the steps in the global framework.

## References

- [1] Gunasekaran, A., Yusuf, Y. Y., Adeleye, E. O., Papadopoulos, T., Kovvuri, D., Geyi, D. A. G., 2019, Agile manufacturing: an evolutionary review of practices, International Journal of Production Research, 57/15-16:5154-5174.
- [2] Fragapane, G., Ivanov, D., Peron, M., Sgarbossa, F., Strandhagen, J. O., 2020, Increasing flexibility and productivity in industry 4.0 production networks with autonomous mobile robots and smart intralogistics. Annals of Operations Research, 1-19.
- [3] Sabattini, L., Digani, V., Secchi, C., Cotena, G., Ronzoni, D., Foppoli, M., Oleari, F., 2013, Technological roadmap to boost the introduction of AGVs in industrial applications. IEEE 9th International Conference on Intelligent Computer Communication and Processing, 203-208.
- [4] De Ryck, M., Versteijhe, F., Debrouwere, 2020, AGV systems, state-of-the-art control algorithms and techniques. Journal of Manufacturing Systems, 54:152-173.
- [5] Krä, M., Hörbrand, S., Schilp, J., 2019, Dynamic production control for flexibility in Cyber-Physical Production Systems using an autonomous transport system. Procedia CIRP, 81:1160-1165.
- [6] Demesure, G., Defoort, M., Bekrar, A., Trentesaux, D., Djemai, M., 2017, Decentralized motion planning and scheduling of AGVs in an FMS. IEEE Transactions on Industrial Informatics, 14/4:1744-1752.
- [7] Pujawan, I. N., 2004, Schedule nervousness in a manufacturing system: a case study. Production planning & control, 15/5:515-524.
- [8] Cardin, O., Ounnar, F., Thomas, A., Trentesaux, D., 2016, Future industrial systems: best practices of the intelligent manufacturing and services systems (IMS<sup>2</sup>) French Research Group. IEEE Transactions on Industrial Informatics, 13/2:704-713.
- [9] Gräßler, I., & Pöhler, A., 2017, Implementation of an adapted holonic production architecture. Procedia CIRP, 63:138-143.
- [10] Van Brussel, H., Valckenaers, P., 1989, Hierarchical control of a generic flexible assembly cell. CIRP Annals, 38/1:33-36.
- [11] Stadnicka, D., Litwin, P., Antonelli, D., 2019, Human factor in intelligent manufacturing systems-knowledge acquisition and motivation. Procedia CIRP, 79:718-723.
- [12] Meissner, H., Ilsen, R., & Aurich, J. C., 2017, Analysis of control architectures in the context of Industry 4.0. Procedia CIRP, 62, 165-169.
- [13] Nikolakis, N., Kousi, N., Michalos, G., Makris, S., 2018, Dynamic scheduling of shared human-robot manufacturing operations. Procedia CIRP, 72:9-14.

- [14] Mezgebe, T. T., Demesure, G., Bril El-Haouzi, H., Pannequin, R., Thomas, A. 2019, CoMM: a consensus algorithm for multi-agent-based manufacturing system to deal with perturbation. *The International Journal of Advanced Manufacturing Technology*, 105/9:3911-3926.
- [15] ElMaraghy, W., ElMaraghy, H., Tomiyama, T., Monostori, L., 2012, Complexity in engineering design and manufacturing. *CIRP Annals*, 61/2:793-814.