Energy 324 (2025) 135793

Contents lists available at ScienceDirect

Energy

journal homepage: www.elsevier.com/locate/energy

Multiobjective eco-driving speed optimisation with real-time traffic: Balancing fuel, NOx, and travel time

Enze Liu [a,b], Zhiyuan Lin [b,*], Haibo Chen [b], Dongyao Jia [c], Ye Liu [b,*], Junhua Guo [d], Tiezhu Li [e], Tangjian Wei [f]

[a] School of Mechanical Engineering, Hefei University of Technology, Hefei, 230009, China
[b] Institute for Transport Studies, University of Leeds, Leeds, LS2 9JT, UK
[c] School of Advanced Technology, Xi'an Jiaotong-Liverpool University, China
[d] School of Transportation Engineering, East China Jiaotong University, China
[e] School of Transportation, Southeast University, Nanjing, China
[f] Department of Civil and Environmental Engineering, Western University, London, Ontario, N6A 3K7, Canada

ARTICLE INFO

Keywords:
Eco-driving speed profile optimisation
Fuel consumption
NOx emission
Multiobjective optimisation
Model predictive control
Deep reinforcement learning

ABSTRACT

Optimising driving velocity profiles is crucial for reducing vehicle fuel consumption and NOx emissions without altering core vehicle components. While many studies have addressed eco-driving, most have focused solely on minimising fuel consumption or have treated NOx emissions separately, resulting in distinct, non-integrated speed profiles, and have often neglected the influence of real-time traffic. To overcome these limitations, this paper introduces a novel Multiobjective Speed Profile Optimisation (MO-SPO) framework for eco-driving that simultaneously minimises fuel consumption, NOx emissions, and travel time while accounting for surrounding traffic. Two solution approaches are developed and compared: a two-phase Model Predictive Control (MPC) method and a newly proposed Deep Reinforcement Learning (DRL) method that directly integrates multiple objectives and real-time traffic constraints into the speed control policy.

Simulation results on a UK highway segment, with vehicle dynamics and engine characteristics derived from GT-SUITE data, demonstrate the benefits of the proposed framework. For instance, at one representative Pareto point, results indicate that the DRL approach achieves up to 10% lower fuel consumption and 16% lower NOx emissions compared to MPC-based methods while reducing travel time by approximately 5%. In addition, the DRL method maintained safer headway distances, offering more robust eco-driving strategies in dynamic traffic environments.

This work is the first to apply multiobjective optimisation to generate integrated speed profiles that consider fuel, NOx, and travel time simultaneously under realistic traffic conditions.

1. Introduction

Rapid urbanisation and the steady increase in global vehicle ownership have heightened concerns about energy consumption and air pollution in the transportation sector [1]. Although research and industry efforts have led to the development of more efficient powertrain systems and alternative-fuel vehicles [2–4], conventional internal combustion engine (ICE) vehicles still dominate the roads and contribute significantly to environmental problems, particularly through emissions of nitrogen oxides (NOx). Prolonged exposure to NOx is linked to photochemical smog, acid rain formation, and particulate matter (PM), such as PM2.5 and PM10, which pose direct risks to public health [5–7]. Additionally, repeated studies indicate that vehicle-related emissions are a major source of air pollution, leading to an estimated 7500 premature deaths annually in the UK alone [8].

Besides the health and environmental concerns, the global rise in fuel prices and the finite nature of petroleum supply have consistently underscored the economic imperative to minimise vehicle fuel consumption [2]. As a result, numerous strategies have been explored to reduce both emissions and energy usage, ranging from traffic signal optimisation [9] and cooperative driving [10,11] to the development of hybrid and electric vehicle technologies. Among these, optimising driving velocity profiles stands out as a highly cost-effective method, since it does not require retrofitting vehicles with new hardware or redesigning powertrains [12]. Instead, it focuses on modifying driver

* Corresponding authors.
E-mail addresses: z.lin@leeds.ac.uk (Z. Lin), y.liu8@leeds.ac.uk (Y. Liu).

https://doi.org/10.1016/j.energy.2025.135793
Received 17 January 2025; Received in revised form 21 March 2025; Accepted 23 March 2025
Available online 4 April 2025

behaviour – speed, acceleration, braking – to achieve more efficient and cleaner operation.

A substantial body of literature has investigated velocity profile optimisation from various angles [13–17]. Most of these studies concentrate on single-objective formulations, typically aiming to minimise fuel consumption under specific constraints such as road safety and rules. Although effective in reducing fuel consumption, these methods often overlook or only superficially address NOx emissions—an omission that is partly attributable to the complexities in accurately modelling and incorporating NOx in optimisation frameworks [18]. The incorporation of NOx is indeed technically more challenging, involving additional engine and aftertreatment parameters whose dynamic behaviour is less straightforward to predict compared to fuel consumption. Consequently, comprehensive studies that jointly optimise fuel and NOx remain sparse.

A few exceptions exist. For instance, Fernández-Yáñez et al. [18] investigated speed profile generation while considering both fuel and NOx, yet it treated each objective separately, yielding distinct profiles optimised exclusively for fuel or NOx. More recently, Yuval et al. [19] employed multi-objective optimisation to integrate fuel consumption and NOx objectives simultaneously. While this represents a meaningful advance, it primarily addresses traffic-free conditions and is built on a shortest-path method [20] without incorporating real-world traffic flow. The absence of traffic considerations limits the real-life applicability of such solutions, since constraints like headway distance, dynamic speed limits, and surrounding vehicles' behaviours significantly influence feasible speed profiles.

Against this backdrop, our work aims to close the gap by proposing a multiobjective speed profile optimisation (MO-SPO) framework that jointly minimises fuel consumption, NOx emissions, and travel time while explicitly accounting for surrounding traffic. Travel time is included as a third objective to reflect practical stakeholder needs, since drivers and freight operators often balance economic, environmental, and time-efficiency goals. By framing these objectives within a multiobjective optimisation perspective, we avoid the pitfalls of blending incommensurable objectives (e.g., fuel vs. NOx) into a single scalar function [21]. Instead, we derive a Pareto front – a set of optimal solutions – where no objective can be improved without compromising at least one other. This approach provides a flexible decision-making tool for diverse user preferences, allowing stakeholders to select solutions that best align with their priorities.

From a methodological perspective, applying multiobjective optimisation to real-time speed generation in the presence of dynamic traffic is notably challenging. While traditional MPC can handle certain multi-objective problems by aggregating objectives into a single cost function, its sequential decision-making nature and reliance on finite-horizon optimisation can limit its capacity to capture global trade-offs [22]. In contrast, reinforcement learning – grounded in the convergence properties of Bellman's equations – offers a holistic, global approach that naturally considers long-term interactions among multiple objectives, making it more suitable for truly complex multi-objective optimisation scenarios. Therefore, we propose and compare two alternative approaches:

(i) **Two-Phase MPC:**

- **Phase-1**: Solve a traffic-free problem to obtain an "ideal" Pareto front that captures trade-offs between fuel consumption, NOx, and travel time in an uncongested environment.
- **Phase-2**: Integrate the sampled Pareto-optimal solutions into a real-time MPC framework, balancing the objectives and constraints in the presence of surrounding traffic.

(ii) **Multiobjective Deep Reinforcement Learning (DRL):**

- Simultaneously considers real-time traffic dynamics and user-defined weight preferences for fuel, NOx, and travel time.
- Exploits the compatibility between multiobjective optimisation and reinforcement learning [23,24], enabling an agent to learn speed control policies that yield different Pareto-optimal solutions.

We demonstrate the practicality and effectiveness of these two approaches using a highway segment in southern England based on both simulated and real-world traffic data. The vehicle's powertrain characteristics and emission rates are modelled based on GT-SUITE simulation data [25], enabling a realistic and detailed representation of fuel consumption and NOx generation. Although the two-phase MPC method offers a comparatively more straightforward integration of multiobjective solutions into an MPC framework, our results indicate that multiobjective DRL provides greater flexibility and superior performance in simultaneously balancing the three objectives. However, its reliance on training data and computational resources may limit its applicability to completely new traffic scenarios without additional training.

By shedding light on the strengths and limitations of these two approaches, this paper aims to contribute both methodologically and practically to the ongoing pursuit of greener transportation. The proposed framework illustrates how multiobjective optimisation can be leveraged to deliver not just a single solution, but an entire spectrum of speed profiles that can be tailored to different priorities and real-time traffic conditions.

To summarise, the primary innovations of our approach are summarised as follows:

- **Integrated Multiobjective Optimisation:** Simultaneously minimises fuel consumption, NOx emissions, and travel time, overcoming the limitations of single-objective approaches.
- **Real-Time Traffic Integration:** Explicitly incorporates real-world traffic dynamics to generate practical, adaptive speed profiles.
- **Dual Methodology:** Proposes both a two-phase MPC and a multi-objective DRL approach, offering flexible solutions to eco-driving challenges.
- **Enhanced Performance:** The DRL method demonstrates significant improvements over MPC, validated through UK highway cases.

The remainder of the paper is organised as follows. Section 2 provides an in-depth literature survey of eco-driving and optimal speed profile generation. Section 3 presents the vehicle modelling and the relationships between engine power, fuel consumption, and NOx emission. Section 4 introduces the multiobjective problem formulation and details the two proposed solution approaches. Section 5 discusses the experimental setup, results, and a comparison of the approaches. Finally, Section 6 draws conclusions and outlines directions for future research, including advanced multiobjective reinforcement learning techniques and the incorporation of other emissions such as PMs and COx.

2. Related work

2.1. Conventional approaches for energy focused eco-driving

Generating an optimised driving speed profile provides an effective way for reducing energy consumption and emission of pollutants. Various approaches have been proposed for eco-driving, in particular for generating speed profiles that minimise total fuel consumption. Typical conventional solution approaches used for generating optimal speed profiles include analytical/exact methods (e.g., mathematical programming or dynamic programming) and optimal control methods (e.g., MPC and its variants).

The concept of "Look-ahead Control" has been widely used in some works [26], which demonstrates the advantage of using available information on future disturbances. For instance, Sharma et al. [27] minimised fuel consumption of a heavy-duty vehicle by predicting the speed of the leading vehicle based on its uphill deceleration, achieving up to 8% fuel savings in real road scenarios. Other similar studies include [28] for minimising fuel consumption of heavy diesel trucks, Kamal et al. [10] for predicting the states of the preceding vehicle in urban scenarios at an adaptive look-ahead time step, etc. This advantage has been further applied in the cooperative driving scenario which employs aerodynamic drag reduction of platoons. For instance, Zhai et al. [29] proposed an ecological cooperative look-ahead control strategy based on distributed model predictive control (DMPC) for a platoon of automated vehicles on freeways with varying slopes, combining eco-driving and platooning technologies to maximise fuel efficiency. Kong and Ma [11] developed a cooperative eco-driving and energy management control strategy for heterogeneous vehicle platoons at multiple signalised intersections, leveraging a soft actor–critic (SAC)-based approach to optimise ecological velocity, safe inter-vehicle distance, and energy efficiency while maximising fuel economy and driving comfort. With the help of emerging vehicular communication technologies, a distributed optimal control scheme [30] is proposed to achieve cooperative highway driving at the level of individual vehicles, which demonstrates the improvement of fuel economy and traffic efficiency.

For long-haul applications, two-stage hierarchical frameworks decouple global route planning from local speed optimisation. For instance, Hamednia et al. [14] proposed a bi-level optimisation approach where gear selection is pre-optimised offline, and a nonlinear dynamic programme is solved online. By leveraging Pontryagin's Maximum Principle and a model predictive control framework, the method achieves up to 11.60% energy savings compared to average driving cycles. Furthermore, integrating advanced ICT technologies, such as cloud-based systems, can enhance real-time perception and decision-making. For example, Schlechtendahl et al. [31] introduced the concept of control system as a service (CSaaS), enabling cloud-deployed optimisation. Jia et al. [15] developed an enhanced cloud-based predictive cruise control (PCC) system, combining deep learning-based traffic prediction with adaptive MPC to optimise speed profiles under varying traffic conditions. Their method, tested on a UK highway segment, demonstrated improved fuel efficiency for heavy-duty vehicles (HDVs) by leveraging real-time traffic data and advanced computational techniques. In addition, Nie et al. [16] coupled gradient-based MPC for speed planning with MPC-based energy allocation in fuel cell hybrids, reducing traction power by 2.65% and battery degradation by 8.14%. Khalatbarisoltani et al. [32] propose a two-level eco-driving strategy for Connected Fuel Cell Vehicles (C-FCVs) to optimise speed trajectories and powertrain operation, addressing computational challenges and real-time traffic complexities. The top layer integrates an LSTM-based traffic predictor and an MPC framework to optimise speed while considering hydrogen consumption, ride comfort, and traffic efficiency, while the bottom layer employs decentralised MPC to allocate power optimally between fuel cells and the battery. Simulation results demonstrate that this strategy enhances ride comfort, reduces hydrogen consumption by 7.28%, and mitigates component degradation by 5.33%.

Drive cycle optimisation was also considered in some researches to minimise vehicle's fuel consumption. Mensing et al. [33] minimise a light-duty vehicle's fuel consumption, which demonstrates a 16% decrease relative to the New European Driving Cycle (NEDC) while preserving travel time and adhering to speed regulations. Cui et al. [13] proposes a Simulated Annealing (SA)-based method to develop driving cycles that better align with real-world speed-acceleration patterns, reducing errors by up to 23% compared to traditional methods and improving fuel consumption estimation accuracy. Additionally, Lot et al. [17] proposed an optimal control formulation for eco-driving in front-wheel drive electric vehicles, integrating driver preferences – such as desired speed, following distances, and smooth acceleration – with energy efficiency goals, using a simplified polynomial approximation of vehicle losses and relaxed regenerative braking constraints. Testing on a 25 km simulated journey shows 21% energy savings with only a 7% reduction in average speed, and 10%–15% energy savings in car-following scenarios without speed reduction.

2.2. Reinforcement learning based approaches

Recent advancements in cloud computing and artificial intelligence have enabled the integration of machine learning techniques, particularly reinforcement learning (RL), with traditional optimisation frameworks to address vehicle energy management challenges. Unlike conventional methods that often rely on heuristic rules or static models, RL-based approaches demonstrate unique capabilities in solving complex optimisation and optimal control problems through data-driven exploration of state–action spaces. This subsection systematically reviews emerging RL methodologies and their applications across diverse energy optimisation scenarios.

Hierarchical control architectures. A prominent trend involves hierarchical frameworks that decompose energy management tasks into coordinated layers. Hu and Li [34] developed an adaptive hierarchical energy management system (EMS) combining deep deterministic policy gradient (DDPG) with equivalent consumption minimisation strategy (ECMS) knowledge. This hybrid approach achieves near-optimal fuel consumption comparable to dynamic programming (DP) benchmarks while outperforming PID-based ECMS and rule-based strategies. The framework's efficient exploration mechanism demonstrates particular promise for real-world applications requiring safe online learning. Extending this concept, Dong et al. [35] proposed a three-layer flexible eco-cruising strategy (FECS) featuring: (1) Dijkstra-based lane planning considering long-term traffic impacts, (2) trigonometric speed optimisation for energy savings, and (3) robust trajectory tracking with safety guarantees. Stochastic simulations reveal significant cost reductions in moderate-flow and free-flow traffic scenarios.

Multi-objective optimisation. Addressing the inherent trade-offs in vehicular energy systems, Yang et al. [36] formulated hybrid electric vehicle energy management as a general-sum stochastic game solved through multi-agent RL (MARL). By modelling the engine-generator set and hybrid energy storage system as competing agents, their framework achieves Nash equilibrium solutions balancing fuel economy, battery degradation, and ultracapacitor state of charge. The MARL approach demonstrates superior performance over single-agent RL and DP in maintaining balanced objective optimisation. Similarly, Xia Jiang and Li [37] established a hierarchical Markov Decision Process (MDP) integrating car-following, lane-changing, and RL policies for electric connected vehicles. SUMO simulations at signalised intersections show substantial energy savings while maintaining safe interactions with human-driven vehicles.

Partial observability and complex environments. For realistic traffic scenarios with limited information, Yang et al. [38] developed autonomous eco-driving strategies using DDPG, PPO, and SAC algorithms combined with hybrid car-following models. Their framework enables connected and automated vehicles (CAVs) to optimise safety, energy efficiency, and ride comfort simultaneously when navigating signalised intersections. Comparative analyses reveal that the Hybrid-SAC variant surpasses human drivers and traditional models (Trigo, IDM) across all performance metrics. Addressing partial observability, Zhu et al. [39] framed multi-power-source CAV control as a Partially Observable MDP (POMDP) solved via proximal policy optimisation (PPO). The developed controller reduces fuel consumption by 17% versus human drivers while maintaining comparable travel times.

Integrated decision-making architectures. Recent innovations emphasise unified frameworks for simultaneous longitudinal and lateral control. Li et al. [40] introduced an attention-enhanced Twin Delayed DDPG (TD3) architecture incorporating multi-head self-attention and hybrid action representation. This integration achieves 42.18% stability improvement over prior methods while delivering 30.25% energy efficiency gains. Building on this, Fan et al. [41] proposed a TD3-based eco-driving strategy combining lane preference scoring with longitudinal speed planning. Their SUMO simulations demonstrate synergistic benefits: longitudinal control alone reduces travel time by 7.94% or energy consumption by 18.15%, while integrated lateral decisions further decrease both metrics by 5.7% and 1.75% respectively.

Customised multi-agent and deep learning techniques. Khalatbarisoltani et al. [42] proposes a decentralised health-conscious learning-based integrated thermal and energy management (ITEM) system for hybrid electric vehicles (HEVs) that optimises fuel consumption, driver comfort, and battery lifetime using a multi-agent deep reinforcement learning (MADRL) framework with long short-term memory (LSTM). The MADRL approach outperforms rule-based and single-agent strategies, reducing battery degradation by 48% while maintaining cabin comfort. Experimental validation through hardware-in-the-loop (HIL) testing confirms the reliability of the proposed method, with battery and cabin temperature deviations from simulation results remaining within 0.45 and 0.85 degrees, respectively. Jia et al. [43] propose a predictive energy management system (PEMS) for fuel cell hybrid electric buses (FCHEBs) using a twin delayed deep deterministic policy gradient (TD3) algorithm, integrating future driving conditions and a predictive passenger model to optimise operational costs. Experimental results show that the TD3-based PEMS reduces comprehensive operational costs by 5.92% compared to conventional TD3-based EMS with a fixed passenger count.

2.3. Energy and emission focused eco-driving

The analysed papers suggest a predominant focus on energy consumption when generating speed profiles for vehicles. Most of the reviewed literature emphasises energy use, sometimes considering travel time, while neglecting the assessment of NOx emissions due to its inherent technical complexities in quantification. In [18] explores speed profile generation considering both fuel and NOx, producing separate optimal profiles for each objective. It shows that optimising for fuel does not necessarily reduce NOx, and the study achieves significant reductions in both fuel consumption and NOx emissions through dynamic programming.

Huang et al. [44] investigate the impact of driver behaviour on real driving emissions (RDE) using a portable emission measurement system with 30 drivers (15 novice, 15 experienced) driving the same diesel vehicle on the same route. Results show that novice drivers are generally more aggressive, leading to slightly higher mean fuel consumption (2%) and significantly higher NOx (17%) and PM (29%) emissions than experienced drivers. However, individual driver differences play a more significant role than experience level, suggesting that adopting eco-driving skills could substantially reduce fuel consumption and emissions for the worst-performing drivers.

Tang et al. [45] present a strategy for managing energy and emissions based on a deep Q-network (DQN) as applied to dynamic programming (DP) as an optimal reference point. Two distributed deep reinforcement learning (DRL) algorithms, namely asynchronous advantage actor–critic (A3C) and distributed proximal policy optimisation (DPPO), were employed to propose EMSs. Afterwards, emission optimisation was incorporated to propose distributed DRL-based E&EMSs. Through simulation results, three control strategies based on deep reinforcement learning (DRL) show outstanding computational efficiency and near-optimal fuel economy. Compared to DQN, two distributed DRL algorithms improve learning efficiency by four times.

Guo et al. [46] introduces an advanced energy management strategy for fuel cell hybrid vehicles based on a duelling-double-deep Q-network (D3QN). The primary challenge addressed is achieving an effective trade-off between system degradation and hydrogen consumption, while minimising computational costs across diverse operational environments.

Yuan et al. [47] quantify the fuel use and emission reduction potential of eco-driving for light-duty gasoline vehicles (LDGVs) using three million seconds of real-world driving data from 160 drivers across eight routes and 199 segments. Using a Vehicle Specific Power modal model, results show that route-level eco-driving can reduce fuel use and emissions by 6% to 40% compared to average driving. While eco-driving generally leads to simultaneous fuel and emission reductions, trade-offs exist, highlighting the need for strategic decision-making in LDGV eco-driving.

Jia et al. [48] propose a novel cost-minimisation energy management strategy that integrates thermal safety, degradation awareness of lithium-ion batteries, and fuel cell aging suppression to balance durability and hydrogen consumption. Using an enhanced self-learning stochastic Markov predictor for speed prediction, the strategy reduces battery aging by 34.8% and total operating costs by 12.3% compared to conventional methods.

Han et al. [49] propose an energy management strategy that integrates a battery preheating technique – supported by a high-precision electro-thermal-aging model, grid- and battery-powered preheating methods, and optimisation algorithms (PSO and PMP) – to determine optimal preheating times and manage energy effectively. Simulation results demonstrate that at −20 °C, preheating can reduce energy usage by approximately 44%–48% compared to non-preheating scenarios.

Wang et al. [50] introduces an advanced energy system combining a solid oxide fuel cell (SOFC) with compressed air energy storage CAES to generate compressed air, electrical power, and heat. The system's performance was assessed and optimised using regression-based machine learning models, focusing on three key process variables: temperature, current density, and utilisation factor.

The closest work relevant to our paper is given by Yuval et al. [19], where an approach using multiobjective optimisation was introduced, aiming to create optimised speed profiles while simultaneously considering fuel consumption, NOx emissions and travel time under traffic-free conditions. This method represents a more favourable approach for handling problems featuring multiple objectives that cannot be directly compared. Rather than combining these objectives into a single weighted metric, the proposed approach offers a collection of non-dominated solutions (Pareto front). Each solution within this set reflects varying preferences concerning the importance of fuel, NOx and time. A standard shortest path model similar to Ozatay et al. [20] was designed to implicitly address several constraints, and was solved using linear programming. By obtaining the Pareto front for the three objectives, this approach provides a range of options for users or driving guidance systems to select tailored strategies according to their specific requirements. However, the study in [19] only considers traffic-free scenarios, which significantly narrows its applicability in real-world situations.

2.4. Contributions of our work

Our work distinguishes itself from the existing literature by addressing a critical gap: while many eco-driving studies focus solely on fuel consumption or treat NOx emissions separately – often overlooking the impact of real-time traffic – our paper presents the first multiobjective framework that optimises speed profiles for fuel consumption, NOx emissions, and travel time in an integrated manner. Unlike conventional approaches, which typically generate isolated or non-integrated speed profiles using methods such as dynamic programming, MPC, or even single-objective reinforcement learning, our Multiobjective Speed Profile Optimisation (MO-SPO) framework incorporates both a two-phase MPC and a novel DRL method that explicitly account for dynamic traffic constraints.

3. Vehicle modelling

In this part, we present the longitudinal vehicle dynamics model, along with employing simulated data to establish connections between engine power, fuel usage, and NOx emissions. By considering the vehicle's dynamics, the power output can be precisely computed by factoring in the road slope, road condition, and the driver's actions, indicated by changes in speed over a specific duration and the resulting acceleration.

3.1. Vehicle dynamics

In our research, we utilise a vehicle's longitudinal dynamics model, following the convention from previous work such as Ozatay et al. [20], Jia et al. [15], Fernández-Yáñez et al. [18] and Yuval et al. [19]. We use M_e to denote the effective mass of the vehicle, which accounts for both the vehicle's actual mass and the rotational inertia of its wheels. The term $\frac{dv}{dt}$ represents the vehicle's acceleration, describing the rate of change of its velocity v over time. The forces acting on the vehicle include F_{eng} for the tractive force generated by the engine, F_{brk} for the braking force, F_{rol} for the rolling resistance force, F_{aro} for the aerodynamic resistance force, and F_{grd} for the road grade resistance force. The rolling resistance force F_{rol} is calculated using the vehicle mass M_v, gravitational acceleration g, rolling resistance coefficient C_r, and the cosine of the road gradient $\theta(t)$. The aerodynamic resistance force F_{aro} depends on the air density ρ, frontal area A_f, aerodynamic drag coefficient C_d, and the square of the vehicle's speed $v(t)$. The road grade resistance force F_{grd} is determined by the vehicle mass M_v, gravitational acceleration g, and the sine of the road gradient $\theta(t)$. Finally, the effective mass M_e incorporates the vehicle mass M_v and the rotational inertia of the wheels, calculated using the number of wheels N_w, rotational inertia of each wheel J_w, and wheel radius R_w. The complete model reads,

$$M_e \frac{dv}{dt} = F_{eng} - F_{brk} - F_{rol} - F_{aro} - F_{grd}. \tag{1}$$

$$F_{rol} = M_v g C_r \cos(\theta(t)). \tag{2}$$

$$F_{aro} = \frac{1}{2}\rho A_f C_d v(t)^2. \tag{3}$$

$$F_{grd} = M_v g \sin(\theta(t)). \tag{4}$$

$$M_e = M_v + N_w \frac{J_w}{R_w^2}. \tag{5}$$

The vehicle's resulting force can be straightforwardly calculated by applying Eq. (1) to (5). The term representing the overall force generated by the vehicle, denoted as $F_{veh} := F_{eng} - F_{brk}$, is established. To calculate the tractive force F_{eng} and braking force F_{brk}, we operate under the assumption that efficient driving avoids simultaneous use of throttle and brake, a premise found in various pertinent studies like [18,19]. This assumption assumes that at any given time, either F_{eng} or F_{brk} must be zero, determined as follows: When $F_{veh} \geq 0$, then $F_{eng} = F_{veh}$ and $F_{brk} = 0$; if $F_{veh} < 0$, then $F_{eng} = 0$ and $F_{brk} = F_{veh}$.

After determining the tractive force F_{eng}, we establish the engine power P_{eng} using the predetermined vehicle specifications. Subsequently, fuel consumption and NOx emissions are derived from this engine power. We will now elaborate on this process.

3.2. Fuel and NOx rate functions based on simulation

Based on of the simulated vehicle, we applied a third-order polynomial fit to establish the relationships between NOx rates $\dot{m}_N$ and engine power P_{eng}, and a first-order polynomial fit was used to simulate between fuel $\dot{m}_f$ and engine power P_{eng}. These relationships are derived from simulation data obtained through experiments conducted using the GT-SUITE [25] package. Appendix A.1 elaborates the simulation environment and vehicle modules used for deriving such relationships.

In this study, the GT-SUITE powertrain and emission model parameters were adopted from the rigorously validated work of Gao et al. [51]. Their validation process included experimental comparisons under diverse driving conditions, such as the Worldwide Harmonised Light Vehicles Test Cycle (WLTC), and covered critical scenarios like cold-start emissions, SCR/ACCT system efficiency, and thermal dynamics of after-treatment systems. Specifically, fuel consumption and NOx emission simulations were benchmarked against experimental data, showing strong agreement (e.g., minor deviations in NOx rates and fuel consumption trends). By leveraging this pre-validated model, we ensure that our eco-driving analysis reflects real-world powertrain and emission behaviours across the operational scenarios examined in this work.

For NOx, the relationship is:

$$\dot{m}_f = \alpha_1 P_{eng}^3 + \alpha_2 P_{eng}^2 + \alpha_3 P_{eng} + \alpha_4, \quad P_{eng} \geq 0, \tag{6}$$

where $\alpha_1 = 9.207 \times 10^{-20}, \alpha_2 = 1.663 \times 10^{-14}, \alpha_3 = 2.076 \times 10^{-10}$, and $\alpha_4 = 4.204 \times 10^{-7}$. The R^2 of the fitting is 0.97.

For fuel, the relationship is:

$$\dot{m}_N = \beta_1 P_{eng} + \beta_2, \quad P_{eng} \geq 0, \tag{7}$$

where $\beta_1 = 5.937 \times 10^{-8}, \beta_2 = 0.0001002$. The R^2 of the fitting is 0.94.

The situation of negative engine power (i.e., $P_{eng} < 0$) did not happen in our experiments, since $F_{eng} \geq 0$ and $v(s) \geq 0$ always hold. If it is to be included, as in several other research cases [52,53], a common practice is to set an additional condition such that $\dot{m}_f = \alpha_0$ and $\dot{m}_N = \beta_0$ if $P_{eng} < 0$, which can be easily incorporated into our model if needed. Based on Eqs. (6) and (7), the static relationships of fuel consumption and NOx emissions with both zero and varying road grades while maintaining a constant vehicle speed are illustrated in Fig. 1. Note that since the road grade typically varies along the observed journey section, these static relationships provide only idealised results.

4. Problem formulation and two alternative solution approaches

In this section, we first introduce the fundamentals of multiobjective optimisation and the overall speed profile generation problem as an optimal control problem. We then delve into the two alternative solution approaches both offering innovative ways to tackle the challenge of simultaneously optimising fuel consumption, NOx emissions, and travel time considering surrounding traffic. The first approach is based on traditional optimisation and control. It divides the problem into two phases, applying multiobjective optimisation in a traffic-free scenario and then using model predictive control to address real-time traffic scenarios. The second approach combines multiobjective deep reinforcement learning with real-time traffic considerations, allowing for direct weighting of preferences to obtain optimised speed profiles. Through these approaches, we aim to enhance eco-driving strategies and promote more sustainable and efficient transportation solutions.

4.1. Multiobjective optimisation

Multiobjective optimisation [21] is a technique used to handle problems with multiple, often incomparable, objectives. Instead of seeking a single optimal solution, it aims to find a set of solutions known as the Pareto front, where no other solution can improve one objective without sacrificing another. This approach provides decision-makers with a range of trade-off options, allowing them to select the most suitable solution according to their preferences and requirements.

The modelling of multi-objective optimisation for eco-driving is of paramount importance due to the diverse and often conflicting preferences of users, as well as the inherent uncertainties in real-world driving scenarios [54,55]. Traditional eco-driving strategies typically prioritise single objectives, which may not adequately address the

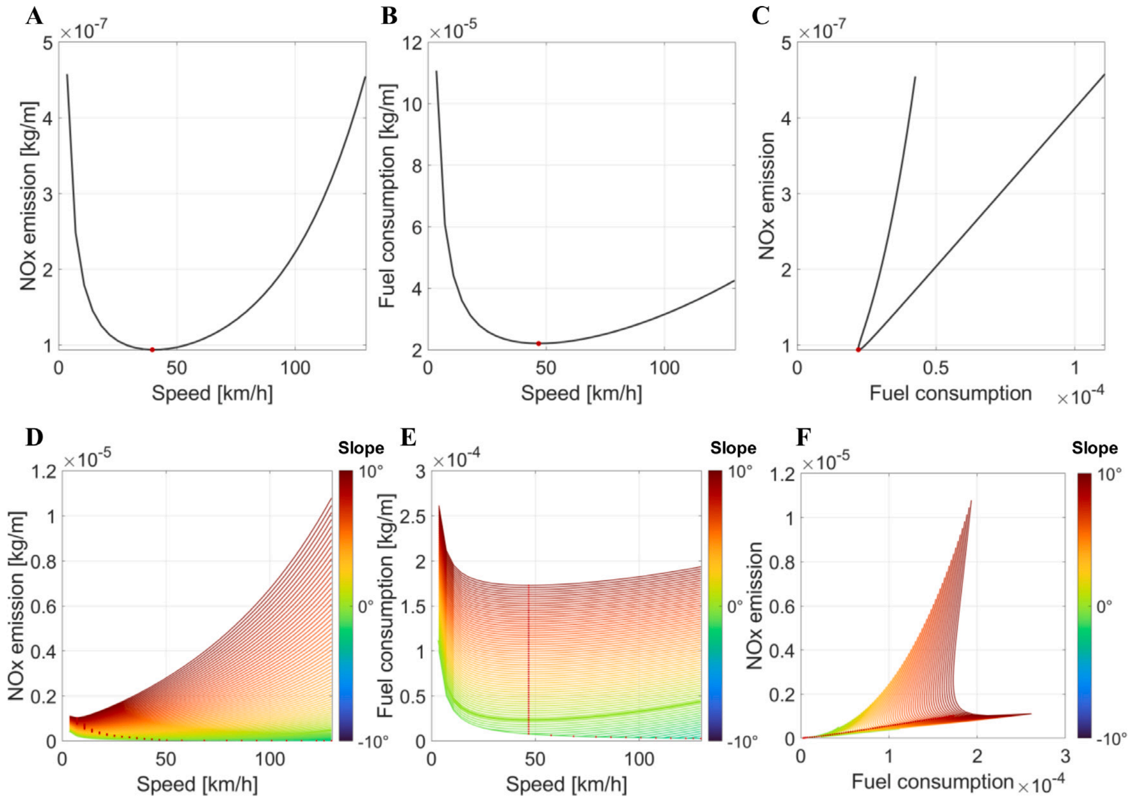


Fig. 1. Investigating the relationship between NOx emissions and fuel consumption at a constant vehicle speed and varying road gradients. A–C: Fuel consumption (A) and NOx emissions (B) are examined with respect to a constant vehicle speed on a flat road. The relationship between fuel consumption and NOx emissions is shown in C. The red dots indicate the minimum values. D–F: Similar analyses to A–C are conducted, but with a range of constant road slopes from −10° to 10°.

multifaceted priorities of drivers. For instance, while some drivers may prioritise energy efficiency, others may place greater emphasis on minimising travel time or enhancing driving comfort. This paper underscores the necessity of integrating multiple objectives, including fuel consumption, NOx emissions, and travel time, into a cohesive framework to deliver tailored eco-driving recommendations. Additionally we highlight the critical role of accounting for real-time dynamics in traffic conditions, where the unpredictable behaviours of leading vehicles can significantly influence energy efficiency, NOx emission and travel time. Addressing these complexities is crucial for the widespread adoption of eco-driving practices, where real-time adaptability and user satisfaction are key to achieving both environmental and operational goals.

Let there be K distinct objectives, each representing an aspect to be minimised and denoted by $z_k(x), k = 1, \ldots, K$, and these objectives are not directly comparable:

$$\text{minimise} \quad \{z_1(x), z_2(x), \ldots, z_K(x)\}. \tag{8}$$

Solution x is said to *dominate* solution x' if x is better than or the same as x' for all objectives, i.e., $z_k(x) \leq z_k(x'), \forall k = 1, \ldots, K$, and there exists at least one objective where x is strictly better than x', i.e., $\exists k : z_k(x) < z_k(x')$. A non-dominated (efficient) solution refers to a feasible solution within a set that is not surpassed by any other feasible solutions. The collection of all these non-dominated solutions is termed the Pareto-optimal set. The boundary delineated by the points derived from this Pareto-optimal set is known as the Pareto front (frontier). In multiobjective optimisation problems, the goal is to discover a diverse set of solutions situated along this Pareto front. Common methods used to generate a Pareto front include techniques like weighted sum, ϵ-constraint, and weighted metric methods [21].

4.2. Speed profile generation as an optimal control problem

4.2.1. Original optimal control problem

In this section, we outline the overall optimal control problem focused in our research. The aim is to generate a speed profile that minimises specific objectives throughout a total distance travelled, denoted as S. As the longitudinal model operates within the spatial domain, we apply the following domain transformations: $dt = \frac{ds}{v(s)}$ and $\frac{dv}{dt} = \frac{dv}{ds}\frac{ds}{dt} = \frac{dv}{ds}v$, in a way that the distance travelled s becomes an independent variable and $\dot{m}(t)dt = \frac{\dot{m}(s)}{v(s)}ds$ corresponds to a rate $\dot{m}(t)$ originally measured with respect to time. When the travel velocity $v(s)$ is known, the engine power ($P_{eng}(s) = \frac{F_{eng}}{v(s)}$) can be exclusively determined using Eqs. (1)–(5) provided that the velocity and acceleration are identifiable. Subsequently, the fuel consumption and NOx emission rates can be computed using Eqs. (6) and (7). Three objectives are identified in our multiobjective optimisation framework:

(i) Total fuel consumption: $J_f = \int_0^S \frac{\dot{m}_f(P_{eng}(s))}{v(s)} ds$,
(ii) Total NOx emission: $J_N = \int_0^S \frac{\dot{m}_N(P_{eng}(s))}{v(s)} ds$, and
(iii) Total travel time: $J_T = \int_0^S \frac{1}{v(s)} ds$.

Our goal is to minimise the three objectives while taking various preferences into account:

$$\text{minimise} \quad \{J_f, J_N, J_T\}. \tag{9}$$

As per convention, necessary normalisation is needed for the three objectives in Eq. (9) in an multiobjective optimisation context. In our MO-SPO framework, we adopt the weighted sum method, one of the most widely used techniques [21]. This approach assigns a weight to each objective and combines them into a single objective function. By systematically varying the weights, different regions of the Pareto frontier can be explored as comprehensively as possible. Specifically, the objective function is formulated as,

$$J = w_f J_f + w_N J_N + w_T J_T, \quad w_f + w_N + w_T = 1 \tag{10}$$

where w_f, w_N and w_T are the weights associated with fuel, NOx and time objectives respectively. In addition to the objectives, the following constraints are included into our model to guarantee practical driving scenarios in real-world.

The speed is restricted within the range of minimum speed limit $v_{\min}$ to maximum speed limit $v_{\max}$ for all velocities $v(s)$. This range ensures adherence to legal speed limits on the motorway. Additionally, a lower bound may be included if specified by the local traffic authority. The vehicle's acceleration is confined within the range of maximum acceleration limit $-a_{\max}$ to $a_{\max}$ for all velocities $v(s)$. This limitation is implemented to prioritise the safety and comfort of the driver and passengers (Table 1). The initial and final states of the journey entail the vehicle being stationary, indicated by the conditions $v(0) = v(S) = 0$. Standing condition $\frac{dv}{dt} \neq 0, \forall v = 0$ is imposed such that when the speed reaches zero, the acceleration must not be zero to prevent the vehicle from remaining stationary indefinitely. When a vehicle navigates through traffic, its movement is influenced by the presence and behaviour of other vehicles nearby. These neighbouring vehicles create constraints that impact how the vehicle can manoeuvre or accelerate, making it essential to consider these limitations when planning or controlling its movement.

We denote the above constraints as a constraint set $\mathscr{D}$. Depending on the specific requirements, more constraints apart from the above ones can be included into $\mathscr{D}$. Note that our driving model focuses on motorway conditions and does not account for signal stop points. Nevertheless, these can be readily integrated into the model depending on the chosen settings.

4.2.2. Discretised optimal control problem based on road position

Similar to Ozatay et al. [20] and Jia et al. [15], the total distance S is discretised into Q equal intervals $\Delta s = S/Q$. This allows a variety of approaches, such as shortest path, MPC and DRL to be applied in practically solving the original optimal control problem. We further make the assumption that the acceleration remains unchanged within each interval $i = 1, \dots, Q$, and denote it as a_i. Then the speed profile can be derived by determining the start and end speed of each interval i (denoted as v_{1i} and v_{2i}), or equivalently, by determining the acceleration $a_i = \frac{v_{2i}^2 - v_{1i}^2}{2\Delta s}$ of interval i if v_{1i} is given. Note that the time needed within interval i is $\Delta t_i = \frac{2\Delta s}{v_{1i}+v_{2i}}$. Therefore the fuel consumption and NOx emission incurred at interval i can be calculated by $\dot{m}_f(P_{eng}(i))\Delta t_i = \frac{2\dot{m}_f(P_{eng}(i))}{v_{1i}+v_{2i}}\Delta s$ and $\dot{m}_N(P_{eng}(i))\Delta s = \frac{2\dot{m}_N(P_{eng}(i))}{v_{1i}+v_{2i}}\Delta s$ respectively, where $P_{eng}(i)$ is the engine power at interval i by applying Eqs. (1)–(5).

Three discretised objective terms on the distance domain can be further defined, corresponding to the original objectives in (9):

(i) Total fuel consumption: $J'_f = \sum_{i=1}^{Q} \frac{2\dot{m}_f(P_{eng}(i))}{v_{1i}+v_{2i}}\Delta s$,
(ii) Total NOx emission: $J'_N = \sum_{i=1}^{Q} \frac{2\dot{m}_N(P_{eng}(i))}{v_{1i}+v_{2i}}\Delta s$, and
(iii) Total travel time: $J'_T = \sum_{i=1}^{Q} \frac{2}{v_{1i}+v_{2i}}\Delta s$.

In a multiobjective optimisation framework, we aim to minimise the three objectives considering different preferences:

$$\text{minimise} \quad \{J'_f, J'_N, J'_T\}. \tag{11a}$$

$$\text{subject to} \quad \mathscr{D}. \tag{11b}$$

The final objective after adopting weighted sum remains in the same form as in Eq. (10).

We propose two alternative solution approaches to deal with the above multiobjective optimal control problem in Sections 4.3 and 4.4 respectively.

4.3. A two-phase approach using shortest path and MPC

As our objective is to generate a speed profile based on real-time traffic conditions, it becomes imperative to consider the influence of surrounding traffic, which sets it apart from traffic-free scenarios. MPC has traditionally been an effective tool for addressing such real-time problems. However, when combined with multiobjective optimisation, MPC encounters significant challenges, and despite considerable efforts made in the past few decades, there is no satisfactory generic method to obtain exact or high-quality solutions [56,57]. Due to the successive computational nature of MPC, the results are often not Pareto optimal [58].

Considering the above challenges mentioned, we propose an approximate two-phase approach, striking a balance between the "desirable" Pareto-efficient speeds obtained from the traffic-free condition (Phase-1) and the adjusted speeds due to surrounding traffic, computed using an MPC model (Phase-2). In Phase-1, in the absence of surrounding traffic, the speed profiles are ideally designed to minimise fuel consumption, NOx emissions, and travel time, based on vehicle specifications and road geometry information. However, these profiles may not be practical or entirely feasible due to the lack of surrounding traffic considerations. In Phase-2, realistic solutions are generated by a conventional MPC model, accounting for other vehicles' presence, while endeavouring to maintain speeds as close to those obtained in Phase-1 as possible. The subsequent sections provide a comprehensive elaboration of both phases.

4.3.1. Phase-1: traffic-free shortest path problem formulation

Fig. 2 gives an illustration of the framework of Phase-1. In Phase-1, given the absence of surrounding traffic, the discretised multiobjective optimal control problem represented by (11) can be further reformulated as a deterministic shortest path problem [19,20], if the speed horizon is also discretised into $[0, \Delta v, 2\Delta v, \dots, v_{max}]$. This yields a shortest path network defined over $[0, \Delta s, 2\Delta s, \dots, S] \times [0, \Delta v, 2\Delta v, \dots, v_{max}]$, where each node (s, v) in the network represents a chosen speed v at a distance s, and an arc represents the costs (NOx, fuel and time) from one node to another, i.e., how speed changes from one distance point to the next. This shortest path problem is solvable using standard mathematical programming. The outcome of Phase-1 yields a Pareto front, illustrating various trade-offs among the preferred weight settings, where each point on the Pareto front corresponds to a complete speed profile. A set of sampled points $p \in \mathscr{P}$ will be collected from the Pareto front and be used as reference points for Phase-2. For details in how to formulate the shortest path problem in the context of generating speed profiles, see examples from [19,20] (see Fig. 2).

4.3.2. Phase-2: MPC problem considering surrounding traffic

In [15], an MPC model is proposed to generate speed profiles that only minimise fuel consumption considering surrounding traffic. Its objective function for an interval i and total prediction horizon n reads,

$$J(i) = \lambda_e \sum_{j=i}^{i+n-1} E_e(j)^2 + \lambda_k \sum_{j=i}^{i+n-1} (E_k(j) - \frac{1}{2} M_e v_d^2(j))^2 + \lambda_s \sum_{j=i}^{i+n-1} (E_e(j) - E_e(j-1))^2. \tag{12}$$

where the first term minimises the engine energy E_e (fuel), the second term minimised the deviation between the actual speed (represented by kinetic energy E_k) and the desired speed v_d and the third term minimises jerk (represented by energy increment) to ensure driver's comfort. λ_e, λ_k and λ_s are the corresponding weights.

We have developed an MPC model based on [15] to account for the surrounding traffic while aiming to keep the speed profile as close as possible to the sampled Pareto solutions from Phase-1. A detailed description of this MPC algorithm can be found in Appendix A.3. Fig. 3 provides an illustration of how our Phase-2 operates: the MPC model is

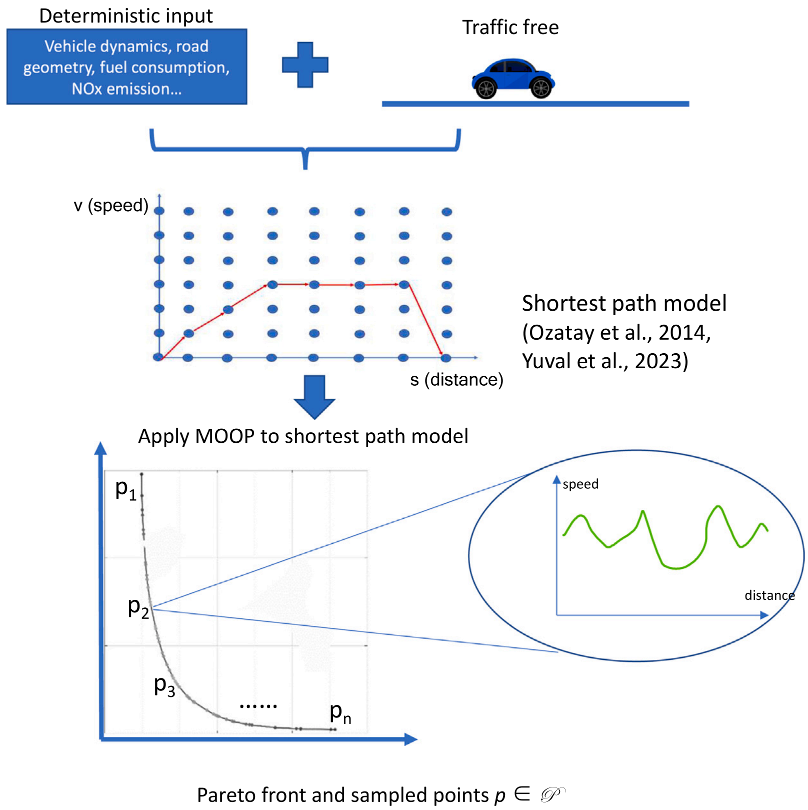


Fig. 2. An illustration of Phase-1 where the traffic-free scenario is modelled as a shortest path problem and applied to a multiobjective optimisation framework. Sampled points $p_1, p_2, \ldots, p_n \in \mathscr{P}$ correspond to different efficient speed profiles with respect to their own preferences.

employed for generating vehicle speed profiles considering surrounding traffic for each sampled points $p \in \mathscr{P}$ from Phase-1. The vehicle's dynamics accounts for various constraints $\mathscr{D}$ including surrounding traffic (headway) and speed/acceleration limits. The MPC indirectly optimises the three objectives (fuel, NOx and time) by minimising the deviation between v_d and the reference Pareto point p. The entire process is conducted over a finite distance, which is divided into discrete steps. The controller predicts the vehicle's future behaviour within the horizon, subject to the constraints $\mathscr{D}$. At each time step, MPC solves an optimisation problem to find the optimal control input sequence. Then, the controller shifts the horizon by one step and updates the information with new measurements.

It should be noted that we adopted a data-driven traffic predictive model for speed prediction which applies the CNN-based deep learning method to capture spatio-temporal dependencies in traffic data [15]. The multi-view CNN processes multiple factors (e.g., traffic flow, speed) separately through convolutional layers, fuses their outputs, and predicts traffic speed via fully connected layers. The model uses a weighted loss function to balance contributions from different traffic factors. Predicted speeds are transformed from the time domain to the space domain for use in predictive control systems, enabling real-time speed optimisation.

To realise the above, for each sampled Pareto point $p \in \mathscr{P}$, we set the desirable speed v_d in the MPC's objective function (see [15]) dynamically depending on the speed of the front vehicle v_f and the Pareto speed $v_P(p)$ derived from Phase-1. Two strategies are designed to address the problem from different aspects: a conservative MPC strategy ("MPC1") and a balanced MPC strategy ("MPC2").

Conservative MPC strategy (MPC1): Under this strategy, the new desirable speed v_d is calculated as follows:

$$v_d = \begin{cases} v_P(p), & \text{if } v_P(p) \le v_f, \\ v_f, & \text{if } v_P(p) > v_f. \end{cases} \tag{13}$$

The justification for the conservative MPC tactic, as described in Eq. (13), is that in order to maintain maximum safety, the speed of the targeted vehicle must not surpass that of the front vehicle at any given time. Furthermore, the vehicle following the targeted one will regulate its speed in tandem with the targeted vehicle, and the whole set of traffic behind them will do likewise. Note that this strategy cannot guarantee that the speed of the current vehicle will never exceed that of the preceding vehicle, as v_d can only be approached as much as possible in objective. However, this approach has the advantage that whenever the Pareto speed $v_P(p)$ is less than the speed of the preceding vehicle v_f, the MPC will attempt to achieve $v_P(p)$, resulting in solutions with higher quality in terms of the three objectives.

Balanced MPC strategy (MPC2): Since safety headway constraints are included in the MPC model [15], it is considered safe to occasionally allow the desired speed to be higher than the speed of the front car. Therefore, in the balanced strategy, the desired speed is calculated as the average value of the Pareto and front car speeds, i.e.,

$$v_d = \frac{v_P(p) + v_f}{2}. \tag{14}$$

This approach increases the likelihood of the current vehicle surpassing the front vehicle's speed when prioritising minimisation of travel time. It provides more realistic speed profiles, but may result in lower solution quality than "MPC1", since the desired speed will only match the Pareto speed if $v_P(p) = v_f$.

4.4. A deep reinforcement learning approach

Reinforcement learning (RL) enables agents to learn decision-making strategies for maximising cumulative rewards in sequential processes [59]. Deep reinforcement learning (DRL) employs multi-layer Artificial Neural Networks (ANNs) for training in simulated environments. Here, the agent interacts with the environment, receives

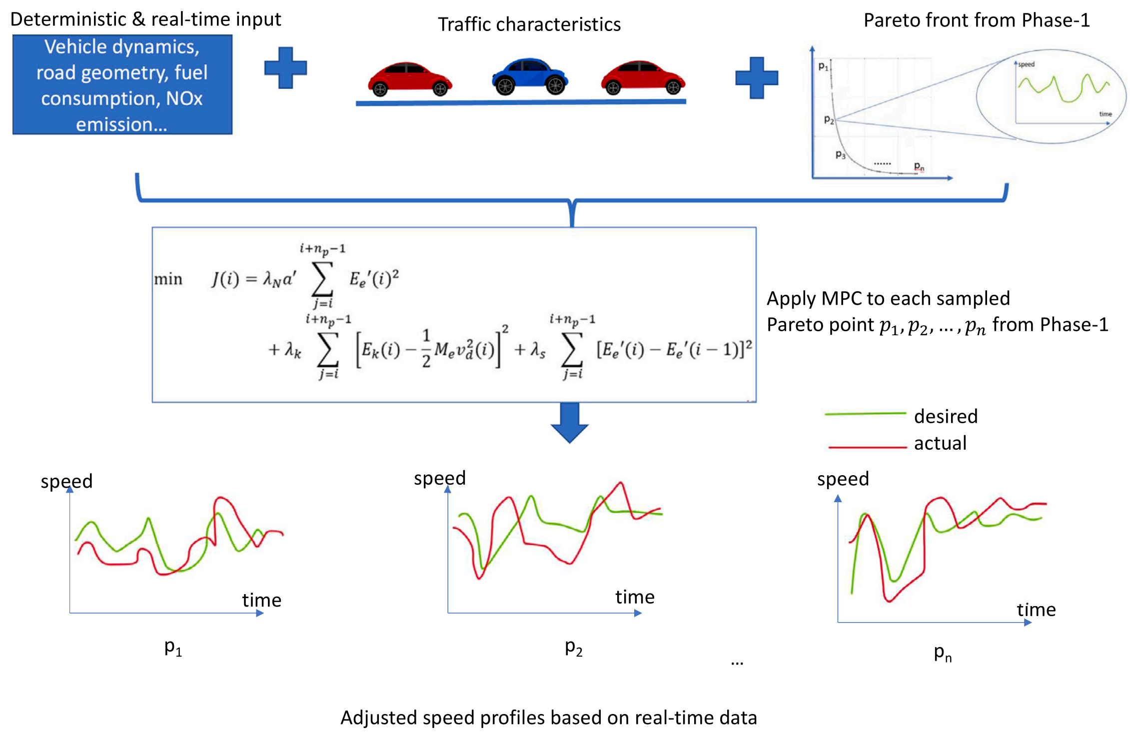


Fig. 3. An illustration of Phase-2 where based on the sampled Pareto points $p_1, p_2, \ldots, p_n \in \mathscr{P}$ from Phase-1, the surrounding traffic is taken into account in an MPC model.

feedback on actions, and improves decision-making through trial and error. This study focuses on continuously controlling the starting point's acceleration in each section to achieve a speed profile that addresses multiple diverse objectives. To address complex control tasks with continuous state and action spaces, we use an actor–critic framework with the deep deterministic policy gradient (DDPG) algorithm [59]. The actor–critic architecture, resembling a Generative Adversarial Network, consists of two ANNs: the "critic" estimates state transition values, guiding decisions, and the "actor" selects optimal actions based on critic feedback. The actor uses a Policy-based method for high-dimensional and continuous action spaces, and the critic employs a Value-based method for efficiency and stability. The iterative interaction in the actor–critic framework is depicted in Fig. 4. The black lines represent the predicting loop, while the red lines represent the training loop. The squares depict the agents and the environment, and the ellipses represent the information flow. The red circle represents to update the weights of ANNs for a given state–action pair.

The DRL approach is shown in Fig. 5. The state is formulated by traffic speed v_f, driving speed v, headway distance δ, and gradient θ. The action determines the speed variance a, which is a continuous value, at the upcoming road section. Negative values indicate deceleration, while positive values indicate acceleration. The action is constrained by the limits specified in Table 1, ensuring the agent's acceleration or deceleration stays within acceptable bounds during the control process. Consequently, the agent can adjust its speed within the speed limit and efficiently navigate through the road section. The agent updates the state at the beginning of each road section and receives the reward after traversing the section with the given speed. The reward is formulated by combining the three objectives of the optimisation in Eq. (11), namely fuel consumption, NOx emissions, and travel time. These values are normalised and combined into the same weighted objective. Due to the nature of minimisation, the reward is inversely proportional to the values of fuel, NOx, and travel time.

In this study, success is defined as the vehicle safely traversing the road without a crash, and failure occurs when a crash happens. The step reward is provided after each action, but they do not distinguish between success and failure. Increasing control accuracy results in more decision points, potentially leading to sparse rewards before task completion. Too few penalties may reduce the agent's cautiousness, resulting in numerous crashes in the initial stages of learning, hindering the ability to successfully complete the task. Conversely, too large penalties may lead the agent to adopt overly conservative actions, such as driving slowly to maintain a safe distance from the front car, which is not desired. Experiencing excessive failures during training can lead the agent to adopt a conservative behaviour, often referred to as the *coward effect* in reinforcement learning [60]. This is primarily attributed to the agent's exploration of the environment resulting in infrequent successes. Over time, the agent begins to perceive the game as consistently ending in failure. Consequently, its strategy shifts towards surviving longer rather than optimising reward acquisition.

To prevent the agent from being stuck in local optimal solutions and to mitigate the coward effect, an episode reward is designed each time the agent completes an episode, whether it is a failure or success. The termination condition is determined by two standard criteria: (i) either the agent crashes the front vehicle during the experiment (indicating failure) or (ii) the agent completes the entire journey through the road (indicating success). When the agent fails, it receives a penalty, which is discounted by the length it travelled. This means that the longer the agent travels before failure, the less severe the penalty. When the agent completes the task, it receives a reward, but the reward is discounted based on how the objective values achieved by the agent compare to the values derived from the ideal condition (representing traffic-free solution). This means that the agent receives a higher reward for achieving objectives closer to the ideal values. Accordingly, the agent receives an evaluation after termination based on its ending state, which is calculated by Eq. (15). γ_+ and γ_- are coefficients to balance the value of episode reward and step reward, which avoids the gradient vanishing during training. β_1 is a parameter to control the discounting rate. Obj_I is the weighted sum of objectives derived from traffic-free solution, and Obj_E is that derived from this episode. The tendency of the episode reward is shown in Fig. 6. The penalty curve (blue) follows an exponential shape, which penalises the agent more when the agent fails early, but imposes only a slight penalty if it fails near the end of the road. On the other hand, the reward curve (green) follows a linear

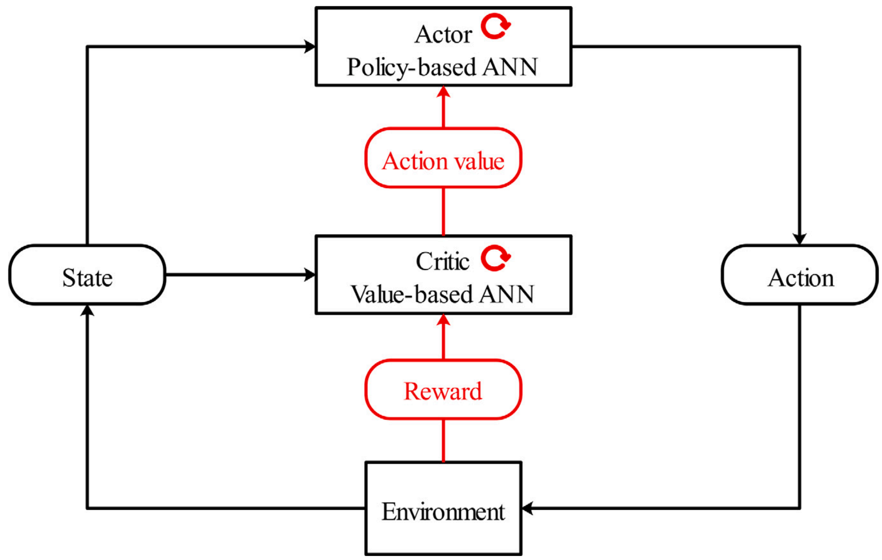


Fig. 4. Actor–critic training framework.

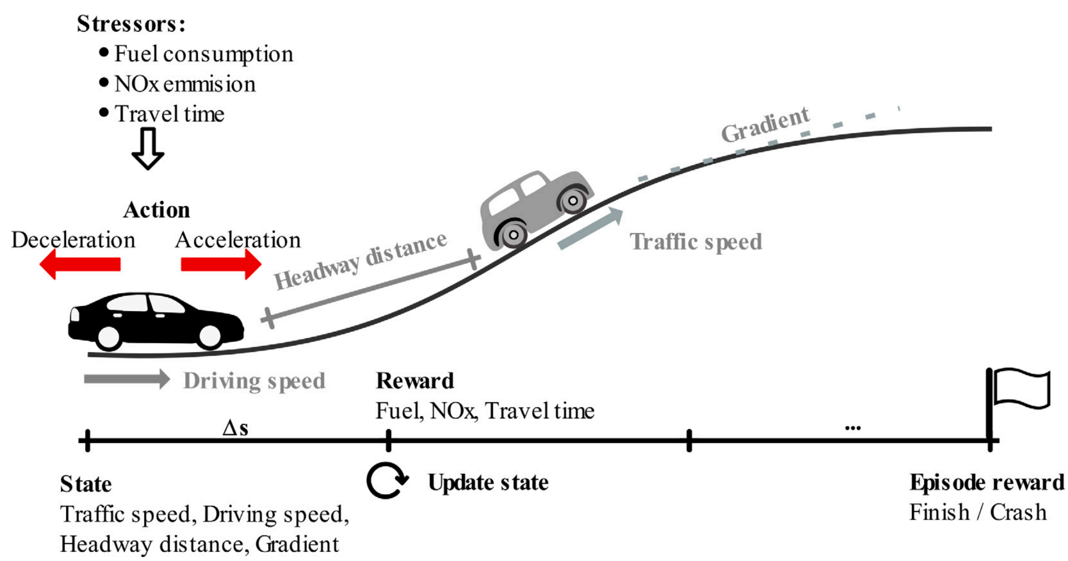


Fig. 5. DRL framework for eco-driving with traffic flow.

shape, which uniformly increases as the objective becomes better. A linear-shaped function imparts a consistent and gradual reward as the agent performs better, thereby reducing the intricacy of stimulation and preventing the agent from getting trapped in local performance optima.

$$\text{Episode reward} = \begin{cases} \gamma_+ \times \dfrac{Obj_I}{Obj_E}, & \text{Finish} \\ \gamma_- \times \left(e^{\beta_1(\sum \Delta s - S)} - 1\right), & \text{Failure} \end{cases} \tag{15}$$

The training process of ANN can be viewed as solving a parametric optimisation problem through stochastic gradient descent, which iteratively updates the parameters of ANN to minimise the loss function. The training process by DDPG algorithm follows the pseudo code in Algorithm 3 in Appendix A.4.

The policy gradient method with time-difference error can be summarised with the following equations. A tuple (s_i, a_i, r_i, s_{i+1}) represents the state, action, reward, and next state, respectively. First, compute the target value y_i by the target critic-network Q_{target} with weight set $\theta^{\text{target\_critic}}$ and target actor-network μ with weight set $\theta^{\text{target\_actor}}$ by Eq. (16).

$$y_i = r_i + \gamma Q_{\text{target}}(s_{i+1}, \mu(s_{i+1}|\theta^{\text{target\_actor}})|\theta^{\text{target\_critic}}) \tag{16}$$

Here, γ is the discount factor, which represents how much importance is given to future rewards.

Then, calculate the loss function of critic network using the memory set containing N samples. This is done by employing the mean square error as shown in Eq. (17). Subsequently, the weights of the critic network are updated using the gradient descent method with the gradient $\nabla\mathscr{L}(\theta^{\text{critic}})$.

$$\mathscr{L}(\theta^{\text{critic}}) = \frac{1}{N}\sum_i^N (Q(s_i, a_i|\theta^{\text{critic}}) - y_i)^2 \tag{17}$$

The loss function of actor network is defined as the negative mean of the expected Q-values for the state–action pairs in the batch in Eq. (18). This means that the actor seeks to minimise the negative Q-values, effectively maximising the Q-values. Then, the gradient decent $\nabla\mathscr{L}(\theta^{\text{actor}})$ is performed to update the weight of actor network.

$$\mathscr{L}(\theta^{\text{actor}}) = -\frac{1}{N}\sum_i^N Q(s_i, \mu(s_i|\theta^{\text{actor}})|\theta^{\text{critic}}) \tag{18}$$

After each training round, the target critic network is updated using a decay rate τ. This update is performed to prevent rapid changes in target Q-values, which can lead to unstable learning.

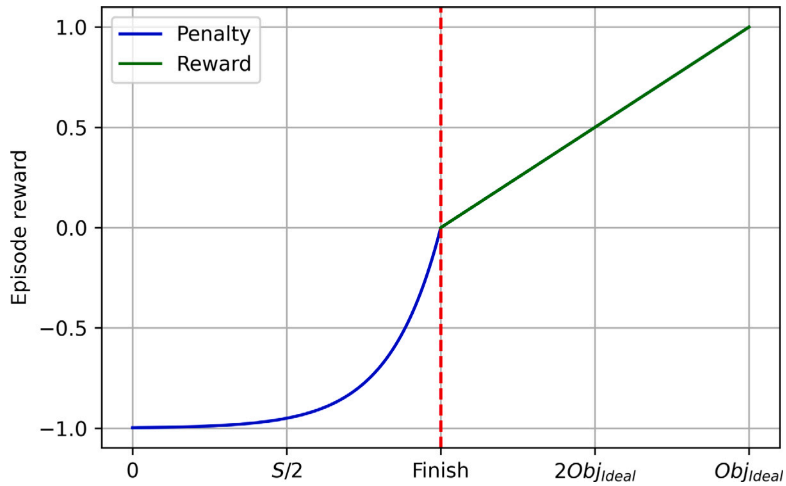


Fig. 6. Episode reward under different termination states.

The action is selected following the ε-greedy method where ε is the explorative parameter. Before each action is selected, a random number is generated according to a uniform distribution between 0 and 1. If the random number is greater than ε, the action with the highest probability is selected. Otherwise, an action is selected randomly. For sufficient exploration at the initial process of the simulation, ε is annealed in a sigmoid shape as

$$\varepsilon = \varepsilon_{\max} - \frac{\varepsilon_{\max} - \varepsilon_{\min}}{1 + e^{-\beta_2[(E-\beta_3)-\beta_4]}} \tag{19}$$

where $\varepsilon_{\max}$ and $\varepsilon_{\min}$ are the lower and upper bounds, respectively. β_2, β_3 and β_4 are the parameters to control the shape of annealing. E stands for the number of experienced episodes. The value of ε decays with the increasing of episode number.

The learning rate of the actor and critic ANNs is also decayed during iteration, following an exponential shape. The decaying learning rate ensures that the networks adapt to changing dynamics and avoids overshooting or getting stuck in local minima during the learning process.

5. Computational experiments

5.1. Experiment environment and dataset

In this section, we present the experimental results obtained by applying both the two-phase approach (shortest-path + MPC) and the novel DRL-based approach to generate optimised speed profiles for our focused passenger car. The surrounding traffic was simulated using SUMO [61], an open-source traffic simulation software that allows modelling and analysing the movement of vehicles, pedestrians, and other road users in urban areas. To validate the ability of the traffic simulator SUMO to replicate real-world traffic scenarios, we utilised loop data collected from April 1, 2015, to December 31, 2015, on a segment of the M25 highway. This segment includes approximately 30 evenly distributed detector points, which recorded average traffic speed and flow at 15 min intervals. The same dataset was employed in [15]. Traffic demand was initially generated using DFROUTER based on historical loop data from entrance point A, as illustrated in Fig. 7a, and subsequently implemented in SUMO with the Intelligent Driver Model (IDM) for car-following behaviour. A validation point C was randomly selected midway along the highway segment to collect simulated traffic flow and average speed data, which were then compared against real-world records. Vehicles in the real dataset were classified into two categories: passenger cars and freight cars. Their parameters, such as speed and acceleration, were configured using default values in the simulation.

We applied our approaches to the same 12 km segment on the M25 motorway in the UK as in [15,19] (see Fig. 7a), including the elevation data for this road segment from the Shuttle Radar Topography Mission (SRTM) [62]. In both the two-phase and MPC approaches, three objectives (fuel, NOx and travel time) were considered to fully explore the potential of these methods.

5.2. Experiments on the two-phase approach

5.2.1. Parameter settings

The experiments for the two-phase approach were conducted using MATLAB 2022a (MathWorks). The phase-1 multiobjective shortest path problem was solved using the default linear programming solver of MATLAB. For the Phase-2 MPC problem, our MPC model was developed based on the OptiTruck model in [15] by updating its objective terms and speed generating logic, and replacing the original heavy duty truck with our simulated car vehicle in Table 1.

5.2.2. Results from phase 1 shortest path multiobjective optimisation

The experiment road section of 12 km is divided into 120 segment of 100 m and thus the available road positions form a finite set $D = \{0, 100, \dots, 12\,000\}$. The speed range from 0 to 120 km/h (33.33 m/s) is divided into 33 levels with a 1 m/s resolution and thus the available speed values form a finite set $V = \{0, 1, 2, \dots, 33\}$. The discrete road positions and speed values form the feasible region of the shortest path problem.

Based on the relationships established in Eqs. (6) and (7) and the multiobjective optimisation shortest path computations, we have obtained the corresponding Pareto front as shown in Figs. 8 and 9. It can be concluded that generally the travel time is conflicting with both NOx and fuel but with different rates and patterns under free-flow. The relationship between NOx and fuel is positively correlated but is linear.

5.2.3. Knee point and sample points

In the two-phase approach, the Phase-1 Pareto front provides valuable guidance points for Phase-2. To ensure the sampled points adequately represent the Pareto front, we consider various types of points.

Firstly, we include the *knee point* [66], where an enhancement in one objective would result in a significantly adequate decline in at least one other objective. These solutions are often referred to as “knees” due to their distinctive characteristics and are often found in the “middle” area of the Pareto front. A knee point is arguably the most “balanced” point on the front [67]. Additionally, we incorporate the boundary points obtained by minimising only one individual objective. These points represent extreme solutions along each objective axis and contribute to a comprehensive understanding of the Pareto front.

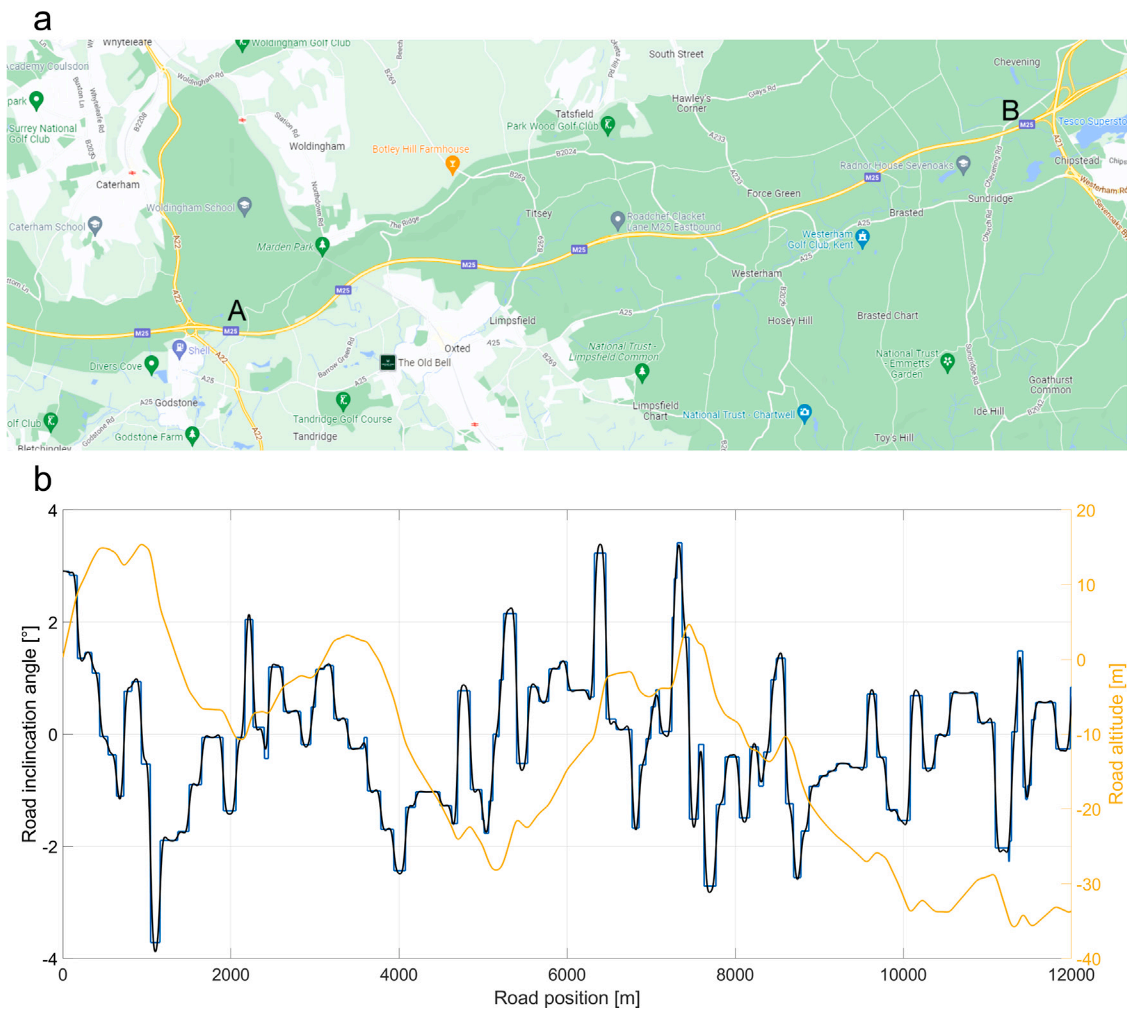


Fig. 7. Road segment for experiments.
a: A 12k road segment on the M25 motorway (marked from A to B) used in the experiments (Source: Google Maps).
b: The plot illustrates the relationship between road position along an 12 km segment of the M25 motorway and both the road slope angle.

Table 1
Parameters settings.

Symbol	Value [unit]	Description	Remarks
g	9.81 [m/s^2]	Gravitational acceleration	
A_f	2 [m^2]	Vehicle frontal area	
M_v	1505 [kg]	Vehicle mass	
N_w	4	Number of wheels	
J_w	15 [kg m^2]	Tire inertia	
R_w	0.6 [m]	Tire radius	
C_r	0.012	Tire rolling resistance coefficient	Wargula et al. [63]
C_d	0.31	Aerodynamic drag coefficient	Windsor [64]
g	9.81 [m/s^2]	Gravitational acceleration	
ρ	0.51 [kg/m^3]	Air density	
$a_{\max}$	1.47 [m/s^2]	Maximum acceleration/deceleration	Bae et al. [65]
$v_{\max}$	120 [km/h]	Maximum speed	Jia et al. [15]
S	12 [km]	Total travel distance	A segment of the M25 motorway

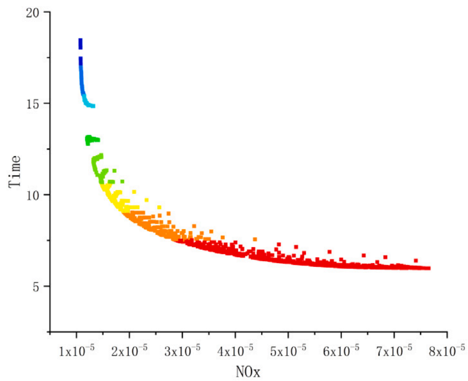
Furthermore, we include the points between the knee and boundary points by averaging the weights. These intermediate points capture the gradual transition in the trade-off relationship and provide a more nuanced representation of the Pareto front. Fig. 10 gives an illustrative example of the knee and boundary points.

Algorithm 1 in Appendix A.2 shows a classical approach in calculating the knee point that is used in our experiments. Note that since the shortest path problem in Phase-1 is convex [68], this standard approach suffices in finding the knee point.

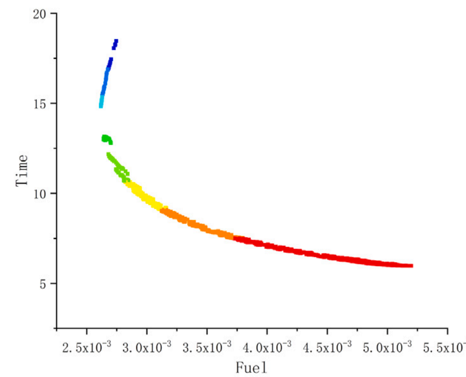
By considering these different types of points, we ensure that the sampled points in Phase-2 are representative enough to guide the MPC and DRL process effectively. The weight sets and Pareto-optimal values of sample points can be found in Table 2 and Fig. 11.

5.2.4. Results from phase-2 MPC approach

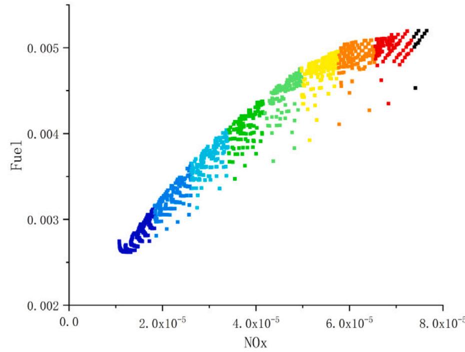
In these Phase-2 experiments, MPC was utilised to account for the surrounding traffic at each of the sampled points. The speed profiles of the seven points sampled from the Pareto front (see Fig. 8 and Table 2) are designated as the Pareto speed v_P. The front car speed v_f is



(a) 2D projection in space of NOx and time.



(b) 2D projection in space of fuel and time.



(c) 2D projection in space of NOx and fuel.

Fig. 8. 2D projections of the 3D Pareto front.

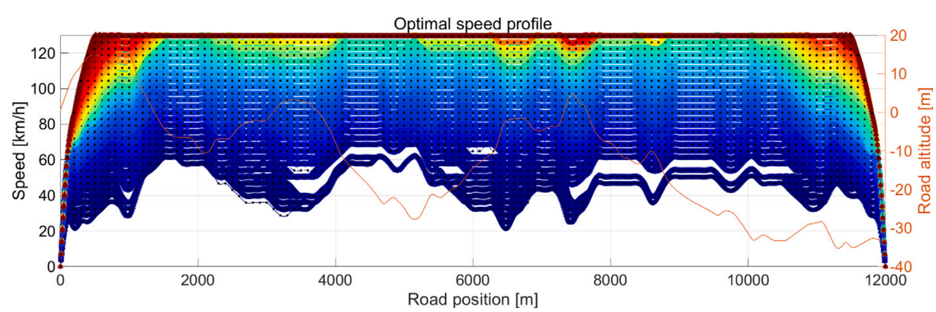


Fig. 9. Speed profiles of three objective Pareto front (with the same colour legend as in Fig. 8).

Table 2
Sampled Pareto points from Phase-1 result (traffic-free).

Sample	Fuel weight	NOx weight	Time weight	Fuel	NOx	Time (s)
p_1	0	1	0	0.00536	2.42×10^{-5}	1094
p_2	1	0	0	0.00529	2.97×10^{-5}	866
p_3	0.68	0.16	0.16	0.00573	3.44×10^{-5}	626
p_4	0.18	0.66	0.16	0.00541	3.48×10^{-5}	716
p_5 (knee)	0.37	0.32	0.31	0.00822	9.26×10^{-5}	419
p_6	0.18	0.16	0.66	0.00824	1.07×10^{-4}	417
p_7	0	0	1	0.00835	1.19×10^{-4}	415

determined using the same simulated traffic as in [15]. The parameter settings in the MPC model remained the same as in [15] except that the series of desired speed v_d was set in accordance with either Eq. (13) (MPC1) or Eq. (14) (MPC2). Both the conservative strategy (MPC1) and the balanced strategy (MPC2) were employed and their outcomes compared.

Results from conservative strategy (MPC1) Results in terms of the objective values of the three criteria from applying MPC1 are reported in Table 3 and the seven resulting speed profiles are depicted in Fig. 12. In the figure, the dark blue line represents the speed profile of sample point p_1, the light blue line represents p_2, the green point represents p_3, the yellow point represents p_4, the orange point represents p_5, the light red point represents p_6, and the dark red point represents p_7. The speed of the traffic flow is indicated by the black dashed line. The speed profiles correspond to the left y-axis, while the road altitude, represented by the light brown line, corresponds to the right y-axis. The influence of minimising travel time gradually becomes more

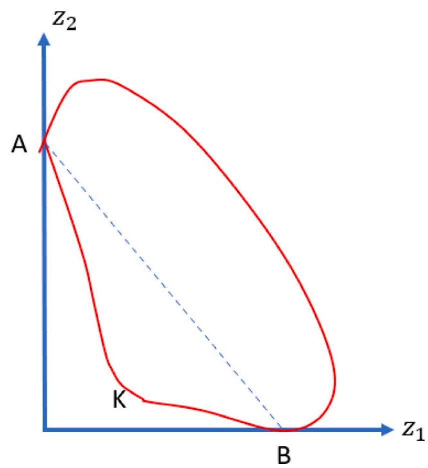


Fig. 10. An illustrative example of the knee point (K) and boundary points (A, B) on the Pareto front in a minimisation problem with two objectives.

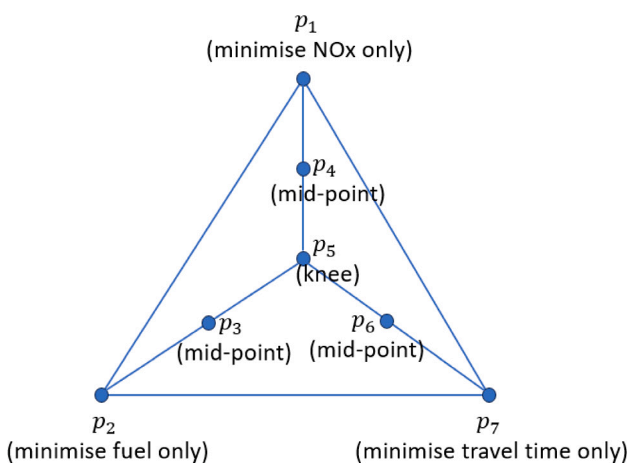


Fig. 11. Illustration of sampled knee, boundary and middle points from the Pareto front with three objectives.

Table 3
Phase-2 results given by MPC1 based on seven sampled points.

Sample	Fuel	NOx	Time (s)
p_1	0.00562	2.83×10^{-5}	1028
p_2	0.00564	3.06×10^{-5}	854
p_3	0.00653	4.28×10^{-5}	629
p_4	0.00599	3.56×10^{-5}	715
p_5 (knee)	0.00919	9.20×10^{-5}	495
p_6	0.00907	8.80×10^{-5}	496
p_7	0.00959	1.00×10^{-4}	493

significant from sampled point p_1 to p_7, leading to higher speeds. The pattern depicted in Fig. 12 remains consistent with the observation that maintaining a low speed approximately between 40 and 60 km/h frequently leads to reduced fuel consumption and NOx emissions (see Fig. 1). As the travel time is further prioritised, both the fuel and NOx get worse values. Note that due to the design of the strategy in Eq. (13), even p_6 or p_7 is set as the reference Pareto speed, MPC1 rarely gives solutions with speeds surpassing the traffic when travel time is more prioritised.

Results from balanced strategy (MPC2) Results regarding the objective values of the three criteria obtained by applying MPC2 are outlined in Table 4, along with the depiction of the seven resulting speed profiles in Fig. 13. Similar to results given by MPC1, the impact of minimising travel time increases gradually from sampled point p_1 to p_7, resulting in higher speeds. On the other hand, since MPC2 uses the average speed between the Pareto and front car speeds, when travel time is more of a priority, the speed of the vehicle can exceed that of the traffic, making the speed profiles more realistic and flexible. In addition, the speed profiles from MPC2 are more centred around the traffic, as opposite to the case in MPC1's result. However, as previously stated, this compromise comes at the expense of lower solution quality since the profiles deviate further from the Pareto samples due to averaging the Pareto and front car speeds.

Table 4
Phase-2 results given by MPC2 based on seven sampled points.

Sample	Fuel	NOx	Time (s)
p_1	0.00670	4.30×10^{-5}	657
p_2	0.00700	4.90×10^{-5}	614
p_3	0.00796	6.48×10^{-5}	548
p_4	0.00745	5.74×10^{-5}	578
p_5 (knee)	0.00774	7.10×10^{-5}	483
p_6	0.00757	6.93×10^{-5}	482
p_7	0.00834	8.81×10^{-5}	441

Overall, in either the solutions from MPC1 or MPC2, given the set of varied results, users have the flexibility to select a speed profile from this collection based on their specific requirements while taking into account of surrounding traffic. For instance, if a driver prioritises reaching their destination earlier and is less concerned about fuel efficiency or NOx emissions, they can opt for p_6 or p_7 as their preferred speed profile. By considering these options, users can tailor their driving experience to align with their individual preferences and priorities.

5.3. Experiments using DRL-based approach

5.3.1. Parameter settings

With the same settings for section division, vehicle parameters, and traffic flow as Phase-2 MPC conditions, we utilised the DRL method to simultaneously generate a speed profile considering both the three objective terms and the traffic flow. The actor and critic networks were formulated using deep neural networks in Python 3.9 with TensorFlow. The actor network consists of two hidden layers with 100 and 50 nodes, utilising ReLU activation functions. The output layer uses tanh activation to ensure the output acceleration remains within the specified accelerate/decelerate limits. On the other hand, the critic network has two hidden layers with 300 and 200 nodes, using SELU activation to maintain the value of the penalty (negative value). For learning rate decay, we employed the exponential decay function from TensorFlow with the initial value of 10^{-6} for the actor network and 10^{-3} for the critic network. The decay step occurs every 1000 episodes, and the decay rate is set to 0.1. For epsilon decay, the parameters are set as follows: $\varepsilon_{min} = 0.1$, $\varepsilon_{max} = 1$, $\beta_2 = 10^{-3}$, $\beta_3 = 5000$, $\beta_4 = 0.5$. The total number of episodes in the training process is set to 10,000. During each episode, the parameters used in the episode reward calculation are specified as follows: $\gamma_+ = 10$, $\gamma_- = 50$, $\beta_1 = 0.01$.

In the three-objective settings of the two-phase method, we have selected 7 points from the Pareto front, $p_1, p_2, \ldots, p_7$. These weighting sets are then utilised in the step reward of the DRL method to generate the multi-objective solutions. Each of these 7 solutions corresponds to a specific combination of weights for the three objectives. In the DRL method, the step reward is computed by normalising and weighting the three objectives, and then summing them up. By leveraging these 7 different sets of weights, the DRL method produces 7 distinct solutions, each offering a unique trade-off among the three objectives. These solutions effectively provide a diverse set of optimised outcomes that cater to different decision-making requirements.

The settings were chosen with specific values to balance stability, convergence speed, and computational efficiency. For example, the actor network's two hidden layers use 100 and 50 nodes with ReLU, a choice that helps achieve quick convergence, while the `tanh` activation

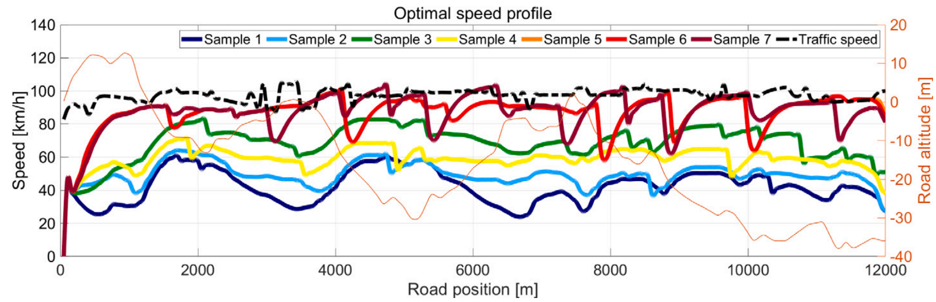


Fig. 12. Phase-2 speed profiles given by MPC1 based on seven sampled Pareto points.

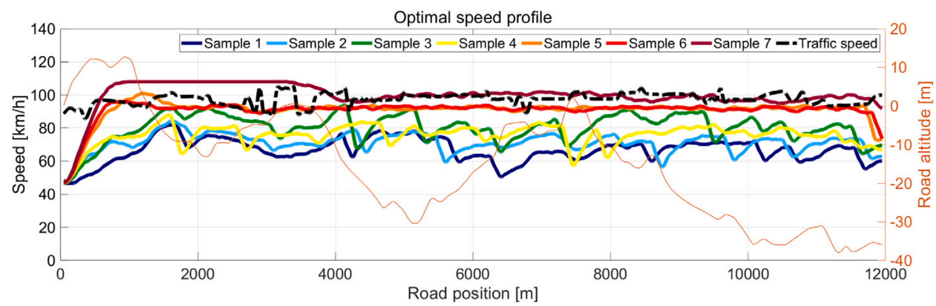


Fig. 13. Phase-2 speed profiles given by MPC2 based on seven sampled Pareto points.

in the output layer ensures the acceleration stays within defined limits. Meanwhile, the critic network's larger layers (300 and 200 nodes) with SELU activation are tailored to accurately capture the negative penalty values. The learning rate decay settings – starting at 10^{-6} for the actor and 10^{-3} for the critic, with a decay step every 1000 episodes and a decay rate of 0.1 – are specifically set to gradually reduce the learning rate as training progresses, preventing overshooting and ensuring fine adjustments in later stages. Additionally, the epsilon decay parameters ($\varepsilon_{min} = 0.1$, $\varepsilon_{max} = 1$, $\beta_2 = 10^{-3}$, $\beta_3 = 5000$, $\beta_4 = 0.5$) are precisely tuned to balance exploration and exploitation over the 10,000 episodes of training. Finally, selecting 7 points from the Pareto front allows the method to cover a range of trade-offs among the three objectives by assigning distinct weight combinations in the step reward calculation, leading to a diverse set of optimised outcomes. Each of these specific settings plays a crucial role in ensuring the DRL method not only trains effectively but also maintains real-time responsiveness in deployment.

The DRL methodology was executed using Python on a high-performance computing system with Intel Xeon Gold 6138 CPUs operating at 2.0 GHz. Each training episode consumed approximately 7 s of computational time, resulting in an overall training duration of approximately 19 h for 10,000 episodes. It is worth noting that while the training process exhibited substantial duration, the application of a pre-trained agent demonstrated prompt responsiveness, effectively responding to a given state within a millisecond timeframe. This responsiveness aligns well with real-time response requisites.

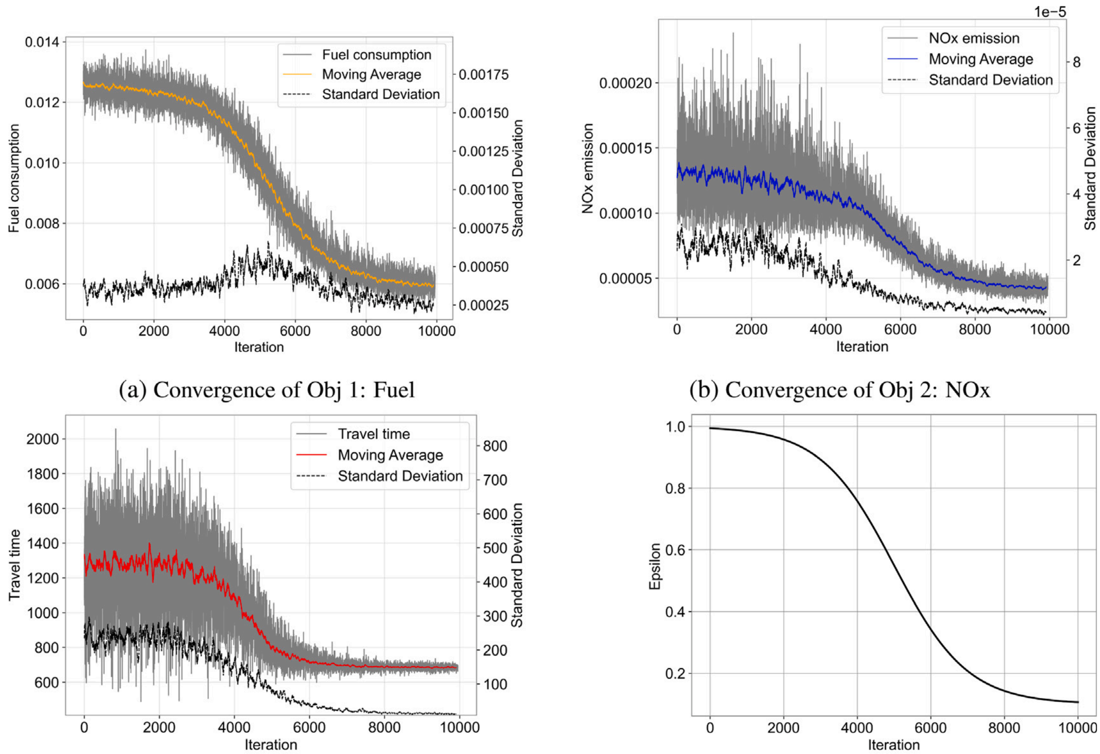
5.3.2. Convergence analysis

Due to space limitations, we present the convergence progress of the DDPG training with epsilon decay for weighting set of the knee point. The convergence patterns for other weighting combinations are similar. The epsilon decay follows the shape depicted in Fig. 14(d). The convergence of the three objectives, namely fuel, NOx, and travel time, is shown in Figs. 14(a)–14(c), respectively. The moving average of 50 solutions is shown in the coloured lines, and the standard deviation is shown in the black dashed line. Based on the convergence figures, the grey lines represent the objective values obtained in each episode, while the coloured lines (red, blue, and yellow) show the moving averages of the 100 nearest values. At the exploration stage (episode 0–5000), the relatively high epsilon indicates that the agent's actions heavily rely on random selection. Consequently, the objective values show significant deviations, and the solutions fluctuate widely as the agent explores different actions to gather rewards in varying states. As the epsilon decays (episode 3000–6000), the agent starts to depend more on its experience rather than random actions. This leads to a better understanding of the environment and rewards, resulting in less deviation among the solutions and more cost-saving solutions. At the exploitation stage (episode 6000–10,000), both epsilon and learning rate are low, indicating that the agent predominantly relies on the trained actor ANN. Consequently, it can consistently provide cost-saving solutions and effectively drive the vehicle on the experiment road, striking a balance among the three objectives.

5.3.3. Application of the energy management system with DRL

Table 5 presents the objective values associated with the same sampled seven points as shown in Table 2, which were obtained using the DRL method. A comparison among the Pareto-optimal solutions and solutions derived from two-phase and DRL approaches is illustrated in Fig. 15, and the projections are shown in Fig. 16. The proximity of a scatter point to the bottom-left corner indicates its superior performance. Notably, the Pareto solutions reflect the best outcomes within a traffic-free context, representing the optimal solutions for given weights. In actual scenarios, the speed profile is controlled by MPC or DRL approaches amidst surrounding traffic flow. Upon comparison, it is evident that the DRL solutions are situated closer to the Pareto front in contrast to the MPC solutions. Across all weight sets, the DRL solutions consistently outperform the two-phase solutions under traffic flow conditions. The integrated DRL approach excels in identifying solutions that yield reduced fuel consumption, NOx emissions, and time savings compared to the two-phase approach.

The speed profiles resulting from the DRL approach, as depicted in Fig. 17, exhibit distinct qualities when contrasted with the optimisation-based method. Notably, the DRL outcomes showcase several notable features. Firstly, the DRL approach offers enhanced flexibility in speed adjustments, a characteristic that stems from its heightened sensitivity to variances in gradient, speed, and acceleration. This heightened adaptability enables it to more effectively address the three



(a) Convergence of Obj 1: Fuel

(b) Convergence of Obj 2: NOx

(c) Convergence of Obj 3: Travel time

(d) Epsilon decay

Fig. 14. Convergence of DRL method using weighting set of the knee point.

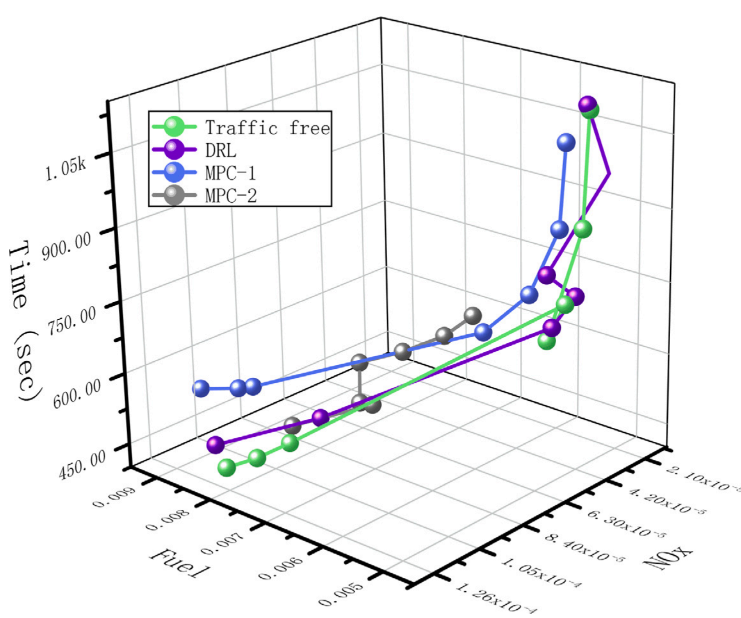
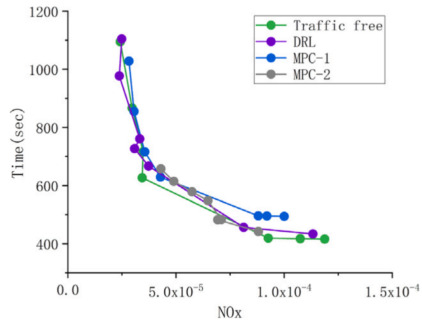


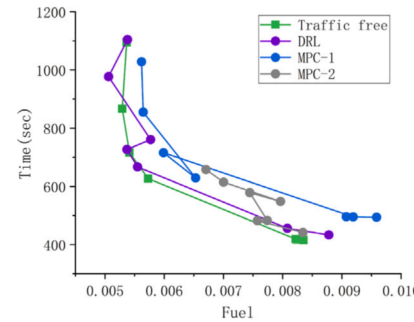
Fig. 15. Comparison of the three objective values derived from Phase-1 Pareto (traffic-free), Phase-2 (MPC-1 and MPC-2) and DRL.

objectives, reacting dynamically to their fluctuations. Importantly, boundary samples (1, 2, and 7) indicate DRL's superiority over MPC1 and MPC2. These boundary cases highlight the DRL agent's ability to adeptly navigate the complex interaction between driving speed and emissions within the dynamic context of traffic flow. This showcases the DRL's capacity to capture nuanced relationships and deliver enhanced performance, setting it apart as a powerful optimisation approach. Another notable attribute is the incorporation of gradient profiles. In scenarios favouring emission reduction over travel time, the DRL agent showcases a strategic behaviour: maintaining a consistent speed on uphill sections while accelerating on downhill stretches. This smart strategy serves to optimise both emission levels and travel time efficiency.

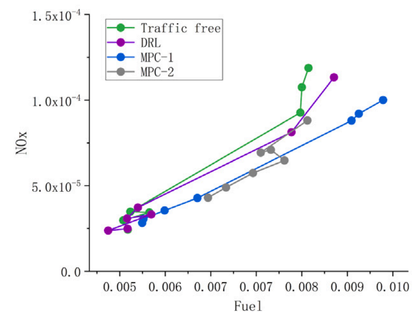
Moreover, the DRL method integrates the concept of headway gap, a safety parameter. In situations where the gap remains within safe limits, the vehicle is allowed to accelerate, leading to instances where driving speed outpaces traffic speed. This approach takes into consideration not only objective optimisation but also road safety. The headway gap of



(a) 2D projection in space of NOx and time.



(b) 2D projection in space of fuel and time.



(c) 2D projection in space of NOx and fuel.

Fig. 16. 2D projections of the 3D objective values.

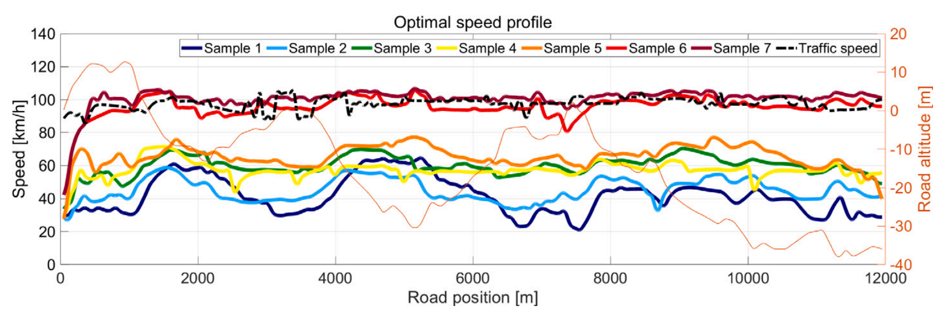


Fig. 17. Speed profile of the seven sample Pareto points by DRL (three objectives case).

Table 5
Results given by DRL based on seven sampled points.

Sample	Fuel	NOx	Time (s)
p_1	0.0538	2.49×10^{-5}	1104
p_2	0.00506	2.38×10^{-5}	976
p_3	0.00577	3.32×10^{-5}	760
p_4	0.00537	3.08×10^{-5}	726
p_5 (knee)	0.00555	3.73×10^{-5}	667
p_6	0.00808	8.13×10^{-5}	456
p_7	0.00878	1.13×10^{-4}	433

the DRL, MPC1 and MPC2 are compared in Fig. 18 where the solutions with optimal travel time (p_7) are selected. Because, when objectives focus on the NOx and fuel consumption, the driving speed is always smaller than the flow speed, following the eco-driving requirements. It is obviously that the MPC methods will induce large headway with the front vehicle. Because the MPC methods control the speed by referring to the traffic speed rather, while headway indicator is not incorporated in such controlling algorithms. The vehicle cannot accelerate even though the headway is safe enough.

The DRL approach outperforms the two-phase strategy due to its methodological advantages. Unlike MPC, which lacks the capability to adjust acceleration based on headway distance, DRL provides a more flexible strategy. MPC's conservative approach, prioritising collision avoidance based on the lead vehicle's speed, becomes inefficient when headway distance is safe. In such cases, if the lead vehicle slows down, MPC responds by decreasing speed, impacting overall efficiency.

Unlike MPC, the DRL approach adapts dynamically, updating headway distance and traffic speed in real-time. Trained to optimise acceleration and deceleration based on accumulated experience, it offers enhanced flexibility, enabling more nuanced movement strategies aligned with optimisation goals.

Additionally, the issue of transferability is crucial. While the two-phase strategy generates optimal solutions in traffic-free conditions and incorporates it into an MPC for real-world scenarios, the dynamic nature of traffic and variable speeds can undermine the effectiveness of this idealised profile. If traffic speed consistently falls below the set of Pareto-optimal solutions, the MPC may predominantly mimic traffic speed, potentially sidelining essential optimisation objectives. In contrast, the DRL approach consistently makes optimal decisions for each state variable, systematically addressing optimisation objectives at each time step.

The experiment currently assumes that the following vehicle never surpasses our own, which is somewhat unrealistic. However, the DRL approach excels, especially in scenarios involving "vehicle insertion". Strategies focused on conserving fuel or reducing NOx emissions often

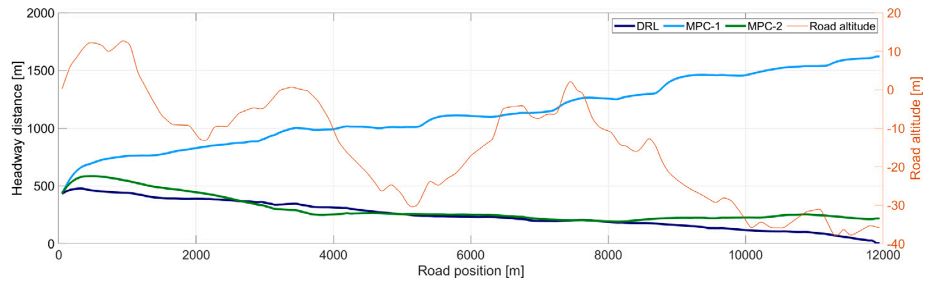


Fig. 18. Comparison of headway between the three methods.

entail maintaining a larger headway distance, leading to situations where vehicles insert themselves between the subject vehicle and the lead vehicle. In contrast, the MPC strategy may struggle to handle such insertions accurately, highlighting the DRL agent's strength in promptly recalibrating the headway distance and implementing suitable braking measures.

Despite its advantages, the DRL approach has a drawback. It requires training for each specific weight set, leading to a time-intensive process before convergence. This slower solving efficiency, compared to the two-phase approach, may limit its suitability for entirely new situations lacking training data. This highlights the trade-off between DRL's enhanced decision-making and the computational time needed for optimal convergence.

Furthermore, the profiles of fuel consumption, NOx emissions, travel time, and headway are compared and analysed in Appendix A.5 to comprehensively demonstrate the different eco-planning strategies resulting from various weight combinations.

5.4. Incorporation of the jerk cost

As highlighted by several studies on eco-driving behaviour [54,55], the jerk cost significantly impacts driver comfort during acceleration and deceleration. To further account for driver comfort, this section evaluates the performance of our approaches by incorporating the jerk cost into the objectives. Methodologically, the jerk cost is calculated as the absolute difference between the previous and current speed across all road segments [15].

To generate speed profiles after incorporating the jerk cost, the weights of the objectives are uniformly set to 1/4 for fuel consumption, NOx emissions, travel time, and jerk cost. To assess the influence of the jerk cost, the knee point weight solution (p_5) from Section 5.3.3 (which does not include the jerk cost in the objectives) is used as a benchmark for both the two-phase approaches and the DRL approach.

The speed profiles with and without the jerk cost in the objective function are illustrated in Fig. 19. The dark and light blue lines represent the DRL solutions, the dark and light green lines represent the MPC1 solutions, and the dark and light red lines represent the MPC2 solutions. As shown in the figure, the speed profiles produced by considering the jerk cost become smoother for the DRL and MPC1 approaches compared to those without the jerk cost. However, since the speed profile for the MPC2 solution is already sufficiently smooth even without considering the jerk cost, the difference is less pronounced. Incorporating the jerk cost effectively reduces severe acceleration and deceleration, thereby enhancing driving comfort.

The objective values for each term are presented in Table 6. For the DRL approach, while fuel consumption, NOx emissions, and jerk cost are reduced when the jerk cost is incorporated, the travel time increases from 658.7 to 718.05. Consequently, no dominated solution is found in the multiobjective optimisation problem, demonstrating the capability of DRL to handle multiobjective optimisation problems effectively. In contrast, for both the MPC1 and MPC2 approaches, incorporating the jerk cost results in a reduction of all four objective values, dominating the solutions that do not consider the jerk cost. This indicates that the two-phase approach is less robust in ensuring solution quality.

5.5. Online application with real-world traffic

MO-SPO face significant challenges in real-time applications due to their high computational complexity, which arises from solving multiple conflicting objectives like fuel consumption, NOx emissions, and travel time simultaneously. These problems are further compounded by the dynamic and unpredictable nature of real-world traffic conditions, such as fluctuating traffic speeds and driver behaviour, which MO-SPOs struggle to adapt to efficiently. Additionally, as the number of objectives and constraints increases, the complexity grows exponentially, making it difficult to scale MO-SPOs for large-scale or complex scenarios. Finally, the lack of real-time feedback mechanisms means that solutions based on static data may become suboptimal or infeasible in dynamic environments, limiting their practicality for online applications.

An advantage of our two-phase and DRL approach is their suitability for online applications, which enable the generation of eco-driving speed profiles using real-time information. To demonstrate the online applicability of our approaches, we applied our pretrained models to a novel real-world traffic scenario. Specifically, we utilised evening peak-hour traffic speed data (19:00–21:00) collected from January 1–7, 2022, on a segment of the M25 highway. This segment includes 12 detector points, approximately evenly distributed, where traffic speed was recorded at 1 min intervals. The average speed of each segment was used to represent the real-world traffic speed.

The comparison between the simulated traffic speed (using SUMO) and the real-world traffic speed is illustrated in Fig. 20. In the figure, the blue line represents the simulated traffic speed, while the red line represents the average real-world traffic speed. This comparison highlights the ability of our approaches to adapt to real-world conditions, ensuring that the generated eco-driving speed profiles are both practical and effective in dynamic traffic environments. By leveraging real-time data, our methods provide a robust solution for optimising speed profiles in real-world applications, particularly during peak traffic hours when efficiency and responsiveness are critical.

By comparison, the real-world traffic speed is slightly higher than the simulated traffic speed. Since the traffic speed serves as the upper bound for speed limitations and influences the headway to the front vehicle, a higher traffic speed does not significantly impact the solutions for p_1 to p_5. This is because the speed profiles for these points are consistently lower than the traffic speed to optimise fuel consumption and NOx emissions. Therefore, to test the online application under real-world traffic conditions, the weights of p_6 and p_7 are applied. These points represent scenarios where the speed profiles are closer to the traffic speed, making them more sensitive to real-world variations and thus better suited for evaluating the performance of our approaches in dynamic environments. This ensures that the solutions remain robust and effective even when applied to real-world traffic data with higher average speeds.

The speed profile of MPC1, MPC2 and DRL approaches are shown in Fig. 21. The blue lines represent the speed profiles of MPC1 approach, the green lines represent the speed profiles of the MPC2, the red lines represent the speed profiles of DRL. The light lines represent

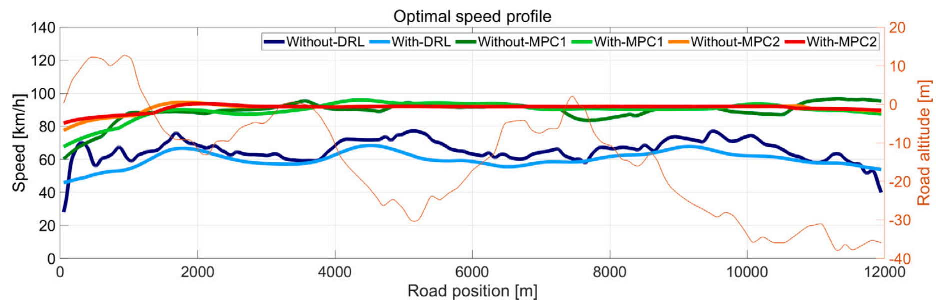


Fig. 19. Comparison of speed profiles with and without jerk cost.

Table 6
Objectives of the speed profiles with and without jerk cost.

	Without jerk cost			With jerk cost		
	DRL	MPC1	MPC2	DRL	MPC1	MPC2
Fuel consumption	0.00555	0.00919	0.00774	0.0052	0.0069	0.0068
NOx emission	3.73E−05	9.20E−05	7.10E−05	2.80E−05	5.35E−05	5.41E−05
Travel time	667	495	483	718.05	481.98	474.86
Jerk cost	88.48	159.83	90.70	28.55	43.76	29.14

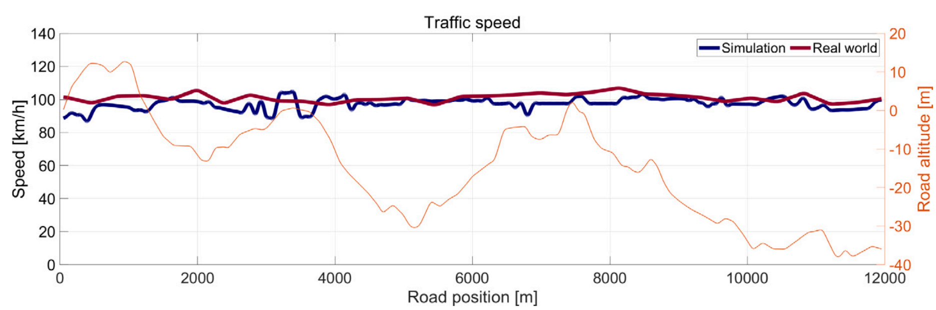


Fig. 20. Comparison of traffic speed of simulating and real-world condition.

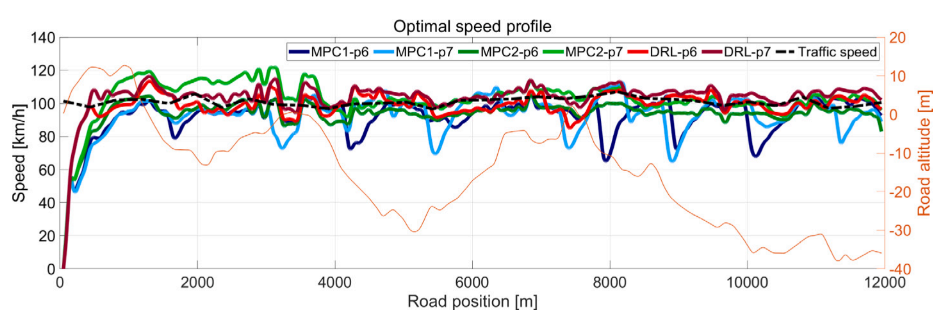


Fig. 21. Speed profiles of p_6, p_7 by different approaches.

the sample p_6, and the dark lines represent the sample p_7. Among the methods, DRL speed profiles demonstrate exceptional smoothness and adaptability, closely aligning with the real-world traffic speed. For p6, the DRL profile is smoother than MPC1 and slightly more adaptive than MPC2, showcasing its ability to balance smoothness and real-world responsiveness. For p7, the DRL profile almost perfectly follows the real-world traffic speed, highlighting its superior capability to handle dynamic conditions. This adaptability makes DRL particularly well-suited for unpredictable environments.

Table 7 compares the performance of three approaches, namely MPC1, MPC2, and DRL, for two sample points, p_6 and p_7, across three objectives: fuel consumption, NOx emissions, and travel time. Among these, DRL stands out as the best-performing approach, particularly when travel time is prioritised as the most important factor. DRL achieves the shortest travel times for both p_6 (440.66) and p_7 (418.68), making it the fastest and most time-efficient method. This exceptional performance in reducing travel time is particularly critical for online applications, where speed and responsiveness are paramount, especially when user preferences prioritise time efficiency. While MPC2 and MPC1 excel in fuel efficiency and environmental performance, they cannot match DRL's speed and responsiveness.

Table 7
Objectives of in online application.

	MPC1		MPC2		DRL	
	p6	p7	p6	p7	p6	p7
Fuel consumption	0.0031	0.0037	0.0028	0.0034	0.0033	0.0041
Nox emission	3.43E−05	5.62E−05	3.38E−05	7.06E−05	5.57E−05	8.02E−05
Travel time	476.69	475.09	464.11	425.04	440.66	418.68

6. Conclusions and future work

Using multiobjective optimisation, this paper addresses the eco-driving problem by generating vehicle speed profiles that consider up to three objectives: fuel consumption, NOx emission, and total travel time in real-world scenarios with surrounding traffic. Unlike traditional approaches that generate a single solution, multiobjective optimisation provides a collection of solutions, each representing unique preferences in weighting different objectives. This approach is particularly suitable for problems with incomparable objectives, as is the case in our study. Simulated data from GT-SUITE are used to derive the relationships between engine power and the rates of fuel consumption and NOx emission, which can be determined analytically by vehicle dynamics.

Two solution approaches are presented and compared. The first involves a two-phase process: Phase-1 solves a traffic-free problem analytically, providing "ideal" Pareto points for Phase-2. In Phase-2, an existing model predictive control approach generates compromised results considering both Pareto points and surrounding traffic. The second approach, designed by the authors from scratch, employs deep reinforcement learning (DRL) to generate speed profiles, considering multiple objectives and surrounding traffic simultaneously. Both approaches use the weighted sum method to generate Pareto fronts, marking the first application of multiobjective optimisation to simultaneously consider fuel consumption and NOx emissions in generating optimised speed profiles.

The DRL approach outperforms the two-phase method in modelling flexibility and solution quality on a real-world highway in southern England. It explicitly considers vehicle headway, leading to more sophisticated eco-driving strategies and optimised objective values across all three criteria. For instance, at one representative Pareto point, results indicate that the DRL approach achieves up to 10% lower fuel consumption and 16% lower NOx emissions compared to MPC-based methods while reducing travel time by approximately 5%. In addition, the DRL method maintained safer headway distances, offering more robust eco-driving strategies in dynamic traffic environments. However, it requires prior training, making it less suitable for entirely new scenarios with limited or no training data.

Building on these promising results, future work could extend the MO-SPO framework to other vehicle classes, such as heavy-duty trucks, electric vehicles, and hybrid models, by re-calibrating vehicle parameters and integrating appropriate powertrain and emission models to better reflect distinct dynamic characteristics. Moreover, adapting the framework to diverse traffic environments – including urban settings, mixed-traffic conditions, or multi-lane roads with varying densities – will enhance its applicability, while integrating richer traffic data, such as real-time signal timings or pedestrian interactions, could further improve its robustness. As the complexity of these extended scenarios may increase computational demands, future research should focus on improving training efficiency through methods like transfer learning or advanced parallel computing strategies to maintain real-time responsiveness. Additionally, incorporating further environmental metrics such as particulate matter (PM) and COx, and integrating robust or stochastic optimisation techniques to manage uncertainties in traffic flow, weather, and driver behaviour, represent promising avenues for further refinement. Overall, these research directions aim to advance the MO-SPO framework towards a more comprehensive, adaptable, and environmentally conscious solution for modern traffic management.

CRediT authorship contribution statement

Enze Liu: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis. **Zhiyuan Lin:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Conceptualization. **Haibo Chen:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization. **Dongyao Jia:** Writing – review & editing, Writing – original draft, Software, Funding acquisition. **Ye Liu:** Writing – review & editing, Validation, Software, Resources, Data curation. **Junhua Guo:** Writing – review & editing, Writing – original draft, Conceptualization. **Tiezhu Li:** Writing – review & editing, Writing – original draft, Project administration, Funding acquisition. **Tangjian Wei:** Writing – review & editing, Writing – original draft, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research was jointly supported by European Union's Horizon 2020 projects MODALES (grant agreement No. 815189) and nPETS (grant agreement No. 954377), and the National Key Research and Development Program of China (No. 2021YFE0112700), the Foreign Expert Project of China: G2022022004L which enables international collaboration in intelligent transport systems and clean air.

Enze Liu is supported by a China Scholarship Council fund (No. 202106560015) for his visiting research in Institute of Transport Studies, University of Leeds. Dongyao Jia is supported in part by the National Natural Science Foundation of China (Grant No. 62372384), and the Research Development Fund of XJTLU, China under Grant RDF-21-02-082.

The authors extend their sincere gratitude to Dr. Jianbing Gao for his foundational work in developing and validating the GT-SUITE model, which served as the cornerstone for this study. His meticulous calibration of powertrain and emission parameters under diverse operational conditions was instrumental to our analysis.

This work was undertaken on ARC4, part of the High Performance Computing facilities at the University of Leeds, UK (https://arc.leeds.ac.uk/).

Appendix

A.1. Simulation environment and vehicle modules

Simulations were carried out using GT-SUITE simulation software, as detailed in [25], employing identical vehicle specifications outlined in [51]. The vehicle under study was a Euro 6 compliant diesel passenger car equipped with a four-cylinder, four-stroke turbocharged diesel engine. It weighed 1505 kg and boasted a maximum power output of 103 kW, correlating to an engine speed of 4000 rpm. The diesel engine featured a compression ratio of 16.5:1. This vehicle model comprised three main modules: vehicle powertrains, emission sources,

Algorithm 1: Calculate Knee Point on Pareto Front

Require: Non-dominated solution set on the Pareto front, S
Ensure: Knee point solution, K
Initialise weights for distance and angle, w_d, w_a
for all Non-dominated solution $s_i = (x_i, y_i, z_i)$ in S **do**
  Initialise the utility (weighted sum of distance and angle) of s_i, i.e., $u_i := w_d \cdot d(s_i) + w_a \cdot \theta(s_i)$
  for all Other non-dominated solutions s_j in S where $s_j \neq s_i$ **do**
    Calculate Euclidean distance d_{ij} between s_i and s_j:
    $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$
    Calculate angle θ_{ij} between s_i and s_j:
    $\theta_{ij} = \arccos\left(\frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\|\mathbf{v}_i\| \cdot \|\mathbf{v}_j\|}\right)$, where $\mathbf{v}_i$ and $\mathbf{v}_j$ are vectors from origin to s_i and s_j, respectively.
    Update $u_i := u_i + w_d \cdot d_{ij} + w_a \cdot \theta_{ij}$
  end for
  if u_i is the current minimum **then**
    Update knee point solution $K = s_i$
  end if
end for
return Knee point solution K

and after-treatment systems. The powertrain system encompassed an engine model, a transmission model, and a control model. The engine model was constructed based on experimental tests, incorporating maps for brake-specific fuel consumption and brake mean effective pressure. Additionally, the emission model integrated maps for exhaust temperature, emission factors, and exhaust flow rates to consider the impact of cold starts on emissions. The after-treatment system encompassed a diesel oxidation catalyst and a diesel particulate filter, as discussed in [51].

A.2. Algorithm for calculating the knee point

See Algorithm 1.

Here the weights on distance and angle (w_d and w_a) are set as 5×10^{-9} and 0.5 respectively, and a discount of 10^{-8} is applied to eliminate the magnitude of distance.

A.3. Algorithm for the MPC method

See Algorithm 2.

A.4. Pseudocode for DDPG

See Algorithm 3.

A.5. Performance comparison between different weights of preference

Figs. 22, 23, 24, and 25 display the profiles of NOx emission, fuel consumption, travel time, and headway distance resulting from the three approaches under various traffic conditions. Each subfigure presents the profile of a specific approach. The comparative analysis among different points obtained from the same approach reveals their sensitivity to weight proportions. Greater deviation signifies heightened sensitivity of an approach. A substantial similarity in profiles under diverse weight sets may fail to cater to different trade-off requirements.

As observed in Figs. 22 and 23, the DRL approach exhibits more pronounced deviations in NOx emission and fuel consumption profiles, while the profiles of MPC-1 and MPC-2 remain similar across different samples. The objective value visualisation in Fig. 16 illustrates that heightened sensitivity to weight proportions corresponds to larger differences between objectives on the Pareto front, allowing for more flexible choices. Significantly, the divergence in the time profile is substantial across all three approaches. Intriguingly, in p_1 and p_2, where NOx and fuel are minimised, the time profile mirrors the road altitude. This similarity suggests that adopting a slower pace uphill and accelerating downhill can effectively conserve fuel and minimise emissions. Referring to Fig. 25, it is evident that the headway distance to the lead vehicle increases notably with higher weights assigned to NOx and fuel. Opting for strategies that minimise NOx or fuel eventually results in a headway gap surpassing 6000 m.

Figs. 26, 27, 29, and 28 provide a comparative analysis of objectives and headway distance among the three approaches with different weight configurations. These samples encompass scenarios, including p_1 (minimum NOx), p_2 (minimum fuel), p_7 (minimum time), and p_5 (knee point). Each subfigure presents profiles of NOx emission, fuel consumption, travel time, and headway distance.

In the cases of the first two samples (p_1 with minimum NOx and p_2 with minimum fuel), MPC-2 exhibits the least favourable performance due to its higher driving speed. The strategy employed by MPC-2 struggles to address scenarios of minimum NOx and fuel, primarily because the driving speed is heavily influenced by the traffic speed. This limitation prevents MPC-2 from ensuring optimal speeds in scenarios where only a single objective is considered. Conversely, MPC-1 and DRL showcase similar performance. However, DRL adopts a more conservative driving approach compared to both MPC methods, prioritising fuel and NOx reduction over higher speeds, irrespective of the traffic conditions.

In Sample 7, we delve into the scenario of minimising travel time, where all three methods are focused on completing the drive as quickly as possible. Starting with the same initial headway distance, the two MPC approaches and the DRL approach adopt distinct strategies. MPC-1 consistently maintains its driving speed at the same pace as the traffic speed, resulting in a stable headway that hovers around its initial value. In contrast, MPC-2 employs a more uniform acceleration strategy, causing the headway to gradually decrease over time. Lastly, DRL opts for an initial speed increase, actively tailing the front vehicle. As the headway narrows, DRL slows down, ensuring that the vehicle maintains a reasonable distance from the vehicle in front. This strategy results in the headway fluctuating between approximately 50 to 300 m.

Examining the outcomes of the knee point (p_5), the fuel consumption and NOx emission exhibit striking similarity across all three approaches. In terms of speed, the DRL approach positions itself between MPC-1 and MPC-2, strategically finding a balance that minimises the combined fuel and NOx values. This measured approach to speed is complemented by the travel time and headway, both of which also fall within the midpoint between the two MPC strategies.

Algorithm 2: Time-Varying Adaptive MPC for Speed Profile Generation

Input: Longitudinal model parameters (e.g. $A(i)$, $B(i)$, $D(i)$), cost weights λ_e, λ_k, λ_s, prediction horizon n_p, initial state $E_k(0)$, vehicle and safety constraints, traffic context data, and desired speed profile v_d

Output: Optimal engine energy sequence $E_e^*(i)$ over the prediction horizon and corresponding speed profile

Initialisation: Set current index $i \leftarrow 0$, obtain initial kinetic state $E_k(i)$, and retrieve initial traffic and road condition information.

while *vehicle has not reached the destination* **do**

1. Update Measurements:
Obtain current state $E_k(i)$, vehicle speed $v(i)$, and updated traffic context (including real-time local and predicted future traffic data).

2. Update Prediction Parameters:
Adapt the spatial step Δs based on the current vehicle speed $v(i)$
Determine the desired speed profile v_d over the horizon using the traffic predictive model.

3. Update Longitudinal Model:
Compute the time-varying matrices $A(i)$, $B(i)$, and offset $D(i)$ from the linearised vehicle dynamics:

$$E_k(i+1) = A(i)\,E_k(i) + B(i)\,U(i) - D(i)$$

where the control input is $U(i) = \begin{bmatrix} E_e(i) \\ E_b(i) \end{bmatrix}$.

4. Solve the MPC Optimisation:
Formulate the quadratic cost function over the prediction horizon:

$$J(i) = \sum_{j=i}^{i+n_p-1} \left(\lambda_e\, E_e(j)^2 + \lambda_k \left(E_k(j) - \frac{1}{2} M_e v_d(j)^2 \right)^2 + \lambda_s\, \left(E_e(j) - E_e(j-1) \right)^2 \right)$$

subject to constraints $\mathscr{D}$ and safety headway requirements.

Compute the optimal control sequence:

$$\{U^*(i), U^*(i+1), \ldots, U^*(i+n_p-1)\} = \arg\min_U J(i)$$

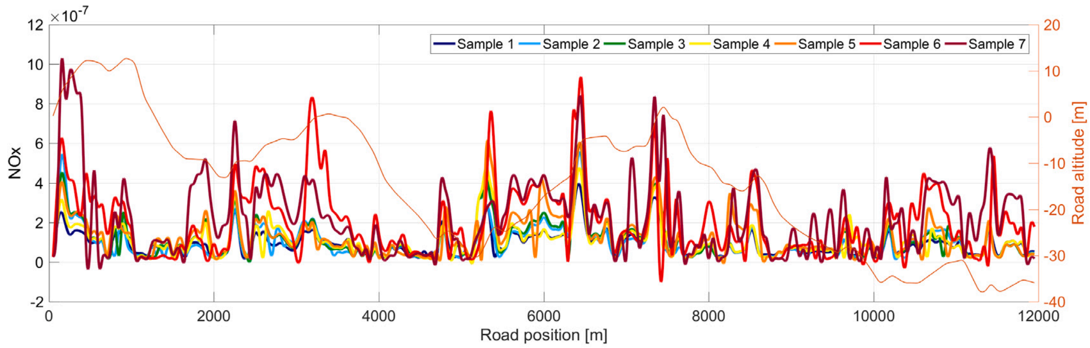
5. Implement Control Action:
Apply the first control input $U^*(i)$ (i.e., use $E_e^*(i)$ and $E_b^*(i)$) to update the vehicle state.

6. Shift Horizon:
Set $i \leftarrow i+1$ and update the prediction horizon accordingly.

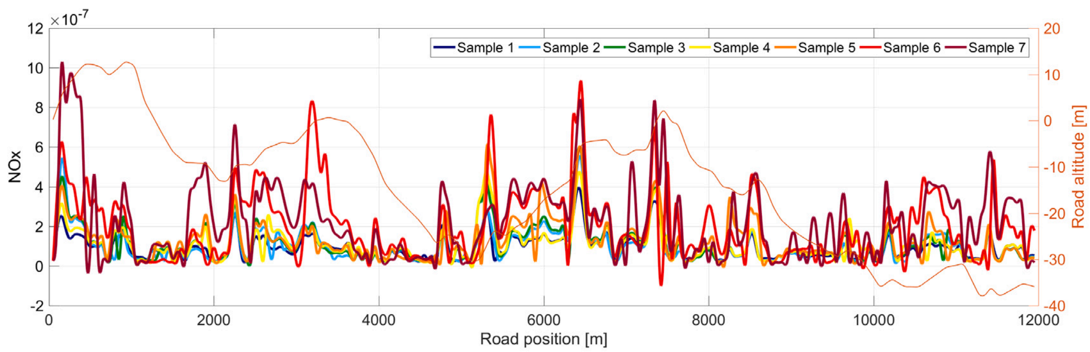
return *Sequence* $\{U^*(0), U^*(1), \ldots, U^*(N-1)\}$ *representing the optimal engine energy inputs and resulting speed profile.*

Algorithm 3: Deep Deterministic Policy Gradient (DDPG)

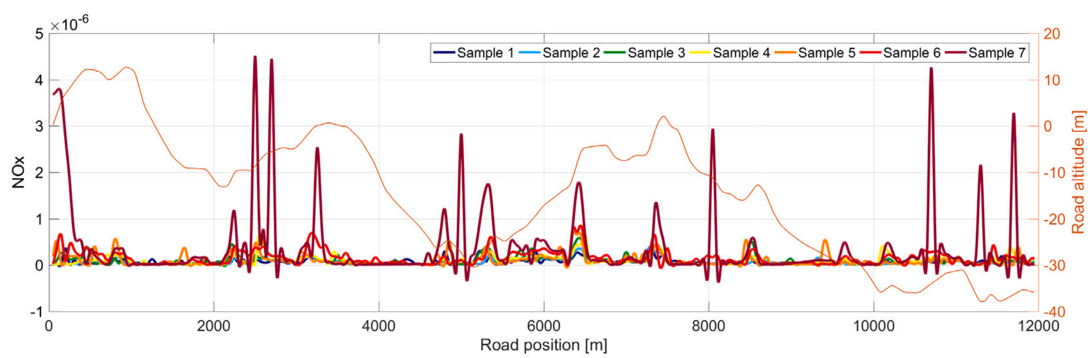
Data: Initialise actor network θ^{actor} and critic network θ^{critic} with random weights
Data: Initialise target actor network $\theta^{\text{target\_actor}} \leftarrow \theta^{\text{actor}}$ and target critic network $\theta^{\text{target\_critic}} \leftarrow \theta^{\text{critic}}$
Data: Initialise memory B
while $e \le E$ **do**
  Receive initial state s_1;
  for $t = 0$ *to* $S/\Delta s - 1$ **do**
    if *random* $\le \epsilon$ **then**
      Choose action $a_t = \mu(s_t|\theta^{\text{actor}})$;
      else
      Random choose an action within the limit;
    end
    Execute action a_t and calculate the driving speed;
    Observe step reward r_t and new state s_{t+1};
    Store transition (s_t, a_t, r_t, s_{t+1}) in B;
  end
  Give the episode reward r_e;
  Store the terminal tuple $(s_e, a_e, r_e, \_)$ in B;
  for (s_t, a_t, r_t, s_{t+1}) *in* B **do**
    Pick a transition (s_i, a_i, r_i, s_{i+1}) from B;
    Compute target value y_i;
    Calculate the loss function of critic network $\mathscr{L}(\theta^{\text{critic}})$;
    Update weights of critic network by gradient descent $\nabla\mathscr{L}(\theta^{\text{critic}})$;
    Calculate the loss function of actor network $\mathscr{L}(\theta^{\text{actor}})$;
    Update weights of actor network by gradient descent $\nabla\mathscr{L}(\theta^{\text{actor}})$
  end
  Update target networks:
  $\theta^{\text{target\_critic}} \leftarrow \tau\theta^{\text{critic}} + (1-\tau)\theta^{\text{target\_critic}}$;
  Decay exploration rate ϵ and learning rates;
end



(a) NOx emission of MPC-1 samples.

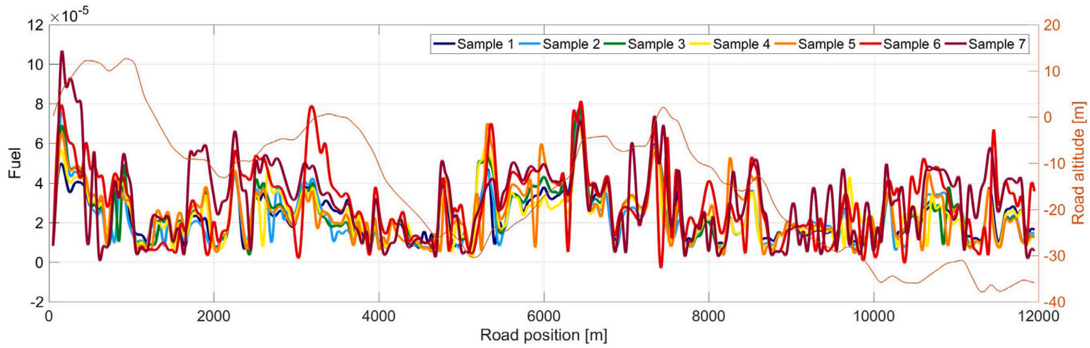


(b) NOx emission of MPC-2 samples.

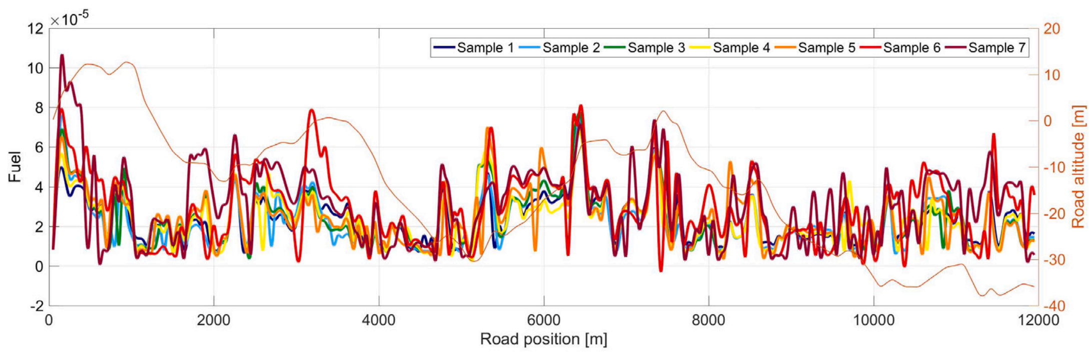


(c) NOx emission of DRL samples.

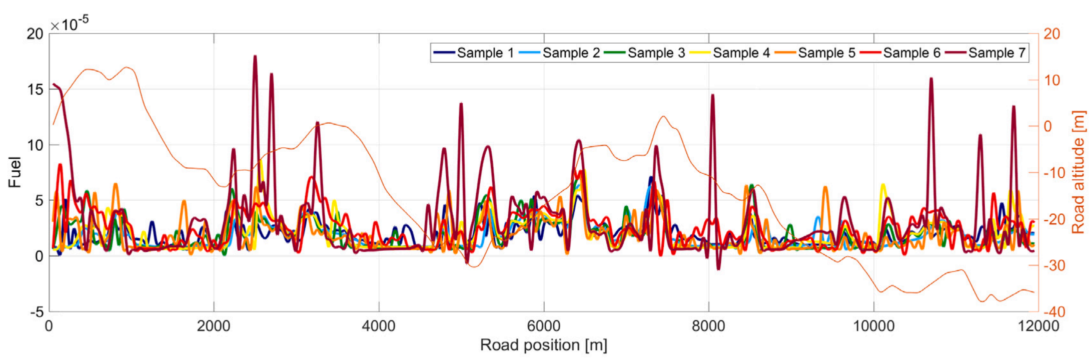
Fig. 22. Comparison of NOx emission.



(a) Fuel consumption of MPC-1 samples.

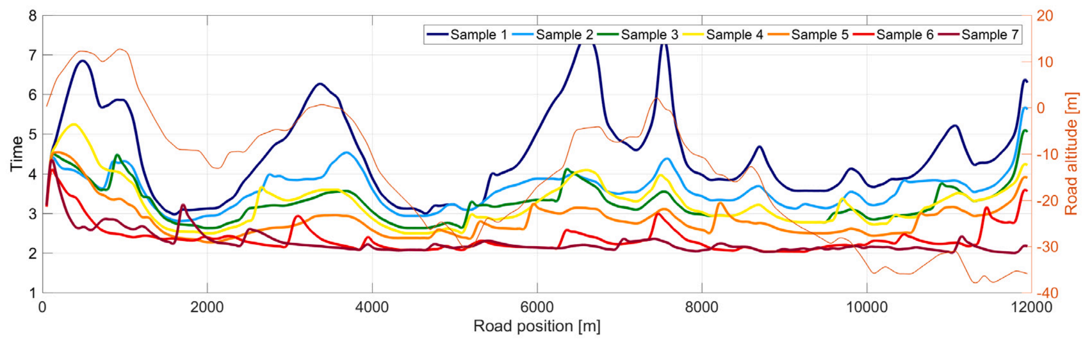


(b) Fuel consumption of MPC-2 samples.

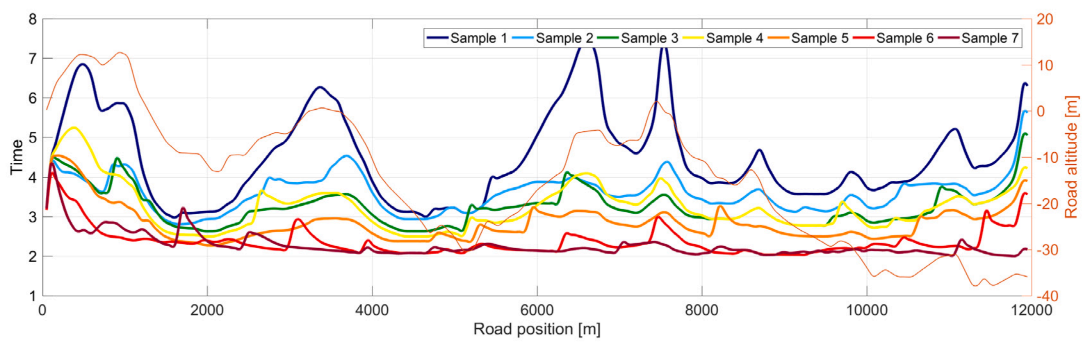


(c) Fuel consumption of DRL samples.

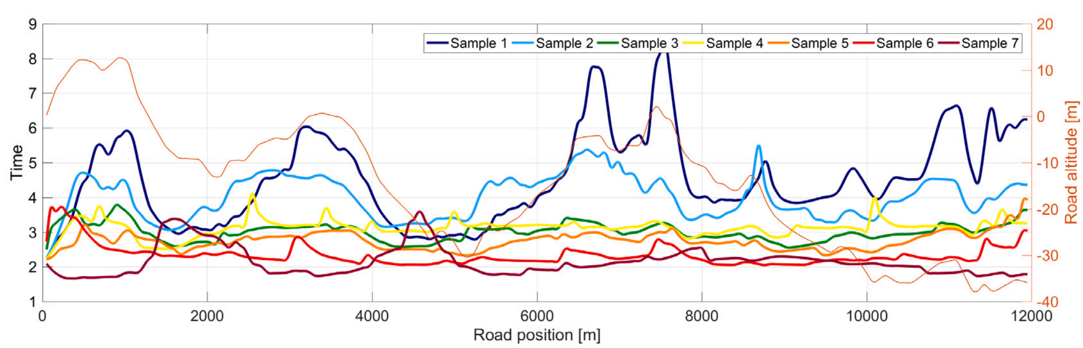
Fig. 23. Comparison of fuel consumption.



(a) Travel time of MPC-1 samples.

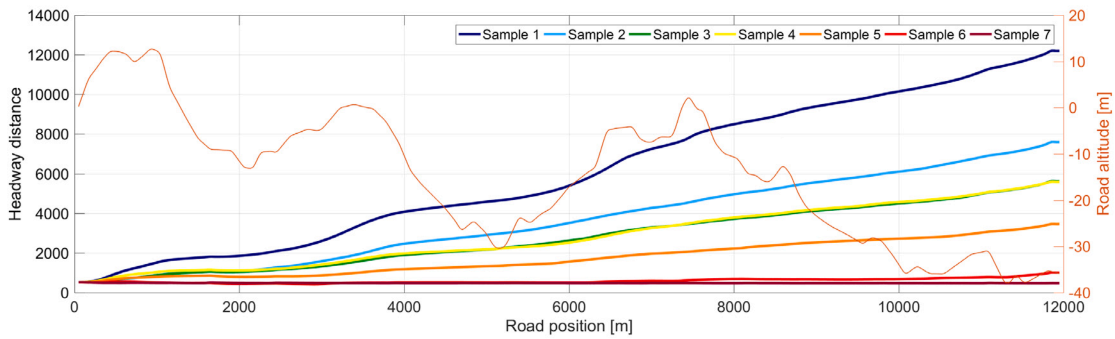


(b) Travel time of MPC-2 samples.

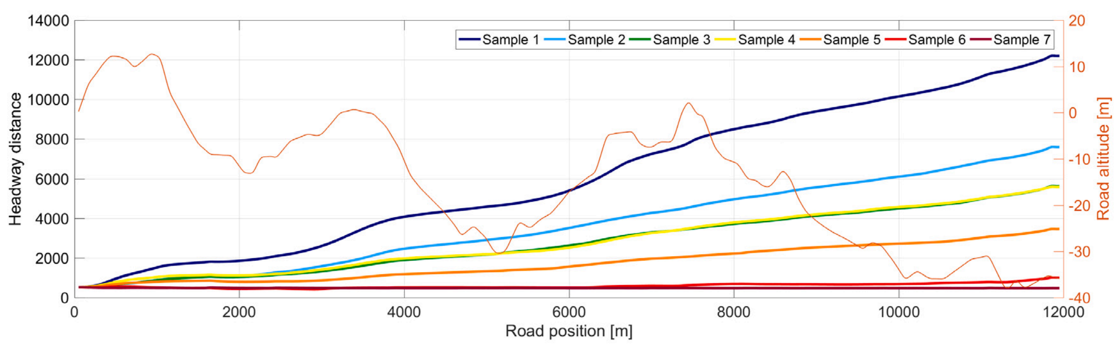


(c) Travel time of DRL samples.

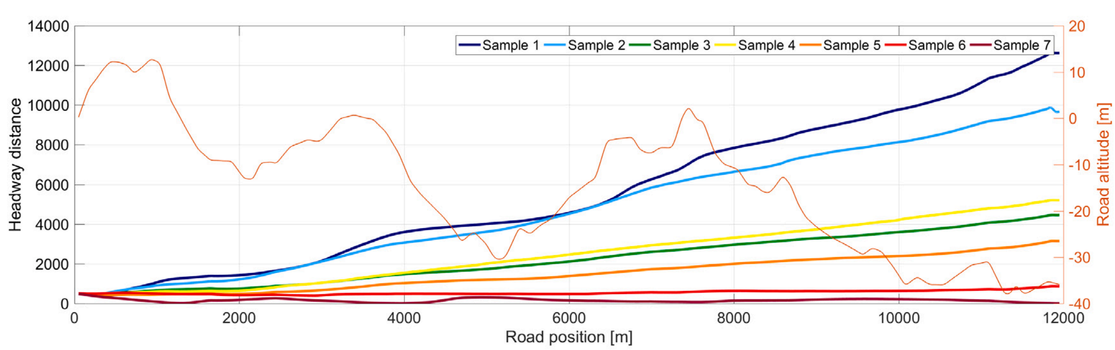
Fig. 24. Comparison of Travel time.



(a) Headway distance of MPC-1 samples.



(b) Headway distance of MPC-2 samples.



(c) Headway distance of DRL samples.

Fig. 25. Comparison of Headway distance.

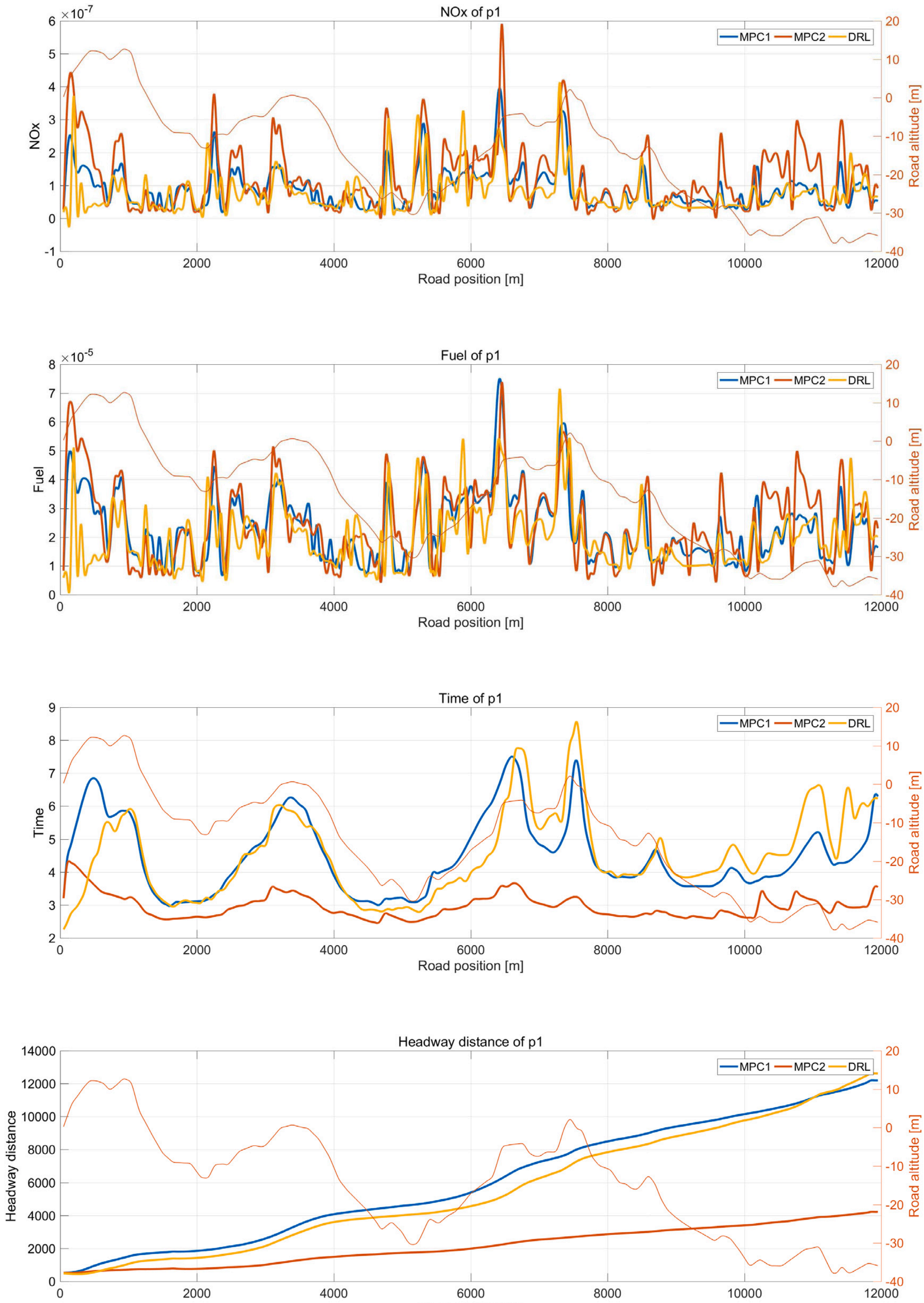


Fig. 26. Comparison of results of p_1 (minimum NOx).

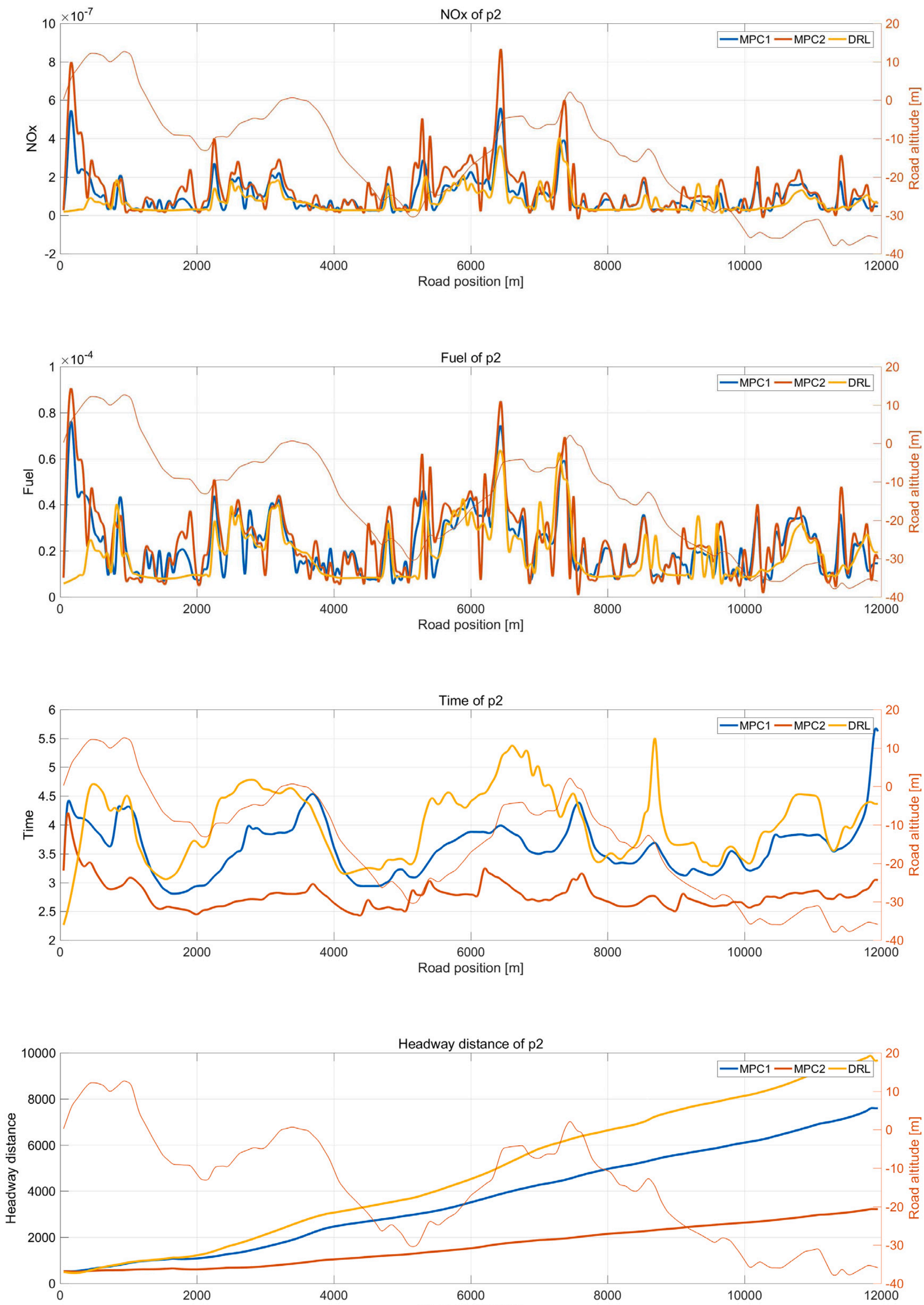


Fig. 27. Comparison of results of p_2 (minimum fuel).

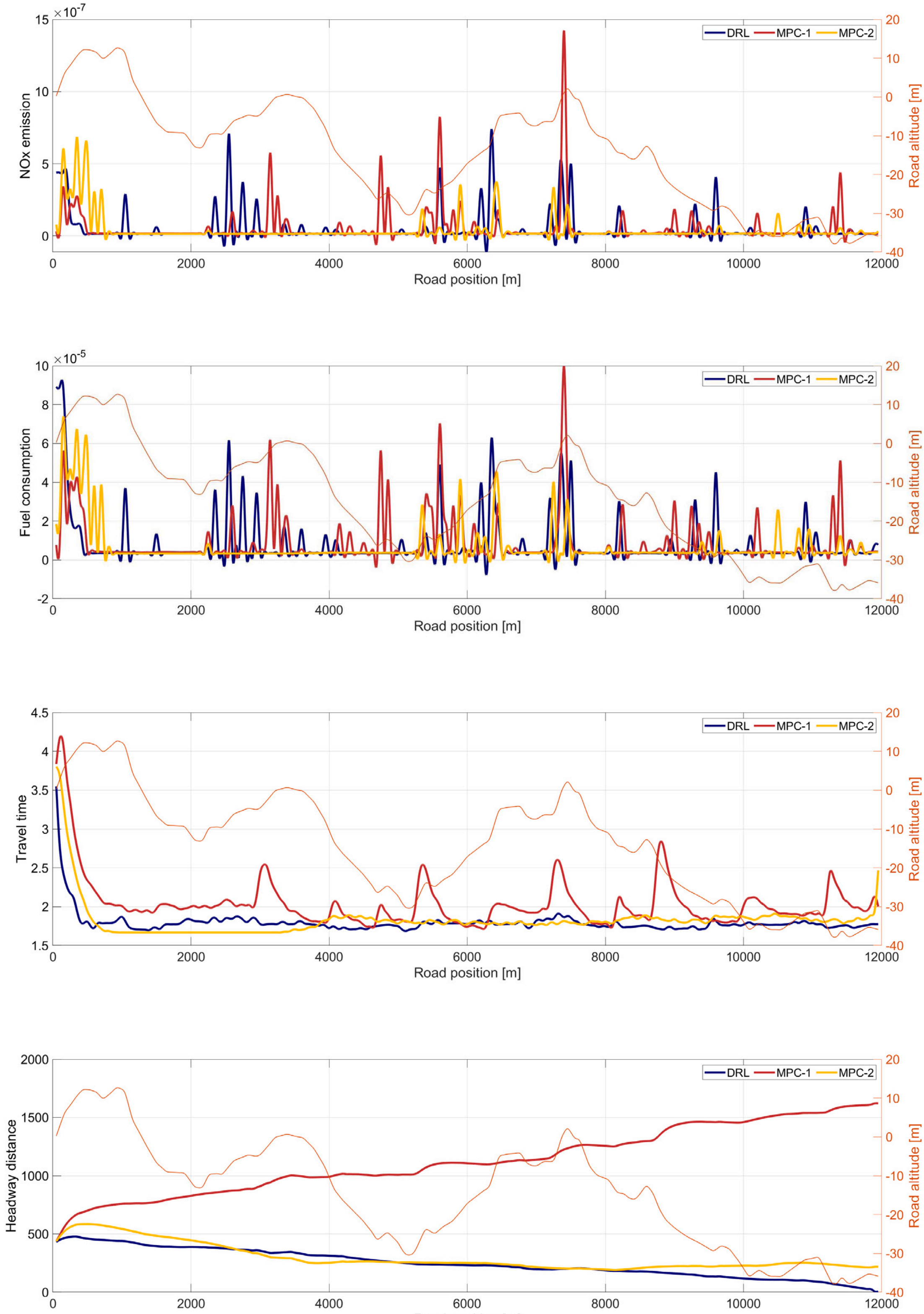


Fig. 28. Comparison of results of p_7 (minimum time).

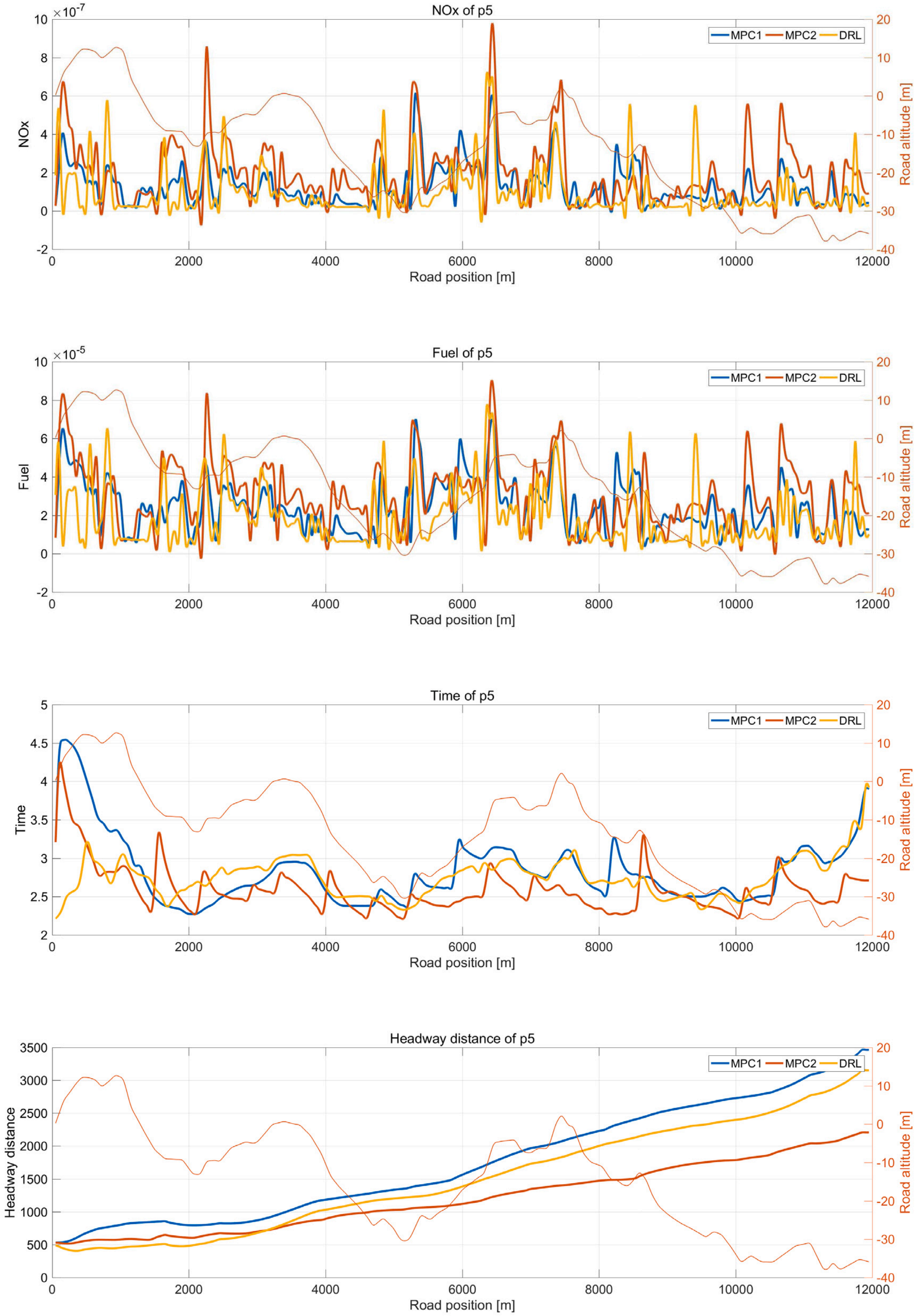


Fig. 29. Comparison of results of p_5 (knee point).

Data availability

The data that has been used is confidential.

References

[1] Placek Martin. Estimated worldwide motor vehicle production from 2000 to 2021. 2022, https://www.statista.com/statistics/262747/worldwide-automobile-production-since-2000/. (Accessed 01 December 2022).

[2] Pickl Matthias J. The renewable energy strategies of oil majors – from oil to energy? Energy Strat Rev 2019;26:100370. http://dx.doi.org/10.1016/j.esr.2019.100370, URL https://www.sciencedirect.com/science/article/pii/S2211467X19300574.

[3] Alam Assad Al, Gattami Ather, Johansson Karl Henrik. An experimental study on the fuel reduction potential of heavy duty vehicle platooning. In: 13th international IEEE conference on intelligent transportation systems. 2010, p. 306–11. http://dx.doi.org/10.1109/ITSC.2010.5625054.

[4] Jia Dongyao, Lu Kejie, Wang Jianping, Zhang Xiang, Shen Xuemin. A survey on platoon-based vehicular cyber-physical systems. IEEE Commun Surv Tutor 2016;18(1):263–84. http://dx.doi.org/10.1109/COMST.2015.2410831.

[5] Haakman Robert, Beenakker Ivo, Geerlings Harry. Reducing vehicle-related NOx and PM emissions in metropolitan areas: A comparison between the randstad and the Rhine-Ruhr area. J Clean Prod 2020;247:119175.

[6] Chossière Guillaume P, Malina Robert, Ashok Akshay, Dedoussi Irene C, Eastham Sebastian D, Speth Raymond L, Barrett Steven R H. Public health impacts of excess NOx emissions from Volkswagen diesel passenger vehicles in Germany. Environ Res Lett 2017;12(3):034014. http://dx.doi.org/10.1088/1748-9326/aa5987.

[7] Liu Ye, Chen Haibo, Li Ying, Gao Jianbing, Dave Kaushali, Chen Junyan, Li Tiezhu, Tu Ran. Exhaust and non-exhaust emissions from conventional and electric vehicles: A comparison of monetary impact values. J Clean Prod 2022;331:129965. http://dx.doi.org/10.1016/j.jclepro.2021.129965, URL https://www.sciencedirect.com/science/article/pii/S0959652621041342.

[8] Ježek I, Katrašnik T, Westerdahl D, Močnik G. Black carbon, particle number concentration and nitrogen oxide emission factors of random in-use vehicles measured with the on-road chasing method. Atmos Chem Phys 2015;15(19):11011–26.

[9] Osorio Carolina, Nanduri Kanchana. Energy-efficient urban traffic management: A microscopic simulation-based approach. Transp Sci 2015;49(3):637–51. http://dx.doi.org/10.1287/trsc.2014.0554.

[10] Kamal Md Abdus Samad, Hashikura Kotaro, Hayakawa Tomohisa, Yamada Kou, Imura Jun-ichi. Look-ahead driving schemes for efficient control of automated vehicles on urban roads. IEEE Trans Veh Technol 2022;71(2):1280–92. http://dx.doi.org/10.1109/TVT.2021.3132936.

[11] Kong Yan, Ma Yao. Connected and automated vehicles: A cooperative eco-driving strategy for heterogeneous vehicle platoon among multiple signalized intersections. IFAC- Pap 2024;58(29):272–7. http://dx.doi.org/10.1016/j.ifacol.2024.11.156, URL https://www.sciencedirect.com/science/article/pii/S2405896324022973. 7th IFAC Conference on Engine and Powertrain Control, Simulation and Modeling E-COSM 2024.

[12] Wang Shaohua, Yu Pengfei, Shi Dehua, Yu Chengquan, Yin Chunfang. Research on eco-driving optimization of hybrid electric vehicle queue considering the driving style. J Clean Prod 2022;343:130985.

[13] Cui Yuepeng, Xu Hao, Zou Fumin, Chen Zhihui, Gong Kuangmin. Optimization based method to develop representative driving cycle for real-world fuel consumption estimation. Energy 2021;235:121434. http://dx.doi.org/10.1016/j.energy.2021.121434.

[14] Hamednia Ahad, Sharma Nalin Kumar, Murgovski Nikolce, Fredriksson Jonas. Computationally efficient algorithm for eco-driving over long look-ahead horizons. IEEE Trans Intell Transp Syst 2022;23(7):6556–70. http://dx.doi.org/10.1109/TITS.2021.3058418.

[15] Jia Dongyao, Chen Haibo, Zheng Zuduo, Watling David, Connors Richard, Gao Jianbing, Li Ying. An enhanced predictive cruise control system design with data-driven traffic prediction. IEEE Trans Intell Transp Syst 2022;23(7):8170–83.

[16] Nie Zhigen, Jia Yuan, Wang Wanqiong, Chen Zheng, Outbib Rachid. Co-optimization of speed planning and energy management for intelligent fuel cell hybrid vehicle considering complex traffic conditions. Energy 2022;247:123476. http://dx.doi.org/10.1016/j.energy.2022.123476, URL https://www.sciencedirect.com/science/article/pii/S0360544222003796.

[17] Lot Roberto, Fleming James, Chen Boli, Evangelou Simos. Eco-driving optimal control for electric vehicles with driver preferences. Transp Eng 2025;19:100302. http://dx.doi.org/10.1016/j.treng.2025.100302, URL https://www.sciencedirect.com/science/article/pii/S2666691X25000028.

[18] Fernández-Yáñez Pablo, Soriano José A, Mata Carmen, Armas Octavio, Pla Benjamín, Bermúdez Vicente. Simulation of optimal driving for minimization of fuel consumption or NOx emissions in a diesel vehicle. Energies 2021;14(17).

[19] Yuval Omer, Lin Zhiyuan, Chen Haibo. Multiobjective speed profile optimisation considering fuel and NOx. In: Proceedings of the 15th ITS European congress. 2023.

[20] Ozatay Engin, Onori Simona, Wollaeger James, Ozguner Umit, Rizzoni Giorgio, Filev Dimitar, Michelini John, Di Cairano Stefano. Cloud-based velocity profile optimization for everyday driving: A dynamic-programming-based solution. IEEE Trans Intell Transp Syst 2014;15(6):2491–505.

[21] Ehrgott Matthias. Multicriteria optimization. Berlin, Heidelberg: Springer-Verlag; 2005.

[22] Rawlings JB, Mayne DQ, Diehl M. Model predictive control: theory, computation, and design. Nob Hill Publishing; 2017.

[23] Moffaert Kristof Van, Nowé Ann. Multi-objective reinforcement learning using sets of pareto dominating policies. J Mach Learn Res 2014;15:3483–512.

[24] Li Kaiwen, Zhang Tao, Wang Rui. Deep reinforcement learning for multiobjective optimization. IEEE Trans Cybern 2021;51(6):3103–14.

[25] Technologies Gamma. GT-SUITE. 2022, https://www.gtisoft.com/gt-suite/. (Accessed 01 December 2022).

[26] Eriksson Lars, Thomasson Andreas, Ekberg Kristoffer, Reig Alberto, Eifert Mark, Donatantonio Fabrizio, D'Amato Antonio, Arsie Ivan, Pianese Cesare, Otta Pavel, Held Manne, Vögele Ulrich, Endisch Christian. Look-ahead controls of heavy duty trucks on open roads — six benchmark solutions. Control Eng Pract 2019;83:45–66. http://dx.doi.org/10.1016/j.conengprac.2018.10.014.

[27] Sharma Nalin Kumar, Hamednia Ahad, Murgovski Nikolce, Gelso Esteban R, Sjöberg Jonas. Optimal eco-driving of a heavy-duty vehicle behind a leading heavy-duty vehicle. IEEE Trans Intell Transp Syst 2021;22(12):7792–803. http://dx.doi.org/10.1109/TITS.2020.3009288.

[28] Hellström Erik, Ivarsson Maria, Åslund Jan, Nielsen Lars. Look-ahead control for heavy trucks to minimize trip time and fuel consumption. Control Eng Pract 2009;17(2):245–54.

[29] Zhai Chunjie, Luo Fei, Liu Yonggui, Chen Ziyang. Ecological cooperative look-ahead control for automated vehicles travelling on freeways with varying slopes. IEEE Trans Veh Technol 2019;68(2):1208–21.

[30] Liu Peng, Ozguner Umit, Zhang Yeqing. Distributed MPC for cooperative highway driving and energy-economy validation via microscopic simulations. Transp Res Part C: Emerg Technol 2017;77:80–95.

[31] Schlechtendahl Jan, Kretschmer Felix, Sang Zhiqian, Lechler Armin, Xu Xun. Extended study of network capability for cloud based control systems. Robot Comput-Integr Manuf 2017;43:89–95.

[32] Khalatbarisoltani Arash, Han Jie, Liu Wenxue, Hu Xiaosong. Speedy hierarchical eco-planning for connected multi-stack fuel cell vehicles via health-conscious decentralized convex optimization. SAE Int J Electr Veh 2023;13(1):93–106.

[33] Mensing Felicitas, Trigui Rochdi, Bideaux Eric. Vehicle trajectory optimization for application in ECO-driving. In: 2011 IEEE vehicle power and propulsion conference. 2011, p. 1–6. http://dx.doi.org/10.1109/VPPC.2011.6042993.

[34] Hu Bo, Li Jiaxi. An adaptive hierarchical energy management strategy for hybrid electric vehicles combining heuristic domain knowledge and data-driven deep reinforcement learning. IEEE Trans Transp Electr. 2021;8(3):3275–88.

[35] Dong Haoxuan, Wang Qun, Zhuang Weichao, Yin Guodong, Gao Kun, Li Zhaojian, Song Ziyou. Flexible eco-cruising strategy for connected and automated vehicles with efficient driving lane planning and speed optimization. IEEE Trans Transp Electr 2023.

[36] Yang Ningkang, Han Lijin, Liu Rui, Wei Zhengchao, Liu Hui, Xiang Changle. Multi-objective intelligent energy management for hybrid electric vehicles based on multi-agent reinforcement learning. IEEE Trans Transp Electr 2023.

[37] Xia Jiang Jian Zhang, Li Dan. Eco-driving at signalized intersections: a parameterized reinforcement learning approach. Transp B: Transp Dyn 2023;11(1):1406–31.

[38] Yang Zhiwei, Zheng Zuduo, Kim Jiwon, Rakha Hesham. Eco-driving strategies using reinforcement learning for mixed traffic in the vicinity of signalized intersections. Transp Res Part C: Emerg Technol 2024;165:104683.

[39] Zhu Zhaoxuan, Gupta Shobhit, Gupta Abhishek, Canova Marcello. A deep reinforcement learning framework for eco-driving in connected and automated hybrid electric vehicles. IEEE Trans Veh Technol 2024;73(2):1713–25.

[40] Li Menglin, Wan Xiangqi, Yan Mei, Wu Jingda, He Hongwen. Attentive hybrid reinforcement learning-based eco-driving strategy for connected vehicles with hybrid action spaces and surrounding vehicles attention. Energy Convers Manage 2024;321:119059.

[41] Fan Jiawei, Wu Xiaodong, Li Jie, Xu Min. Deep reinforcement learning based integrated eco-driving strategy for connected and automated electric vehicles in complex urban scenarios. IEEE Trans Veh Technol 2024;73(4):4621–35.

[42] Khalatbarisoltani Arash, Han Jie, Liu Wenxue, Liu Cong-zhi, Hu Xiaosong. Health-consciousness integrated thermal and energy management of connected hybrid electric vehicles using cooperative multi-agent deep reinforcement learning. IEEE Trans Intell Veh 2024;1–12.

[43] Jia Chunchun, He Hongwen, Zhou Jiaming, Li Jianwei, Wei Zhongbao, Li Kunang, Li Menglin. A novel deep reinforcement learning-based predictive energy management for fuel cell buses integrating speed and passenger prediction. Int J Hydrog Energy 2025;100:456–65.

[44] Huang Yuhan, Ng Elvin CY, Zhou John L, Surawski Nic C, Lu Xingcai, Du Bo, Forehead Hugh, Perez Pascal, Chan Edward FC. Impact of drivers on real-driving fuel consumption and emissions performance. Sci Total Environ 2021;798:149297.

[45] Tang Xiaolin, Chen Jiaxin, Liu Teng, Qin Yechen, Cao Dongpu. Distributed deep reinforcement learning-based energy and emission management strategy for hybrid electric vehicles. IEEE Trans Veh Technol 2021;70(10):9922–34.

[46] Guo Xiaokai, Yan Xianguo, Chen Zhi, Meng Zhiyu. Research on energy management strategy of heavy-duty fuel cell hybrid vehicles based on dueling-double-deep Q-network. Energy 2022;260:125095. http://dx.doi.org/10.1016/j.energy.2022.125095, URL https://www.sciencedirect.com/science/article/pii/S0360544222019909.

[47] Yuan Weichang, Frey H Christopher, Wei Tongchuan. Fuel use and emission rates reduction potential for light-duty gasoline vehicle eco-driving. Transp Res Part D: Transp Environ 2022;109:103394.

[48] Jia Chunchun, Zhou Jiaming, He Hongwen, Li Jianwei, Wei Zhongbao, Li Kunang, Shi Man. A novel energy management strategy for hybrid electric bus with fuel cell health and battery thermal- and health-constrained awareness. Energy 2023;271:127105.

[49] Han Jie, Khalatbarisoltani Arash, Yang Yalian, Hu Xiaosong. Energy management in plug-in hybrid electric vehicles: Preheating the battery packs in low-temperature driving scenarios. IEEE Trans Intell Transp Syst 2024;25(2):1978–91.

[50] Wang Yongfeng, Li Shuguang, Bu sinnah Zainab Ali, Ghandour Raymond, Khan Mohammad Nadeem, Ali H Elhosiny. Optimizing energy efficiency and emission reduction: Leveraging the power of machine learning in an integrated compressed air energy storage-solid oxide fuel cell system. Energy 2024;313:133962. http://dx.doi.org/10.1016/j.energy.2024.133962, URL https://www.sciencedirect.com/science/article/pii/S036054422403740X.

[51] Gao Jianbing, Chen Haibo, Liu Ye, Li Ying. Impacts of de-NOx system layouts of a diesel passenger car on exhaust emission factors and monetary penalty. Energy Sci Eng 2021;9(12):2268–80.

[52] Rakha Hesham A, Ahn Kyoungho, Moran Kevin, Saerens Bart, den Bulck Eric Van. Virginia tech comprehensive power-based fuel consumption model: Model development and testing. Transp Res Part D: Transp Environ 2011;16(7):492–503. http://dx.doi.org/10.1016/j.trd.2011.05.008, URL https://www.sciencedirect.com/science/article/pii/S1361920911000782.

[53] Wang Jinghui, Rakha Hesham A. Fuel consumption model for heavy duty diesel trucks: Model development and testing. Transp Res Part D: Transp Environ 2017;55:127–41.

[54] Li Daofei, Jiang Yangye, Shen Yijie. Intersection eco-driving for automated vehicles: SMPC-based strategies for handling leading vehicle starting-up uncertainties. Energy 2024;302. http://dx.doi.org/10.1016/j.energy.2024.131781.

[55] Chen Chen, Zhao Xiaohua, Yao Ying, Zhang Yunlong, Rong Jian, Liu Xiaoming. Driver's eco-driving behavior evaluation modeling based on driving events. J Adv Transp 2018. http://dx.doi.org/10.1155/2018/9530470.

[56] Gambier Adrian, Badreddin Essameddin. Multi-objective optimal control: An overview. In: 2007 IEEE international conference on control applications. 2007, p. 170–5. http://dx.doi.org/10.1109/CCA.2007.4389225.

[57] Bemporad Alberto, Muñoz de la Peña David. Multiobjective model predictive control. Automatica 2009;45(12):2823–30.

[58] Grüne Lars, Stieler Marleen. Performance guarantees for multiobjective model predictive control. In: 2017 IEEE 56th annual conference on decision and control. CDC, 2017, p. 5545–50. http://dx.doi.org/10.1109/CDC.2017.8264482.

[59] Sutton Richard S, Barto Andrew G. Reinforcement learning: an introduction. 2nd ed.. Cambridge, MA: MIT Press; 2018.

[60] Bakos Steve, Davoudi Heidar. Mitigating cowardice for reinforcement learning agents in combat scenarios. In: 2022 IEEE conference on games. CoG, IEEE; 2022, p. 377–84. http://dx.doi.org/10.1109/CoG51982.2022.9893546.

[61] Lopez Pablo Alvarez, Behrisch Michael, Bieker-Walz Laura, Erdmann Jakob, Flötteröd Yun-Pang, Hilbrich Robert, Lücken Leonhard, Rummel Johannes, Wagner Peter, Wießner Evamarie. Microscopic traffic simulation using SUMO. In: The 21st IEEE international conference on intelligent transportation systems. IEEE; 2018, URL https://elib.dlr.de/124092/.

[62] The shuttle radar topography mission. Rev Geophys 2007. http://dx.doi.org/10.1029/2005RG000183.

[63] Wargula Łukasz, Wieczorek Bartosz, Kukla Mateusz. The determination of the rolling resistance coefficient of objects equipped with the wheels and suspension system – results of preliminary tests. MATEC Web Conf 2019. http://dx.doi.org/10.1051/matecconf/201925401005.

[64] Windsor S. Real world drag coefficient – is it wind averaged drag? In: The international vehicle aerodynamics conference. 2014, http://dx.doi.org/10.1533/9780081002452.1.3.

[65] Bae Il, Moon Jaeyoung, Seo Jeongseok. Toward a comfortable driving experience for a self-driving shuttle bus. Electron (Switzerland) 2019. http://dx.doi.org/10.3390/electronics8090943.

[66] Das Indraneel. On characterizing the "knee" of the Pareto curve based on normal-boundary intersection. Struct Optim 1999;18:107–15. http://dx.doi.org/10.1007/BF01195985.

[67] Branke Jürgen, Deb Kalyanmoy, Dierolf Henning, Osswald Matthias. Finding knees in multi-objective optimization. In: Parallel problem solving from nature. 2004, URL https://api.semanticscholar.org/CorpusID:9883277.

[68] Ahuja RK, Magnanti TL, Orlin JB. Network flows: theory, algorithms, and applications. Prentice Hall; 1993.