Unlocking the Potentials of Modularity in Railways, a Heuristic Framework for Pods Scheduling

Ximeng Liao[1], Jihee Han[1], Aaron Paz Martinez[2] and Mahnam Saeednia[1]

***Abstract*— Under the pressing need for increasing sustainability of freight transport, the rail sector faces challenges in competing with its road counterpart which can be contributed to lower flexibility and reliability. To overcome these challenges, intermodal rail freight systems should undergo a transformative evolution to enhance flexibility and seamlessly integrate with broader mobility services. Crucial to the success of any such innovative solution is integration with the existing railway infrastructure. This study explores the concept of modular vehicles (MVs) in rail systems, focusing particularly on the innovative 'Pod' system introduced in the Pods4Rail project. It introduces a framework for Pods scheduling on the railway network, incorporating an overlap-level-based platooning. Experiments show that implementing this system can significantly reduce the makespan and optimize railway capacity utilization, especially when dealing with larger problem sizes.**

I. INTRODUCTION

Intermodal freight transportation involves moving goods using two or more modes, such as rail and truck, from origin to destination. Increasing the share of rail-based intermodal freight transport is a critical step in reducing emissions in this sector [1]. Nonetheless, this form of transport has experienced a declining trend over the past decade, primarily due to factors such as reduced flexibility and reliability compared to its primary competitor, i.e. the road sector.

To mitigate this impact, it is imperative to evolve intermodal, rail-based freight transport systems to offer increased flexibility, while utilizing the existing infrastructure. Historically, rail freight transport has predominantly catered to bulk and mining commodities [2]. Enabling the transportation of smaller loads, including less than container/container loads, expands the range of freight demand suitable for rail transport. Hence, the rail industry is actively exploring innovative solutions to address this challenge. Advancements in vehicle-to-vehicle communication and fully automatic operation technologies, such as virtual coupling, have paved the way for the emergence of modular vehicles (MVs) in rail systems [3]. A “Pod” system is a MV-concept characterized by its detachable capsule-carrier architecture (Figure 1) operating within a decentralized and autonomous transport network. It allows for the seamless transshipment of capsules -so called transport units (TU)- within identical or between diverse transport modes, including railway and road, thereby enabling the utilization of a single carrier (also referred to as moving infrastructure, MI) to transport different TUs at different times. The handling system, the TU-carrier coupling and the mobility management system (MMS) are crucial subsystems of the Pod. As part of the Europe´s Rail Joint Undertaking's (EU-Rail) as the new European partnership on rail research and innovation established under the Horizon Europe programme, the ”Pods4Rail” [3] project commits to conceptualizing the fundamental groundwork for this system. The primary objective of this project is to establish fully autonomous intermodal mobility systems catering to both passengers and freight, characterized by sustainability, collaboration, interconnectivity, digitization, standardization, and compatibility with different transport modes. As a basic requirement of MMs, carriers should be assigned to TUs enabling their transport on the railway network. Complete Pods (combined carriers and TUs) then create Pod-swarms or platoons of virtually coupled Pods, enhancing capacity utilization. This paper proposes a heuristic framework for carrier scheduling to pickup and deliver TUs, which is compared against single pod pick-up and delivery.

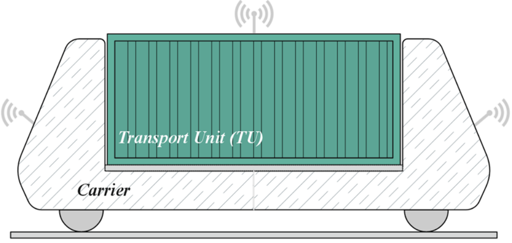

Fig. 1. Sketch of an initial Pod concept with one (freight) transport unit (TU) coupled to the carrier (DLR, 2024).

The paper is structured as follows: Section II elaborates on literature review and research gaps, and section III introduces the functioning of modular vehicle platooning on railways. Section IV describes problem and assumptions for Pod scheduling on railway. Section V presents the framework of the modular vehicle scheduling with platooning on railway. Section VI summarizes the simulation results and conducts a sensitivity analysis. Finally, Section VII concludes the findings of this study.

II. LITERATURE REVIEW

Modularity can be defined as a group of (Pods or modular) vehicles capable operating independently or as a interconnected Pod platoons [4], demonstrating potential advantages such as reduced operational costs and increased vehicle and

[1] Faculty of Civil Engineering, Transport and Planning Department, TU Delft, Stevinweg 1, The Netherlands `X.Liao,j.han-1,m.saeednia@tudelft.nl`

[2] Institute of Vehicle Concepts, German Aerospace Center (DLR), Rutherfordstraße 2, Berlin, Germany `aaron.pazmartinez@dlr.de`

infrastructure utilization in different transportation applications [5], [6]. The investigation into platooning and modularization within the railway sector is still emerging, enabled by the advancements in virtual coupling (VC) technology. This concept has been studied from a technical viewpoint [23], [7], [18], [24], [25], [8], [16]. Additionally, [26] analyzed the concept of slip coaching. Within this operational concept, not only can a part of a train be left behind at intermediate stations without the main platoon reducing its speed, but it can also accelerate again after stopping and be picked up by a following platoon. Additional studies have highlighted its impact on railway operations, e.g. [10], [11] explored urban metro rail platooning to enhance passenger satisfaction, [12], [13] analyzed modular vehicle (MV) scheduling along urban rail corridors, highlighting cost reductions and improved utilization rates over traditional operations, and [14], [15] investigated variable-capacity MV operations in urban railways, demonstrating notable cost savings.

Much of the existing research on virtual train coupling focuses on microscopic control aspects like speed adjustments, with fewer studies addressing operational scheduling challenges. Current models often overlook the heterogeneity and traceability of trains or modular vehicles in platoons, generally treating these as indistinct entities for demand fulfillment. This contrasts with road truck platooning, where individual trucks remain identifiable. Additionally, there's a noted research gap on how modularity affects rail operations. This paper aims to fill this gap by exploring modular vehicle (Pods) operations on railways, especially at intersections of road and rail networks. Here, freight Pods can dynamically join or leave rail platoons, remaining identifiable throughout the process. The study focuses on carrier scheduling, TU assignment, platoon formation, and Pod sequencing decisions.

III. DESCRIPTION OF THE OPERATING SYSTEM

As described by [24], virtually coupled train sets (VCTS) will bring self-organised operation with short vehicles onto the tracks. However, the nature of railways allows fewer track changes than when changing lanes or roads. As a result, there is always a tendency to couple railway vehicles together to form longer units (platoons). The operation of the entire Pods system on the railway network depends upon the convergence of several key factors: availability of carriers, their assignment to the influx of freight demand originating from road, and scheduling the movement of the carrier-TU combination. This paper introduces a heuristic framework to handle the complexities of coordinating these phases. In this framework, the carrier departs from its current location to pick up TUs. The combination, referred to as a Pod, waits for other Pods for the formation of the platoon. Eventually, TUs are delivered at their corresponding destinations.

IV. PROBLEM DESCRIPTION AND ASSUMPTIONS

Taking into account the planning horizon T_h, each TU ($p \in P$) is specified by its origin, destination and time constraints. The railway network is represented as a graph $G = (N, A)$, where N and A are sets of stations (nodes) and sections (arcs). Each section is represented as a tuple $(i, j), i, j \in N$, and divided into a set of segments, SC_{ij}. A set of carriers C facilitate the movement of TUs on the network. Each carrier is associated with its last scheduled location and job-finishing time (marking its availability for the next pick-up), where 'job' refers to: MP (pre-movement for pickup), PI (pick-up), PL (platooning), MD (movement for delivery), DR (drop-off), and SS (stopping for separation). Pre-movement refers to cases that the assigned carrier needs to move to the origin station of the TU to pick it up. Waiting for platooning signifies the waiting of a TU for all carriers to be ready for platooning.

A. Assumptions and notation

Considering the notations outlined in Table I, the following assumptions underlie the system modeling and the heuristic approach detailed in Section V: From a technical standpoint, availability of only one type of carrier is considered, particularly regarding dimensions and axle load, with each carrier capable of transporting only one TU at a time. It is assumed as a necessary simplification, that the battery maintains a full state of charge (SOC). All TUs carry an identical payload to ensure uniform dynamic forces. Pods are assumed to operate at a maximum speed of 2.5 km/min, but carrier speeds will comply with EN 15528 standards (route classes to manage the interface between vehicle load limits and infrastructure) during actual operations.

From an operational point of view, if the number of carriers is not sufficient, the corresponding TUs need to wait until carriers are available (limited by a waiting threshold). There exists a pick-up and drop-off time at stations (i.e. the service time). In case of platooning, time for platooning and separation is also considered. Regarding the scheduling of carriers, firstly, given a section divided into a set of segments, maximum one individually dispatched carrier can occupy each segment at a time. Secondly, movement of carriers/platoons on each segment should respect the headway constraint, assumed to be a constant value (h). Note that, The current scope leaves out detailed scheduling and train control aspects on the railway, as well as empty movement of carriers and Pods to be integrated in future works.

V. FRAMEWORK DESCRIPTION

This section describes the heuristic framework for Pods dispatching (including platooning decisions) and carrier assignment.

A. Pods dispatching and carrier assignment heuristic

To ensure maximum capacity utilization of the railway infrastructure, the overarching objective of this heuristic is to minimize the makespan, i.e. to transport all Pods to be dispatched within the planning horizon T_h.

Ensuring the availability of carriers is a fundamental prerequisite for facilitating the timely dispatching of Pods. The assignment of carriers to Transport Units (TUs) is determined based on several factors, including availability and proximity to the TUs' pick-up locations. This proximity is measured

TABLE I

NOTATIONS

Notation	Description
N	Set of stations (nodes)
A	Set of sections (arcs)
SC_{ij}	Set of segments on section $(i, j) \in A$
P	Set of TUs (={0, 1, 2,..., $\|P\|$})
C	Set of carriers (={0, 1, 2,..., $\|C\|$})
dt_p	Due date of Pod p, $p \in P$
o_p	Origin station of Pod p, $p \in P$
d_p	Destination station of Pod p, $p \in P$
loc_c	Last scheduled location of carrier c, $c \in C$
$time_c$	Last scheduled finishing-time of carrier c, $c \in C$
T_{wait}	Time threshold to wait for carrier availability
$route_{i,m}$	Route from i to m $\forall i, m \in N$
$time_p$	time when the movement of Pod p is completed, $p \in P$
l_{ij}^k	Length of segment k on section $(i, j) \in A$, $k \in SC_{ij}$
h	Headway length
v	Carrier speed (km/min)
T_h	Time horizon for scheduling
ns_{ij}	Maximum number of time slots on section $(i, j) \in A$
ns_{ij}^{used}	Used number of time slots on section (i, j), $(i, j) \in A$

by the Estimated Time of Arrival (ETA) of carriers at the pick-up locations. Carriers are then assigned to an ordered set of TUs based on their Earliest Due Date (EDD).

The detailed pseudocode presented in Algorithm 1 determines the schedule of carriers. It starts with identifying a sorted set of individual set of Pods, P^{unpl}. The first Pod in P^{unpl}, with the earliest due date is referred to as the standard Pod, sp. At each iteration, TUs that have been platooned are removed from P^{unpl}. The platooning function (Function 1), decides a platoon formation which consists of lists of Pods and of the corresponding carriers. Lines 9-17 address the pre-movement of carriers (in case the carrier's current location and TU's origin are different). Note that carrier's last scheduled location and time (loc_c, $time_c$), are updated every time they are assigned a new job. For a platoon to be formed, carriers arriving at different times at the origin station of sp, o_{sp}, may need an additional 'waiting' time (Lines 23-32).

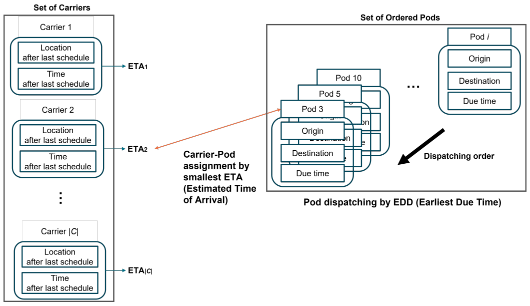

Fig. 2. Pod dispatching and carrier assignment

B. Overlap-level-based platooning

Platooning is expected to reduce makespan and enhance capacity utilization. An overlap-based platooning concept is introduced to the framework to examine this. Referring to the route of sp as the 'base route' (Figure 3), a Pod may join the platoon on three occasions: as part of the 'base platoon' where all Pods share the same route (Lines 33-48). Pods that are placed in the front have a longer route and those Pods that share part of the base route (shorter than the base route) are placed in the rear. The latter separate from the base platoon (Lines 39-46, (sct_{end} indicates the end station of the section). The algorithm is presented in Function 1.

In Line 3, a list of available carriers within T_{wait}, the time threshold to wait for carriers, will be generated. Beyond this threshold, the assignment of carriers to TUs will not be made, rendering those carriers as unavailable for the pick-up activity. An exception is made for sp where at least one carrier will be assigned for it regardless of T_{wait}. Once carrier availability is confirmed, the base platoon is formed (Lines 6-11). Then, front and rear Pods are inserted (Lines 18-20 and 21-31, respectively). Sequencing of Pods has been considered in the rear section, by inserting the Pods that share the least number of overlapping sections with the base Pod farthestfrom the base platoon.

VI. RESULTS

To validate the proposed algorithm, railway network in Miao [22] (Figure 4) is used. This network comprises four stations, interconnected by three railway tracks: The track between stations 0 and 1 is divided into 40 segments of varying lengths, while the tracks between stations 1 and 2, and stations 1 and 3, are divided into 15 and 10 identical segments, each with a length of 1 km. The information related to TUs and carriers is randomly generated. The base scenario includes 100 carriers and 200 TUs, each with their respective origin, destination, and due date. All other relevant experiments parameters are summarized in table II. Additionally, it is assumed that the distance between pods in a platoon is negligible and the safety margin applies only for independently dispatched pods and between platoons. In addition to makespan, the performance of the algorithm is evaluated considering capacity utilization, u_{capa}. It is calculated as the number of occupied slots in the timetable (see Equation 1).

$$u_{capa} = \frac{\sum_{(i,j)\in A} ns_{ij}^{used}}{\sum_{(i,j)\in A} ns_{ij}} \quad (1)$$

A. Experiments

This section presents a sensitivity analysis of the impact of key simulation parameters, including the number and ratio of TUs/carriers, service time (pick-up, delivery), headway, and carrier speed and availability on makespan and capacity utilization. Firstly, an experiment is conducted to examine

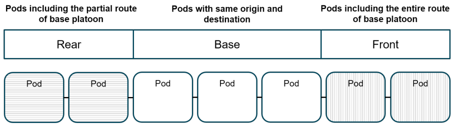

Fig. 3. Overlap-level-based platooning

Algorithm 1: Heuristic for Carrier Scheduling

```
Data: Transport unit (TU) data (origin/destination/due time),
      Initial carrier data (starting time, location)
Result: Carrier schedule
$P^{unpl} \leftarrow$ A list of Pods sorted in ascending order of due time
Initialize $loc_c$ and $time_c, \forall c \in C$, to the given carrier location and starting time
$tt \leftarrow GenerateTimetable()$
while $P^{unpl}$ is not empty do
    $sp \leftarrow P^{unpl}[0]$
    ETA $\leftarrow$ Calculate the estimated time of arrival for all carrier $c \in C$ to $o_{sp}$
    $pl^p, pl^c \leftarrow$ GeneratePlatoon($P^{unpl}$, $sp$, ETA)
    $P^{unpl} \leftarrow P^{unpl} \setminus pl^p$
    for each Pod $p \in pl^p$ and carrier $c \in pl^c$ do
        if $loc_c \neq o_p$ then
            $start \leftarrow$ Get start time from $tt$
            Update $tt$ by adding the schedule for movement from $loc_c$ to $o_p$
            Update $loc_c$ and $time_c$
        end
        Update $tt$ by adding the schedule for pick-up at $o_p$
        Update $time_c$
    end
    if $|pl^p| == 1$ then
        $start \leftarrow$ Get start time from $tt$
        Update $tt$ by adding the schedule for movement from $o_{sp}$ to $d_{sp}$
        Update $loc_c$ and $time_c$, where $c = pl^c$
    else
        $latest \leftarrow \max(time_c), \forall c \in pl^c$
        for each Pod $p \in pl^p$ and carrier $c \in pl^c$ do
            $wait \leftarrow latest - time_c$
            if $wait > 0$ then
                Update $tt$ by adding the schedule for waiting for platooning at $o_{sp}$
                Update $time_c$
            end
            Update $tt$ by adding the schedule for platooning
            Update $time_c$
        end
        for each section $sct \in route_{o_{sp},d_{sp}}$ do
            if separation is not required then
                $start \leftarrow$ Get start time from $tt$
                Update $tt$ by adding the schedule for movement along with $sct$
                Update $loc_c$ and $time_c$, $\forall c \in pl^c$
            else
                $pl^p_s, pl^c_s \leftarrow$ Pods and carriers separated at $sct_1$
                $pl^p, pl^c \leftarrow pl^p \setminus pl^p_s, pl^c \setminus pl^c_s$
                Update $tt$ by adding the schedule for separation at $sct_{end}$
                for $p \in pl^p_s$ and $c \in pl^c_s$ do
                    $start \leftarrow$ Get start time from $tt$
                    Update $tt$ by adding the schedule for movement from $sct_{end}$ to $d_p$
                    Update $loc_c$ and $time_c$
                end
            end
        end
        for each Pod $p \in pl^p$ and carrier $c \in pl^c$ do
            $start \leftarrow$ Get start time from $tt$
            Update $tt$ by adding the schedule for movement from $d_{sp}$ to $d_p$
            Update $loc_c$ and $time_c$
        end
    end
    for each Pod $p \in pl^p$ and carrier $c \in pl^c$ do
        Update $tt$ by adding the schedule for drop-off at $d_p$
        Update $time_c$
    end
end
```

Function 1: GeneratePlatoon

```
Function GENERATEPLATOON ($P^{unpl}$, $sp$, $ETA$):
    $base \leftarrow$ A list of Pods $p$ that $o_p = o_{sp}$ and $d_p = d_{sp}$, where $p \neq sp$, $\forall p \in P^{unpl}$
    $cr_{all} \leftarrow$ Get a list of available carriers
    pl, cr $\leftarrow P^{unpl}.pop(), cr_{all}.pop()$
    $n \leftarrow \min(|base|, |cr_{all}|)$
    if $n > 0$ then
        pl.$append(base[: n])$
        cr.$append(cr_{all}[: n])$
        $P^{unpl}.remove(base[: n])$
        $cr_{all}.remove(cr_{all}[: n])$
    end
    $overlap_{all} \leftarrow []$
    $overlap_{part}$, counts $\leftarrow [], []$
    $i \leftarrow 0$
    for $i < \min(|P^{unpl}|, |cr_{all}|)$ do
        $p \leftarrow P^{unpl}[i]$
        if $route_{(o_p,d_p)}[0] == route_{(o_{sp},d_{sp})}[0]$ then
            if $route_{(o_{sp},d_{sp})} \subset route_{(o_p,d_p)}$ then
                $overlap_{all}.append(p)$
            else
                cnt $\leftarrow 0$
                for $j \leftarrow [0, 1, .., min(|route_{(o_p,d_p)}|, |route_{(o_{sp},d_{sp})}|)]$ do
                    if $route_{(o_p,d_p)}[j] == route_{(o_{sp},d_{sp})}[j]$ then
                        cnt $\leftarrow$ cnt + 1
                    else
                        break
                    end
                end
                $overlap_{part}.append(p)$
                counts.$append$(cnt)
            end
        end
    end
    if $|overlap_{all}| > 0$ then
        pl.$append(overlap_{all})$
        cr.$append(cr_{all}[: |overlap_{all}|])$
        $cr_{all}.remove(cr_{all}[: |overlap_{all}|])$
    end
    if $|overlap_{part}| > 0$ then
        $overlap_{part} \leftarrow$ Sort $overlap_{part}$ by ascending order of counts
        pl.$insert(0, overlap_{part})$
        cr.$insert(0, cr_{all}[: |overlap_{part}|])$
    end
    return pl, cr
```

the impact of fleet size by varying the number of TUs and carriers.

As shown in Table III and Figure 5, when the problem size is relatively small (i.e. $|P| \leq 50$), using platooning is not beneficial to reduce makespan. This is related to the waiting, coupling and decoupling time of Pods at origin and destination stations. However, as the problem size increases ($|P| \leq 400$), platooning reduces makespan compared to non-platooning case. This shows the effectiveness of platooning in better capacity utilization of the railway network.

Table IV and Figure 6 illustrate the impact of varying TU-carrier ratio, with a fixed $|P|$ of 200. When carrier availability is insufficient, typically ranging from 10% to 30% ($|P|/|C|$), platooning does not effectively reduce makespan. This limitation arises as TUs may need to be dispatched individually to

TABLE II

PARAMETERS FOR EXPERIMENTS

Parameters	value
Speed (km/min)	2.5
Service time (pickup and delivery, min)	1
Safety margin (headway, km)	0.1
Carriers' waiting time threshold (min)	15
Platooning time (min)	2
Separation time (min)	1
Maximum platoon size	10
Time horizon (min)	100

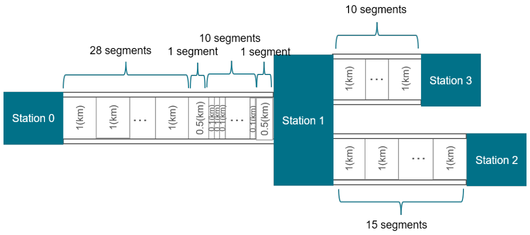

Fig. 4. Railway network case, Miao [22]

TABLE III

PLATOONING IMPACT OVER PROBLEM SIZE (CARRIER-TU RATIO: 50%)

		20,10	50,25	100,50	200,100	300,150	400,200
	No. of platoons	5	12	16	26	36	46
Platoon	Makespan	00:55:12	01:01:38	01:07:21	01:12:00	01:18:50	01:27:12
	u_{capa}	2.42%	4.25%	7.63%	10.71%	16.28%	19.51%
	No. of platoons	0	0	0	0	0	0
WithoutPlatoon	Makespan	00:51:45	00:56:36	00:57:26	01:12:40	01:24:00	01:41:09
	u_{capa}	3.30%	7.92%	15.25%	27.35%	38.72%	49.50%

TABLE IV

PLATOONING IMPACT OVER CARRIER-TU RATIO) $[|P| = 200]$

		$\|C\|/\|P\|$=10% ($\|C\|$=20)	$\|C\|/\|P\|$=30% ($\|C\|$=60)	$\|C\|/\|P\|$=50% ($\|C\|$=100)	$\|C\|/\|P\|$=100% ($\|C\|$=200)
	No. of platoons	30	26	26	24
Platoon	Makespan	03:53:14	01:36:38	01:12:00	01:08:12
	u_{capa}	13.93%	12.25%	10.71%	11.95%
	No. of platoons	0	0	0	0
WithoutPlatoon	Makespan	02:55:02	01:17:07	01:12:40	01:12:40
	u_{capa}	26.47%	27.21%	27.35%	27.35%

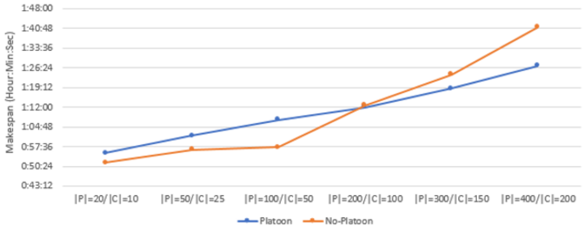

Fig. 5. Makespan over problem size

TABLE V

IMPACT OF SERVICE TIME(PICKUP AND DELIVERY) (MIN)) $[|P|, |C|]$

service time(min)		1	2	3	4	5
	No of platoons	0	0	0	0	0
Withoutplatoon	Makespan	01:12:40.80	01:12:40.00	01:26:21.60	01:28:40.80	01:33:57.60
	Capacity utilization	27.35%	28.16%	28.75%	29.11%	29.26%
	No of platoons	26	26	26	27	24
Platoon	Makespan	01:12:00.00	01:27:36.00	01:33:36.00	01:28:31.20	01:32:38.40
	Capacity utilization	10.71%	12.69%	13.71%	15.55%	15.40%

meet waiting thresholds. However, when carrier availability improves, ranging from 50% to 100% ($|P|/|C|$), platooning can offer benefits despite the associated overhead times.

The impact of service time is summarized in Table V: as the service time increases, a higher capacity utilization and a longer makespan are observed for both platoon and non-platoon scenarios, while the number of formed platoons remains almost constant. Interestingly, when service time is limited to 1 minute, the makespan of the non-platoon scenario becomes nearly identical to that of the platooning scenario, albeit with lower capacity utilization. This suggests that any drawbacks associated with platooning (which would have a trend of increasing the makespan) could be mitigated by reducing the service time. Furthermore, increasing the train speed yields a similar trend (see Table VI) in terms of capacity utilization and makespan for both platoon and non-platoon scenarios. However, altering speed may lead to slight fluctuations in the number of platoons, whereas adjusting the service time maintains a constant number. This discrepancy arises as changes in speed can affect the scheduling of moving infrastructures. To include different design parameters, Figure 7 shows the impact of the lower speed (=60km/hour) and larger safety margins (0.8 km) on the makespan. It indicates that makespan is approximately three times longer than the origianl experiment (Figure 5). The break-point between platooning and no-platooning scenarios in terms of makespan is also obtained in lower problem sizes compared to the original experiment. As a final note, when comparing the platooning and non-platoon scenarios in all cases, one can observe an increased efficiency in capacity utilization. While there is an overall decreasing trend in makespan in platooning cases, minor fluctuations are observed, which can be attributed to a combination of factors that are crucial in the analysis of makespan. As shown in Figure 8, larger problem sizes and longer platooning/separation times result in increased makespans.

VII. CONCLUSIONS

This paper introduces an innovative and disruptive autonomous, intermodal, and modular transport system. Through a heuristic algorithm and a thorough sensitivity analysis, it models the operational concept of the system

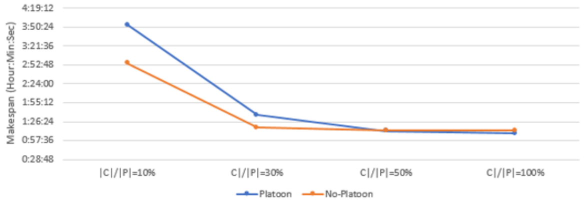

Fig. 6. Makespan over carrier-TU ratio

TABLE VI

IMPACT OF SPEED (KM/MIN)) $[|P|, |C|]$

speed(km/min)		1	1.5	2	2.5	3	3.5
	No of platoons	0	0	0	0	0	0
Withoutplatoon	Makespan	02:51:06.00	01:55:48.00	01:27:36.00	01:12:40.80	01:00:46.00	00:52:20.57
	Capacity utilization	67.10%	44.49%	33.73%	27.35%	22.79%	19.70%
	No of platoons	25	23	24	26	26	24
Platoon	Makespan	2023-12-01 02:35:24.00	01:48:20.00	01:26:36.00	01:12:00.00	01:01:00.00	00:52:17.14
	Capacity utilization	23.83	16.74%	13.48%	10.71%	9.11%	7.96%

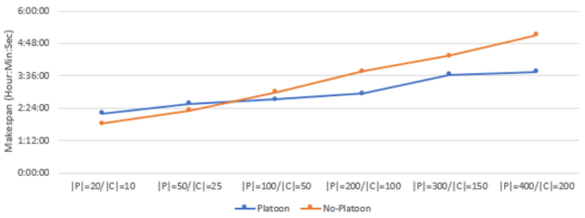

Fig. 7. Platooning impact over different parameter values

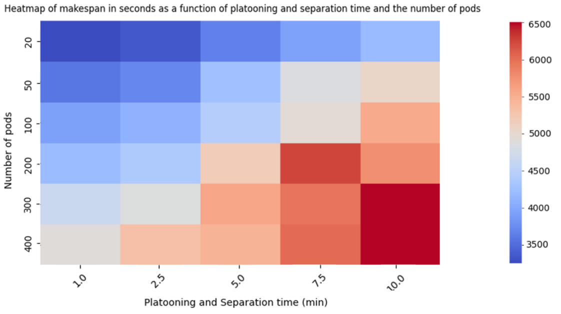

Fig. 8. Heatmap: impact of platooning/separation time and problem size on makespan

and studies the impact of dispatching and the platooning decisions on its performance. Results show that platooning can be useful in terms of reducing makespan when the problem size is relatively large. Also, the number of carriers needs to be sufficient to make the platooning effective. Future work will address adding complementary modules such logistic and fleet management, detailed railway scheduling algorithm, and extending the network size.

ACKNOWLEDGMENT

This project has received funding from the Europe's rail Joint Undertaking (JU) under GA No. 101121853. It is supported by the Europe's Rail JU and its members funded by the European Union. Views and opinions expressed are those of author(s) only and do not necessarily reflect those of the European Union or the Europe's Rail Joint Undertaking. Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

[1] Inboundlogistics.(2022) Available at: https://www.inboundlogistics.com/articles/intermodal-transportation/

[2] Eurostat. (2022). Railway freight transport statistics.

[3] Pods4Rail. (2024, January). Pods4Rail. Available at: https://Pods4rail.eu/.

[4] Lin, J., Nie, Y. M., & Kawamura, K. (2022). An autonomous modular mobility paradigm. IEEE Intelligent Transportation Systems Magazine, 15(1), 378-386.

[5] Hatzenbühler, J., Jenelius, E., Gidófalvi, G., & Cats, O. (2023). Modular vehicle routing for combined passenger and freight transport. Transportation Research Part A: Policy and Practice, 173, 103688.

[6] Fu, Z., & Chow, J. Y. (2023). Dial-a-ride problem with modular platooning and en-route transfers. Transportation Research Part C: Emerging Technologies, 152, 104191.

[7] Liu, L., Wang, P., Zhang, B., & Wei, W. (2019). Coordinated control method of virtually coupled train formation based on multi-agent system. Advances in Smart Vehicular Technology, Transportation, Communication and Applications: Proceedings of the Second International Conference on Smart Vehicular Technology, Transportation, Communication and Applications, October 25-28, 2018, Mount Emei, China, Part 1 2 (pp. 225-233). Springer International Publishing.

[8] Quaglietta, E., Wang, M., & Goverde, R. M. (2020). A multi-state train-following model for the analysis of virtual coupling railway operations. Journal of Rail Transport Planning & Management, 15, 100195.

[9] Ning, Z., Ou, D., Zhang, L., (2022). Virtual Coupling Formation Scheduling Method for Y-type Rail Transit Lines. In 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC (pp. 3673-3678). Macau, China. doi: 10.1109/ITSC55140.2022.9922029.

[10] Chai, S., Yin, J., D'Ariano, A., Liu, R., Yang, L., & Tang, T. (2024). A branch-and-cut algorithm for scheduling train platoons in urban rail networks. Transportation Research Part B: Methodological, 181, 102891.

[11] Chai, S., Yin, J., D'Ariano, A., Samà, M., & Tang, T. (2023). Scheduling of coupled train platoons for metro networks: A passenger demand-oriented approach. Transportation Research Record, 2677(2), 1671-1689.

[12] Chen, Z., & Li, X. (2021). Designing corridor systems with modular autonomous vehicles enabling station-wise docking: Discrete modeling method. Transportation Research Part E: Logistics and Transportation Review, 152, 102388.

[13] Chen, Z., Li, X., & Qu, X. (2022). A continuous model for designing corridor systems with modular autonomous vehicles enabling station-wise docking. Transportation Science, 56(1), 1-30.

[14] Shi, X., Chen, Z., Pei, M., & Li, X. (2020). Variable-capacity operations with modular transits for shared-use corridors. Transportation Research Record, 2674(9), 230-244.

[15] Shi, X., & Li, X. (2021). Operations design of modular vehicles on an oversaturated corridor with first-in, first-out passenger queueing. Transportation Science, 55(5), 1187-1205.

[16] Schumann, T. (2016). Increase of capacity on the Shinkansen high-speed line using virtual coupling.

[17] Ling, LIU, Ping, WANG, Ying, WANG, Wei, WEI, & Zhoufan, WANG. (2019). Dynamic train formation and dispatching for rail transit based on virtually coupled train set. 2019 IEEE Intelligent Transportation Systems Conference (ITSC) (pp. 2823–2828). IEEE.

[18] Cao, Y., Wen, J., & Ma, L. (2021). Tracking and collision avoidance of virtual coupling train control system. Future Generation Computer Systems, 120, 76–90.

[19] Bhoopalam, A. K., Agatz, N., & Zuidwijk, R. (2018). Planning of truck platoons: A literature review and directions for future research. Transportation Research Part B: Methodological, 107, 212-228.

[20] Liatsos, V., Golias, M., Hourdos, J., & Mishra, S. (2024). The capacitated hybrid truck platooning network design problem. Transportation Research Part A: Policy and Practice, 181, 103999.

[21] Boysen, N., Briskorn, D., & Schwerdfeger, S. (2018). The identical-path truck platooning problem. Transportation Research Part B: Methodological, 109, 26-39.

[22] Miao, R. (2022). A capacity-effective scheduling approach for virtual coupling train operations.

[23] Parise, R., Dittus, H., Winter, J., Lehner, A. (2019). Reasoning Functional Requirements for Virtually Coupled Train Sets: Communication. IEEE Communications Magazine, Vol. 57, No. 9, 12-17.

[24] Schöne, S., Schumann, T. (2018). Next Generation Train: Dynamisches Flügeln, Beschreibung eines Betriebskonzepts auf Basis der zukünftigen LST, MS 2420 1712. DLR-Forschungsbericht. DLR. Deutsches Zentrum für Luft- und Raumfahrt e. V. 27 S.

[25] Stickel, S., Schenker, M., Dittus, H., Unterhuber, P., Canesi, S., Riquier, V., Parrilla, F., Berbineau, M., Goikoetxea, J. (2022). Technical feasibility analysis and introduction strategy of the virtually coupled train set concept. Scientific Reports. Nature Publishing Group. doi: 10.1038/s41598-022-08215-y. ISSN 2045-2322.

[26] Flamm, L., Mönsters, M., Weik, N. (2023). Increasing Capacity and Reducing Travel Times in High-speed Train Operations through Virtual Coupling. In: Proceedings of the Fifth International Conference on Railway Technology: Research, Development and Maintenance, 1, 13.3. Civil-Comp Press. Railways 2022.