

# Multi-Agenten KI-Ansatz für ein ganzheitliches Verkehrsmanagementsystem

Towards a holistic traffic management system  
with Multi-Agent AI

Stefan Schneider | Anne Roets | Ciprian Stirbu | Farhad Safaei | Faten Ghriess | Jan Wülfing | Mehmet Güral | Nima Siboni | Roman Liessner | Thomas Hustache | Thomas Lecat | Umashankar Deekshith | Valerii Markin | Victor Le | Wissam Bejjani | Michael Küpper | Irene Sturm

**E**in intelligentes Verkehrsmanagementsystem ist als „planendes Gehirn“ ein zentraler Baustein für einen hochautomatisierten Netz- und Fahrbetrieb. Der hier vorgestellte Ansatz wendet die Methode des Multi-Agent Reinforcement Learning (MARL), ein Teilgebiet der Künstlichen Intelligenz (KI), an. Eine KI agiert auf einer Eisenbahnsimulation und erstellt hochdetaillierte Ablaufpläne für den Zugverkehr. Im Vergleich zu klassischen Optimierungsmethoden wird deutlich kürzere Rechenzeit benötigt. Somit ist der Ansatz vielversprechend für eine Anwendung auf großen Netzbereichen.

## 1 Motivation

Für eine nachhaltige Mobilität muss künftig noch mehr Verkehr von der Straße auf die Schiene verlagert werden. Ohne tiefgreifende Innovationen ist jedoch attraktiver Bahnverkehr mit mehr Zügen in einer höheren Qualität auf dem Schienennetz nicht zu realisieren. Die Sektorinitiative „Digitale Schiene Deutschland“ (DSD) hat es sich zur Aufgabe gemacht, das Potenzial digitaler Technologien zu heben, um die Kapazität, Qualität und Effizienz des Bahnsystems zu steigern [1]. Ein Kernstück des neuen digitalen Bahnsystems ist ein intelligentes Verkehrsmanagementsystem („Capacity & Traffic Management System“, CTMS) für die vollautomatische Planung und Steuerung des Bahnbetriebs.

Die Erstellung von Fahrplänen ist schon im heutigen Bahnsystem mit 40 000 Zugfahrten täglich eine komplexe Aufgabe, die monatelange Vorbereitung von Experten mit Spezialsoftware erfordert. Der laufende Bahnbetrieb kann nur funktionieren, wenn permanent Tausende von Fachkräften in Betriebszentralen und Stellwerken ständig den Verkehrsfluss überwachen und Störungen behandeln. Hierbei haben sie einen räumlich begrenzten Fokus auf einen Dispositionsbereich bzw. eine Betriebsstelle. Die Koordination von Maßnahmen über größere Netzbereiche erfordert viel mündliche Kommunikation nicht nur zwischen Disponenten und Fahrdienstleitern, sondern auch mit Fahrzeugdisponenten verschiedener Verkehrsunternehmen. Mit noch mehr Verkehr auf dem Schienennetz und dem Wunsch nach mehr Flexibilität kommen die heutigen personalintensiven Prozesse zur Planung und Verkehrssteuerung an ihre Grenzen. Das künftige CTMS soll eine holistische Disposition mit Blick auf das Gesamtnetz ermöglichen.

Das CTMS ist grundsätzlich als universelles Werkzeug zu verstehen, welches Planungsaufgaben über den gesamten Planungshorizont des Eisenbahnsystems erledigt, von der langfristigen Fahr- und Bauplanung bis hin zur Disposition der Zugfahrten im Betrieb. In allen Planungsstadien erzeugt CTMS Pläne, die detailliert genug sind, um

**A**n intelligent traffic management system constitutes a core component for highly automated railway network operations and plays the role of a “planning brain”. The approach presented here uses the Multi-Agent Reinforcement Learning (MARL) method, a subfield of Artificial Intelligence (AI). AI acts on a railway simulation and creates highly detailed schedules for train operations. Significantly shorter computing time is required compared to classic optimisation methods. The approach therefore promises to be applicable to large network areas.

## 1 Motivation

The goal of sustainable mobility will require even more traffic to shift from road to rail in the future. Only profound innovation can lead to more attractive rail transport with more trains on the network at a higher quality. The “Digitale Schiene Deutschland” (DSD) sector initiative has set itself the task of leveraging the potential of digital technologies in order to increase the capacity, quality and efficiency of the rail system [1]. An intelligent Capacity & Traffic Management System (CTMS) constitutes a core part of the new digital rail system for fully automated planning and the control of rail operations.

Even in today’s rail system with 40,000 train runs a day, timetable creation is already a complex task that requires months of preparation by experts supported by specialised software. Ongoing rail operations can only function if the thousands of specialists in the operation control and signalling centres constantly monitor the traffic flow and deal with any disruptions. In doing so, their focus is spatially limited to a dispatch area or station. Coordinating across larger network areas requires a considerable amount of verbal communication, not only between dispatchers and signallers, but also with vehicle dispatchers from different railway undertakings. With even more traffic on the rail network to come and the desire for more flexibility, the current personnel-intensive planning and traffic control processes are reaching their limits. The future CTMS will enable holistic scheduling with a view of the entire network.

The CTMS can be understood as a universal tool that performs planning tasks over the entire planning horizon of a railway system, from long-term traffic and construction planning to (re) scheduling train runs during operations. The CTMS generates plans that are detailed enough to enable highly or fully automated train operations and automated control of the interlockings, and it does so at any of the planning stages. The CTMS supports

hoch- oder vollautomatisierte Zugfahrten und automatisierte Bedienung von Stellwerken zu ermöglichen. Dabei unterstützt CTMS sowohl heutige blockbasierte Leit- und Sicherungssysteme als auch künftige zugzentrische Sicherungstechnik, die eine Schlüsselrolle für mehr Kapazität im Zielbild DSD spielt [2]. Umfang und Aufgaben des CTMS werden ausführlicher in [3] beschrieben.

Im Folgenden wird ein bei DSD entwickelter Ansatz zur Fahrplan-konstruktion mit MARL vorgestellt. MARL ist ein Spezialgebiet des Deep Reinforcement Learning (DRL). DRL ist eine Methode der KI, in der KI-Modelle durch Erfahrung lernen, komplexe Entscheidungsprobleme zu lösen. In [4] konnte gezeigt werden, dass DRL bei vergleichbarer Qualität klassischen Optimierungsmethoden des Operation Research (OR) in der Rechengeschwindigkeit deutlich überlegen ist. Mit dem MARL-Ansatz sollen nun die Skalierbarkeit und Generalisierungsfähigkeit weiter gesteigert werden. Der vorgestellte MARL-Ansatz ist grundsätzlich sowohl zur Planung als auch zur Disposition anwendbar.

## 2 Lösungsansatz

Beim DRL lernen neuronale Netze durch Interaktion mit einer simulierten dynamischen Umgebung, komplexe Aufgaben zu lösen. DRL hat sich als vielversprechend auf dem Gebiet der kontinuierlichen Echtzeitsteuerung und strategischen Planung erwiesen [5]. Für das hier vorgestellte CTMS ist diese dynamische Umgebung eine im Kontext von DSD entwickelte, mikroskopisch scharfe Eisenbahnsimulation, die alle relevanten Aspekte des Bahnbetriebes abbildet. Dazu gehören das reale Schienennetz, physikalische Fahrzeugeigenschaften, Leit- und Sicherungstechnik (LST) in verschiedenen Varianten sowie Prozesse wie das Ausführen von Passagierhalten. Durch diese Umgebung kann ein Steuersystem, egal ob mit oder ohne KI, Züge fahren lassen. Dazu bestimmt es zu jedem Zeitpunkt alle individuellen Fahrwege, Geschwindigkeiten, Haltepunkte und -zeiten der Fahrzeuge – wobei die Simulation die Einhaltung sowohl physikalischer als auch sicherheitstechnischer Grenzen einfordert. Zu beachten ist hierbei, dass CTMS nicht als sicherheitskritisches System angelegt ist, das kollisionsfreien Eisenbahnverkehr gewährleisten muss. Die Sicherheitsverantwortung liegt bei dem System, das die Bedienung der Stellwerke leistet, im vorliegenden Fall das Advanced Protection System (APS) [2]. Nichtsdestotrotz muss auch CTMS die Bedingungen der LST beachten, um ausführbare Pläne zu erzeugen. In der vorliegenden CTMS-Anwendung erhält die KI eine Menge von Fahrplanfragen, die nur sparsame betrieblich relevante Eckdaten enthalten: Ein- / Austritt in den Netzabschnitt und eine Folge an kommerziellen Halten, jeweils mit Zeitinformation. Die KI hat die Aufgabe, durch gute Entscheidungen Zugfahrten in der Simulation so ablaufen zu lassen, dass die Fahrplanfragen für alle Züge wunschgemäß realisiert und gegebene Optimierungsziele erreicht werden. Ein erfolgreich von der KI gesteuerter Simulationslauf erzeugt einen Ablaufplan, der alle Informationen für Stellwerk und Zugsteuerung enthält und so perspektivisch automatisierten Betrieb von Infrastruktur (Stellwerke) und Zügen (Automatic Train Operation, ATO) ermöglicht.

Für einen einzelnen Zug ist die oben beschriebene Aufgabe ein einfaches Wegfindungsproblem, das mit Standard-Algorithmen gelöst werden kann. Mit wachsender Anzahl von Zügen steigt die Anzahl von Optionen für Wege und Reihenfolgen jedoch exponentiell. Gleichzeitig wird der Raum für fahrbare Lösungen im stark belasteten Netz geringer. Das Fahren im Minimalabstand (Moving Block) fügt hier zusätzliche Komplexität hinzu, weil das Fahrverhalten eines Zuges durch vorausfahrende Züge mitbestimmt wird.

both today's block-based safety logic as well as any future train-centric safety technology, which is a key lever to increasing capacity in the DSD target scenario [2]. The scope and tasks of the CTMS are described in more detail in [3].

The following presents an approach developed at DSD for timetable construction using MARL. MARL is a subfield of Deep Reinforcement Learning (DRL). DRL is an AI method, in which AI models learn to solve complex decision problems through experience. DRL has been shown to outperform the classic Operations Research (OR) optimisation methods in computing speed while maintaining a comparable quality [4]. The MARL approach presented here aims to further increase the scalability and generalisation capability. It can be used for both planning and dispatching.

## 2 Method

In DRL, neural networks learn to solve complex tasks by interacting with a simulated dynamic environment. DRL has been demonstrated to be highly efficient in the field of continuous real-time control and strategic planning [5]. For the CTMS presented here, this dynamic environment is a microscopic railway simulation developed within the context of DSD that maps all the relevant aspects of railway operations. This includes the actual rail network, the physical vehicle characteristics, the safety system in different variants as well as processes such as the execution of passenger stops. A control system – be it with or without AI – can execute train runs within this environment. It determines all the individual routes, speeds, stopping points and times of the vehicles at all times, while the simulation ensures compliance with both the physical and the safety-related limits. Importantly, the CTMS is not being designed as a safety-critical system that must guarantee collision-free rail traffic. The responsibility for safety lies with the system that operates the interlockings, in this case the Advanced Protection System (APS) [2]. Nevertheless, the CTMS must integrate the safety system's constraints in order to generate executable plans.

In this application inside the CTMS, the AI receives a set of requests for train runs, which are only sparsely defined by operationally relevant key data: the entry / exit to / from the network section and a sequence of commercial stops, each with time information. The AI's task is to ensure that all the train runs are performed in the simulation in such a way that fulfils the requests in the desired manner and meets the given optimisation targets by making appropriate decisions. When successfully controlled by the AI, a simulation run generates a schedule that contains all the information needed by the interlocking and train control systems. In the future, this will enable the automated operation of the infrastructure (interlockings) and trains (Automatic Train Operation, ATO).

For a single train, this task is a simple path-finding problem that can be solved using standard algorithms. As the number of trains increases, however, the number of options for possible train paths and sequences increases exponentially. At the same time, the solution space on the heavily utilised network becomes smaller. Moving at a minimum distance (moving block) adds additional complexity here, because the driving behaviour of one train depends on that of the trains ahead.

In contrast to classic mathematical optimisation, the DRL approach chosen here approximates an optimum solution, i.e. it searches for an acceptable solution, but does not guarantee that it will be optimal. The strength of DRL lies in its ability to gen-

Im Gegensatz zu klassischer mathematischer Optimierung arbeitet der hier gewählte DRL-Ansatz approximativ, d.h. er sucht eine akzeptable Lösung, garantiert aber nicht die optimale Lösung. Seine Stärken liegen darin, durch kluge Reduktion der Informationsmenge sowie paralleles Rechnen in kurzer Zeit eine Lösung erzeugen zu können. Kurze Rechenzeiten sind essenziell für den Einsatz in der betrieblichen Echtzeitdisposition.

Für den Anwendungsfall der Disposition wurde in [6] gezeigt, wie mit DRL auf Graphenstrukturen, die den Fahrplan repräsentieren, Lösungen erzeugt wurden, die qualitativ mit Methoden des OR vergleichbar sind, jedoch deutlich schneller errechnet wurden. Der hier vorgestellte Ansatz benutzt ebenso die Methode des DRL, aber unterscheidet sich hinsichtlich der Problemformulierung. Hier steuern KI-Modelle direkt Züge durch einen Simulationslauf und erzeugen so einen Ablaufplan. Im Gegensatz zu [6], wo die KI lernt, einen bestehenden Fahrplan durch optimale (sparsame) Auswahl von Dispositionshandlungen zu modifizieren, ist der hier beschriebene Ansatz konstruktiv: Hier erstellt die KI bei maximaler Ausnutzung der bestehenden Freiheiten der Steuerung einen grundsätzlich neuen Ablaufplan. Dieser Ansatz ist zum einen sowohl für Disposition als auch für Planung anwendbar. Zum anderen bietet der Fokus auf die Steuerung des einzelnen Zugs (unter Berücksichtigung der globalen Situation) Möglichkeit zu Parallelisierung und somit Skalierbarkeit auf größere Netze und mehr Züge.

DRL-Modelle interagieren mit der Simulation durch Aktionen und lernen durch Beobachtungen und Belohnungen (Bild 1).

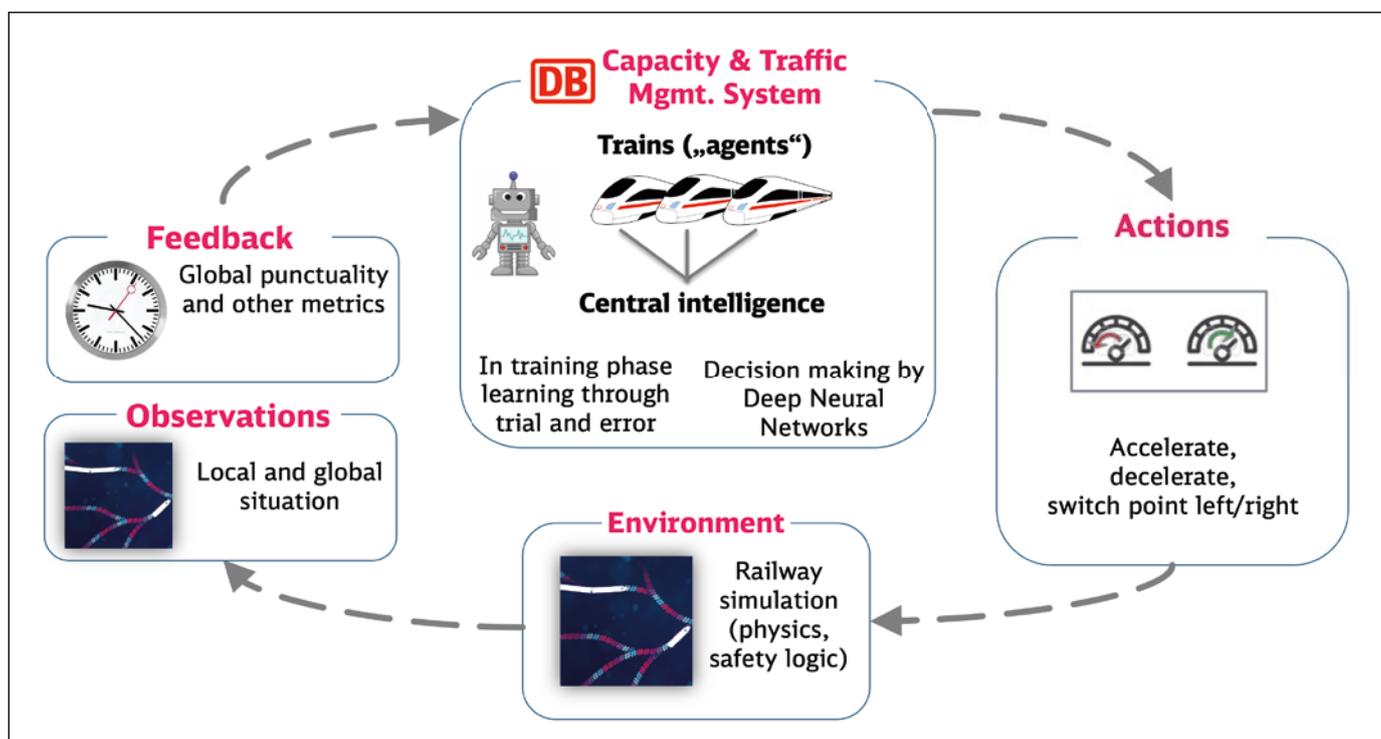
- **Aktion:** In jedem Simulationsschritt entscheidet das KI-Modell über Aktionen, die zwei Komponenten haben: Geschwindigkeit und eine Schaltung der nächsten Weiche.
- **Beobachtung:** In jedem Simulationsschritt erhält das KI-Modell für jeden Zug Information, die die lokale Perspektive des Zugs auf sei-

erate a solution in a short time by means of the intelligent compression of information and parallel processing. Short computing times are essential for use in real-time operations.

[6] has shown that DRL produces results significantly faster than OR methods, but at a comparable quality, for the dispatching use case. These DRL models worked on graph structures representing the global timetable. While the approach presented here also uses DRL, it formulates the problem in a different way. Here, AI models directly steer the trains through a simulation run and thereby generate a schedule. In contrast to [6], where the AI learns to modify an existing timetable by optimally selecting (a limited number of) dispatching actions, the approach described here is constructive, i.e. the AI creates an entirely new schedule while making maximum use of the existing degrees of freedom. This approach can be used for both scheduling and planning. At the same time, the concept of controlling the individual train (while taking into account the global situation) offers the possibility of parallelisation and thus scalability to larger networks and more trains.

DRL models interact with the simulation environment through actions and they learn through observation and rewards (fig. 1).

- **Action:** in each simulated time step, the AI model decides on an action with two components: the speed and the next point setting.
- **Observation:** the AI model receives information about the local situation of the train in each time step and for each train, as well as compressed information about the global state of the railway network.
- **Reward:** the feedback function rewards the reaching of commercial stops and the exiting of the network as desired. The success of a completed simulation run is measured as a per-



**Bild 1: Problemformulierung DRL: Neuronale Netze interagieren mit einer Simulationsumgebung durch Aktionen („actions“), die in der Simulation ausgeführt werden, erhalten Informationen über die aktuelle Situation durch Beobachtungen („observations“) und lernen durch Feedback („reward“).**

Fig. 1: DRL problem formulation: neural networks interact with a simulation environment through actions that are executed in the simulation. They receive information on the state of the environment (observation) and learn through feedback (reward).

ne Umgebung abbildet und zusätzlich komprimierte Information über den globalen Zustand des Gesamtnetzes.

- **Belohnung:** Die Feedback-Funktion belohnt in einfacher Weise das Erreichen von Halten und Austrittspunkten. Der Erfolg eines Simulationslaufes wird in Anteil vollständig ausgeführter Zugfahrten gemessen. In dieser ersten prototypischen Implementierung ist die Formulierung des Feedbacks bewusst einfach gehalten. Eine detaillierte Beschreibung findet sich in [7].

In der Trainingsphase beginnt ein untrainiertes KI-Modell, Züge mit zufälligen Aktionen durch den Netzabschnitt fahren zu lassen. Durch Feedback lernt es über viele Trainingsläufe hinweg, Fahrzeuge und Infrastruktur so zu steuern, dass ein Fahrplan entsteht, der alle Fahrplananfragen erfüllt. Um diesen Zustand zu erreichen, muss das KI-Modell im Training tausende von Simulationsläufen mit einer großen Bandbreite an repräsentativen Situationen erlebt haben.

Um das Potenzial der Skalierbarkeit zu realisieren, wird im vorgestellten Ansatz die DRL-Methode in einer Multi-Agenten-Formulierung (MARL) verwendet (vgl. auch [8], zum Vergleich: Der in [6] vorgestellte Ansatz ist ein Single-Agent Modell). In einer einfacher strukturierten Single-Agent RL-Variante würde ein KI-Modell mit einer einzigen, komplexen Aktion pro Zeitschritt alle Züge in der Simulation steuern. Dies ist vom Ablauf her einfach, jedoch ist die Aktion selbst hochkomplex und verändert sich mit der Anzahl der Züge. Die Grenzen des Lernbaren sind schnell erreicht. In der Multi-Agenten-Formulierung gibt es ebenfalls nur ein einziges KI-Modell. Dieses trifft für jedes einzelne Fahrzeug Entscheidungen, spezifisch für dessen Situation, aber mit einer einzigen gelernten Entscheidungsstrategie für alle Fahrzeuge. Dies kann parallelisiert werden, wodurch die Skalierung auf eine höhere Anzahl von Zügen möglich wird [9]. Außerdem ist die gelernte Entscheidungsstrategie anwendbar für unterschiedliche Anzahlen von Zügen.

Die Grundidee des beschriebenen Ansatzes ist die der Fahrplankonstruktion: Für sparsam definierte Fahrplananfragen werden mikroskopische Ablaufpläne erzeugt. Auch in der Disposition generiert ein Steuersystem Fahrpläne, jedoch mit anderen Randbedingungen und Optimierungszielen. Wenn eine Störung eintritt, beginnt ein Simulationslauf mit einer Momentaufnahme des laufenden Betriebs, welche die bereits auf dem Netz fahrenden Züge an ihren jeweiligen Positionen mit ihrer jeweiligen Geschwindigkeit abbildet. Zudem ist der aktuelle, möglicherweise gestörte Betriebszustand der Infrastruktur und des Sicherungssystems enthalten. Die Fahrplananfragen sind im Dispositionsfall durch den letzten gültigen Fahrplan gesetzt: Jeder Zug soll nach Möglichkeit seine in diesem Fahrplan enthaltenen, noch zu bedienenden Halte sowie seinen korrekten Austrittspunkt wahrnehmen. Im Folgenden wird der MARL-Ansatz für Planung und Disposition evaluiert.

### 3 Evaluation

Der vorgestellte MARL-Ansatz wird auf einem realen Ausschnitt des deutschen Schienennetzes um den Hauptbahnhof Magdeburg evaluiert. Für diesen Netzausschnitt (Bild 2) wurden verschiedene Anwendungsfälle betrachtet:

In Anwendungsfall 1 ist die Koordination von dichtem Verkehr in einer einzelnen Betriebsstelle von Interesse. Es wurden 100 verschiedene Szenarien aus Fahrplananfragen generiert, in denen jeweils zehn Züge in verschiedenen Variationen in den Bahnhof ein- und ausfahren sowie einen Halt absolvieren. CTMS muss die Durchfahrt durch den Bahnhof sowie die Verteilung der Halte auf Bahnsteige koordinieren.

Anwendungsfall 2 untersucht die Erstellung von Plänen für den gesamten gezeigten Netzausschnitt (Bild 2). Es wurden 72 Szenarien generiert mit 17–41 Zügen pro Szenario (Mischverkehr: alle Verkehrsklassen mit sechs verschiedenen Zugtypen und Charakteristika). Hier ist von Interesse, wie CTMS mit der Größe des Problems umgeht (viele simultane

centage of the fully executed train runs. The reward function has been deliberately kept simple in this prototypical implementation.

A detailed description can be found in [7].

When undergoing training, an untrained AI model steers the trains through the network section using random actions. It uses the feedback over many training runs to learn to control the vehicles and infrastructure in such a way that the created schedule fulfils all the requests. In order to achieve this, the AI model needs to experience thousands of simulation runs with a wide range of representative situations during the training phase.

The presented approach uses the DRL method in a multi-agent formulation (MARL, also see [8], for a comparison: the approach presented in [6] is a single-agent model), in order to exploit the potential for scalability. In a simpler single-agent RL variant, an AI model would control all the trains in the simulation with a single, complex action per time step. This is a straightforward control process, but the action itself is highly complex and it changes with the number of trains in the simulation. The scope of what such a single-agent AI model can learn is very limited. There is also only a single AI model in the multi-agent formulation. However, the AI model makes individual decisions for each vehicle that are specific to its situation. One learned decision strategy is applied to each vehicle. The decision-making processes for individual trains can be parallelised, which allows the AI model to be scaled to a larger number of trains [9]. Furthermore, the learned decision strategy is applicable to any arbitrary number of trains.

The idea of schedule construction lies at the core of the proposed approach: microscopic schedules are generated from sparsely defined requests for train runs. The control system also generates schedules in dispatching, but with different boundary conditions and optimisation goals. In the case of a disruption, the simulation run is initiated with a snapshot of the current state of operations containing all the trains already running on the network at their respective positions and speeds. The current, possibly disrupted operating status of the infrastructure and the safety system is also included. The requests for train runs are set using the last valid timetable: each train will perform its remaining stops and exit the network as previously planned as far as possible. The following evaluates the MARL approach to planning and dispatching.

### 3 Evaluation

The MARL approach has been evaluated on a real section of the German railway network located around the Magdeburg central station. The following use cases have been addressed for this section of the network (fig. 2):

Use case 1 focussed on the coordination of dense traffic in a single operating station. One hundred different scenarios containing sets of train run requests were generated. In each scenario, ten trains had to enter and exit the station in different variations and perform one stop. The task of the CTMS was to coordinate the passage through the station and the use of the platforms in the station.

Use case 2 evaluated the CTMS on the entire network section (fig. 2). Here, 72 scenarios were generated with 17–41 trains per scenario (mixed traffic: all traffic classes with six different train types and characteristics). This was to assess how the CTMS performed on a considerably larger network sec-



**Bild 2: Großes Bild:** Netzausschnitt um Magdeburg Hauptbahnhof mit 29 Bahnhöfen, 343 Gleiskilometern (Anwendungsfall 2a und 2b), **kleines Bild links:** geographische Darstellung, **kleines Bild rechts:** Betriebsstelle Biederitz (Anwendungsfall 1)

Fig. 2: The large image: the network section around the Magdeburg main station with 29 stations and 343 km of track. (use cases 2a and 2b), the small image on the left-hand side: the geographical map, the small image on the right-hand side: Biederitz station (use case 1) Quelle / Source: Digitale Schiene Deutschland, OpenRailwayMap

Entscheidungen, viele Gleise und Weichen) und über längere Distanzen und Zeit den Verkehr koordiniert. Für diesen Netzausschnitt wurde sowohl die Anwendung für Planung (2a) als auch für Disposition (2b) evaluiert. Für den Anwendungsfall Disposition wurde jeweils ein vielbefahrenes Gleis im Zentrum des Netzausschnitts unbefahrbar gemacht und somit eine Neuberechnung des Plans erzwungen. Die Störungen wurden so platziert, dass Züge zwingend umgeleitet werden mussten. Ferner wurden in der Evaluation die Störungen auf anderen Gleisen platziert als im KI-Training. Das war wichtig, um die Generalisierungsfähigkeit der KI-Modelle nachzuweisen, also deren Fähigkeit, auf unbekannte Situationen zu reagieren.

Als Referenz diente eine Lösung, die Pläne mit einer einfachen Heuristik erstellt, im Folgenden „Greedy Agent“ genannt. Der Greedy Agent erstellt Pläne für jeden Zug einzeln nach folgenden Regeln:

- Ein Wegfindungsverfahren bestimmt mögliche Routen.
- Bei mehreren möglichen Routen zum Ziel entscheidet der Zufall.
- Jeder Zug fährt so schnell wie möglich.

Diese Heuristik erzeugt für den einzelnen Zug Pläne, die garantiert und schnell zum Ziel führen, koordiniert jedoch nicht zwischen verschiedenen Zügen.

## 4 Ergebnisse

### 4.1 Anwendungsfall 1: Verkehr in einer Betriebsstelle

Tab. 1 zeigt, dass der MARL-Ansatz auf kleinem, aber dichtbefahrenem Raum 100 % der Szenarien löst, während der Greedy Agent nur in 69 % der Fälle für alle Züge eine Lösung findet. In den ungelösten Beispielen enden Züge mangels Koordination im sog. „deadlock“, also blockieren sich gegenseitig.

### 4.2 Anwendungsfall 2: Gesamter Netzausschnitt, Planung und Disposition

Tab. 1 zeigt, dass der MARL-Ansatz über 72 Szenarien mit je 17–41 Zugfahrten hinweg 94 % aller Zugfahrten konfliktfrei planen kann. Diese Erfolgsquote gilt gleichermaßen für den Planungs- (2a) und

tion (many simultaneous decisions, many tracks and points) and coordinated traffic over longer distances and times. On this larger section of the network, the CTMS was evaluated for both planning (2a) and dispatching tasks (2b). A heavily utilised track in the centre of the network section was rendered unavailable for the dispatching task, thereby forcing the CTMS to create a new schedule. The disruptions were positioned in such a way that at least some trains had to be re-routed. During the evaluation, the disruptions were placed on different tracks than in the AI training. This was important in order to demonstrate the generalisation capability of the AI models, i.e. their ability to react to previously unseen situations.

A simple heuristic, referred to below as “Greedy Agent”, was used as a reference. The Greedy Agent creates plans for each train individually according to the following rules:

- a path-finding procedure determines the possible routes to the next destination.
- the route is chosen randomly if more than one route exists.
- each train drives as fast as possible.

This heuristic produces plans that guarantee that each individual train reaches its destination quickly. However, the heuristic does not coordinate between the trains.

## 4 Results

### 4.1 Use case 1: Scheduling train runs at a single station

Tab. 1 shows that the MARL approach solved 100 % of the scenarios in a small area with dense traffic. In contrast, the Greedy Agent produced complete solutions in only 69 % of the cases. Trains ended in deadlock in the unresolved examples due to the lack of coordination.

### 4.2 Use case 2: The full network section, planning and dispatching

| Anwendungsfall   | MARL-Ansatz | Greedy Agent |
|--|-------------|--------------|
| Anwendungsfall 1: einzelne Betriebsstelle                                | 100 %       | 69 %         |
| Anwendungsfall 2a: gesamter Netzausschnitt, ohne Störung                 | 94 %        | 14 %         |
| Anwendungsfall 2b: gesamter Netzausschnitt, mit Störung                  | 94 %        | 16 %         |
| Skalierbarkeitsanalyse: Erfolgsquote für eine bestimmte Anzahl von Zügen |             |              |
| 19 Züge  | 100 %       | 32 %         |
| 41 Züge  | 99 %        | 10 %         |

**Tab. 1: Erfolgsrate für verschiedene Anwendungsfälle (MARL-Modell und Referenzlösung). Der Erfolg einer Lösung pro Szenario wird in Anteil vollständig ausgeführter Zugfahrten gemessen. Eine Zugfahrt ist vollständig ausgeführt, wenn der Zug alle Halte in vorgesehener Reihenfolge absolviert und das Netz an der geforderten Austrittsstelle verlässt.**

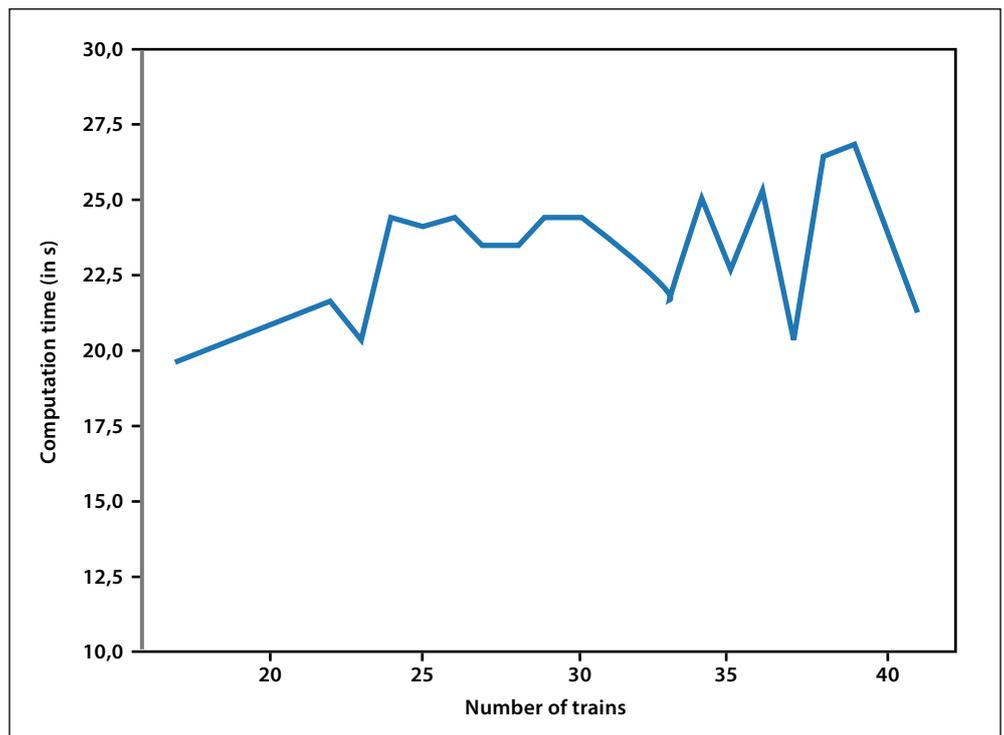
Dispositionsfall (2b). Das KI-Modell kommt also mit unerwarteten Änderungen in der Infrastruktur zurecht. Beim Greedy Agent, mit Erfolgsquoten von 14 % bzw. 16 %, war kein Unterschied zwischen 2a und 2b zu erwarten. Der starke Abfall der Erfolgswahrscheinlichkeit des Greedy Agent gegenüber Anwendungsfall 1 zeigt, wie sehr die Notwendigkeit der räumlich-zeitlichen Koordination von Zugfahrten auf dem komplexeren Netzbereich steigt. Auch in den Ergebnissen des MARL-Ansatzes ist dies, trotz des im Mittel sehr hohen Erfolgs, sichtbar: Bis zu einer Anzahl von 19 Zügen werden alle Szenarien perfekt gelöst, erst für höhere Anzahlen von Zügen nimmt die Qualität der Ergebnisse leicht ab (Tab. 1).

### 4.3 Skalierbarkeit

Bild 3 zeigt die mittleren Rechenzeiten, die im MARL-Ansatz benötigt werden, um einen kompletten Ablaufplan für den gesamten Netzausschnitt Magdeburg zu berechnen. Für alle Szenarien liegen die Rechenzeiten unter 30 Sekunden, für durchschnittlich 42 Minuten simulierten Zugverkehr. Trotz einer hohen Varianz, die zum einen auf unterschiedliche Komplexität der einzelnen Szenarien, teils auf fluktuierende Hardware-Auslastung zurückzuführen ist, legen

**Bild 3: Rechenzeit des MARL-Ansatzes (in Sekunden) in Abhängigkeit von der Anzahl der Züge**

Fig. 3: The computation time for the MARL approach (in seconds) as a function of the number of trains in a scenario



| Use case  | MARL approach | Greedy Agent |
|---|---------------|--------------|
| Use case 1: single station  | 100 %         | 69 %         |
| Use case 2a: entire network section, schedule construction without disruption | 94 %          | 14 %         |
| Use case 2b: entire network section, schedule construction after disruption   | 94 %          | 16 %         |
| Scalability analysis: success rate for specific numbers of trains             |               |              |
| 19 trains   | 100 %         | 32 %         |
| 41 trains   | 99 %          | 10 %         |

**Tab. 1: The success rate per use case (MARL versus reference). The success rate is defined as the ratio of completed train runs over all the train runs in the scenario. A train run counts as having been completed if all the commercial stops have been performed in the correct order and the network is exited where requested.**

Tab. 1 shows that the MARL approach produced a conflict-free schedule for 94 % of all train runs averaged across 72 scenarios with 17-41 train runs each. The success rate was the same for planning (2a) and dispatching (2b). This means the AI model coped well with any unexpected changes in the infrastructure. No difference between 2a and 2b had been anticipated for the Greedy Agent, with success rates of 14 % and 16 % respectively. However, the sharp drop in the Greedy Agent’s success rate with respect to use case 1 highlights the importance of the spatial-temporal coordination of the train runs in increasingly large networks. Despite the very high average success of the MARL model, a similar effect can also be seen here: all the scenarios were solved perfectly for up to 19 trains, while the quality of the results decreased slightly for larger numbers of trains (tab. 1).

### 4.3 Scalability

Fig. 3 shows the average times required in the MARL approach to calculate a complete schedule for the full Magdeburg network section. The calculation times for all the scenarios are below 30 seconds for an average of 42 minutes of simulated traffic. Despite a high variance, which can be attributed to varyingly complex scenarios and

die Ergebnisse einen linearen Zuwachs an Rechenzeit für steigende Anzahl von Zügen dar. Damit hebt sich der MARL-Ansatz deutlich von Ansätzen aus der OR ab, bei denen die Rechenzeit typischerweise exponentiell mit der Anzahl der Züge wächst. Ein detaillierter Vergleich findet sich in [6]. Insbesondere diese potenziell kurzen Rechenzeiten machen den MARL-Ansatz zum geeigneten Kandidaten für den Einsatz in der Echtzeit-Disposition.

## 5 Zusammenfassung und Ausblick

Der vorgestellte Ansatz benutzt die KI-Methode des MARL für die Erstellung von räumlich und zeitlich hochdetaillierten Ablaufplänen für den Eisenbahnbetrieb. Der Ansatz wird sowohl für Planungsaufgaben als auch für das sofortige Umplanen im laufenden Betrieb (heute: Disposition) entwickelt. Die Ergebnisse der Evaluation mit realistischem (simuliertem) Eisenbahnverkehr zeigen, dass der MARL-Ansatz für kleine Betriebsstellen mit starker Auslastung verlässlich fahrbare Ablaufpläne erzeugt. Auf einem größeren Netzausschnitt sind die Ergebnisse noch nicht perfekt, aber insgesamt vielversprechend. Hier wird die Herausforderung der Koordination von Zugfahrten auf größeren Gebieten noch an Einzelfällen deutlich. Das bisher erreichte Qualitätsniveau und die gegenüber anderen Ansätzen signifikant kürzeren Rechenzeiten machen den MARL-Ansatz zu einer wegweisenden Technologie für eine schnelle, flexible und gesamthafte automatische Verkehrsplanung und -steuerung in einem künftigen digitalisierten Bahnsystem.

Auf dem Weg zur Produktreife muss dieser Ansatz methodisch verfeinert werden, um Skalierbarkeit und Qualität der Lösungen weiter zu steigern. Gleichzeitig muss die KI in die Lage gebracht werden, alle relevanten Details des Anwendungsfalls (zusätzliche Dispositionsmaßnahmen, Randbedingungen und weitere Optimierungskriterien) verarbeiten zu können. ■

fluctuating hardware utilisation, the results show a linear increase in the computing time with the number of trains. This clearly distinguishes the MARL approach from classic OR approaches, where the computing time typically grows exponentially with the number of trains. A detailed comparison can be found in [6]. These potentially small computation times make the MARL approach a suitable candidate for real-time scheduling.

## 5 Summary and outlook

The presented approach proposes the MARL AI method for generating spatially and temporally highly detailed schedules for railway operations. The approach can be applied to planning tasks as well as to real-time re-scheduling in railway operations (today: dispatching). The evaluation of realistic (simulated) rail traffic has shown that the MARL approach reliably generates feasible schedules for small stations, even with dense traffic. In general, the results have also shown promise on a larger network section, although they are not yet perfect. The challenge of train-to-train coordination in larger networks is still evident in individual cases. The level of quality achieved to date and the significantly shorter computing times compared to other approaches make MARL a promising, pioneering technology for fast, flexible and holistic automatic traffic planning and control in a future digitalised rail system.

In order to become a mature product for use in live operations, this approach must be methodically refined towards even better scalability and solution quality. The AI models need to be extended in parallel in order to process all the relevant details of their respective use cases (further-reaching dispatching measures, boundary conditions and additional optimisation criteria). ■

### AUTOREN | AUTHORS

#### Stefan Schneider

Senior AI & Software Engineer  
Digitale Schiene Deutschland  
DB InfraGO  
Anschrift / Address: EUREF Campus 17, Gasometer, D-10829 Berlin  
E-Mail: stefan.sh.schneider@deutschebahn.com

#### Anne Roets

Artificial Intelligence Researcher – CTMS  
DB Systel GmbH  
Anschrift / Address: Kynaststraße 1, D-10317 Berlin  
E-Mail: anne-helene-paul.Roets@deutschebahn.com

#### Ciprian Stirbu

Senior Research Engineer  
InstaDeep  
Anschrift / Address: Kemperplatz 1, Mitte D, D-10785 Berlin  
E-Mail: c.stirbu@instadeep.com

#### Farhad Safaei

Berufsbezeichnung Deutsch und *English*  
Digitale Schiene Deutschland  
DB InfraGO  
Anschrift / Address: EUREF Campus 17, Gasometer, D-10829 Berlin  
E-Mail: farhad.safaei@deutschebahn.com

#### Faten Ghriss

Engineering Manager  
InstaDeep  
Anschrift / Address: 40 bis rue du Faubourg Poissonnière, F-75010 Paris  
E-Mail: f.ghriss@instadeep.com

#### Jan Wülfing

Senior Expert in Artificial Intelligence  
Digitale Schiene Deutschland  
DB InfraGO  
Anschrift / Address: EUREF Campus 17, Gasometer, D-10829 Berlin  
E-Mail: jan.wuefing@deutschebahn.com

#### Mehmet Güral

DevOps Engineer  
Digitale Schiene Deutschland  
DB InfraGO  
Anschrift / Address: EUREF Campus 17, Gasometer, D-10829 Berlin  
E-Mail: mehmet-ugur.gueral@deutschebahn.com

#### Nima Siboni

Senior Research Engineer  
InstaDeep  
Anschrift / Address: Kemperplatz 1, Mitte D, D-10785 Berlin  
E-Mail: n.siboni@instadeep.com

#### Roman Liessner

Senior Expert in Artificial Intelligence  
Digitale Schiene Deutschland  
DB InfraGO  
Anschrift / Address: EUREF Campus 17, Gasometer, D-10829 Berlin  
E-Mail: roman.liessner@deutschebahn.com

#### Thomas Hustache

Research Engineer  
InstaDeep  
Anschrift / Address: Kemperplatz 1, Mitte D, D-10785 Berlin  
E-Mail: t.hustache-marmou@instadeep.com

**Thomas Lecat**

Staff Research Engineer  
 InstaDeep  
 Anschrift / Address: 40 bis rue du Faubourg Poissonnière, F-75010 Paris  
 t.lecat@instadeep.com

**Umashankar Deekshith**

Artificial Intelligence Researcher – CTMS  
 DB System GmbH  
 Anschrift / Address: Kynaststr. 1, D-10317 Berlin  
 Umashankar.Deekshith@deutschebahn.com

**Valerii Markin**

Research Engineer  
 InstaDeep  
 Anschrift / Address: 40 bis rue du Faubourg Poissonnière, F-75010 Paris  
 v.markin@instadeep.com

**Victor Le**

Machine Learning Engineer  
 InstaDeep  
 Anschrift / Address: 40 bis rue du Faubourg Poissonnière, F-75010 Paris  
 v.le@instadeep.com

**Wissam Bejjani**

Research Engineer  
 InstaDeep  
 Anschrift / Address: 40 bis rue du Faubourg Poissonnière, F-75010 Paris  
 w.bejjani@instadeep.com

**Michael Küpper**

Capacity & Traffic Management Ambassador  
 Digitale Schiene Deutschland  
 DB InfraGO  
 Anschrift / Address: EUREF Campus 17, Gasometer, D-10829 Berlin  
 michael.m.kuepper@deutschebahn.com

**Irene Sturm**

Product Manager Capacity & Traffic Management System und Expertin Künstliche Intelligenz  
 Digitale Schiene Deutschland  
 DB InfraGO  
 Anschrift / Address: EUREF Campus 17, Gasometer, D-10829 Berlin  
 irene.sturm@deutschebahn.com

**LITERATUR | LITERATURE**

[1] Digitale Schiene Deutschland [Online]. Available: <https://digitale-schiene-deutschland.de/en/What-is-Digitale-Schiene-Deutschland>. [accessed: 24 09 2024]

[2] Digitale Schiene Deutschland is working intensively on the development of an „Advanced Digital Infrastructure“ [Online available: <https://digitale-schiene-deutschland.de/en/news/2024/special%20article-ADI>] [accessed: 08.11.24]

[3] Küpper, M.: Das Capacity & Traffic Management System für die Digitale Schiene, DER EISENBAHNINGENIEUR 10/2023 [Online available: [https://digitale-schiene-deutschland.de/Downloads/2310\\_EI\\_Artikel\\_CTMS\\_K%C3%BCpper.pdf](https://digitale-schiene-deutschland.de/Downloads/2310_EI_Artikel_CTMS_K%C3%BCpper.pdf), accessed: 08.11.24]

[4] Sturm, I.; Vienken, G.; Söhlke, A.; Hartleb, J.; Gorsane, R.; Gorsan Mestiri, K.; Martinez, D.; Coyette, V.; Truong, M. T.; Küpper, M.: Künstliche Intelligenz für das Verkehrsmanagementsystem der Zukunft, SIGNAL+DRAHT (Bd. 116) 1+2/2024, pp. 24-30 [Online available: [https://digitale-schiene-deutschland.de/Downloads/24\\_01\\_02\\_SignaluDraht\\_Fachartikel\\_CTMS\\_Sturm.pdf](https://digitale-schiene-deutschland.de/Downloads/24_01_02_SignaluDraht_Fachartikel_CTMS_Sturm.pdf), accessed: 8.11.24]

[5] Wang, H.-N.; Liu, N.; Zhang, Y.-Y.; Feng, D.-W.; Hunag, F.; Li, D.-S.; Zhang, Y.-M.: Deep reinforcement learning: a survey, Frontiers of Information Technology & Electronic Engineering, Bd. 21, Nr. 12, p. 1726–1744, 2020

[6] Gorsane, R.; Mestiri, K.; Martinez, D.; Coyette, V.; Makhlouf, B.; Vienken, G.; Truong, M.; Söhlke, A.; Hartleb, J.; Kerkeni, A.; Sturm, I.; Küpper, M.: Reinforcement learning based train rescheduling on event graphs, in 2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), Bilbao, Spanien, 2023

[7] Schneider, S.; Ramesh, A.; Roets, A.; Stirbu, C.; Safaei, F.; Ghrris, F.; Wülfing, J.; Güral, M.; Siboni, N.; Gentry, R.; Liessner, R.; Hustache-Marmou, T.; Lecat, T. D.; Markin, V.; Le, V.; Bejjani, W.; Küpper, M.; Sturm, I.: Intelligent Railway Capacity and Traffic Management Using Multi-Agent Deep Reinforcement Learning, in IEEE 27th International Conference on Intelligent Transportation Systems (ITSC), Edmonton, CA, 2024

[8] Mohanty, S.; Nygren, N.; Laurent, F.; Schneider, M.; Scheller, C.; Bhattacharya, N.; Watson, J.; Egli, A.; Eichenberger, C.; Baumberger, C.; Vienken, G.; Sturm, I.; Sartoretti, G.; Spigler, G.: Flatland-RL: Multi-Agent Reinforcement Learning on Trains, arXiv:2012.05893, 2020

[9] Liang, E.; Liaw, R.: Scaling multi-agent reinforcement learning, 2018 [Online]. Available: <https://bair.berkeley.edu/blog/2018/12/12/rllib/>. [accessed: 24 09 2024]

**Eurailpress**  
 Archiv

- über 44.000 Beiträge
- laufende Aktualisierung
- individuelle Suchoptionen
- Volltextsuche
- Sofort-Download

Abonnenten erhalten bis zu **50% Rabatt**

**Ohne Umwege zu Ihren Fachartikeln**

www.eurailpress.de/fachartikel

Logos: EI DER EISENBAHN INGENIEUR, ETR, EIK, Raii, GÖTTERGEBILDE, DER NAHVERKEHR, bahn manager