Adaptive Multi-Objective Predictive Cruise Control With Digital Map Using a Utopia Tracking Method

Yongjun Yan, Ziyou Song, *Senior Member, IEEE*, Bingzhao Gao, *Member, IEEE*, Hong Chen, *Fellow, IEEE*, and Jing Sun, *Fellow, IEEE*

***Abstract*— The integration of look-ahead information into Model Predictive Control (MPC) frameworks has shown promise for intelligent transportation systems. However, transitioning Predictive Cruise Control (PCC) system research into practical application poses challenges due to numerous weighting parameters and increased computational demands in complex driving environments. Although the Weighted Sum Method is commonly used in PCC system research to balance fuel consumption and trip time objectives, it requires time-consuming weight tuning and often results in suboptimal performance due to fixed weighting parameters. To address this, this paper proposes a Utopia-tracking Model Predictive Control (UTM-MPC) controller, where the cost function is reformulated as the sum of the distances between the objectives and the average Utopia point over the prediction horizon. By analyzing the Pareto front of the PCC optimization problem under varying slope profiles extracted from digital map data, we demonstrate that the proposed UTM-MPC effectively leverages the geometric characteristics of the Pareto front to identify preferred trade-off solutions. The adaptive weighting mechanism—derived from the online-calculated Utopia point—enhances the robustness of the PCC system under complex and dynamic driving conditions. To mitigate the computational burden associated with integrating UTM-MPC into the MPC framework, we introduce a tailored neighboring extremal-based solving algorithm. Leveraging the receding horizon nature of MPC, this method requires only minimal updates to efficiently identify an optimal solution near the nominal trajectory from the previous sampling instance. Simulation results show that the UTM-MPC controller, with its adaptive weighting strategy, consistently outperforms the traditional Weighted Sum Method in terms of both fuel efficiency and trip time.**

***Index Terms*— Multi-objective predictive control, predictive cruise control, utopia-tracking.**

Received 8 March 2021; revised 13 November 2021, 25 March 2023, 17 May 2024, and 7 April 2025; accepted 8 April 2025. Date of publication 30 April 2025; date of current version 2 June 2025. The work of Yongjun Yan was supported in part by Chinese Scholarship Council and in part by the Chinese Scholarship Council under Grant 201906170182. The Associate Editor for this article was S. Sacone. *(Corresponding authors: Ziyou Song; Bingzhao Gao.)*

Yongjun Yan is with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong, SAR, China, and also with the Multi-Scale Medical Robotics Center, Hong Kong (e-mail: yanyongjun18@gmail.com).

Ziyou Song is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA, and also with the Department of Mechanical Engineering, National University of Singapore, Singapore 117575 (e-mail: ziyou@umich.edu).

Bingzhao Gao is with the Clean Energy Automotive Engineering Center, Tongji University, Shanghai 201804, China, and also with the State Key Laboratory of Automotive Simulation and Control, Jilin University, Changchun 130022, China (e-mail: gaobz@jlu.edu.cn).

Hong Chen is with the College of Electronic and Information Engineering, Tongji University, Shanghai 201804, China (e-mail: chenhong2019@tongji.edu.cn).

Jing Sun is with the Department of Naval Architecture and Marine Engineering, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: jingsun@umich.edu).

Digital Object Identifier 10.1109/TITS.2025.3559918

NOMENCLATURE

$\alpha_{1\sim6}$	Coefficients of fuel consumption model, [-].
β	Weight of weighted sum method, [-].
$\dot{m}_f$	Instantaneous fuel consumption rate, $[g/s]$.
η	Gearbox efficiency, [-].
ω_{e}	Rotational velocity of the engine, $[rpm]$.
ω_{w}	Rotational velocity of wheel, $[rpm]$.
λ	Co-states variables, [-].
μ	Lagrange multiplier, [-].
Φ	Terminal penalty function, [-].
ϕ	Slope data, [°].
Φ_1	Fuel consumption in one sampling period, $[g]$.
Φ_2	trip time in one sampling period, $[s]$.
ρ	Air density, $[kg/m^3]$.
ε	Tolerance constant, [-].
A	Frontal projected area of vehicle, $[m^2]$.
C_d	Coefficient of drag, [-].
$C(\cdot)$	State-input constraints, [-].
F_{r}	Resistance force, $[Nm]$.
F_{t}	Traction force, $[Nm]$.
f_r	Coefficient of rolling resistance, [-].
$f(\cdot)$	Vehicle dynamic equation, [-].
g	Gravitational constant, [-].
I_{final}	Gear ratio of final reduction drive, [-].
I_g	Gear ratio of gearbox, [-].
J	Cost function, [-].
J_1	Accumulated fuel consumption, $[g]$.
J_2	Accumulated trip time over the horizon, $[s]$.
H	Hamiltonian function, [-].
m	Vehicle mass, $[kg]$.
r	Wheel radius, $[m]$.
s	Space coordinate, $[m]$.
x	State variable, [-].
T_{b}	Brake torque, $[N]$.
T_{e}	Engine torque, $[N]$.
T_{final}	Final driveshaft torque, $[N]$.
v	Vehicle longitudinal velocity, $[m/s]$.

ACRONYMS

ACC	Adaptive cruise control.
ADAS	Advanced driver-assistance system.
ECUs	Electronic control units.
IPA-SQP	Integrated perturbation analysis and sequential quadratic programming.
MOOP	Multi-objective optimization problem.
MPC	Model predictive control.
NE	Neighboring extremal.
NLP	Nonlinear programming.
OEMs	Original equipment manufacturers.
SQP	Sequential quadratic programming.
SUV	Sports utility vehicle.
UTM	Utopia-tracking method.
WSM	Weighted sum method.

I. INTRODUCTION

IN THE field of intelligent transportation systems, Predictive Cruise Control (PCC) has emerged as a next-generation concept that enables vehicles to anticipate and respond to upcoming road conditions without driver intervention. This advancement has been particularly prominent in the commercial vehicle sector, where major Original Equipment Manufacturers (OEMs) have integrated predictive capabilities into their highway fleets through the use of an electronic horizon [1]. These systems commonly employ rule-based strategies that adjust vehicle acceleration or deceleration in anticipation of road gradients [2], or they follow an optimal velocity profile precomputed offline based on slope information [1].

Extending PCC to passenger vehicles presents notable challenges, primarily due to the complexity and variability of real-world traffic and driving environments. These factors limit the effectiveness of rule-based strategies or offline-computed velocity profiles [3]. While earlier studies have demonstrated the fuel-saving potential of integrating static look-ahead information—such as road slope—into control strategies [4], [5], [6], [7], more recent research has identified Model Predictive Control (MPC) frameworks as a promising solution for achieving real-time, adaptive PCC performance in dynamic driving conditions.

Two main factors hinder the practical implementation of PCC systems in passenger vehicles: the complexity of weight tuning and the high computational demand. The numerous weighting parameters make calibration time-consuming and error-prone, while the computational burden exceeds the capabilities of standard electronic control units (ECUs). Our previous field tests in Chongqing, China, demonstrated that an MPC-based PCC controller utilizing look-ahead information (slope, speed limits, and curvature) can reduce fuel consumption by 8.73% compared to a conventional Adaptive Cruise Control (ACC) system [8], [9]. However, integrating such a system as a sub-function in mass-produced vehicles remains impractical. Its reliance on fixed weights limits adaptability to varying traffic scenarios, and its computational load is too demanding for typical ECUs.

At the core of the PCC problem is a trade-off between fuel consumption and trip time, forming a multi-objective optimization problem [5], [10], [11], [12]. The commonly used Weighted Sum Method (WSM) [7], [13] seeks to balance these objectives, but selecting appropriate weights is nontrivial. Fixed weights optimized for one scenario often degrade performance in others, leading to instability and reduced robustness [14], [15].

To overcome the limitations of the weighted sum method in the receding horizon control framework for PCC on hilly roads, a fast fuzzy-tuned MPC framework has been proposed [16]. This framework considers the notion of Pareto optimality, which refers to solutions that cannot be improved upon in one objective without compromising another [17]. Bemporad et al. introduced an MPC where the control action is selected from a set of Pareto optimal solutions, based on state-dependent scaling weights that vary over time [18]. Another recent development is the explicit multi-objective MPC, which reduces computational complexity by precomputing a library of Pareto optimal solutions [19]. These solutions form the Pareto front, which represents a preferred trade-off between conflicting objectives [20]. Zavala et al. proposed a Utopia-tracking method (UTM) to compute a compromise solution by minimizing the distance between a set of objective functions and its steady-state Utopia point [21]. However, this method is limited to systems with slow dynamics, such as microgrids [22] and building energy management [15]. In the PCC problem with an economic cost function, the steady-state Utopia point [21] is unsuitable, as vehicle cruising lacks a steady state under dynamic driving conditions like acceleration and deceleration.

To address the computational complexity challenges in real-time MPC implementations, various methods have been proposed, with a significant focus on the sequential quadratic programming (SQP) framework [23], [24], [25]. The fundamental idea behind an SQP-based model predictive controller is to apply a finite number of SQP iterations at each sampling instant, warming up the iterations with the solution estimate from the previous sampling instant. Diehl et al. [26], [27] introduced a real-time iteration scheme to enhance computational efficiency by performing only one SQP iteration at each sampling time with a Gauss-Newton Hessian approximation. However, this approach may not ensure stability in the presence of inequality constraints. A time-distributed optimization approach in MPC is proposed in [24]. This MPC controller distributes the optimization iterates over time by maintaining a running solution estimate for the optimal control problem and updating it at each sampling instant.

To improve computational efficiency for MPC with complexity on the order of the prediction horizon N, the Integrated Perturbation Analysis and Sequential Quadratic Programming (IPA-SQP) algorithm has been applied to solve constrained MPC problems [28]. The IPA-SQP algorithm updates the solution at time t by treating it as a perturbation of the solution at time $(t - i)$, and refines it through SQP updates to maintain optimality. The computational efficiency of the IPA-SQP algorithm has been validated across various applica-

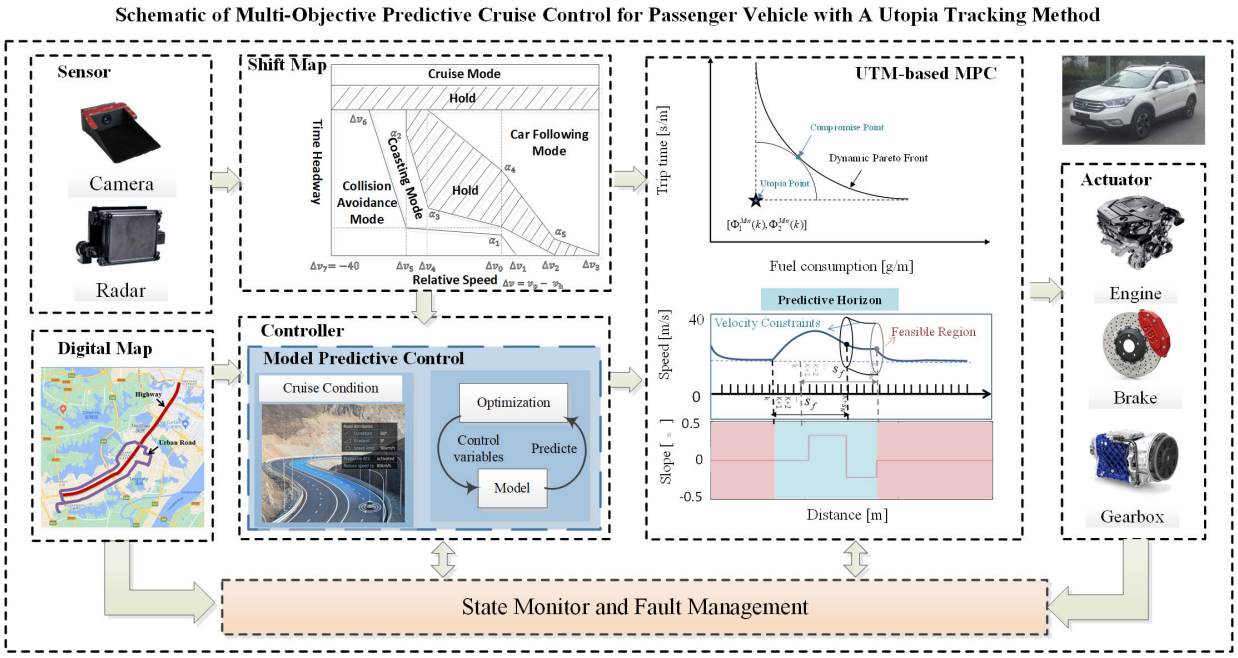


Fig. 1. Schematic of multi-objective predictive cruise control for a passenger vehicle with a utopia tracking method.

tions, including voltage regulation of a DC/DC converter [29] and shipboard power management [30].

To bridge the gap between research and practical applications of PCC systems on passenger vehicles with digital maps, this work addresses two major challenges: (1) the non-trivial weight tuning process and (3) the computational burden. A Utopia tracking-based MPC (UTM-MPC) method is proposed, which eliminates the need for manual weight tuning by employing a Utopia point chase mechanism to overcome the suboptimality associated with fixed weighting parameters. Additionally, a neighboring extremal method is introduced for predictive cruise control problems to reduce the computational load of the UTM-MPC controller. Comprehensive simulation results highlight the limitations of fixed weights used in the Weighted Sum Method under challenging driving conditions. In contrast, the UTM-MPC controller, with its adaptive weighting strategy, achieves superior performance without the need for manual tuning.

The novelties of this paper are summarized below:

1) Predefined weights for conflicting objectives (i.e., fuel consumption and trip efficiency) often lead to suboptimal performance under varying driving conditions, such as changes in slope and speed limits. The proposed UTM-MPC method employs a Utopia point chase mechanism by calculating the Utopia point over the horizon in advance to determine the best trade-off based on current conditions.

2) Considering the computational burden, we introduce a neighboring extremal method for PCC problems subject to inequality constraints from speed limits. Given the receding horizon nature of MPC, this method only requires minimal research to find the optimal solution near the nominal solution from the last sampling instance. This significantly reduces UTM-MPC computation time to under 0.1 seconds, even when solving two optimal control problems in advance to formulate the UTM-based cost function.

3) The proposed UTM-MPC controller demonstrates advanced control performance and real-time implementation capability under various driving conditions, which is validated through simulation results.

The remainder of this paper is organized as follows: Section II discusses the vehicle modeling and PCC problem statement; Section III describes the multi-objective optimization problem formulation; Section IV presents the traditional WSM, the proposed UTM-MPC, and the IPA-SQP algorithm; Section V demonstrates its performance with simulation results; Section VI concludes and presents a plan for future work.

II. Problem Statement and Vehicle Models

This section provides a detailed discussion on the scope, assumptions, and inherent trade-offs in modeling the PCC system. The objective is to clarify the rationale behind the PCC controller design while highlighting the balance between model accuracy, computational feasibility, and practical applicability.

A. Predictive Cruise Control Problem Statement

In this study, we use a conventional sports utility vehicle (SUV) equipped with a combustion engine and a 6-speed automatic transmission. The PCC strategy is implemented as a subsystem of the advanced driver-assistance system (ADAS), activated in cruise mode (Fig. 1), where the ego vehicle operates at relatively high speeds with a large time headway from the preceding vehicle. Subsequently, the system optimizes the vehicle's engine torque and braking profile to minimize fuel consumption and improve transportation efficiency.[1] This is

[1]Transportation efficiency refers to completing the entire trip as quickly as possible.

achieved by using real-time data on road topology and traffic conditions within a receding horizon framework, enabling continuous updates and adjustments based on the latest information. Detailed assumptions and considerations for the PCC problem are outlined below:

1) Road Topology Awareness: The model assumes access to accurate road topology data, including gradients and curvature, to forecast upcoming driving conditions and plan appropriate acceleration or deceleration maneuvers. The system uses GPS data to query a digital map for slope and speed limit information over the prediction horizon. Although this assumption may not fully capture real-world uncertainties, it provides a necessary approximation for tractable optimization.

2) Traffic Conditions: Given the variability of real-world traffic, the PCC system operates within a receding horizon framework, utilizing relative distance and speed charts (Fig. 1). Sudden traffic changes, such as a vehicle cutting in at a substantial relative distance, are accounted for by adjusting the feasible velocity zone based on the maximum speed vector when planning the optimal speed profile. This enables dynamic adaptation to changing traffic conditions.

3) Vehicle Model Simplification: To enhance computational efficiency, the model simplifies vehicle dynamics, focusing primarily on longitudinal behavior while omitting complex lateral dynamics. This abstraction ensures real-time implementation without significantly compromising performance, though we acknowledge that it may miss some aspects of vehicle behavior.

4) Safety Constraints: The PCC system assumes that safety-critical scenarios, such as dangerous cut-ins, are handled by other ADAS subsystems (Fig. 1). As such, these scenarios are excluded from the PCC optimization, simplifying the problem to focus on optimizing engine torque and braking profiles under normal cruising conditions.

B. Vehicle Model

1) Vehicle Longitudinal Model: Control-oriented longitudinal vehicle models are initially developed for numerical optimization to capture the vehicle's inertial dynamics and predict energy consumption. Table I lists the vehicle's key characteristics and parameters. The longitudinal dynamic model is described as

$$\begin{aligned} \dot{s} &= v, \\ \dot{v} &= \frac{1}{m}\left[F_{\mathrm{t}} - F_{\mathrm{r}}(s, v)\right], \end{aligned} \tag{1}$$

where s is the distance, v is the velocity, F_{t} is the traction force and F_{r} is the total resistance force.

2) Resistance Force: F_{r} consists of the aerodynamic resistance, the grade resistance, and the rolling resistance, as calculated by

$$F_{\mathrm{r}}(s, v) = mg\left(\sin(\phi(s)) + f_{\mathrm{r}}\cos(\phi(s))\right) + \frac{C_{\mathrm{d}}\rho A}{2}v^2, \tag{2}$$

where m is the vehicle mass, g is the gravitational constant, $\phi(s)$ is the slope data in the distance domain, f_{r} is the coefficient of rolling resistance, C_{d} is the coefficient of drag, ρ is the air density, A is the frontal projected area of the vehicle.

TABLE I
PARAMETERS OF VEHICLE MODEL

Symbol	Description	Value
m	Vehicle mass	1870 kg
C_d	Coefficient of drag	0.373
A	Frontal projected area of vehicle	2.58 m^2
ρ	Air density	1.205 kg/m^3
f_r	Coefficient of rolling resistance	0.011
g	Gravitational constant	9.8 m/s^2
η	Gearbox efficiency	0.94
r	Wheel radius	0.364 m
I_g	Gear ratio of gearbox	1.159
I_{final}	Gear ratio of final reduction drive	4.103
α_1	Coefficient of fuel consumption model	0.003851
α_2	Coefficient of fuel consumption model	1.19E-6
α_3	Coefficient of fuel consumption model	2.95E-10
α_4	Coefficient of fuel consumption model	5.45E-5
α_5	Coefficient of fuel consumption model	-1.59E-9
α_6	Coefficient of fuel consumption model	1.74E-7

Assuming no slip on the wheel, the wheel model is given by the following algebraic relation

$$\begin{aligned} \omega_{\mathrm{w}} &= \frac{v}{r}, \\ F_{\mathrm{t}} &= \frac{1}{\mathrm{r}}T_{\mathrm{final}}, \end{aligned} \tag{3}$$

where T_{final} is the final driveshaft torque, ω_{w} is the rotational velocity of wheel, and r is the wheel radius.

3) Drivetrain and Braking System: When the vehicle cruises on a highway, the lock-up clutch of the torque converter is engaged, and the gear remains at the 6th with a constant transmission efficiency. Therefore, the drivetrain model is given by

$$\begin{aligned} T_{\mathrm{final}} &= T_e\eta I_g I_{\mathrm{final}} - T_b, \\ \omega_{\mathrm{w}} &= \frac{\omega_e}{I_g I_{\mathrm{final}}}, \end{aligned} \tag{4}$$

where T_{final}, T_e, T_b are the torque of the final driveshaft, engine, and brake, respectively, ω_e is the rotational velocity of the engine, η is the gearbox efficiency, I_g, I_{final} are the gear ratios of gearbox and final driveshaft, respectively.

4) Engine: A quasi-static engine model is considered to predict fuel consumption with acceptable accuracy, as provided by the following polynomial equation [31]

$$\begin{aligned} \dot{m}_f(T_e) = {} & \alpha_1\omega_e + \alpha_2\omega_e^2 + \alpha_3\omega_e^3 + \alpha_4\omega_e T_e \\ & + \alpha_5\omega_e^2 T_e + \alpha_6\omega_e T_e^2, \end{aligned} \tag{5}$$

where $\dot{m}_{\mathrm{f}}$ is the instantaneous fuel consumption rate, $\alpha_{1\sim6}$ are the constant coefficients identified through engine dynamometer test [8].

III. MULTI-OBJECTIVE OPTIMIZATION IN PCC

This section presents the formulation of the PCC optimization problem as a multi-objective optimization problem in the distance domain, and introduces the receding horizon framework for solving the problem in each sampling step, which leverages the preview information to improve the system's performance.

A. Spatial Coordinates Transformation

The vehicle dynamic equations presented in (1) are based on the time coordinate, but to incorporate the location-specific slope and speed limit information from the digital map more naturally over the prediction horizon, we adopt a spatial-based model and sample it in spatial coordinates. Thus, denoting time as t, we have:

$$\frac{dv}{dt} = \frac{dv}{ds}\frac{ds}{dt} = \frac{dv}{ds}v, \tag{6}$$

based on which the model in the spatial domain can be given as

$$\begin{aligned}\frac{dv}{ds} &= \frac{1}{mv}\left[F_t - F_r(s,v)\right] \\ &= \frac{\eta I_g I_{\text{final}} T_e - T_b}{rmv} - \frac{C_d \rho A v}{2m} - \frac{g}{v}(f_r \cos(\phi(s)) + \sin(\phi(s))).\end{aligned} \tag{7}$$

As shown in Fig. 1, the predictive cruise control system, as a subsystem of the ADAS, is activated only when the vehicle speed is at or above 30 km/h. This threshold also helps prevent numerical instability and model prediction errors caused by singularities in the spatial-based formulation. Compared to the problems formulated in [32], which treat both the speed and distance as system states, the modified model in (7) only uses speed as a state, resulting in reduced dimensionality that reduces computational complexity and facilitates real-time implementation.

Applying the Euler forward method and discretizing the vehicle dynamic model in the distance domain, we obtain:

$$\begin{aligned}v(k+1) &= g\left(v(k), T_e(k), T_b(k)\right) \\ &= v(k) + \frac{1}{mv(k)}\left[F_t - F_r(kds, v(k))\right]\Delta s,\end{aligned} \tag{8}$$

where Δs is the sampling distance in each step.

Considering the objective of the PCC system is to minimize energy consumption while finishing the whole trip as fast as possible, trip time and fuel consumption are considered as the objectives for optimization, and their incremental values in one sampling period are represented as

$$\begin{aligned}\Phi_1(k) &= \frac{\dot{m}_f\left(T_e(k), v(k)\right)}{v(k)} \\ \Phi_2(k) &= \frac{1}{v(k)},\end{aligned} \tag{9}$$

where $\Phi_1(k)$ and $\Phi_2(k)$ are the fuel consumption and trip time in one sampling period.

B. Multi-Objective Optimization Problem Within MPC Framework

MPC is employed in this study to address the predictive cruise control problem for the following reasons: 1) Slope and speed limit data from digital maps can be seamlessly incorporated into the MPC optimization as preview information. 2) MPC effectively manages constraints on engine torque, brake torque, and velocity, which are critical for precise vehicle speed control and fuel efficiency. 3) MPC can be implemented online, enabling real-time, adaptive feedback control that responds to dynamic driving conditions and optimizes fuel consumption.

TABLE II
PARAMETERS OF WSM-BASED MOOP

Parameter	Description	Value
$T_{e,\min}$	Minimum of engine torque	0 Nm
$T_{e,\max}$	Maximum of engine torque	120 Nm
$T_{b,\min}$	Minimum of brake torque	0 Nm
$T_{b,\max}$	Maximum of brake torque	6000 Nm
ε	Tolerance	1.0e-6
v_0	Initial velocity	20 m/s

Let $x(i|k)$ denotes the value of variable x at distance $k+i$ while the prediction is made at the current location k. An MPC with an economical cost function defined over a finite-distance horizon N is formulated as

$$\min_{T_e(\cdot|k), T_b(\cdot|k)} \left\{J_1(k), J_2(k)\right\}, \tag{10}$$

and

$$\begin{aligned}J_1(k) &= \sum_{i=0}^{N} \Phi_1(i|k)\Delta s \\ &= \sum_{i=0}^{N} \frac{\dot{m}_f\left(T_e(i|k), v(i|k)\right)}{v(i|k)}\Delta s, \quad i = 0:N, \\ J_2(k) &= \sum_{i=0}^{N} \Phi_2(i|k)\Delta s \\ &= \sum_{i=0}^{N} \frac{1}{v(i|k)}\Delta s, \quad i = 0:N,\end{aligned} \tag{11}$$

where $J_1(k)$, $J_2(k)$ are the accumulated fuel consumption and trip time over the optimization horizon in the formulated Multi-objective Optimization Problem (MOOP).

The constraints of the MPC for PCC are given as

$$\begin{aligned}&v(i+1|k) = g\left(v(i|k), T_e(i|k), T_b(i|k)\right), \quad i = 0:N \\ &v_{\min}(k) \le v(i|k) \le v_{\max}(k), \quad i = 0:N \\ &T_{e,\min} \le T_e(i|k) \le T_{e,\max}, \quad i = 0:N-1 \\ &T_{b,\min} \le T_b(i|k) \le T_{b,\max}, \quad i = 0:N-1 \\ &(T_e(i \mid k) - \varepsilon)\cdot(T_b(i \mid k) - \varepsilon) < 0, i = 0:N-1 \\ &v(0|k) = v_0,\end{aligned} \tag{12}$$

where $v_{\min}(k)$ and $v_{\max}(k)$ are the row vector representing the lower and upper limits of the speed with dimension $N \times 1$ as

$$\begin{aligned}v_{\min}(k) &:= [v_{\min}(1 \mid k), \ldots, v_{\min}(N \mid k)]^T \\ v_{\max}(k) &:= [v_{\max}(1 \mid k), \ldots, v_{\max}(N \mid k)]^T.\end{aligned} \tag{13}$$

$T_{e,\min}$ and $T_{e,\max}$ are constraints on the engine torque, while $T_{b,\min}$ and $T_{b,\max}$ are constraints on the brake torque. To avoid simultaneous application of both engine torque and brake torque, a tolerance constant ε is introduced, and the constraint $(T_e(i \mid k) - \varepsilon)\cdot(T_b(i \mid k) - \varepsilon) < 0$ is added, assuming T_e and T_b are both positive. The velocity at sampling distance k is denoted by $v(0|k)$. The parameters used to define the constraints on the states and inputs are listed in Table II.

At each update instant for the control variables, the first elements of the vectors T_e and T_b are applied to the vehicle, where T_e and T_b are the solutions of the MOOP (10)-(12).

IV. Utopia Tracking Method Based PCC System

In this section, we evaluate the performance of the PCC system using the WSM-MPC approach through simulation. To overcome the drawbacks of the WSM in predictive cruise control systems in various driving conditions, we propose the utopia point tracking method based on the geometric properties of the knee point on the Pareto front. Moreover, recognizing that the MPC controller in the PCC framework treats the current vehicle state as a perturbation of the previous step under input-state constraints, we employ a neighboring extremal method to compute the optimal control sequence across the horizon, facilitating the real-time implementation of the PCC system.

A. Weighted Sum Method

The Weighted Sum Method (WSM) is a widely used approach for solving multi-objective optimization problems by combining multiple objectives into a single scalar objective [17], [33]. In the context of PCC, the problem can be formulated as follows:

$$\min_{T_e(\cdot|k)T_b(\cdot|k)} \sum_{i=0}^{N} \left(\beta\Phi_1\left(v(i|k), T_e(i|k)\right) + (1-\beta)\Phi_2\left(v(i|k)\right)\right)\Delta s, \tag{14}$$

where β is the weight that needs to be tuned. However, selecting the appropriate weight is often done by trial and error, and the computationally expensive tuning process makes it difficult to update the weight online. For the PCC system, the weight β in (14) is tuned offline and fixed for online implementation to achieve a balance between the two objectives. In this case study, we select the weight parameter β as 0.04.

Two slope profiles, representing highway sections in Chongqing and Wuhan, China, and provided by the map supplier KOTEI, are used to evaluate PCC system performance (Fig. 2). To prevent vehicle shuffle dynamics caused by powertrain model sensitivity, a simple filter is applied to remove high-frequency quantization noise from the digital map [34] (Fig. 3). The resulting velocity is then adjusted to account for slope variations, while engine torque, brake torque, and velocity remain within their respective constraints, as illustrated by the yellow line in Fig. 3.

By varying β from 0 to 1, as shown in Fig. 4, the performance of the PCC system is evaluated using WSM-MPC. To assess the impact of incorporating look-ahead slope information, the same MPC control is applied with the assumption that the road grade is constant. The results indicate that multiple solutions dominate the PCC system's performance without look-ahead slope information, whereas the one using preview road grade information yields superior performance. This finding suggests that incorporating slope information significantly improves the PCC system's performance compared to not using any prediction information.

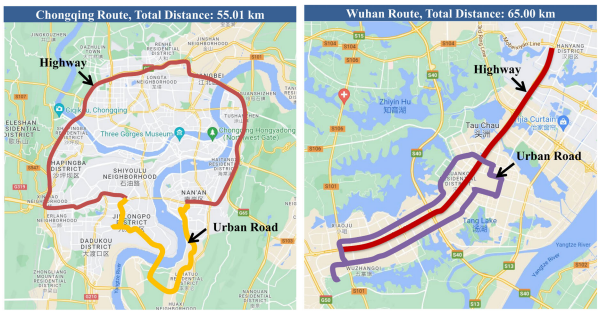


Fig. 2. Route in Chongqing and Wuhan, China.

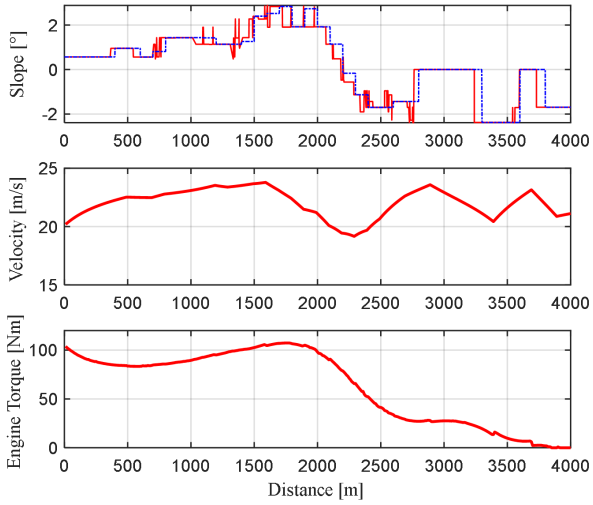


Fig. 3. The simulation results of PCC based on the WSM. (a) Filtered and original slope data. (b) Vehicle speed. (c) Engine torque.

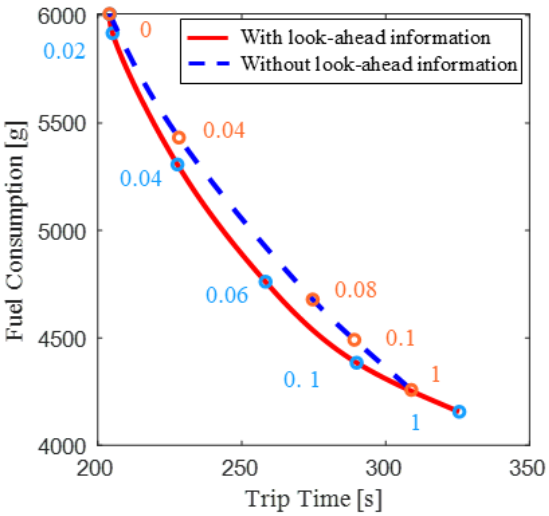


Fig. 4. The trade-off between fuel consumption and trip time with and without look-ahead slope information.

B. Utopia Tracking Method

In multi-objective optimization, no single solution can simultaneously optimize all objectives due to inherent conflicts between them. By varying the weighting factor β, different levels of importance can be assigned to each objective, reveal-

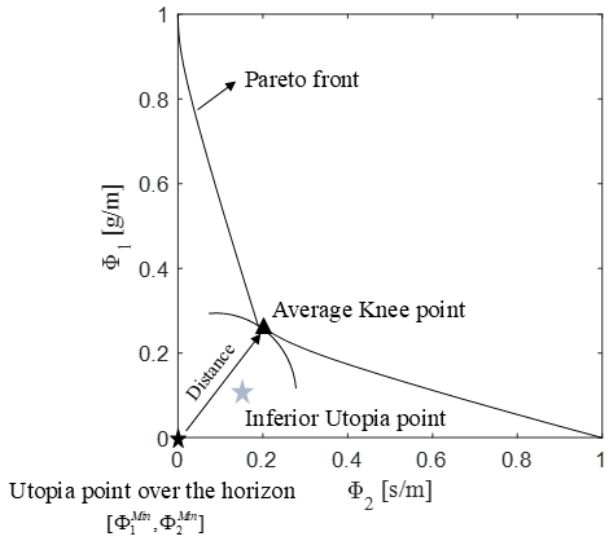


Fig. 5. Pareto front featuring the conflicting criteria and the Knee point after averaging. The costs Φ_1 and Φ_2 over the horizon are all normalized. This positioning signifies that the average Utopia point calculated over the horizon represents both lower fuel consumption and shorter trip time to finish the horizon, as compared to the inferior Utopia point defined in [21].

ing the trade-offs captured by the Pareto front. This front illustrates the balance between trip time and fuel consumption, where no solution can be improved in one objective without compromising the other [35].

The knee point in a multi-objective optimization problem is defined as the point on the Pareto front where the slope changes abruptly, indicating a significant trade-off between the conflicting objectives. If a knee point exists, it is often a preferred trade-off solution. Various definitions of knee points have been proposed in the iterature [20], such as the point having the maximum reflex angle or bend angle compared to its neighbors, but these approaches are mainly used in offline design and analysis and are not suitable for real-time implementation of the PCC system. To that end, we can leverage the Utopia point to exploit the benefits of the Pareto front in the PCC system. In the context of a multi-objective optimization problem, the Utopia point represents the ideal point in the objective space where all objectives are minimized simultaneously. It is the point where the best possible trade-off between the objectives is achieved. As depicted in Fig. 5, the Utopia point is often represented as the intersection of the axes where all objectives reach their minimum values. In this study, we define the knee point as the Pareto point that has the closest distance to the Utopia point. For simplicity and computational efficiency, the Euclidean distance in the normalized space is used.

Remark 1: The Euclidean distance is so-called the L2-norm whose quadratic form is widely used as a cost function since it is differentiable, compared with L1-norm [36].

For the PCC system with economic costs, there is no obvious steady state as shown in Fig. 3. So, the traditional steady-state Utopia point in [21] can not be applied in this research consequently. Based on the fact that the MPC is performed through a repeated process of prediction and optimization. An alternative Utopia Point is proposed by averaging the objectives over the horizon. As depicted in Fig. 5, the utopia points defined by the average performance of ultimate transient processes outperform those defined by steady-state ones, indicating the superiority of using the average Utopia point. Therefore, throughout the rest of the text, the average Utopia point over the prediction horizon is calculated and employed for the PCC system. This point is defined as follows:

$$\Phi_i^{\text{Min}}(k) = \frac{J_i^{\text{Min}}(k)}{N \cdot \Delta s}, i = 1, 2, \tag{15}$$

where $\left(\Phi_1^{\text{Min}}(k), \Phi_2^{\text{Min}}(k)\right)$ is the average Utopia point. N are the predictive steps, Δs is the sampling distance. Consequently, the averaging of the upper bounds which are used in the normalization is defined as follows:

$$\Phi_i^{\text{Max}}(k) = \frac{J_i^{\text{Max}}(k)}{N \cdot \Delta s}, i = 1, 2. \tag{16}$$

Inspired by the existence of the Knee point on the Pareto front, which represents a balance between fuel consumption and trip time, the UTM is proposed by modifying the cost function as the accumulation of the normalized distance from the current objectives to the average Utopia point, which can be given as

$$\begin{aligned} &\min_{T_e(\cdot|k), T_b(\cdot|k)} \sum_{i=0}^{N} (\Phi_1^{\text{Norm}}(i|k)^2 + \Phi_2^{\text{Norm}}(i|k)^2)^{\frac{1}{2}} \Delta s, \\ &\text{s.t.} \quad (12), \end{aligned} \tag{17}$$

where $\Phi_1^{\text{Norm}}(i|k)$, $\Phi_2^{\text{Norm}}(i|k)$ are the normalized fuel consumption and normalized trip time at the distance instant $k+i$, defined as:

$$\begin{aligned} \Phi_1^{\text{Norm}}(i|k) &= \frac{\Phi_1(i|k) - \Phi_1^{\text{Min}}(k)}{\Phi_1^{\text{Max}}(k) - \Phi_1^{\text{Min}}(k)} \\ \Phi_2^{\text{Norm}}(i|k) &= \frac{\Phi_2(i|k) - \Phi_2^{\text{Min}}(k)}{\Phi_2^{\text{Max}}(k) - \Phi_2^{\text{Min}}(k)}, \end{aligned} \tag{18}$$

where normalization is used to scale the objectives without affecting their properties. The pseudocode of UTM is detailed in Algorithm 1. At the k-th sampling instant, the upper bounds and lower bounds used for normalization are calculated using (14) with constraint (12), setting the weight β as 0 and 1. The UTM-based cost function is then adjusted by utilizing the pre-calculated upper and lower bounds. Subsequently, the optimal control sequences T_e and T_b are derived using (17) with constraint (12).

Remark 2: The optimal control problem may become infeasible due to the constraints (12), and the optimization will stop without a solution in such a situation. In order to handle the infeasibility, the least important constraints $(T_e(i \mid k) - \varepsilon) \cdot (T_b(i \mid k) - \varepsilon) < 0$ *will be removed and the optimization will be repeated. If the infeasibility still exists, the last feasible solution will be applied in the sequel. This ad-hoc solution was able to address the feasibility issue of this application.*

C. Weights Adaption Mechanism of the UTM-MPC

A detailed analysis is conducted to investigate the weight adaptive mechanism of the UTM-MPC under varying slope

Algorithm 1 UTM-MPC

Input: Current vehicle velocity at the sampling step $v_{0|k}$, slope information over the horizon $\phi_{k,\dots,k+N}$
Output: Optimal engine torque and brake torque sequence T_e^*, T_b^* at the current sampling step

Compute the minimum trip time $J_2^{\mathrm{Min}}(k)$ and maximum fuel consumption $J_1^{\mathrm{Max}}(k)$ over the horizon using equations (12) and (14) with Algorithm 2, where trip time is the sole objective in the cost function;
Calculate the average trip time $\Phi_2^{\mathrm{Min}}(k)$ and average fuel consumption $\Phi_1^{\mathrm{Max}}(k)$ using equations (15) and (16);
Compute the minimum fuel consumption $J_1^{\mathrm{Min}}(k)$ and maximum trip time $J_2^{\mathrm{Max}}(k)$ over the horizon using equations (12) and (14) with Algorithm 2, where fuel consumption is the sole objective in the cost function;
Calculate the average fuel consumption $\Phi_1^{\mathrm{Min}}(k)$ and average trip time $\Phi_2^{\mathrm{Max}}(k)$ using equations (15) and (16);
Obtain the averaging Utopia point $\left[\Phi_1^{\mathrm{Min}}(k), \Phi_2^{\mathrm{Min}}(k)\right]$;
Update T_e^*, T_b^* using equations (17) and (12) with Algorithm 2;
$k \leftarrow k+1$;

profiles. The first step is to eliminate the square root of the sum of squares of objectives in (17) to facilitate the extraction of the variable part from the cost function and analyze the impact of normalization using the pre-calculated upper and lower bounds. The rationality of this transformation is proved in [36], which states that both are distance functions with similar performance in the Utopia tracking formulation. Consequently, the cost function in (17) is amended as follows:

$$\min_{T_e(\cdot|k),T_b(\cdot|k)} \sum_{i=0}^{N} \left(\left|\frac{\Phi_1(i|k)-\Phi_1^{\mathrm{Min}}(k)}{\Phi_1^{\mathrm{Max}}(k)-\Phi_1^{\mathrm{Min}}(k)}\right| + \left|\frac{\Phi_2(i|k)-\Phi_2^{\mathrm{Min}}(k)}{\Phi_2^{\mathrm{Max}}(k)-\Phi_2^{\mathrm{Min}}(k)}\right|\right)\Delta s. \tag{19}$$

Rearranging the cost function so the objectives with coefficient will be isolated, given in the form as below:

$$\min_{T_e(\cdot|k),T_b(\cdot|k)} \sum_{i=0}^{N} \left(A\Phi_1(i|k) + B\Phi_2(i|k) - C\right)\Delta s, \tag{20}$$

where $A = \left(\Phi_1^{\mathrm{Max}}(k)-\Phi_1^{\mathrm{Min}}(k)\right)^{-1}$, $B = \left(\Phi_2^{\mathrm{Max}}(k)-\Phi_2^{\mathrm{Min}}(k)\right)^{-1}$, and $C = A\Phi_1^{\mathrm{Min}}(k) + B\Phi_2^{\mathrm{Min}}(k)$. Considering the equivalent weight $\beta_e = A/(A+B)$ by normalization, the UTM can be transformed to WSM by

$$\min_{T_e(\cdot|k)T_b(\cdot|k)} \sum_{i=0}^{N} \left(\beta_e\Phi_1(i|k) + (1-\beta_e)\Phi_2(i|k)\right)\Delta s. \tag{21}$$

Therefore, the UTM can be regarded as WSM with adaptive weights.

D. Tailored Algorithm for Solving the UTM-MPC

For the real-time implementation of the PCC system, computing the Utopia point online may increase computational cost. However, under highway cruising conditions, the vehicle's velocity and the road topology are expected to remain relatively stable between sampling steps. By leveraging the gradual variation in slope and vehicle speed, we apply an integrated perturbation analysis and sequential quadratic programming (IPA-SQP) algorithm to solve the UTM-MPC problem. With the warm-starting feature of the IPA-SQP method, an optimal solution can be efficiently refined within a few iterations. For the UTM-MPC problem formulated in Section IV-B, it can be reformulated as a common form

$$\min_{u(\cdot)} \quad J = \Phi(x(N \mid k)) + \sum_{i=0}^{N-1} \phi(x(i \mid k), u(i \mid k)) \tag{22}$$

$$\begin{aligned} \text{subject to } & x(i+1 \mid k) = f(x(i \mid k), u(i \mid k)) \\ & C(x(i \mid k), u(i \mid k)) \le 0 \\ & x(0 \mid k) = x(k), \quad i = 0, 1, \dots, N-1 \end{aligned} \tag{23}$$

where the state variables of the UTM-MPC problem are denoted as $x = \left[v_1 \cdots v_{N_\star}\right]^T \in \mathbb{R}^{N_h}$, while the control variables are $u = \left[T_{e,0}\ T_{b,0} \cdots T_{e,N_h-1}\ T_{b,N_h-1}\right] \in \mathbb{R}^{N_h \times 2}$. The vehicle dynamic equation is represented by $f(\cdot)$, and the state-input constraints are defined by the function $C(\cdot)$.

Algorithm 2 IPA-SQP Algorithm for UTM-MPC

Input: Current vehicle velocity $x(k)$, previous step velocity $x(k-1)$, and previous optimal sequences $x^*(i|k-1)$, $u^*(i|k-1)$ for $i = 1, \dots, N$
Output: Optimal engine and brake torque sequence $u^*(i|k)$ for $i = 1, \dots, N$

Set $j = 0$;
Initialize perturbation: $\delta x^j(k) = x(k) - x(k-1)$;
Set nominal solution: $x^j(i|k) = x^*(i|k-1)$, $u^j(i|k) = u^*(i|k-1)$ for $i = 1, \dots, N$;
repeat
  Calculate $\delta x^j(i|k)$ and $\delta u^j(i|k)$ using PA for the perturbation $\delta x^j(k)$ and nominal solution $x^j(i|k)$, $u^j(i|k)$ via equation (29);
  Find smallest $\alpha_j \in [0, 1]$ that causes constraint activation at least at one step;
  Update:
    - $u^{j+1}(i|k) = u^j(i|k) + \alpha_j \delta u^j(i|k)$
    - $x^{j+1}(i|k) = x^j(i|k) + \alpha_j \delta x^j(i|k)$
    - $\delta x^{j+1}(k) = (1-\alpha_j)\delta x^j(k)$
    - $x^{j+1}(k) = x^j(k) + \alpha_j \delta x^j(k)$
    - $j = j+1$
until $0 \le \alpha_j < 1$;
repeat
  Set $\delta x^j(k) = 0$;
  Compute $\delta x^j(i|k)$, $\delta u^j(i|k)$ via SQP using equation (31);
  Update:
    - $u^{j+1}(j|k) = u^j(j|k) + \delta u^j(j|k)$
    - $x^{j+1}(j|k) = x^j(j|k) + \delta x^j(j|k)$
    - $j = j+1$ for $i = 1, \dots, N$
until $\|H_u\| \approx 0$;

$\Phi(x(N \mid k))$ is the terminal penalty function whose weight parameter is the same as that of the stage cost in this research.

Let the optimal solution at the last instance of the UTM-MPC, $x(\cdot \mid k-1)$ and $u(\cdot \mid k-1)$ serve as candidates for the optimal solution at the current instance k. The current vehicle velocity is treated as a perturbation of the vehicle velocity from the previous sampling step, denoted by $\delta x(0 \mid k) = x(0 \mid k) - x(0 \mid k-1)$. We then find perturbations $\delta x(\cdot \mid k)$ and $\delta u(\cdot \mid k)$ near the candidates for the optimal solution at the current instance, ensuring they satisfy the first-order necessary conditions for optimality when the initial state is perturbed. For the augmented cost. For the augmented cost

$$\bar{J} = \Phi(x(N \mid k)) + \sum_{i=0}^{N-1} (H(i \mid k) - \lambda(i+1 \mid k) x(i+1 \mid k)), \tag{24}$$

it can be expanded to the second order around a candidate solution

$$\delta^2 \bar{J} = \frac{1}{2} \delta x(N)^T \left(\Phi_{xx}(N) + \frac{\partial}{\partial x} \left(C_x^T(x(N)) \mu(N) \right) \right) \delta x(N) + \frac{1}{2} \sum_{k=0}^{N-1} \begin{bmatrix} \delta x(k) \\ \delta u(k) \end{bmatrix}^T \begin{bmatrix} H_{xx}(k) & H_{xu}(k) \\ H_{ux}(k) & H_{uu}(k) \end{bmatrix} \begin{bmatrix} \delta x(k) \\ \delta u(k) \end{bmatrix}. \tag{25}$$

$H(i \mid k)$ is a Hamiltonian function for the UTM-MPC problem, defined as:

$$H(i \mid k) = \phi(x(i \mid k), u(i \mid k)) + \lambda^{\mathrm{T}}(i+1 \mid k) f(x(i \mid k), u(i \mid k)) + \mu^{\mathrm{T}}(i \mid k) C^a(x(i \mid k), u(i \mid k)). \tag{26}$$

and the second-order partial derivatives of the Hamiltonian function are represented by $H_{xx}(k)$, $H_{xu}(k)$, $H_{ux}(k)$, and $H_{uu}(k)$, while $\Phi_{xx}(N)$ denotes the second-order partial derivative of the terminal penalty. $\lambda^{\mathrm{T}}(i+1 \mid k)$ is the sequence of co-states associated with vehicle dynamics $f(\cdot)$ (i.e., the equality constraints). $\mu^{\mathrm{T}}(i \mid k)$ is the vectors of Lagrange multiplier. $C^a(x(i \mid k), u(i \mid k))$ is the vector composed of the active inequality constraints.

Using this second variation of the augmented functional, the so-called accessory minimum problem as shown in [37] and [38], can be formulated as to minimize the cost

$$\min_{\delta u(\cdot), \delta x(\cdot)} J_{\text{new}} = \min_{\delta u(\cdot), \delta x(\cdot)} \delta^2 J, \tag{27}$$

while subject to constraints

$$\begin{aligned} &\delta x(i+1 \mid k) = f_x(i \mid k) \delta x(i \mid k) + f_u(i \mid k) \delta u(i \mid k) \\ &\delta x(0 \mid k) = \delta x(k) \\ &C_x^a(i \mid k) \delta x(i \mid k) + C_u^a(i \mid k) \delta u(i \mid k) = 0. \end{aligned} \tag{28}$$

$C_x^a(i \mid k)$ and $C_u^a(i \mid k)$ represent the partial derivatives of the active constraint vector, and $f_x(i \mid k)$ and $f_u(i \mid k)$ represent the partial derivatives of the vehicle dynamic equation. To obtain the explicit relation between the state and input variations of the perturbed solution, we apply the Karush-Kuhn-Tucker (KKT) conditions to the problem (25) and (28), yielding the following equation:

$$\begin{aligned} &\delta u(i \mid k) = -\begin{bmatrix} I & 0 \end{bmatrix} K_0(i \mid k) K_1(i \mid k) \\ &K_0(i \mid k) = \begin{bmatrix} Z_{uu}(i \mid k) & C_u^{a\mathrm{T}}(i \mid k) \\ C_u^a(i \mid k) & 0 \end{bmatrix} \\ &K_1(i \mid k) = \begin{bmatrix} Z_{ux}(i \mid k) \delta x(i \mid k) \\ C_x^a(i \mid k) \delta x(i \mid k) \end{bmatrix} \\ &\delta x(i+1 \mid k) = f_x(i \mid k) \delta x(i \mid k) + f_u(i \mid k) \delta u(i \mid k) \end{aligned} \tag{29}$$

where

$$\begin{aligned} Z_{uu}(i \mid k) &= H_{uu}(i \mid k) + f_u^{\mathrm{T}}(i \mid k) \Phi_{xx}(i+1 \mid k) f_u(i \mid k) \\ Z_{ux}(i \mid k) &= H_{ux}(i \mid k) + f_u^{\mathrm{T}}(i \mid k) \Phi_{xx}(i+1 \mid k) f_x(i \mid k) \end{aligned} \tag{30}$$

To handle significant perturbations in velocity that result in modifications to the set of active constraints, a method is proposed that involves traversing the line connecting the nominal initial condition $x(0)$ to the point $x(0) + \delta x(0)$ until a constraint's status changes. Subsequently, the optimal correction is computed, and this procedure is repeated for all intermediate points up to the point $x(0) + \delta x(0)$.

It is important to note that the NE method provides only a first-order approximation of the optimal solution, and cannot be used as the initial solution for the next sampling step. Therefore, an SQP algorithm based on the active set method is proposed to refine the solution obtained from the neighboring extremal method [39], which yields

$$\begin{aligned} &\delta u(i \mid k) = -\begin{bmatrix} I & 0 \end{bmatrix} K_0(i \mid k) K_1(i \mid k) \\ &K_0(i \mid k) = \begin{bmatrix} Z_{uu}(i \mid k) & C_u^{a\mathrm{T}}(i \mid k) \\ C_u^a(i \mid k) & 0 \end{bmatrix} \\ &K_1(i \mid k) = \begin{bmatrix} Z_{ux}(i \mid k) \delta x(i \mid k) + f_u^{\mathrm{T}}(i \mid k) T(i+1 \mid k) + H_u(i \mid k) \\ C_x^a(i \mid k) \delta x(i \mid k) \end{bmatrix} \\ &\delta x(i+1 \mid k) = f_x(i \mid k) \delta x(i \mid k) + f_u(i \mid k) \delta u(i \mid k) \end{aligned} \tag{31}$$

where $T(i+N \mid k) = 0$ and $T(i \mid k) = f_x^{\mathrm{T}}(i \mid k) T(i+1 \mid k)$ is the argument matrix used for backward iteration.

SQP refines the solution by solving a series of optimization subproblems. The Hessian and Jacobian matrices of the Lagrangian are symbolically generated and assembled using MATLAB's symbolic toolbox. The proposed algorithm is outlined in pseudo-code in Algorithm 2. This active-set-based SQP method further improves the solution obtained from the neighboring extremal approach.

These recursive updates allow us to calculate a first-order approximation to the optimal solution in the form of $x(\cdot \mid k-1) + \delta x(\cdot \mid k)$ and $u(\cdot \mid k-1) + \delta u(\cdot \mid k)$, without solely relying on traditional second-order optimization methods, such as SQP, to solve a nonlinear programming problem. With the neighboring extremal solution in (32), only a few iterations are required to find the optimal solution near the nominal optimal solution, taking into account the sequential nature of MPC. In contrast, the SQP method requires hundreds of iterations to meet the terminal condition, with each iteration involving numerous quadratic programming iterations [23], making the process time-consuming overall.

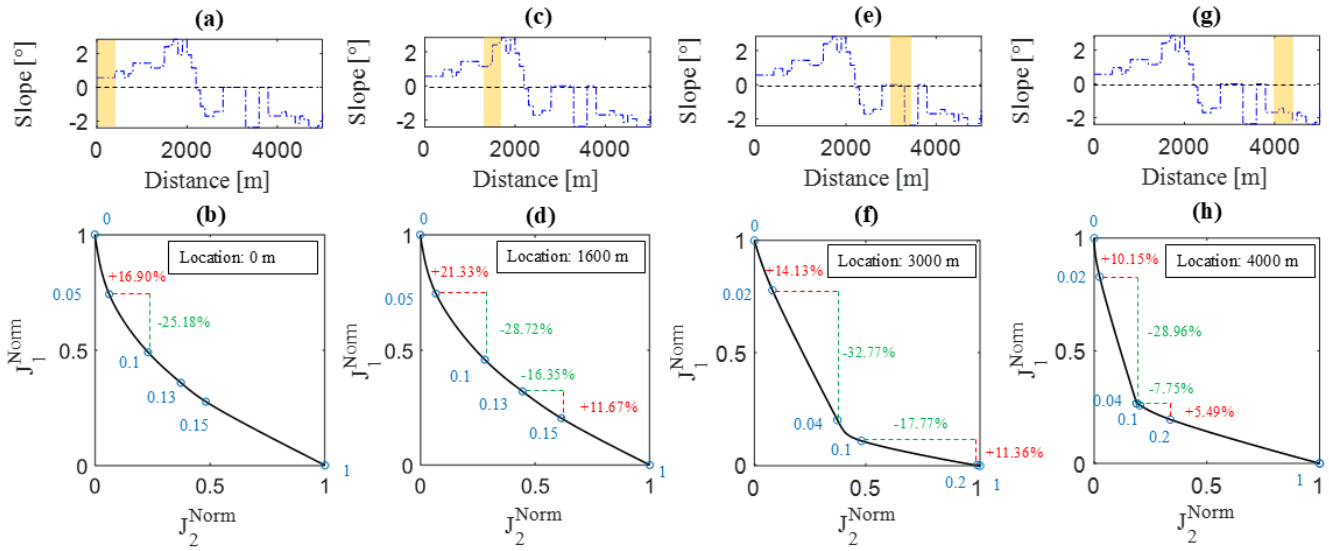


Fig. 6. The Pareto fronts of the MOOPs at different sampling distances. (a) Slope profile at 0 m. (b) The Pareto fronts at the location of 0 m. (c) Slope profile at 1600 m. (d) The Pareto fronts at the location of 1600 m. (e) Slope profile at 3000 m. (f) The Pareto fronts at the location of 3000 m. (g) Slope profile at 4000 m. (h) The Pareto fronts at the location of 4000 m.

V. Simulation Results

In this section, we assess the performance of the UTM-MPC under varying driving conditions. The simulation results demonstrate that the UTM-MPC outperforms the WSM in terms of robustness and balanced performance without requiring a weights-tuning process. We analyze the Knee-point-chasing mechanism to explain the benefits of the UTM-MPC. Moreover, for the speed limit and surrounding vehicle scenario, the results show that the UTM-MPC consumes less fuel while completing the entire trip within the same time frame.

A. Road Grade

To unveil the benefits of the geometric properties of the knee point on the Pareto front under varying road grades, we implement the proposed eco-coasting controller on a realistic driving cycle in Wuhan as shown in Fig. 2. The Pareto front of the multi-objective optimal control problem of the PCC system, which needs to be solved at each control interval, is depicted in Fig. 6. The following remarks can be drawn through the Pareto front analysis.

Remark 1 (Diverse shapes of Pareto fronts under different driving conditions): As illustrated in Fig. 6 (g) and (h), the Pareto fronts at the locations of 3000 m and 4000 m are convex with an evident knee point. These knee points lead to a balanced performance in terms of two competing objectives. On the other hand, the Pareto fronts at the locations of 0 m and 1600 m are non-steep, as shown in Fig. 6 (e) and (f), indicating that the shape of the Pareto front varies with the slope over the prediction horizon.

Remark 2 (Benefits of the knee point on the Pareto front): The simulation results in Fig. 6 show that the knee point represents the point on the Pareto front where the rate of improvement in one objective is small relative to the increase in another objective. In other words, it is the point where trade-offs between objectives become more significant. At the location of 4000 m, there is an unfavorably large sacrifice in fuel consumption as the penalty to gain a small amount of trip time when the Pareto point moves away from the knee point on the Pareto front. Specifically, the decrease in fuel consumption is 28.96% while the increase is 10.15% in trip time as shown in Fig. 6.

Remark 3 (The Euclidean distance from the objectives to the Utopia point is an effective criterion to formulate the Knee point): For the real-time implementation of the PCC system, online construction of the Pareto front and knee point is computationally prohibitive. As shown in Fig. 6, the Pareto point which has a shorter distance to the Utopia point entails a better comprehensive performance as shown. It is reasonable to formulate the cost function of the PCC system as the Utopia tracking form.

The effectiveness of the Utopia tracking mechanism of the UTM method for MOOP in each control interval is validated using the look-ahead slope profile at the location of 4000 m in Fig. 6. As shown in Fig. 7(a), the objectives in each sampling step of the proposed algorithm over the prediction horizon scatter around the average Knee point instead of being located at the knee point due to the dynamic characteristics present in the vehicle powertrain. The controller exploits these dynamics to leave the average Pareto front and approach the average Utopia point. This is the reason why the alternative utopia point in this research, obtained by averaging the objectives over the predictive horizon, is superior to the traditional steady-state utopia point in [21]. The accumulated objectives are precisely located at the Knee point of the Pareto front, as shown in Fig. 7(b), indicating a balanced performance, as shown in Fig. 6(g) and (h).

The effectiveness of the UTM-MPC for PCC is demonstrated using new downhill and uphill slope profiles, as illus-

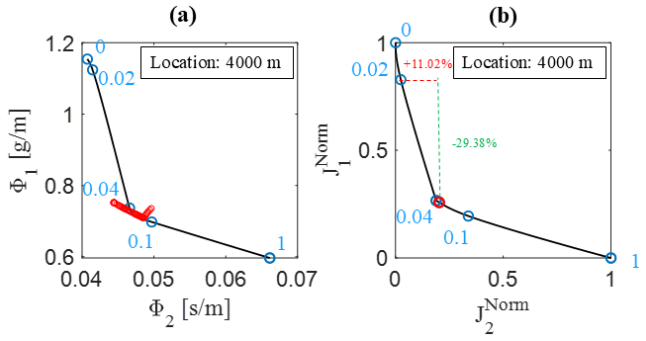


Fig. 7. Performance of UTM-based MOOP at 4000 m. (a) Locations of objectives in each sampling period represented by red circles. (b) Normalized accumulated objectives represented by red circles at the Pareto fronts. For a fair comparison, the same weights as in Fig. 6 (g) and (h) are used in this figure.

trated in Fig. 8. The balanced performance achieved by the UTM-MPC is evident from the results in Fig. 9. In contrast, the WSM sacrifices one objective significantly to achieve a small improvement in another objective, as compared to the objectives of the UTM-MPC. The reason for this improvement is the adaptive weights mechanism of the UTM-MPC, as shown in Fig. 8. The adaptive weights β_e shift from 0.04 to 0.03 under the downhill condition. To investigate the reason for this shift, we plot the Pareto fronts at two locations (0 m and 1000 m) in Fig. 8. We observe that the distribution of the same weights on these Pareto fronts is different. For instance, the Pareto point with the weight $\beta = 0.04$ is located at the knee point of the Pareto front at the location of 0 m. However, it shifts downwards at the location of 1000 m, away from the knee point of the new Pareto front. The shift of the knee point leads to the variation of the weight, aiming to track the knee point on the Pareto front at different locations to achieve comprehensive performance under various driving conditions. The same conclusion can be drawn under the uphill condition, as shown in the right column of Fig. 8.

In addition, the relationship between the shape of the Pareto front and performance is demonstrated in Fig.9. As shown in Fig. 8, the Pareto front at 1000 m has a less steep shape than the one at 0 m, resulting in reduced benefits for all constant weights from 0 m to 1000 m, when compared to the results obtained by the UTM from 0 m to 500 m, as shown in Fig. 9(a). This observation indicates that the existence of a non-steep Pareto front can diminish the performance gained from a steep one.

B. Surrounding Vehicle

Collision avoidance is crucial in two scenarios: (1) dangerous cut-ins requiring immediate braking and (3) the appearance of a preceding vehicle within a certain distance. The PCC system is designed to handle the latter, which is more common in cruise conditions. As shown in Fig. 1, like other ADAS solutions, dangerous cut-ins are not explicitly addressed within the PCC subsystem, with the emergency braking subsystem ensuring safety in such cases. When a preceding vehicle merges with a substantial relative distance, the MPC framework dynamically adjusts the feasible velocity zone over the prediction horizon to optimize the speed profile. The inter-vehicle dynamics in the car-following scenario can be modeled as:

$$\frac{s(k) - s(k-1)}{\Delta t} = v_p(k-1) - v_h(k-1). \tag{32}$$

Here, s denotes the inter-vehicle distance, which is always greater than zero ($s > 0$), and v_p is the speed of the preceding vehicle. This model can be rewritten in terms of the spatial domain as:

$$\frac{s(k) - s(k-1)}{\Delta s} = \left(v_p(k-1) - v_h(k-1)\right)/v(k-1). \tag{33}$$

A spacing policy with a constant reaction time is adopted, and the speed constraints of the host vehicle considering the receding vehicle can be expressed as:

$$\begin{aligned} v_h(k) \le \big[&s_p(k) - s_h(k-1) - \\ &v_h(k-1)\left[s_h(k) - s_h(k-1)\right]/v_h(k)\big] T_{\text{react}}^{-1}. \end{aligned} \tag{34}$$

Here, s_p represents the traveling distance of the preceding vehicle, s_h is the distance of the host vehicle, and T_{react} is the constant reaction time.

Following the approach in [40], a minimum speed constraint is imposed on the host vehicle to maintain a safe distance with the vehicle behind

$$\begin{aligned} v_h(k) \le [&s_h(k) - s_b(k-1) \\ &- v_h(k-1)\left[s_h(k) - s_h(k-1)\right]/v_h(k)\big] T_{\text{react}}^{-1}, \end{aligned} \tag{35}$$

where s_b is the traveling distance of the vehicle behind.

Assuming the preceding vehicle maintains constant acceleration over the prediction horizon, safety constraints can be expressed as speed bounds. Accordingly, surrounding vehicles are incorporated into the PCC formulation by updating the maximum and minimum speed limits in (13) to define a feasible speed range.

C. Speed Limit

In real-world driving, cruising speed limits vary depending on road type and location—such as freeway entry or transitioning to a fast lane on a multi-lane highway. The ego vehicle's feasible speed range is thus constrained by both the road's speed limits and the presence of surrounding vehicles. To evaluate the performance of the PCC system using the UTM-MPC formulation, we reconstruct a traffic scenario based on a digital map of Wuhan, China. In this scenario, the vehicle cruises along a highway with varying speed limits (Fig. 10(b)). We assume surrounding vehicles maintain constant speeds, and that when the ego vehicle changes lanes, the lower speed bound adjusts to prevent collisions with vehicles behind.

Fig. 10(a) illustrates the ego vehicle's feasible speed range as a function of distance, bounded by two solid black lines. Dashed lines represent constant-weight trajectories: $\beta = 0$ (favoring travel time) and $\beta = 1$ (favoring fuel economy). These illustrate that the vehicle tends to follow the upper or lower speed bounds depending on the chosen trade-off. When the feasible speed range narrows—such as when the upper

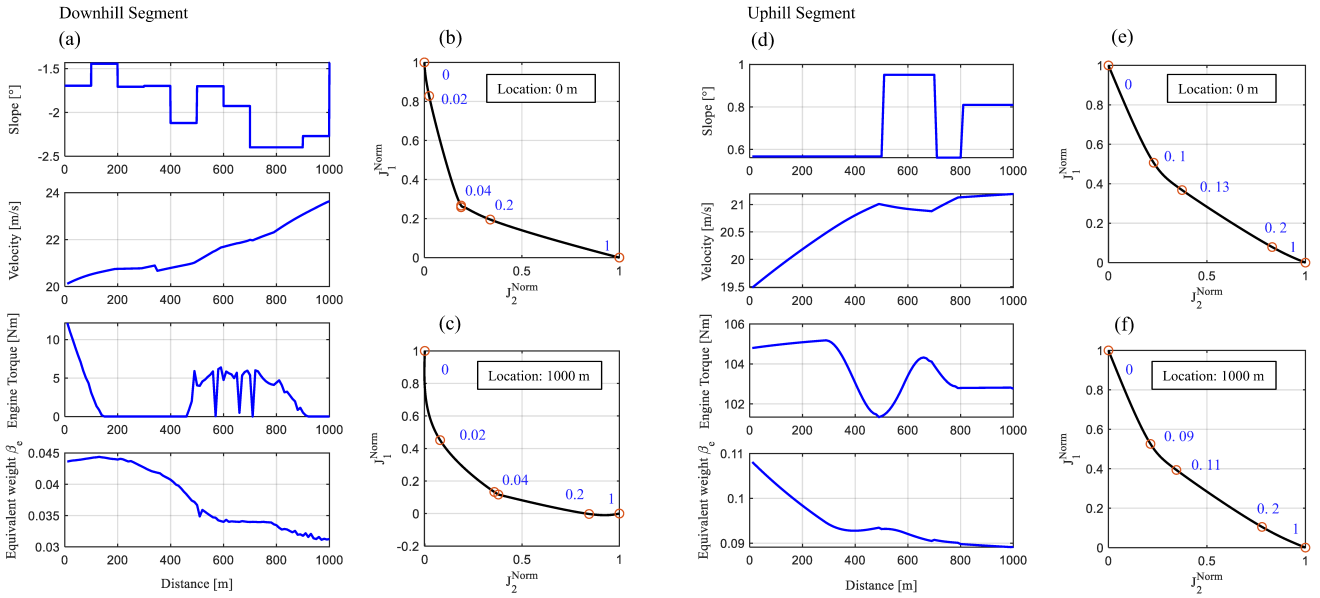


Fig. 8. The validation of UTM-MPC under downhill and uphill. (a) Results over the downhill segment. (b) The Pareto fronts at the location of 0 m of the downhill segment. (c) The Pareto fronts at the location of 1000 m of the downhill segment. (d) Results over the uphill segment. (e) The Pareto fronts at the location of 0 m of the uphill segment. (f) The Pareto fronts at the location of 1000 m of the uphill segment.

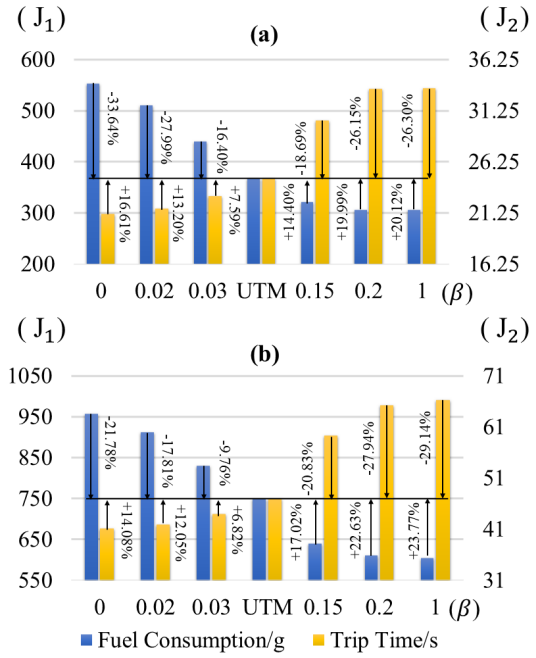


Fig. 9. The comparison between the UTM-MPC and the WSM-MPC. The percentage is calculated by $\left(J_i^{UTM} - J_i^{WSM}\right)/J_i^{WSM} \times 100\%, i = 1, 2$, where the $J_i^{WSM}, i = 1, 2$ and $J_i^{UTM}, i = 1, 2$ are the fuel consumption and trip time of the given trip with WSM-MPC and the UTM-MPC. (a) The accumulated objectives are from 0 m to 500 m. (b) The accumulated objectives from 0 m to 1000 m.

speed limit drops from 90 km/h to 70 km/h—a higher weight on fuel economy is desirable to avoid aggressive driving.

The equivalent weight generated by UTM-MPC is shown in Fig. 10(a). Thanks to its adaptive weighting mechanism, the controller dynamically balances fuel consumption and travel time. Unlike WSM-MPC, which uses a fixed offline-tuned weight (typically assigning higher weights at low speeds and lower weights at high speeds), UTM-MPC produces smoother engine torque profiles, avoiding harsh throttle inputs—evident at the 2000 m mark in WSM-MPC.

As a result, UTM-MPC achieves a 2.9% improvement in fuel economy compared to WSM-MPC, while maintaining the same trip time—without requiring weight tuning. Notably, both UTM-MPC and WSM-MPC with tuned weights outperform the basic WSM approach with a constant weight, as shown in Fig. 10(c).

D. Discussion on the Length of the Prediction Horizon

In real-time implementation, the length of the predictive horizon depends on the performance of the predictive cruise control system and the computational resources available. While a longer predictive horizon could theoretically improve performance in an ideal scenario without disturbances, real-world traffic conditions introduce uncertainties that may limit its benefits [3]. As shown in Fig. 11, a longer predictive horizon can potentially provide better performance in terms of balancing conflicting objectives, but it also increases the complexity and computation time of the optimization problem. It should be noted that the computations are performed on the MATLAB R2019a platform running on an AMD Ryzen 5 2600X 3.60-GHz with 16.0-GB RAM. The simulation results show that increasing the prediction horizon from 50 m to 100 m and 200 m can decrease fuel consumption by 3.97% and 9.20%, respectively. However, the trip time will increase by 0.21% and 1.08%. The PCC controller with a 400 m prediction horizon slightly improves fuel efficiency, but the computation time increases dramatically, making it more susceptible to model mismatch. Thus, the optimal prediction horizon length for real-world applications is 200 m.

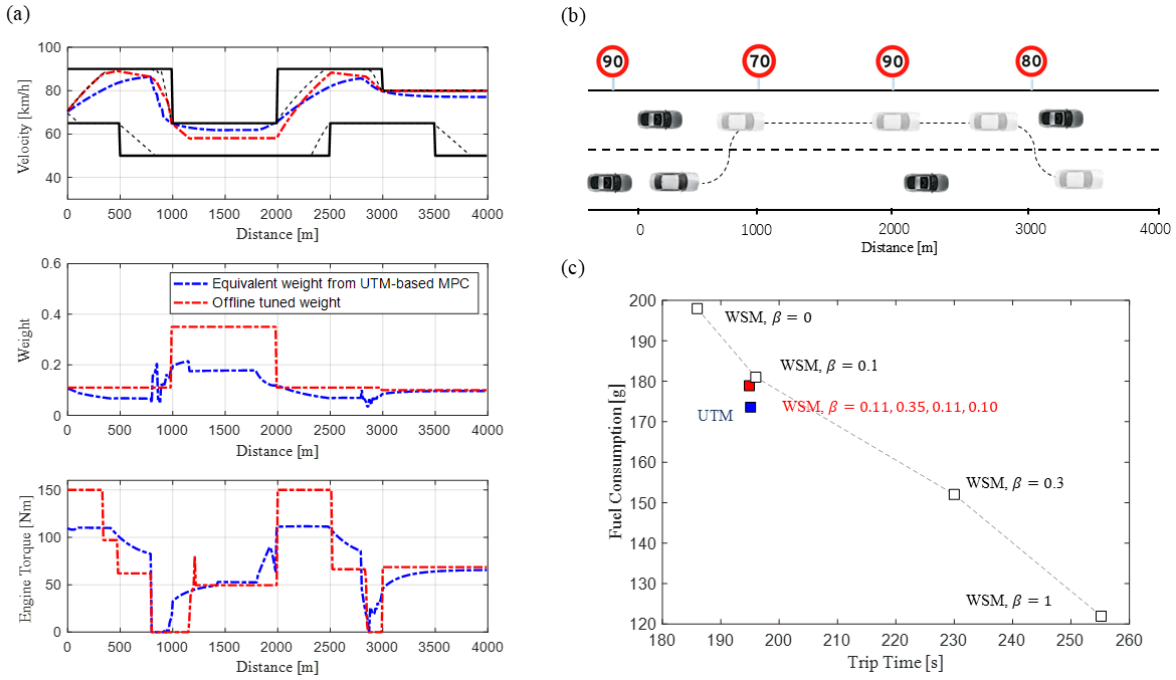


Fig. 10. Comparison of PCC system performance under varying traffic conditions using different MPC formulations. (a) Vehicle speed trajectories: UTM-MPC (blue dashed), WSM-MPC with offline-tuned weights (red dashed), and speed constraints (black solid). (b) Simulation scenario with location-dependent speed limits. (c) Fuel consumption and trip time comparison across controller formulations.

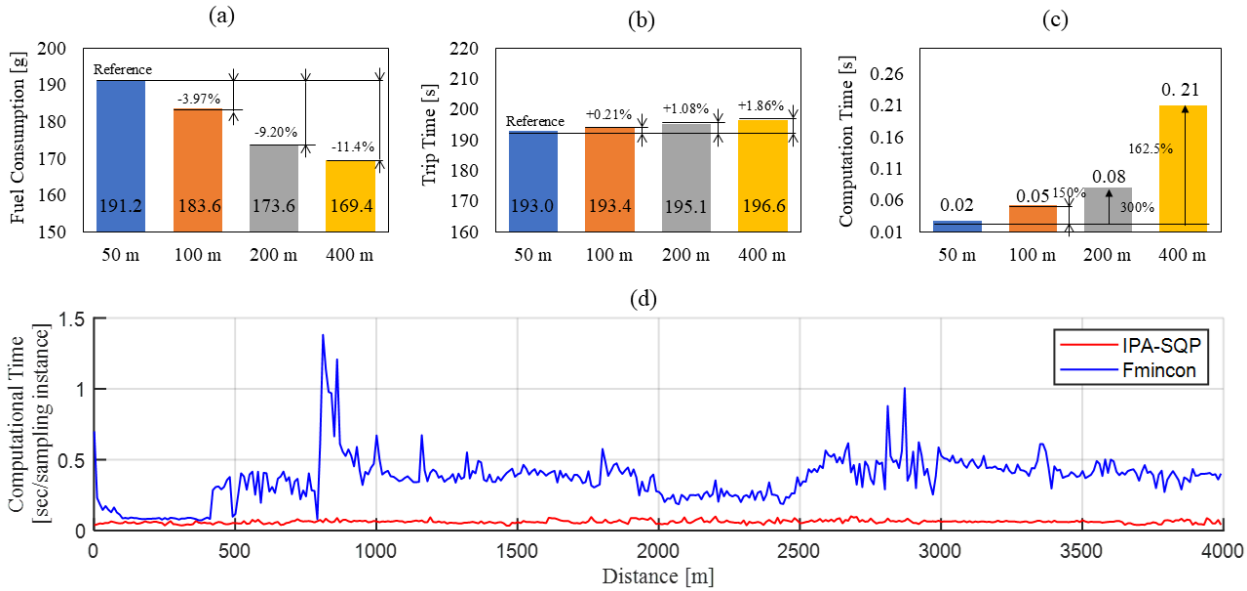


Fig. 11. Impact of prediction horizon length on PCC system performance. (a) Fuel consumption. (b) Trip time. (c) Average computation time. (d) Computation time per sampling step for a 200 m horizon using different solvers.

The computation time per sampling instance for the PCC controller with a 200m prediction horizon is shown in Fig.11(d). Compared to the Fmincon solver, the IPA-SQP method significantly reduces computation time, enabling the controller to operate at a 10Hz control rate—sufficient for highway cruise control. This also highlights the advantage of the IPA-SQP method's warm-starting capability, which is particularly effective for the UTM-MPC framework, where two optimization problems with different cost functions must be solved. Given the gradual changes in vehicle speed and the predefined slope and speed limits, as shown in Fig.10, the optimal solution can be efficiently refined from the previous candidate with only a few iterations.

VI. Conclusion

This paper proposes a UTM-MPC approach for PCC using a digital map. This method addresses the performance limitations of the WSM, which relies on predefined constant weights. First, the Knee point on the Pareto front and its

properties are analyzed based on simulation results of WSM with fixed weights, using slope profiles from a digital map. The analysis reveals that the shape of the Pareto front varies significantly under different driving conditions. To leverage the benefits of the Knee point, UTM is introduced, minimizing the distance from the objectives in each sampling interval to the precomputed average Utopia point. Simulation results using slope profiles from field tests demonstrate that, compared to WSM, UTM-MPC achieves lower fuel consumption while maintaining the same trip time, even under varying speed limits. Additionally, compared to the Fmincon solver, the UTM-based PCC system—implemented with the neighboring extremal-based IPA-SQP method—significantly reduces computation time per sampling instance. This efficiency enables the PCC controller to operate at a 10 Hz control cycle, making it suitable for highway cruise control.

Future work will focus on experimentally validating the proposed approach using the test bed shown in Fig. 1. Additionally, further research will explore more computationally efficient methods for deriving the Utopia point.

ACKNOWLEDGMENT

The authors would like to thank Xun Gong and Chao Shen for their comments on this article. The work of Yongjun Yan was completed when he was with the Department of Naval Architecture and Marine Engineering, University of Michigan, Ann Arbor, MI, USA.

REFERENCES

[1] R. Varnhagen, "Electronic horizon: A map as a sensor and predictive control," SAE, Warrendale, PA, USA, Tech. Paper 2017-01-1945, 2017.

[2] M. A. S. Kamal, M. Mukai, J. Murata, and T. Kawabe, "Ecological vehicle control on roads with up-down slopes," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 783–794, Sep. 2011.

[3] M. R. Amini, I. Kolmanovsky, and J. Sun, "Hierarchical MPC for robust eco-cooling of connected and automated vehicles and its application to electric vehicle battery thermal management," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 1, pp. 316–328, Jan. 2021.

[4] E. Hellström, M. Ivarsson, J. Åslund, and L. Nielsen, "Look-ahead control for heavy trucks to minimize trip time and fuel consumption," *Control Eng. Pract.*, vol. 17, no. 2, pp. 245–254, Feb. 2009.

[5] B. Saerens, "Optimal control based eco-driving," in *Theoretical Approach and Practical Applications*. Leuven, Belgium: Katholieke Universiteit Leuven, 2012.

[6] H. Chu, L. Guo, Y. Yan, B. Gao, and H. Chen, "Self-learning optimal cruise control based on individual car-following style," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 10, pp. 6622–6633, Oct. 2021.

[7] E. Hellström, J. Åslund, and L. Nielsen, "Design of an efficient algorithm for fuel-optimal look-ahead control," *Control Eng. Pract.*, vol. 18, no. 11, pp. 1318–1327, Nov. 2010.

[8] H. Chu, L. Guo, B. Gao, H. Chen, N. Bian, and J. Zhou, "Predictive cruise control using high-definition map and real vehicle implementation," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 11377–11389, Dec. 2018.

[9] H. Chu, L. Guo, Y. Yan, B. Gao, H. Chen, and N. Bian, "Energy-efficient longitudinal driving strategy for intelligent vehicles on urban roads," *Sci. China Inf. Sci.*, vol. 62, no. 6, p. 64201, Jun. 2019.

[10] S. Bae, Y. Kim, J. Guanetti, F. Borrelli, and S. Moura, "Design and implementation of ecological adaptive cruise control for autonomous driving with communication to traffic lights," in *Proc. Amer. Control Conf. (ACC)*, 2019, pp. 4628–4634.

[11] A. Sciarretta and A. Vahidi, *Energy-Efficient Driving of Road Vehicles*. Cham, Switzerland: Springer, 2020.

[12] X. Gong et al., "A benchmark study for real-time optimal control of connected HEVs: Design, integration, and evaluation," *IEEE Trans. Transport. Electrific.*, vol. 10, no. 3, pp. 7591–7603, Sep. 2024.

[13] V. V. Monastyrsky and I. M. Golownykh, "Rapid computation of optimal control for vehicles," *Transp. Res. B, Methodol.*, vol. 27, no. 3, pp. 219–227, Jun. 1993.

[14] M. Vallerio, J. Van Impe, and F. Logist, "Tuning of NMPC controllers via multi-objective optimisation," *Comput. Chem. Eng.*, vol. 61, pp. 38–50, Feb. 2014.

[15] V. M. Zavala, "Real-time resolution of conflicting objectives in building energy," *Proc. SimBuild*, vol. 5, no. 1, pp. 1–8, Jan. 2012.

[16] A. S. M. Bakibillah, M. A. S. Kamal, C. P. Tan, T. Hayakawa, and J.-I. Imura, "Fuzzy-tuned model predictive control for dynamic eco-driving on hilly roads," *Appl. Soft Comput.*, vol. 99, Feb. 2021, Art. no. 106875.

[17] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Struct. Multidiscip. Optim.*, vol. 26, no. 6, pp. 369–395, 2004.

[18] A. Bemporad and D. M. de la Peña, "Multiobjective model predictive control," *Automatica*, vol. 45, no. 12, pp. 2823–2830, Dec. 2009.

[19] S. Ober-Blöbaum and S. Peitz, "Explicit multiobjective model predictive control for nonlinear systems with symmetries," *Int. J. Robust Nonlinear Control*, vol. 31, no. 2, pp. 380–403, Jan. 2021.

[20] K. Deb and S. Gupta, "Understanding knee points in bicriteria problems and their implications as preferred solution principles," *Eng. Optim.*, vol. 43, no. 11, pp. 1175–1204, Nov. 2011.

[21] V. M. Zavala and A. Flores-Tlacuahuac, "Stability of multiobjective predictive control: A utopia-tracking approach," *Automatica*, vol. 48, no. 10, pp. 2627–2632, Oct. 2012.

[22] T. Schmitt, T. Rodemann, and J. Adamy, "Multi-objective model predictive control for microgrids," *Automatisierungstechnik*, vol. 68, no. 8, pp. 687–702, Aug. 2020.

[23] Y. Yan et al., "Eco-coasting controller using road grade preview: Evaluation and online implementation based on mixed integer model predictive control," *IEEE Trans. Veh. Technol.*, vol. 72, no. 10, pp. 12508–12523, 2023.

[24] D. Liao-Mcpherson, M. M. Nicotra, and I. Kolmanovsky, "Time-distributed optimization for real-time model predictive control: Stability, robustness, and constraint satisfaction," *Automatica*, vol. 117, Jul. 2020, Art. no. 108973.

[25] M. Diehl, H. G. Bock, and J. P. Schlöder, "A real-time iteration scheme for nonlinear optimization in optimal feedback control," *SIAM J. Control Optim.*, vol. 43, no. 5, pp. 1714–1736, Jan. 2005.

[26] M. Diehl et al., "Real-time optimization for large scale processes: Nonlinear model predictive control of a high purity distillation column," in *Online Optimization of Large Scale Systems*. Berlin, Germany: Springer-Verlag, 2001, pp. 363–383.

[27] S. Gros, M. Zanon, R. Quirynen, A. Bemporad, and M. Diehl, "From linear to nonlinear MPC: Bridging the gap via the real-time iteration," *Int. J. Control*, vol. 93, no. 1, pp. 62–80, Jan. 2020.

[28] R. Ghaemi, J. Sun, and I. V. Kolmanovsky, "Neighboring extremal solution for nonlinear discrete-time optimal control problems with state inequality constraints," *IEEE Trans. Autom. Control*, vol. 54, no. 11, pp. 2674–2679, Nov. 2009.

[29] H. Park, J. Sun, and I. Kolmanovsky, "A tutorial overview of IPA-SQP approach for optimization of constrained nonlinear systems," in *Proc. 11th World Congr. Intell. Control Autom.*, Jun. 2014, pp. 1735–1740.

[30] H. Park et al., "Real-time model predictive control for shipboard power management using the IPA-SQP approach," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 6, pp. 2129–2143, Nov. 2015.

[31] H. A. Rakha, K. Ahn, K. Moran, B. Saerens, and E. Van den Bulck, "Virginia tech comprehensive power-based fuel consumption model: Model development and testing," *Transp. Res. D, Transport Environ.*, vol. 16, no. 7, pp. 492–503, 2011.

[32] W. Dib, L. Serrao, and A. Sciarretta, "Optimal control to minimize trip time and energy consumption in electric vehicles," in *Proc. IEEE Vehicle Power Propuls. Conf.*, Chicago, IL, USA, Sep. 2011, pp. 1–8.

[33] R. T. Marler and J. S. Arora, "The weighted sum method for multi-objective optimization: New insights," *Struct. Multidisciplinary Optim.*, vol. 41, no. 6, pp. 853–862, Jun. 2010.

[34] A. Myklebust and L. Eriksson, "Road slope analysis and filtering for driveline shuffle simulation," *IFAC Proc. Volumes*, vol. 45, no. 30, pp. 176–183, 2012.

[35] U. Sadek, A. Sarjaš, A. Chowdhury, and R. Svečko, "FPGA-based optimal robust minimal-order controller structure of a DC–DC converter with Pareto front solution," *Control Eng. Pract.*, vol. 55, pp. 149–161, Oct. 2016.

[36] P.-L. Yu, "A class of solutions for group decision problems," *Manage. Sci.*, vol. 19, no. 8, pp. 936–946, 1973.

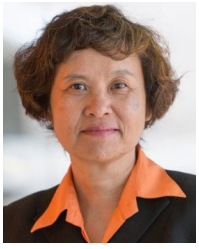
[37] M. E. Fisher, W. J. Grantham, and K. L. Teo, "Neighbouring extremals for nonlinear systems with control constraints," *Dyn. Control*, vol. 5, no. 3, pp. 225–240, Jul. 1995.
[38] S. Gros, B. Srinivasan, B. Chachuat, and D. Bonvin, "Neighbouring-extremal control for singular dynamic optimisation problems. Part I: Single-input systems," *Int. J. Control*, vol. 82, no. 6, pp. 1099–1112, Jun. 2009.
[39] R. Ghaemi, "Robust model based control of constrained systems," Ph.D. dissertation, Dept. Elect. Eng., Univ. Michigan, Ann Arbor, MI, USA, 2010.
[40] R. Rajamani, *Vehicle Dynamics and Control*. Cham, Switzerland: Springer, 2011.

Yongjun Yan received the B.S. degree in vehicle engineering from Hefei University of Technology, Hefei, China, in 2016, and the Ph.D. degree in vehicle engineering from Jilin University, Changchun, China, in 2022. He was a Joint Ph.D. Student at the University of Michigan, Ann Arbor, from 2019 to 2021, where he was co-advised by Prof. J. Sun. He is currently a Post-Doctoral Research Fellow at The Chinese University of Hong Kong. His current research interests include optimal control and applications in automobile and continuum robots.

Ziyou Song (Senior Member, IEEE) received the bachelor's and Ph.D. degrees (Hons.) in automotive engineering from Tsinghua University, China, in 2011 and 2016, respectively. He is currently an Assistant Professor at the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor. Prior to joining the University of Michigan, he was an Assistant Professor at the National University of Singapore and was a Battery Algorithm Engineer at Apple. His research interests include modeling, estimation, optimization, and control of energy storage systems, especially for the electrified transportation sector. He has received several paper awards, including the Automotive Innovation Best Paper Award, the Applied Energy Highly Cited Paper Award, the NSK Outstanding Paper Award of Mechanical Engineering, and the IEEE VPPC Best Student Paper Award. He serves as an Associate Editor and an Editorial Member for IEEE TRANSACTIONS ON TRANSPORTATION ELECTRIFICATION, IEEE TRANSACTIONS ON POWER ELECTRONICS, *Applied Energy*, and *eTransportation*.

Bingzhao Gao (Member, IEEE) received the B.S. and M.S. degrees in vehicle engineering from Jilin University, Changchun, China, in 1998 and 2002, respectively, the Ph.D. degree in mechanical engineering from Yokohama National University, Yokohama, Japan, and the Ph.D. degree in control engineering from Jilin University, in 2009. He is currently a Professor with Tongji University. His current research interests include vehicle powertrain control and vehicle stability control.

Hong Chen (Fellow, IEEE) received the B.S. and M.S. degrees in process control from Zhejiang University, Hangzhou, China, in 1983 and 1986, respectively, and the Ph.D. degree in system dynamics and control engineering from the University of Stuttgart, Stuttgart, Germany, in 1997. In 1986, she joined Jilin University of Technology, Changchun, China. From 1993 to 1997, she was a Wissenschaftlicher Mitarbeiter with the Institut für Systemdynamik und Regelungstechnik, University of Stuttgart. Since 1999, she has been a Professor with Jilin University, Changchun, and hereafter a Tang Aoqing Professor. From 2015 to 2019, she was the Director of the State Key Laboratory of Automotive Simulation and Control, Jilin University. In 2019, she joined Tongji University, Shanghai, China, as a Distinguished Professor. Her current research interests include model predictive control, nonlinear control, and applications in mechatronic systems focusing on automotive systems.



Jing Sun (Fellow, IEEE) received the B.S. and M.S. degrees from the University of Science and Technology of China and the Ph.D. degree from the University of Southern California in 1989. From 1989 to 1993, she was an Assistant Professor at the Electrical and Computer Engineering Department, Wayne State University. She joined Ford Research Laboratory in 1993, working at the Powertrain Control Systems Department. After spending almost ten years in the industry, she returned to academia and joined as a Faculty Member at the College of Engineering, University of Michigan, in 2003. She is currently a Michael G. Parsons Collegiate Professor at the Naval Architecture and Marine Engineering Department, University of Michigan. She holds 44 U.S. patents and has published over 300 peer-reviewed journals and conference papers. She co-authored the textbook "*Robust Adaptive Control*," which received over 7500 citations. Her research interests include modeling, control, and optimization of dynamic systems with applications to marine and automotive systems. Her current research focuses on real-time optimization and decision-making for energy and transportation systems. She is a fellow of the National Academy of Inventors, IFAC, and the Society of Naval Architects and Marine Engineers. She was a recipient of the 2003 IEEE Control System Technology Award.