Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 91 (2025) 329–336

Transportation Research Procedia

www.elsevier.com/locate/procedia

The Science and Development of Transport - TRANSCODE 2025

Passenger centred Transfer Coordination in disturbed Public Transport using Reinforcement Learning

Lukas Hösch[a,1], Robert Alms[a]

[a]German Aerospace Center, Institute of Transport Systems, Rutherfordstr. 2, 12489 Berlin

Abstract

Public transport has proven as an efficient and ecological solution to provide mobility to a large part of the society. Disturbances can impede public transport operation and reduce user acceptance. Current disturbance management strategies primary focus on minimising impacts on operational planning deeming passengers' needs less important. As a development step towards passenger-centred intermodal disturbance management system, we present a minimal reinforcement learning example for transfer coordination. The environment builds up on the microscopic traffic simulator SUMO. While we consider reinforcement learning as a valuable approach to find novel solutions to passenger-centred disturbance management in larger setups, the goal of this work is to demonstrate the applicability of this method. To this end, it is applied in a small public transport network, demonstrating comparable performance to that of a deterministic transfer coordination strategy.

Keywords: Passenger-centred transfer coordination; Reinforcement Learning; SUMO; Public transport resilience

1. Introduction

Public transport (PT) is an efficient and ecological way to ensure the mobility of a possibly large part of the population especially in urban environments. Unfortunately, PT journeys are exposed to risks of disturbances. In case feeding vehicles are delayed, planned interchanges can break, resulting in increased passenger travel time. Many existing works focus on the optimisation of operational metrics during major disruptions Hayat (2010); Pender et al. (2013). In these events, disturbance mitigation can become a challenging task due to the extent of the disruption (e.g., mitigation options are limited if an entire network section is flooded). Furthermore, passengers' concerns during minor disturbances (e.g., missed connections at interchanges) are treated less important compared to operational impacts. Still, even such disruptions

[1] Corresponding author. Tel.: +49 30670 558-061; E-mail address: lukas.hoesch@dlr.de

2352-1465
10.1016/j.trpro.2025.10.043

can affect user experience and lead to lower acceptance rate Rocha et al. (2023), mitigating the efficiency of PT.

The challenge of intermodal PT journeys Babić et al. (2022) can be addressed by a passenger-centric intermodal disturbance management system (PC-IDMS). For the development of such PC-IDMS, regular intermodal transport control system (ITCS) data are enriched with measured real-time passenger locations via Bluetooth Low Energy (BLE) Hösch et al. (2024). Knowing passengers' locations and destinations in real-time is key to account for individual travel plans during disturbance management. By introducing Reinforcement Learning (RL) into PC-IDMS we want to develop it as an assistant system, that expands the action space of human dispatchers while keeping their workload reasonable. In this work, we address three main challenges:

1. Firstly, passengers' real-time locations, combined with ITCS, generate large volumes of data. RL is a feasible approach for such data-intensive problems, as it prioritizes frequently occurring states while downplaying rare ones – a principle proven effective in other applications like Vrbanić et al. (2023); Gregurić et al. (2020).
2. We aim to find novel, innovative solutions relevant to PC-IDMS that have not been considered by human dispatchers.
3. Finally, in this work, we focus on classical regular PT service Sommer and Saighani (2019), where vehicles operate independently from the current demand. Therefore, we aim to answer the question whether it is feasible to construct a minimal RL example (environment and agent), that can achieve comparable performance with respect to deterministic approaches in a simple transfer coordination task.

The remainder of this paper is organised as follows. Section 2 provides an overview of recent research in the field of RL and PT disturbance management. Section 3 presents the methodology of the minimal RL example. Results from this setup in a controlled simulation environment are provided in section 4 and discussed in section 5. Finally, section 6 provides concluding remarks and an outlook for future work.

2. State of the art

Integrating passengers' interests to increase the PT service level has gathered substantial interest over the past few years. For example, Duarte et al. (2021) show that service level increases can be achieved by efficient maintenance and management that is enhanced through customer participation. Customer satisfaction can also be improved by increasing PT resilience. Methods to improve schedule reliability are summarised in the concept suggested by Santos et al. (2020). Another approach consists of real-time operational control strategies. From the plethora of control strategies presented by Gkiotsalitis et al. (2023), we apply transfer coordination. As stated by Corman et al. (2012); Dollevoet and Huisman (2014), minimising both passengers' missed transfers and vehicle delays in case of disturbance can become a complex problem in real-world scenarios.

Vehicle holding is one of the most common control strategies for minor disturbances Desaulniers and Hickman (2007). Generally, a holding strategy increases passenger travel time, so other operational control means are necessary to mitigate this effect Rezazada et al. (2024). It can either be headway-based (aiming for even headways) Cats et al. (2012) or schedule-based (aiming for schedule adherence) Berrebi et al. (2018).

RL is a machine learning paradigm for decision making under uncertainty. Applying it to PT disturbance management requires the formulation of a Markov decision process (MDP) Yu et al. (2022). In comparison to existing, rule-based approaches, RL can deploy real-time vehicle holding in a computationally more efficient manner while considering network-wide effects Alesiani and Gkiotsalitis (2018). RL also proved to be effective for online bus scheduling in cases when offline schedules become infeasible Liu et al. (2023). Similarly, RL can adapt existing PT networks to changed demand conditions Yoo et al. (2023). Many real-time operational control strategies, such as Wang (2023); Low et al. (2024) rely on multi-agent, rather than single-agent RL. An essential feature of PC-IDMS is to incorporate passengers' concerns. Modelling passenger satisfaction

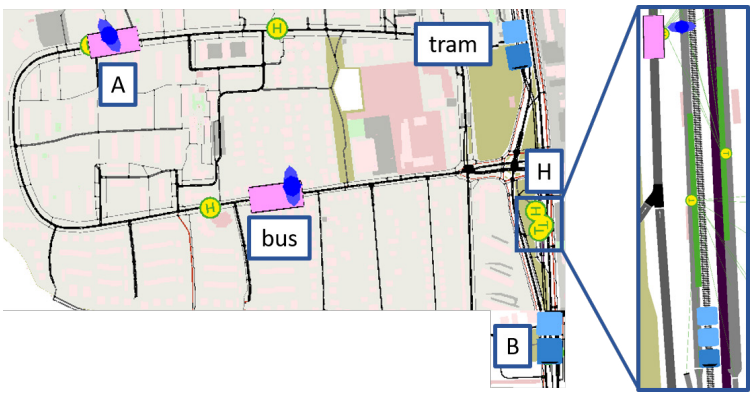


Fig. 1. SUMO implementation of the disturbance situation. A small delay of the feeding vehicle (pink bus) causes the passenger travelling from A to miss the connecting vehicle (blue tram) during the interchange in H. Therefore, the destination B cannot be reached.

Table 1. Travel itinerary of base line scenario: the connecting vehicle (tram) leaves the interchange hub H on time. Therefore, the passenger misses the connection at the interchange and cannot reach the destination B.

Location	Time [min]	Vehicle	Annotation
A	01:00	Bus (Feeding)	Journey start
H	09:00	Bus (Feeding)	Transfer at local hub
H	09:00	Tram (Connecting)	Connection lost
B	10:00	Tram (Connecting)	Tram on time

as MDP has been addressed by Chu and Guo (2023) in a single-agent and by Vikharev et al. (2019) in a multi-agent RL system.

Deep neural networks approximating non-linear functions extend RL to deep RL Sutton (2018). Deep RL can effectively apply holding control at an initial bus stop to dispatch buses according to the real-time passenger demand Zhao et al. (2022). Several works rely on deep RL to model lane changing for autonomous cars driving in mixed traffic. Activities focus on comparison to other algorithms Yao et al. (2024) and different components of the reward function Karalakou et al. (2023). Modelling these multi-agent systems with factor graphs provides a more realistic communication framework for asynchronous decision making Troullinos et al. (2025). Integrating crowd-shipping tasks into a centralised vehicle network, deep RL can outperform existing approaches Rodriguez et al. (2023).

3. Methodology

To demonstrate the feasibility of RL in the context of PC-IDMS, we implement the minimal example for transfer coordination depicted in Fig. 1. Table 1 provides a synthetic travel itinerary of the passenger in the baseline situation (without transfer coordination). We assume a small delay of the bus, so the passenger misses the transfer to the tram, if it leaves on time. We built this scenario in the simulator SUMO (Simulation of Urban Mobility, Lopez et al. (2018)). To keep computational requirements low, this SUMO scenario is limited to the three mentioned entities (two PT vehicles and one person).

3.1. Reinforcement Learning Task

We define the problem as an episodic task. Initially, passenger and bus are at location A. An episode terminates if the passengers successfully reaches location B. If the passenger misses the connection at H, B cannot be reached and the episode ends after 900s (max. number of time-steps). The initial state of the

next episode does not depend on the terminal state of the current one. We model the dispatcher of the local transport provider as a single RL agent. The agent's goal is to learn the optimal departure time of the connecting vehicle, that minimises the passenger's travel time.

3.2. Markov Decision Problem

The MDP has the properties state, actions and reward $\langle S, A, R \rangle$ Sutton (2018). For S, we assign each of the $k = 23$ PT stops inside the SUMO network with a unique integer value. The discrete state space consists of the passenger's p and the tram's t next stop.

$$S = \{(p,t) | p \in \{0,1,...,N-1\}, t \in \{0,1,...,k-1\}\} \tag{1}$$

The discrete action space is defined by two possible actions: Action 0 causes the tram to leave H immediately. Action 1 postpones the tram's departure, imposing a holding interval of 60 s as the shortest time entity used for scheduling (PT schedules usually don't use time information in the order of seconds).

$$A = \{0,1\} \tag{2}$$

Actions can only be selected if the tram has already entered the network and has the connecting stop H downstream. Since too frequent action changes can introduce noise, we add an action skip interval $i_{a,s}$, which restricts the agent to select actions only at discrete time-steps after $i_{a,s}$. We define the reward function as follows.

$$R = \begin{cases} 200 + t_{s,r} & \text{passenger arriving at B} \\ -100 & \text{passenger waiting at H on episode termination} \\ t_{n,i} - t_{n,w} + d_{n,i} & \text{else} \end{cases} \tag{3}$$

Parameter $t_{s,r}$ is the remaining simulation time, $t_{n,i}$ the normalised interchange time, $t_{n,w}$ the waiting time, and $d_{n,i}$ the normalised walking distance between passenger and connecting vehicle at the interchange hub. Introducing $t_{s,r}$ into the first case encourages the agent to minimise the delay of the connecting vehicle even if the connection is ensured.

3.3. Reinforcement Learning Agent

We implement a tabular actor-critic RL agent as introduced in (Sutton, 2018, chap. 13.5), outlined in algorithm 1. The combination of actor and critic enables learning a policy (regulation to map states to actions, learned by the actor) and the value function (assigning a value to each state, learned by the critic) at the same time. As the state space is relatively small and discrete (size= 23×23), policy and value functions can be represented directly instead of using function approximation. Finally, the transition function P depicts how the agent's actions affect the state. In our case, P is governed by the behaviour of the SUMO simulation. The interaction to the environment is realised via the TraCI API.

4. Results

We train the RL agent for 200 episodes. The computation time on a 13th Gen Intel(r) Core(TM) i7-1370P CPU amounts to 4 minutes. We choose a random seed value at

every new episode. The passenger's walking speed is adjusted to a value between 2 and 7 km/h following a uniform random distribution at every time-step. Traffic lights operate in fixed phases. We parametrise the agent by setting both actor α^{θ} and critic $\alpha^{\mathbf{w}}$ learning rates to 0.1. The action skip interval $i_{a,s} = 10s$ was deemed to be the most promising value from the initial experiment set of {10, 30, 60} s.

Fig. 2 depicts the return (sum over all rewards in one episode). All return values are smoothed over 20 episodes. Because the RL training process can be highly stochastic, we average the return over 100 iterations. The figure illustrates the lower return limit of a successful episode at 72 (dotted line) and the upper return limit of an unsuccessful episode at −168 (dashed line). Within the first 25 episodes, returns are low with narrow deviation but improve quickly. In most episodes, returns keep rising until they reach a plateau at a mean return value of 100 at episode 75. We also observe an increasing standard deviation from episode 25 onwards. Overall, 78 training episodes terminate successfully resulting in a mean final return of 111.

Algorithm 1 Actor–Critic Update Algorithm

Input: policy $\pi(a|s, \boldsymbol{\theta})$
Input: value estimate $\hat{v}(s, \boldsymbol{w})$
for each episode do
  Initialise S, $c_s \leftarrow 0$
  while S not terminal do
    $c_s \leftarrow c_s + 1$
    if $c_s = i_{a,s}$ then
      $c_s \leftarrow 0$
      $A \sim \pi(\cdot|S, \boldsymbol{\theta})$
      Take A, observe S', R
      $\delta \leftarrow R + \hat{v}(S', \boldsymbol{w}) - \hat{v}(S, \boldsymbol{w})$
      $\boldsymbol{w} \leftarrow \boldsymbol{w} + \alpha^{w} \cdot \delta$
      $\boldsymbol{\theta} \leftarrow \alpha^{\theta} \cdot \delta$
      $S \leftarrow S'$
    end if
  end while
end for

After training, we test the RL agent by performing 100 episodes in the training setup. We compare the performance of the RL agent to a deterministic approach, that holds the tram for 119 s after the arrival of the bus. Fig. 3a shows the departure delay of the tram at H.

The RL agent's transfer coordination strategy results in a departure delay between 89 and 123 s with a median value of 100 s. The dashed line indicates the departure delay assigned by the deterministic policy. We measure the passenger travel time as the time between beginning the trip at A and terminating it at B. At all knots (A, B and H), the passenger has to cover short walking distances, that influence the travel time. Fig. 3b depicts the box plots of the passenger travel time for both (RL and deterministic) transfer

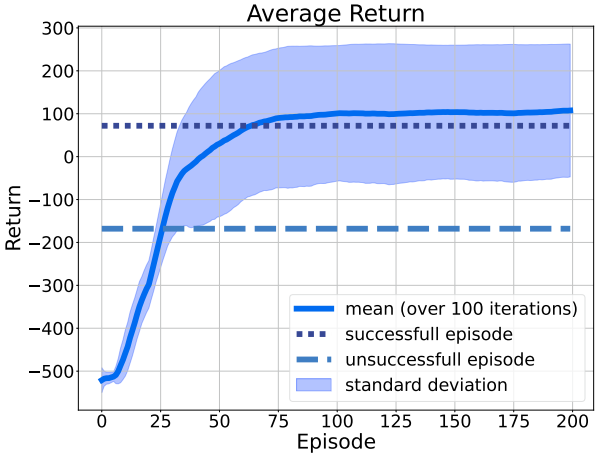
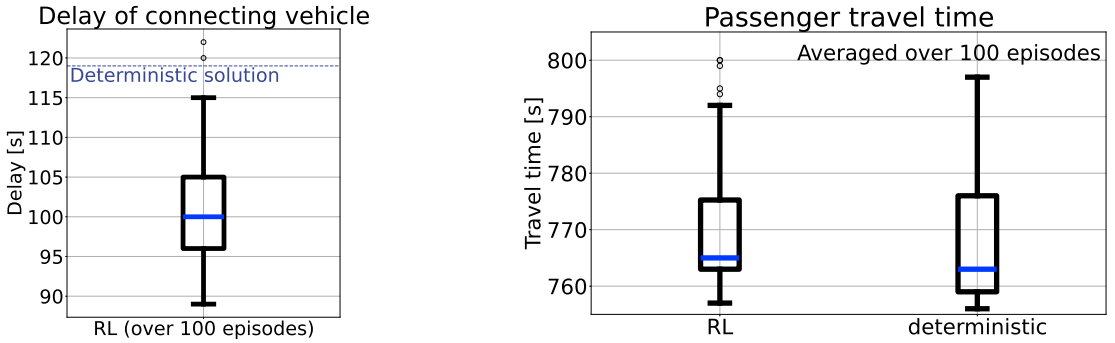


Fig. 2. Average return (sum of all rewards in one episode) over all 200 training episodes. Values are averaged over 100 iterations.

coordination strategies. Surprisingly, despite the higher departure delay, the deterministic approach achieves a slightly lower median passenger travel time than the RL approach. Both box plots are skewed positively (the majority of values is low). The median values of the deterministic (765 s) and the RL (769 s) differ only slightly.



(a) Departure delay of the connecting vehicle (tram) over 100 evaluation episodes.

(b) Passenger travel time from A to B, influenced by walking speed. The deterministic method may outperform the RL agent.

Fig. 3. Evaluation of RL-based transfer coordination based on departure delay (panel a) and passenger travel time (panel b).

5. Discussion of the results

5.1. Training progress

The increasing standard deviation in Fig. 2 from episode 25 onwards indicates that in some training runs, the RL agent learned a suboptimal policy. This suboptimal policy holds the tram until the end of an episode, so the passenger cannot reach the destination. The policy can surpass the lowest possible return, because it maximises the immediate reward during the interchange (see third case in equation 3). It remains suboptimal because the final goal is never reached. Such behaviour is a risk resulting from reward shaping (rewarding intermediate goals). It could potentially be counteracted by discounting future rewards Sutton (2018), which leads the agent to focus more on long-term return rather than immediate reward. Adapting the reward function to sparse rewards only (first two cases in equation 3) would decrease sample efficiency though.

An optimal policy can be learned in 78% of all training runs within 200 episodes (4 minutes computation time on a CPU). We conclude, that the RL agent was able to efficiently handle the data volume and find an optimal solution in most cases. Since states in unseen events were not part of the evaluation, we cannot assess the RL agent's capabilities to generalise to states that have not been part of the training process. With a potentially more complex RL environment, also computational requirements will increase in future implementations.

In particular, the reward function should be designed to model the RL agent's goal while still leaving room to explore novel solutions. "Thinking outside the box" is one of the key motivations to apply RL to PC-IDMS. Since the problem is limited in this minimal example, RL did not find novel solutions to the transfer coordination problem. Instead, the strict boundaries of the RL environment were designed to demonstrate the capabilities of the RL agent to learn a conventional transfer coordination strategy. We thus leave the opportunity of finding novel, innovative solutions to future RL approaches relevant to PC-IDMS.

5.2. PC-IDMS evaluation

From the almost symmetric appearance of the box plot in Fig. 3a, we conclude that the RL agent learned to adapt the tram's departure time to the (uniform random) walking speed of the passenger. We can see that the RL agent outperforms the deterministic solution in almost all tested cases. This strong performance results, inter alia, from an information advantage of the RL agent. The reward function 3 includes the distance between passenger and tram at H. This information is not part of the deterministic solution. Still, it is impressive that the RL agent can learn a policy in 200 episodes, that can minimise the tram's delay to only 75% in some cases, compared to the deterministic approach.

The lower passenger travel time achieved by the deterministic approach in Fig. 3b is governed by a traffic light in front of the destination B. Following a fixed-time phase plan, this traffic light usually shows green when the tram approaches according to deterministic transfer coordination. The tram can then pass the traffic light with maximum velocity, minimising its travel time between H and B. In contrast, the earlier departure time in H assigned by the RL agent causes the tram to stop at the traffic light. Waiting for the green phase and accelerating again requires more time (in the order of seconds) than passing the traffic light with maximum velocity. The large extent of the upper whisker in the deterministic approach results from the random walking speed of the passenger. Despite the slightly higher travel time achieved by the RL approach, we can conclude that the RL training was successful. The positive skew of the RL box plots indicates that the RL agent learned that a low passenger travel time is desirable.

5.3. Limitations of the simulation environment

In this minimum example, we assigned a random walking speed at each time-step to the passenger to introduce randomness. In practice, this is not a realistic approach. Instead, more realistic walking speeds could be assigned at the beginning of an episode following a Gaussian distribution. An action skip interval of $i_{a,s} = 10s$ is lower than the time intervals most transport providers adapt their real-time operational means in. In practice it would not be feasible to send (possibly contradictory) operational instructions to the PT drivers in an interval of 10 s. While the main idea of PC-IDMS is to integrate passengers' concerns into real-time disturbance management, this minimal example ignores this specific concern of transport providers. Consequently, to make the presented framework applicable to multiple passengers and vehicles, the state representation (equation 1) and reward function (equation 3) would need significant adaptions. To evaluate the generalisation ability of the RL agent, additional testing in an unseen environment would be necessary.

Despite the pointed out shortcomings, the presented RL framework achieved decent results already after 200 training episodes and outperforms a deterministic transfer coordination strategy. By demonstrating comparable performance of RL in this minimal example, we are confident to find relevant solutions for even more complex problems in the future.

6. Conclusion and Outlook

With this minimal example, we substantiated our claim that RL can find a transfer coordination strategy which outperforms a deterministic approach. Though this achievement partially results from an information advantage, we demonstrated that RL is a suitable method for PC-IDMS. The RL agent learned an optimal policy in 78% of the training runs. The implementation could be adapted towards scalability to larger networks and a more realistic simulation environment. Also, more operational means could be incorporated. With these features, RL can find innovative solutions to PC-IDMS, which can minimise disturbance effects from passengers' point of view and contribute to passenger satisfaction and user acceptance in PT.

Acknowledgements

This work results from the joint research project STADT:up (grant number 19A22006x). The project is supported by the German Federal Ministry for Economic Affairs and Climate Action (BMWK), based on a decision of the German Bundestag.

References

Alesiani, F., Gkiotsalitis, K., 2018. Reinforcement learning-based bus holding for high-frequency services, in: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), IEEE. pp. 3162–3168.

Babić, D., Kalić, M., Janić, M., Dožić, S., Kukić, K., 2022. Integrated door-to-door transport services for air passengers: from intermodality to multimodality. Sustainability 14, 6503.

Berrebi, S., Hans, E., Chiabaut, N., Laval, J.A., Leclercq, L., Watkins, K., 2018. Comparing bus holding methods with and without real-time predictions. Transportation Research Part C: Emerging Technologies 87, 197–211.
Cats, O., Larijani, A., Ólafsdóttir, Á., Burghout, W., Andréasson, I., Koutsopoulos, H., 2012. Bus-holding control strategies: simulation-based evaluation and guidelines for implementation. Transportation Research Record 2274, 100–108.
Chu, K.F., Guo, W., 2023. Deep reinforcement learning of passenger behavior in multimodal journey planning with proportional fairness. Neural Computing and Applications 35, 20221–20240.
Corman, F., D'Ariano, A., Pacciarelli, D., Pranzo, M., 2012. Bi-objective conflict detection and resolution in railway traffic management. Transportation Research Part C: Emerging Technologies 20, 79–94.
Desaulniers, G., Hickman, M., 2007. Public transit. Handbooks in operations research and management science 14, 69–127.
Dollevoet, T., Huisman, D., 2014. Fast heuristics for delay management with passenger rerouting. Public Transport 6, 67–84.
Duarte, S.P., Campos Ferreira, M., Pinho de Sousa, J., Freire de Sousa, J., Galvão, T., 2021. Improving mobility services through customer participation, in: Advances in Mobility-as-a-Service Systems: Proceedings of 5th Conference on Sustainable Urban Mobility, Virtual CSUM2020, June 17-19, 2020, Greece, Springer. pp. 654–663.
Gkiotsalitis, K., Cats, O., Liu, T., 2023. A review of public transport transfer synchronisation at the real-time control phase. Transport reviews 43, 88–107.
Gregurić, M., Vujić, M., Alexopoulos, C., Miletić, M., 2020. Application of deep reinforcement learning in traffic signal control: An overview and impact of open traffic data. Applied Sciences 10, 4011.
Hayat, S., 2010. Disturbances modelling for improving the traffic operation for the public transport networks. IFAC Proceedings Volumes 43, 644–650.
Hösch, L., Käthner, D., Christ, T., 2024. Creating a traveller digital twin: Bluetooth low energy for fine-granular traveller localisation. Advances in Transdisciplinary Engineering 27, 284.
Karalakou, A., Troullinos, D., Chalkiadakis, G., Papageorgiou, M., 2023. Deep reinforcement learning reward function design for autonomous driving in lane-free traffic. Systems 11, 134.
Liu, Y., Zuo, X., Ai, G., Liu, Y., 2023. A reinforcement learning-based approach for online bus scheduling. Knowledge-Based Systems 271, 110584.
Lopez, P.A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., Wießner, E., 2018. Microscopic traffic simulation using sumo, in: The 21st IEEE International Conference on Intelligent Transportation Systems, IEEE. URL: https://elib.dlr.de/124092/.
Low, V.J.M., Khoo, H.L., Khoo, W.C., 2024. Robust dynamic real-time control strategies for high-frequency bus service: a multi-agent reinforcement learning framework. Journal of Intelligent Transportation Systems , 1–20.
Pender, B., Currie, G., Delbosc, A., Shiwakoti, N., 2013. Disruption recovery in passenger railways: International survey. Transportation research record 2353, 22–32.
Rezazada, M., Nassir, N., Tanin, E., Ceder, A., 2024. Bus bunching: a comprehensive review from demand, supply, and decision-making perspectives. Transport Reviews , 1–25.
Rocha, H., Filgueiras, M., Tavares, J.P., Ferreira, S., 2023. Public transport usage and perceived service quality in a large metropolitan area: the case of porto. Sustainability 15, 6287.
Rodriguez, J., Koutsopoulos, H.N., Wang, S., Zhao, J., 2023. Cooperative bus holding and stop-skipping: A deep reinforcement learning framework. Transportation Research Part C: Emerging Technologies 155, 104308.
Santos, V., Fontes, T., Ribeiro, J., de Sousa, J.P., de Sousa, J.F., 2020. A decision support system for collaborative management of multimodal public transport services. Proceedings of TRA2020, the 8th Transport Research Arena: Rethinking transport-towards clean and inclusive mobility .
Sommer, C., Saighani, A., 2019. 3.4. 8. 7 öpnv-angebotsformen im ländlichen raum. Handbuch der kommunalen Verkehrsplanung - 84. Ergänzungs-Lieferung 84, 1–29.
Sutton, R., 2018. Reinforcement learning: An introduction. A Bradford Book .
Troullinos, D., Chalkiadakis, G., Papamichail, I., Papageorgiou, M., 2025. Conditional max-sum for asynchronous multiagent decision making. arXiv preprint arXiv:2502.13194 .
Vikharev, S., Lyapustin, M., Mironov, D., Nizovtseva, I., Sinitsyn, V., 2019. Modeling of passengers' choice using intelligent agents with reinforcement learning in shared interests systems; a basic approach. Transport problems 14.
Vrbanić, F., Gregurić, M., Miletić, M., Ivanjko, E., 2023. Reinforcement learning-based dynamic zone positions for mixed traffic flow variable speed limit control with congestion detection. Machines 11, 1058.
Wang, J., 2023. Multi-agent Deep Reinforcement Learning for Public Transport Vehicle Fleet Control. Ph.D. thesis. McGill University.
Yao, X., Hou, S., Hoogendoorn, S.P., Calvert, S.C., 2024. Performance comparison of deep rl algorithms for mixed traffic cooperative lane-changing. arXiv preprint arXiv:2407.02521 .
Yoo, S., Lee, J.B., Han, H., 2023. A reinforcement learning approach for bus network design and frequency setting optimisation. Public Transport 15, 503–534.
Yu, X., Khani, A., Chen, J., Xu, H., Mao, H., 2022. Real-time holding control for transfer synchronization via robust multiagent reinforcement learning. IEEE Transactions on Intelligent Transportation Systems 23, 23993–24007.
Zhao, Y., Chen, G., Ma, H., Zuo, X., Ai, G., 2022. Dynamic bus holding control using spatial-temporal data–a deep reinforcement learning approach, in: Australasian Joint Conference on Artificial Intelligence, Springer. pp. 661–674.