



HAL
open science

A Manufacturing-as-a-Service Scheduling Problem and its Tripartite Decision Perspective

Cristian Duran-Mateluna, Valeria Borodin, Alexandre Dolgui, Simon Thevenin

► **To cite this version:**

Cristian Duran-Mateluna, Valeria Borodin, Alexandre Dolgui, Simon Thevenin. A Manufacturing-as-a-Service Scheduling Problem and its Tripartite Decision Perspective. 11th IFAC Conference on Manufacturing Modelling, Management and Control (MIM), Jun 2025, Trondheim, Norway. pp.1558-1563, <10.1016/j.ifacol.2025.09.262>. <hal-05184767>

HAL Id: hal-05184767

<https://hal.science/hal-05184767v1>

Submitted on 24 Jul 2025

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 CC BY-NC-ND 4.0 - Attribution - Non-commercial use - No Derivative Works - International License

A Manufacturing-as-a-Service Scheduling Problem and its Tripartite Decision Perspective^{*}

Cristian Duran-Mateluna^{*,**} Valeria Borodin^{*}
Alexandre Dolgui^{*} Simon Thevenin^{*}

^{*} *IMT Atlantique, LS2N-CNRS, Nantes, France*
(e-mail: cristian.duran-mateluna@imt-atlantique.fr)

^{**} *University of Santiago of Chile (USACH), Faculty of Engineering,
Industrial Engineering Department, Santiago, Chile*

Abstract: The Manufacturing-as-a-Service (MaaS) paradigm offers flexible and on-demand manufacturing services through digital platforms, including the case where customer facilities outsource tasks to providers of manufacturing resources (i.e., machines). A key challenge lies in matching the tasks and resources requested by customers with suitable providers according to their specific requirements. Within the limits of the available requirements and their level of accuracy, task-machine eligibility can be represented in various ways, such as binary values or score-based weights. These requirements are then integrated into the subsequent scheduling problem to be solved by the MaaS platform. In this paper, we extend a scheduling model by integrating task-machine eligibility score-based weights as input to the scheduling problem via eligibility level constraints. Numerical experiments are conducted on randomly generated small instances based on benchmark schemes to study the implications of the introduced task-machine eligibility levels. The quality of sequences proposed within the MaaS framework is empirically evaluated for customers and providers in both aggregate and individual manners. Copyright ©2025 IFAC.

Keywords: Manufacturing-as-a-Service, Scheduling Problem, Task-machine eligibility, Tripartite decision perspective.

1. CONTEXT AND MOTIVATIONS

Manufacturing-as-a-Service (MaaS) is a transformative trend in modern manufacturing, that relies on the servitization of manufacturing resources enabled by cloud manufacturing (Ren et al., 2017; Henzel and Herzwurm, 2018; Bulut et al., 2021; Ivanov et al., 2022). Companies are supposed to access manufacturing resources through digital platforms (Garcia et al., 2019; Santos et al., 2024). Rather than investing in capital-intensive production infrastructure, customers can now procure manufacturing capabilities from service providers, who offer their resources in a flexible, scalable, and efficient manner. There are different MaaS business models for on-demand platforms, which can be categorized in marketplaces with operators or single manufacturers, hybrid models, or federated open collaborative models¹. A MaaS system includes three decision

perspectives: **(i)** Consumers of services who prioritize on-time completion and low costs, **(ii)** service providers who seek to maximize their use of available resources and profitability, and **(iii)** the MaaS platform that centralizes and coordinates the provisioning and release of services to optimize overall system performance (Chen et al., 2024). MaaS also involves important logistical aspects to be coordinated (Liu et al., 2024; Zhao et al., 2024; Wang et al., 2024). A critical challenge for MaaS platforms is to solve the task-machine matchmaking and scheduling problems efficiently. Hertwig et al. (2024) propose a general framework for these problems. Once a task is matched with one or several eligible shared machines by providers, the challenge is to schedule this task on these machines. While substantial research has been conducted on matchmaking and scheduling problems in related areas, these problems are often treated independently. In most cases, matchmaking is only approached through eligibility constraints, ensuring that a provider can technically fulfill the task requirements. Current MaaS platforms can generate a ranking among eligible service providers.

This paper addresses the following research question: How can the tripartite decision perspectives that the MaaS context entails be included and characterized in scheduling problems? In this regard, we contribute by **(i)** extending a state-of-the-art problem of Maecker et al. (2023) by considering task-resource eligibility scores generated by

^{*} The present work was conducted in the framework of the project ACCURATE (Achieving Resilience through Manufacturing as a Service, Digital Twins and Ecosystems), supported by European Union's Horizon Europe research and innovation program under grant agreement No. 101138269, HORIZON-CL4-2023-TWIN-TRANSITION-01-07. However, the views and opinions expressed are those of the author(s) only and do not necessarily reflect those of the European Union or European Health and Digital Executive Agency (HADEA). Neither the European Union nor the granting authority can be held responsible for them.

¹ <https://www.masst2040.eu/knowledge-hub/>

a matchmaking process as input, and **(ii)** analyzing the quality of the solutions through a set of key performance indicators on randomly generated benchmark instances. This article is organized as follows. Section 2 introduces and formulates the problem under study, and discusses the proposed extensions. Section 3 presents the results of the numerical experiments and analyzes the obtained solutions from a tripartite decision perspective. Conclusions and perspectives are presented in Section 4.

2. MAAS SCHEDULING PROBLEM

As a starting point for this study, we considered the model proposed by Maecker et al. (2023) for the unrelated parallel machine scheduling problem with eligibility constraints and delivery times, aiming to minimize total weighted tardiness. The definition of their problem was motivated by a cloud-based manufacturing services provider for Printed Circuit Boards (PCB). In this context, machine customers can upload and edit their PCB designs through an online tool and place production orders. The service provider is supposed to have access to a wide network of PCB companies with different capabilities. Building on this idea, this paper adopts a tripartite perspective of MaaS, wherein one group of users provides services, another group requests them, and a centralized MaaS framework facilitates and coordinates the exchange of these services.

2.1 Problem formulation

Given a set of n service orders (i.e., tasks) $j \in \mathcal{J}$ and a set of m resource providers (i.e., machines) $i \in \mathcal{M}$, the MaaS platform coordinates the assignment and sequence of tasks to the most qualified machines. Each task j has a due date d_j , a weight w_j , a processing time p_{ij} , and a delivery time q_{ij} for each machine i . All tasks are available at time zero, each task j needs to be processed by one and only one machine without interruption, and each machine i can handle at most one task at a time. The delivery time q_{ij} occurs immediately after completing the task j on the respective machine i . The objective function minimizes the total weighted tardiness $\sum w_j T_j$, where $T_j = \max\{C_j - d_j, 0\}$ and C_j is the completion time defined as the time by which a task reaches the customer. The mixed-integer linear formulation includes binary assignment variables y_{ij} of task j to machine i and precedence variables z_{jk} between tasks j and k . Variables s_j denote the start time of task j . The model is formulated as follows:

$$\min \sum_{j \in \mathcal{J}} w_j T_j \quad (1)$$

$$\sum_{i \in \mathcal{M}} y_{ij} = 1 \quad j \in \mathcal{J} \quad (2)$$

$$s_k \geq s_j + p_{ij} - H(3 - y_{ik} - y_{ij} - z_{jk}) \quad i \in \mathcal{M}, j, k \in \mathcal{J}, j \neq k \quad (3)$$

$$s_j \geq s_k + p_{ik} - H(2 - y_{ik} - y_{ij} + z_{jk}) \quad i \in \mathcal{M}, j, k \in \mathcal{J}, j \neq k \quad (4)$$

$$C_j \geq s_j + (p_{ij} + q_{ij})y_{ij} \quad i \in \mathcal{M}, j \in \mathcal{J} \quad (5)$$

$$T_j \geq C_j - d_j \quad j \in \mathcal{J} \quad (6)$$

$$T_j, C_j \geq 0 \quad j \in \mathcal{J} \quad (7)$$

$$y_{ij}, z_{jk} \in \{0, 1\} \quad i \in \mathcal{M}, j, k \in \mathcal{J}$$

Constraints (2) state that each task must be assigned to exactly one machine. Disjunctive Constraints (3) and (4) manage the precedence relationship between tasks on machines. Task completion times and tardiness values are computed according to Constraints (5)-(7). Constant H is an upper bound on task start times and can be computed by: $H = \max_{i \in \mathcal{M}} \left\{ \sum_{j \in \mathcal{J}} p_{ij} - \min_{j \in \mathcal{J}} p_{ij} \right\}$.

2.2 New task-machine eligibility score-based constraints

In Maecker et al. (2023), the problem operates with eligible machine sets M_j to identify the subsets of machines that can process task j . Typical MaaS platforms match technical capabilities to compute the eligibility of providers based on the requirements of the customers. However, it is also possible to consider a ranking of eligible providers under other complementary criteria, such as sustainability and resilience (see, e.g., Schuseil et al. (2024)). Based on these eligibility scores $e_{ij} \in [0, 1]$ between machine $i \in \mathcal{M}$ and task $j \in \mathcal{J}$, let us consider a minimal eligibility level θ for each task using the following new constraints:

$$\sum_{i \in \mathcal{M}} e_{ij} y_{ij} \geq \theta \quad j \in \mathcal{J} \quad (8)$$

Task-machine eligibility satisfaction can also be imposed in an aggregated manner at the customer network level. One approach is to aggregate the eligibility levels using, for instance, the average:

$$\sum_{j \in \mathcal{J}} \sum_{i \in \mathcal{M}} e_{ij} y_{ij} / n \geq \theta \quad (9)$$

From a practical standpoint, Constraints (8) and (9) allow the MaaS platform to take into account the information provided by machine customers and providers by specifying an eligibility threshold. This approach thereby addresses explicitly the satisfaction of customer requirements either on a task-specific basis or at the customer network level. Note that task-machine eligibility can also be included in the set of MaaS optimization criteria. This is left for future work. However, we briefly examine the maximum achievable eligibility levels under various total tardiness values in Section 3.

2.3 Denotation according to the Graham notation

Consider the three-field notation $\alpha|\beta|\gamma$ proposed by Graham et al. (1979), where α refers to the manufacturing environment and more general scheduling settings, β refers to problem constraints, γ specifies optimization criterion/criteria. As formalized in Hertwig et al. (2024), a MaaS scheduling problem includes the available specifications of machine consumers and providers that are combined by the MaaS platform $\langle \alpha_{cp}, \beta_{cp}, \gamma_{cp} \rangle$, where c stands for resource consumers and p for resource providers. Note that the unification of distributed specifications is not a trivial task and relies on the interoperability capability of a MaaS framework (Hertwig et al., 2024). Consequently, the proposed MaaS scheduling problem under study corresponds to $Rm|q_{ij}, \theta|\sum w_j T_j$, where **(i) Environment and settings**: Rm indicates unrelated parallel machines, **(ii) Constraints and characteristics**: q_{ij} Delivery time q_{ij} and different forms of levels of task-machine eligibility θ , and **(iii) Optimization criterion**: Total weighted tardiness.

3. NUMERICAL EXPERIMENTS

This section presents experiments that **(i)** make explicit the interdependence between the task-machine eligibility levels and total tardiness, and **(ii)** evaluate and discuss the solution quality from the perspectives of machine customers, providers, and the MaaS platform itself. All experiments in this paper are conducted on an Apple M3 Pro computer with 4.05 GHz and 18 GB RAM. The models are coded in the Julia language using the JuMP libraries and CPLEX as the optimization solver.

3.1 Instance generation

First experimental results consider small instances with 4 machines and 10 tasks. In future research, we plan to develop efficient solution approaches for industrial-scale instances that consider the distributed nature of a MaaS system coordinated in a centralized way. The problem parameters were generate similarly as by Maecker et al. (2023), in which the due dates follows the scheme of Potts and Van Wassenhove (1982):

$$d_j \sim U\left(\bar{p}\left(1 - T - \frac{R}{2}\right) + q^{min}, \bar{p}\left(1 - T + \frac{R}{2}\right) + q^{min}\right)$$

where T is the tardiness factor, R is the relative range of due dates, and \bar{p} and q^{min} are calculated as follows:

$$\bar{p} = \frac{1}{m} \sum_{j \in \mathcal{J}} \min_{i \in \mathcal{M}} p_{ij} \quad \text{and} \quad q^{min} = \min_{i \in \mathcal{M}, j \in \mathcal{J}} q_{ij}.$$

All task weights w_j are considered equal to 1, for easier interpretation of the tardiness values and in the absence of relevant considerations in this sense at this stage. All parameters are summarized in Table 1.

Table 1. Instance generation: *Parameters*

Parameter	Values
Processing times	$p_{ij} \sim U(1, 100)$
Delivery times	$q_{ij} \sim U(1, 30)$
Eligibility coefficients	$e_{ij} \sim U(0, 1.0)$
Due dates	$T \in \{0.4, 0.6\}$ $R \in \{0.4, 0.6, 0.8\}$

15 instance samples have been generated for each pair of parameters (T, R) .

3.2 Design of experiments

Numerical experiments have been conducted according to the following two experimental settings:

Min $\sum T_j$, given θ . Under these problem settings, let us distinguish the satisfaction level θ imposed in the model from the actual eligibility level associated with a given feasible solution S . Given an eligibility value θ and the best-known solution S^* , we can identify the actual eligibility value $\theta(S^*) := \theta^*$.

$\theta^*(S) \geq \theta$, and even if $\theta = 0$. Before finding a feasible solution, we can identify for each task $j \in \mathcal{J}$ the maximal possible level of eligibility among eligible machines $e_j^+ = \max_{i \in \mathcal{N}} e_{ij}$. These reference values serve as lower and upper bounds of eligibility levels per task, which are exploited in the next section.

To illustrate the shape of the relationship between $\sum T_j$ and θ , Model (1)-(7) subject to Constraints (8) (respectively, Constraint (9)) has been firstly solved with a minimal eligibility level $\theta = 0$. Let the associated solution obtained during the first iteration $i = 0$ be denoted by S_0^* . Then, θ is iteratively increased by 10% from $\theta^*(S_i^*)$ until no feasible solution exists. This process leads to the generation of a total of 682 instances.

Max θ , given $\sum T_j$. To assess the extent to which eligibility levels can be improved, we consider a variant of the model that maximizes the eligibility level given a total tardiness level. To do this, at each iteration, once a solution with an objective value \mathcal{T} is found, we solve $\{\max \theta, \text{given } \sum T_j \leq \mathcal{T}\}$. In this way, we can find an alternative solution with the same total tardiness \mathcal{T} but the maximum reachable eligibility level, denoted by θ^+ .

3.3 Solution quality assessment and illustrative example

One of the main goals of this study is to assess the solutions provided by the model for each of the three perspectives involved, i.e., those of customers, providers, and the MaaS platform itself, that is intended to conciliate the interest of all stakeholders. In this paper, the MaaS platform generates solutions that minimize total tardiness. The quality of these solutions is ambiguous and requires further clarification. In this sense, we analyze the satisfaction of:

- *Task customers* in terms of tardiness and number of On Time Delivered (OTD) tasks,
- *Machine providers* in terms of the number of assigned tasks and machine active/idle times. The idle time is calculated with respect to the total completion time.

The list of considered indicators is summarized in Table 2.

Table 2. Key performance indicators

[MaaS, customers] Tardiness	
$\sum T_j$	Total tardiness
%OTD	Percentage of tasks on time
T_j^+	Maximal tardiness
\bar{T}_j	Average tardiness
σ_{T_j}	Standard deviation of the tardiness
[MaaS, customers] Task-machine eligibility	
θ	Minimal eligibility level
θ^*	Actual eligibility level
θ^+	Maximal possible eligibility level
$\bar{\theta}_j$	Average task eligibility value
$\Delta\theta_j^*$	Average of the differences with the e_j^+
[MaaS, providers] Task assignment	
$\#_i^-$	Minimal number of assigned tasks
$\#_i^+$	Maximal number of assigned tasks
$\sigma_{\#_i}$	Standard deviation of number of assigned tasks
[MaaS, providers] Machine active time	
C_i^-	Minimal machine active time time
C_i^+	Maximal machine active time time
\bar{C}_i	Average machine active time time
$\Delta_{C_i^+}$	Average of idle time

For illustration purposes of these KPIs, consider a randomly generated instance of 10 tasks, 4 machines, and $\theta = 0$ with Constraint (8). Figure 1 and Figure 2 present the corresponding initial solution of the problem, and the solution with the maximum possible of θ^* , respectively.

Figures 1-2 show that Constraint (8) has a significant impact on the solution by changing its characteristics for multiple tasks and doubling the associated total tardiness.

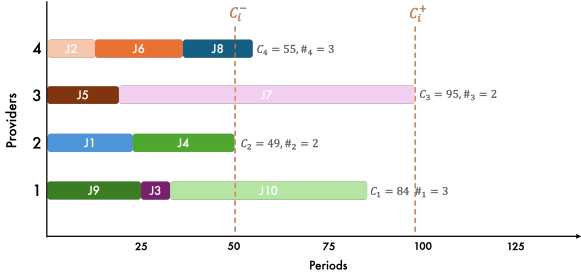


Fig. 1. Solution $\theta = 0, \theta^* = 0.11, \sum T_j = 162$

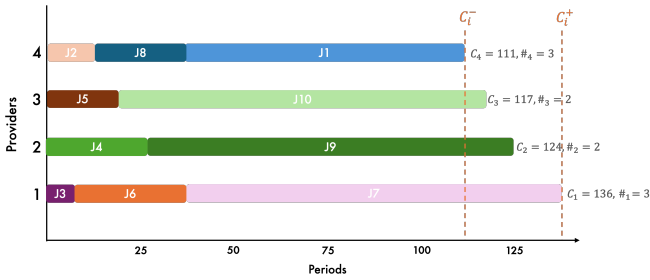


Fig. 2. Solution $\theta = 0.48, \theta^* = 0.53, \sum T_j = 353$

3.4 Analysis of results

Let us discuss the quality of the solutions from the perspectives of the decision-makers involved in the MaaS framework. Table 3 summarizes all KPIs averaged by group of instances depending on parameters (T, R) .

MaaS perspective. Note that the initial solution of both constraints with $\theta = 0$ is the same. Figure 3 shows the distribution of the percentage of OTD tasks. Given the parameters used, most of the tasks have already been delivered with tardiness without considering any eligibility constraint. Consequently, as parameters (T, R) increase, the tardiness-related KPIs also increase. Figure 4 and Figure 5 present the value of the objective function in relation to θ for Constraints (8) and (9), respectively. As expected, the model under Constraints (9) is more flexible, and delays increase more slowly as the corresponding θ increases.

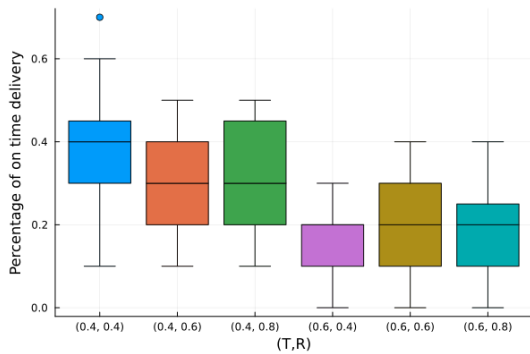


Fig. 3. Distribution of OTD of tasks with $\theta = 0$

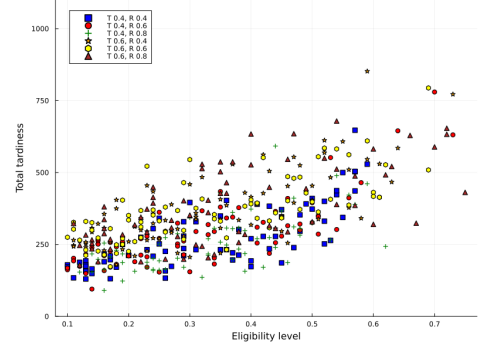


Fig. 4. Relationship $\sum T_j$ and θ^* using Constraints (8)

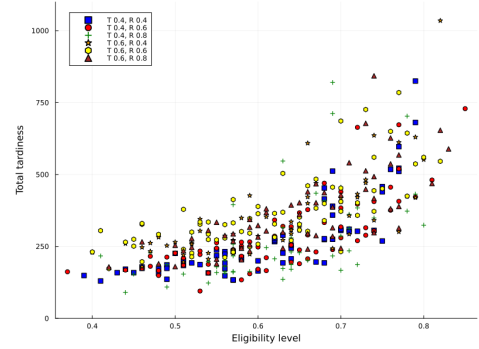


Fig. 5. Relationship $\sum T_j$ and θ^* using Constraints (9)

Figures 6-7 illustrate the distributions of the total tardiness for both Constraints (8) and (9), respectively. Instances with higher values of (R, T) can induce higher total tardiness values and eligibility flexibility for Constraint (9).

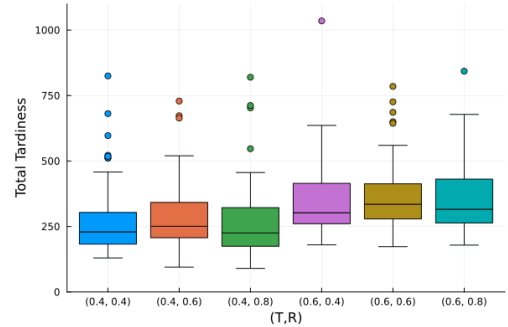


Fig. 6. Distribution of $\sum T_j$ using Constraints (8)

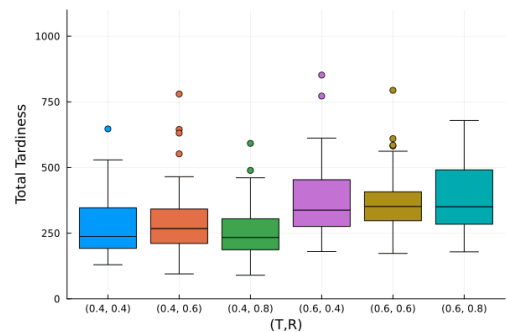


Fig. 7. Distribution of $\sum T_j$ using Constraints (8)

Table 3. Averaged KPIs per parameter pair (T, R) and type of eligibility constraints

Averaged KPI	Model under Constraints (8)						Model under Constraint (9)					
	(0.4, 0.4)	(0.4, 0.6)	(0.4, 0.8)	(0.6, 0.4)	(0.6, 0.6)	(0.6, 0.8)	(0.4, 0.4)	(0.4, 0.6)	(0.4, 0.8)	(0.6, 0.4)	(0.6, 0.6)	(0.6, 0.8)
$\sum T_j$	276	290	254	374	369	384	273	289	272	346	368	361
%OTD	0.32	0.25	0.31	0.15	0.16	0.18	0.30	0.25	0.30	0.16	0.15	0.17
T_i^+	90	95	83	96	105	104	92	88	85	96	103	101
\bar{T}_j	28	29	25	37	37	38	27	29	27	35	37	36
σ_{T_j}	33	32	29	34	35	36	32	31	30	33	35	35
θ	0.26	0.24	0.24	0.26	0.26	0.28	0.49	0.50	0.47	0.48	0.50	0.50
θ^*	0.33	0.32	0.33	0.34	0.34	0.36	0.61	0.63	0.62	0.62	0.61	0.64
θ^+	0.33	0.32	0.33	0.34	0.34	0.36	0.61	0.63	0.62	0.62	0.61	0.64
$\bar{\theta}_j$	0.59	0.61	0.60	0.61	0.61	0.64	0.61	0.63	0.62	0.62	0.61	0.64
$\Delta_{\theta_j^*}$	0.18	0.16	0.15	0.17	0.17	0.16	0.16	0.15	0.13	0.17	0.16	0.15
$\#i^-$	1.6	1.8	1.7	1.5	1.6	1.5	1.6	1.7	1.6	1.5	1.7	1.5
$\#i^+$	3.6	3.3	3.6	3.5	3.4	3.6	3.5	3.4	3.6	3.5	3.4	3.8
$\sigma_{\#i}$	1.0	0.8	0.9	1.0	0.9	1.0	0.9	0.8	1.0	1.0	0.9	1.1
C_i^-	52	51	47	47	48	47	49	48	46	41	48	38
C_i^+	112	113	105	108	111	110	115	110	108	107	112	106
\bar{C}_i	81	80	75	78	78	79	79	78	77	73	78	72
σ_{C_i}	27	28	26	28	28	28	30	28	28	30	29	31
$\Delta_{C_i^+}$	32	33	30	30	33	31	36	33	31	34	34	34

Machine provider perspective. The number of tasks assigned and the active time are similar for both constraints and instance types. Figures 8-9 illustrate the differences in idle times for both constraints. While the average idle times are comparable, the maximum values and standard deviations differ significantly between the groups of instances, leading to more extreme values in the first case.

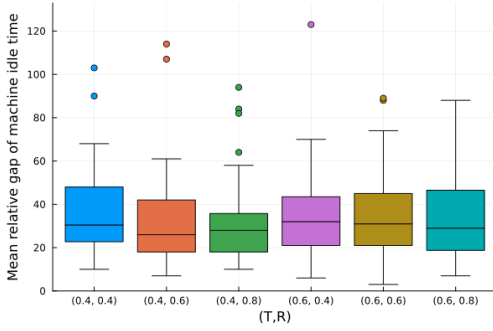


Fig. 8. Distribution of $\Delta_{C_i^+}$ using Constraints (8)

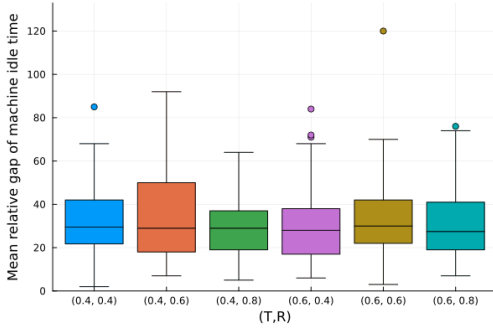


Fig. 9. Distribution of $\Delta_{C_i^+}$ using Constraints (9)

Task customer perspective. Figures 10-11 present the relationship between θ and θ^* . For several instances, given a level θ , a solution with a higher actual value θ^* is found, which is more accentuated in the model under Constraints (8). These constraints can effectively guarantee a level of task-machine eligibility defined per task. However, this will have an impact on the overall performance.

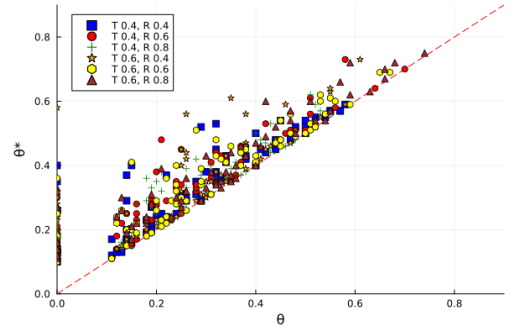


Fig. 10. Relation between θ and θ^* with Constraints (8)

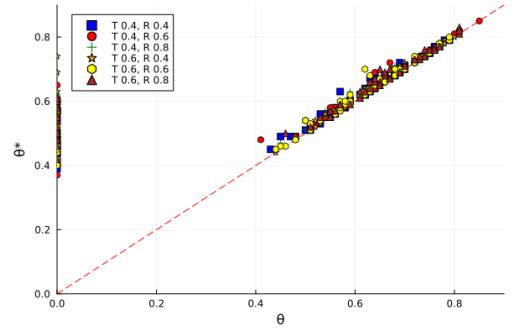


Fig. 11. Relation between θ and θ^* with Constraints (9)

Similarly, Figures 12-13 present the relation between θ^+ and θ^+ . Although there are some exceptions for lower levels of θ^+ in both cases, the initial model finds, in most instances, the solution that maximizes the level of eligibility at the same time. These results motivate further exploration of alternative model variants to uncover solutions that address multiple perspectives simultaneously.

4. CONCLUSION AND PERSPECTIVES

This work addresses the MaaS scheduling problem with task-machine eligibility levels, coordinating the sharing of manufacturing resources in a centralized way guided by total tardiness as an optimization criterion. Extensive numerical experiments are conducted on instances generated

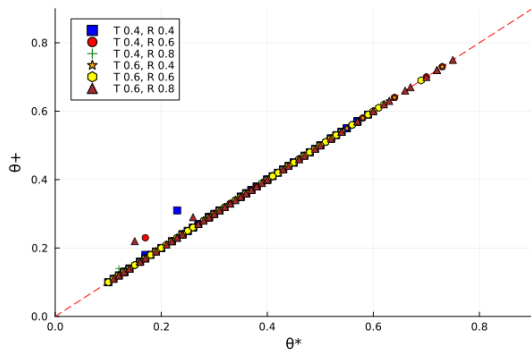


Fig. 12. Relation between θ^* and θ^+ with Constraints (8)

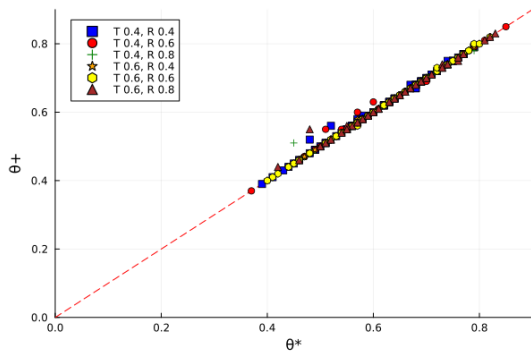


Fig. 13. Relation between θ^* and θ^+ with Constraints (8)

based on state-of-the-art schemes to characterize the solutions from the perspectives of task customers and machine providers. The numerical experiments and the related analysis provide preliminary guidelines for MaaS platforms to align the interests of all stakeholders. A higher level of task-machine eligibility implies a greater increase in total tardiness, with greater flexibility when task-machine eligibility constraints are integrated in a joint way. These relationships are clear from an aggregate perspective but remain obscure at a more detailed level. This study sheds light on the quality and fairness of sequences proposed within the MaaS framework, which are evaluated from both global and individual perspectives for customers and providers.

In future research, we plan to extend this initial work as follows:

- Propose and analyze other MaaS-related optimization criteria to balance the interests of customers and providers at both network and individual levels.
- Incorporate additional constraints to the problem, such as the time availabilities of providers or the availability of tasks at different starting times or logistical aspects.
- Consider collaborative and dynamic settings where users can accept or reject proposals.
- Extend the model to include negotiation mechanisms, including cost configurations and dynamic pricing.

REFERENCES

Bulut, S., Wende, M., Wagner, C., and Anderl, R. (2021). Impact of manufacturing-as-a-service: Business model adaptation for enterprises. *Procedia CIRP*, 104, 1286–1291.

- Chen, W., Feng, P., Luo, X., and Nie, L. (2024). Task-service matching problem for platform-driven manufacturing-as-a-service: A one-leader and multi-follower Stackelberg game with multiple objectives. *Omega*, 129, 103157.
- Garcia, F., Grabot, B., and Paché, G. (2019). Adoption mechanisms of a supplier portal: A case study in the european aerospace industry. *Computers & Industrial Engineering*, 137, 106105.
- Graham, R.L., Lawler, E.L., Lenstra, J.K., and Kan, A.R. (1979). Optimization and approximation in deterministic sequencing and scheduling: a survey. In *Annals of discrete mathematics*, volume 5, 287–326. Elsevier.
- Henzel, R. and Herzwurm, G. (2018). Cloud manufacturing: A state-of-the-art survey of current issues. *Procedia CIRP*, 72, 947–952.
- Hertwig, M., Schuseil, F., Lentés, J., Borodin, V., Duran-Mateluna, C., Dolgui, A., and Thevenin, S. (2024). Ontology-based matchmaking and scheduling for manufacturing as a service [Accepted]. In *Procedia CIRP*.
- Ivanov, D., Dolgui, A., and Sokolov, B. (2022). Cloud supply chain: Integrating Industry 4.0 and digital platforms in the “Supply Chain-as-a-Service”. *Transportation Research Part E: Logistics and Transportation Review*, 160, 102676.
- Liu, S., Deng, Q., Liu, X., Luo, Q., Li, F., and Jiang, C. (2024). Dual-service integrated scheduling of manufacturing and logistics for multiple tasks in cloud manufacturing. *Expert Systems with Applications*, 235, 121129.
- Maecker, S., Shen, L., and Mönch, L. (2023). Unrelated parallel machine scheduling with eligibility constraints and delivery times to minimize total weighted tardiness. *Computers & Operations Research*, 149, 105999.
- Potts, C. and Van Wassenhove, L. (1982). A decomposition algorithm for the single machine total tardiness problem. *Operations Research Letters*, 1(5), 177–181.
- Ren, L., Zhang, L., Wang, L., Tao, F., and Chai, X. (2017). Cloud manufacturing: key characteristics and applications. *International Journal of Computer Integrated Manufacturing*, 30(6), 501–515.
- Santos, R., Piqueiro, H., Dias, R., and Rocha, C.D. (2024). Transitioning trends into action: A simulation-based digital twin architecture for enhanced strategic and operational decision-making. *Computers & Industrial Engineering*, 198, 110616.
- Schuseil, F., Hertwig, M., Lentés, J., Zimmermann, N., and Hölzle, K. (2024). A semantic matchmaking approach to empower human decision-making in manufacturing-as-a-service scenarios. *Human Factors in Design, Engineering, and Computing. AHFE (2024) International Conference. AHFE Open Access*, 159, 106105.
- Wang, X., Lin Zhang, Y.L., and Laili, Y. (2024). An improved deep reinforcement learning-based scheduling approach for dynamic task scheduling in cloud manufacturing. *International Journal of Production Research*, 62(11), 4014–4030.
- Zhao, T., Lu, S., Cheng, H., Ren, M., and Liu, X. (2024). Coordinated production and delivery scheduling for service-oriented manufacturing systems with deterioration effect. *International Journal of Production Research*, 62(20), 7598–7617.