A double-layer data-driven motion planning and control method for parallel parking

Author Shaoyu Song *(Presenter), Hui Chen

Department School of Automotive Studies

University Tongji University, Shanghai, 201804, China

E-mail s.song@tongji.edu.cn

1 Introduction

2 Preliminaries

3 Double-layer data-driven parking

4 Comparison of simulation

5 Summary

Automated Parking System(APS) for Wireless Power Transfer

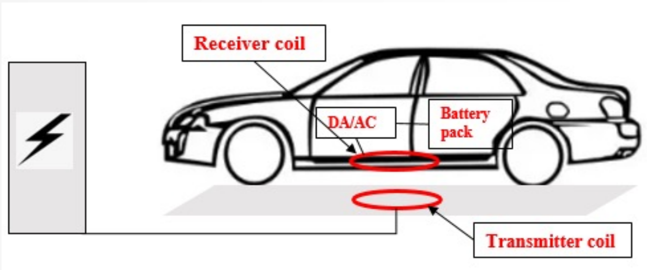
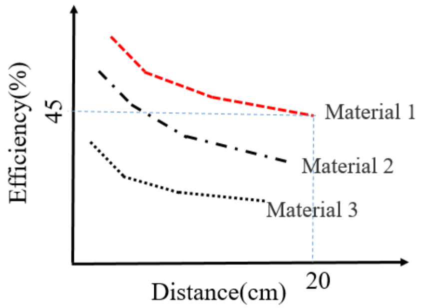


Diagram of a typical wireless power transfer



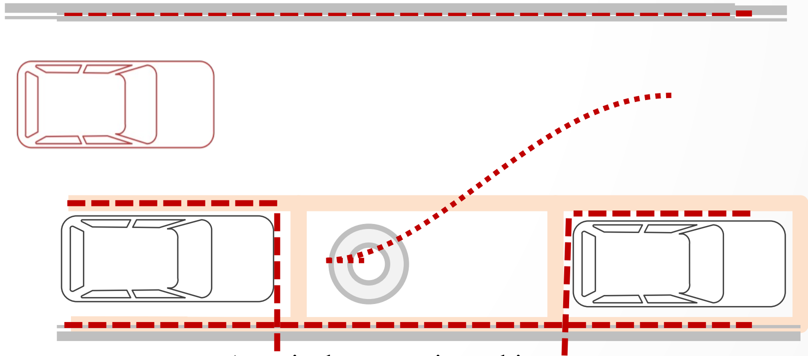
Power transfer efficiency for different coil materials[2]

Wireless Power Transfer: transmission of electrical energy without wires as a physical link

Fewer wires, increasing the mobility, convenience, and safety

Key Challenge: technical challenges such as the **low transfer efficiencies** as the distances increase made this WPT develop very slowly

Detection » **Path planning** » **Path following control**



A typical automatic parking system

Park and Charge: vehicles are parked 96 % of their time

Critical Needs: high precision of parking position

Modules: slot detection, path planning, path following control, ego-vehicle’s posture estimation, and chassis control

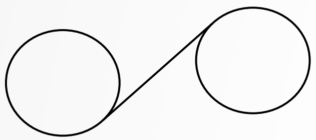
Directly related to **parking accuracy**: planning, following control

[2] Machura P, Li Q. A critical review on wireless charging for electric vehicles [J]. Renewable & Sustainable Energy Reviews, 2019, 104: 209-234.

Related work

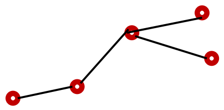
Conventional method

Path planning



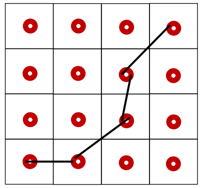
Circular arc, line, spline curve…

Curve-based method



RRT, RRT*, BiRRT,…

Sampling-based method



A*, D*,…

Search-based method

Advantage: simple, easy to implement

Disadvantage: parameter adjustment rely on experience

Path-following control

$$J=\frac{1}{2}\int_0^\infty\left[x^T(t)Qx(t)+u^T(t)Ru(t)\right]dt$$

$$J^*=\frac{1}{2}e_{x0}(s_d)^T P e_{x0}(s_d)$$

$$u^*(t)=-Kx(t),\quad K=R^{-1}B^TP$$

Linear quadratic regulator [6]

$$\delta = tan^{-1}(kL)$$

$$\delta = tan^{-1}\left(\frac{2Lsin\theta}{L_f}\right)$$

Pure pursuit [7]

$$J=\sum_{n=1}^{N}(q_{R_i}-q_i)$$

$$q_0=q_{P_C}$$

$$q_{i+1}=q_i+f(q_i,u_i)\Delta t$$

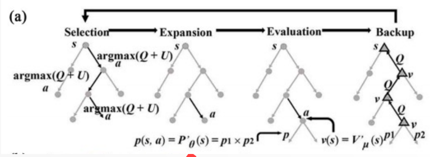
Model predictive control[2]

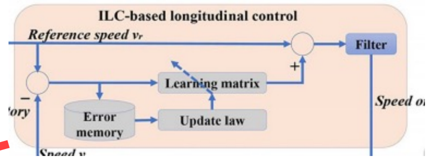
Advantage: simple, easy to implement

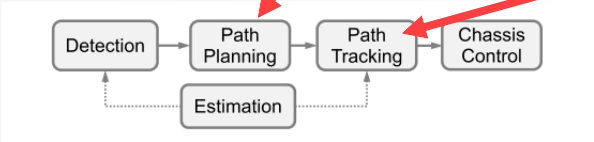
Disadvantage: rely on model with fixed parameters, hard to adjust parameter

AI-based method:

Characteristic: high on-line computational efficiency, working condition self-adaptation, parameter self-adjustment







In previous study:

motion planning: data-driven learnable Monte Carlo tree search(MCTS)-**open-loop control**

tracking control: iterative learning control and **MPC-could not alter the trajectory**

How could the two parts work together to further improve performance?

[1] Song S Y, Chen H, Sun H W, et al. Time-Optimized Online Planning For Parallel Parking With Nonlinear Optimization and Improved Monte Carlo Tree Search [J]. Ieee Robotics and Automation Letters, 2022, 7(2): 2226-2233.
[2] Song S Y, Zhang S K. Data-driven trajectory-tracking in automated parking system via iterative learning compensation and model predictive control [J]. Proceedings of the Institution of Mechanical Engineers Part D-Journal of Automobile Engineering.

1 Introduction

2 Preliminaries

3 Double-layer data-driven parking

4 Comparison of simulation

5 Summary

Markov decision process

- A mathematical model described by a five tuple $< S, A, P, r, \mu >$
  - ✓ state space-S:
    - ■ vehicle state $(x, y, \theta, v, \delta)$
  - ✓ action space-A:
    - ■ Discrete steering wheel angle increment
    - ■ Discrete speed increment
  - ✓ reward function-r:
    - ■ $r(s_k, a_k) = R_y + R_\theta + R_a + R_{safe}$

$$R_y = \frac{-20000}{1+e^{-c_1\times|y-yt|}} + 20000, R_\theta = \frac{-20000}{1+e^{-c_2\times|\theta-\theta t|}} + 20000,$$

$$R_a = c_3 \times \sum_{i=1}^{k} | a_i - a_{i-1} |, R_{safe} = \{0, -10000\}$$

  - ✓ State transition function -P:
    - ■ vehicle single track
  - ✓ discount factor - : μ
    - ■ Reflect the ability of the algorithm to observe the future state
- **How to control the speed and steering angle given vehicle state?**

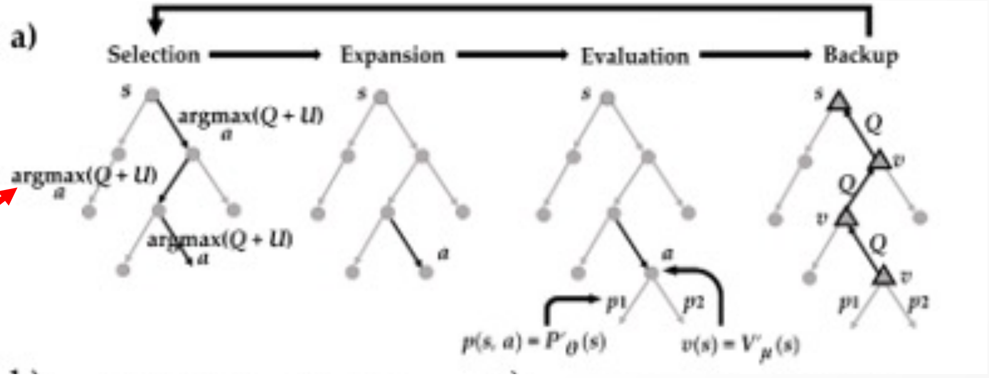
Monte Carlo tree search

$$q(s,a) \leftarrow \frac{1}{N(s,a)} \sum_{i=1}^{N(s)} I_i(s,a) z_i$$

- Iteratively performs 4 steps: selection - expansion - evaluation-backup
- Tree policy:

$$a_t(s) = \mathrm{argmax}(q(s,a) + c_{puct} P(s,a)^{\mu} \sqrt{\frac{\sum_b N(s,b)}{1+N(s,a)}})$$

$$\pi(a|s) = \frac{N(s,a)^{1/T_{em}}}{\sum_b (N(s,b)^{1/T_{em}})}$$



Monte Carlo Tree Search

Model Predictive Control

- **General form**

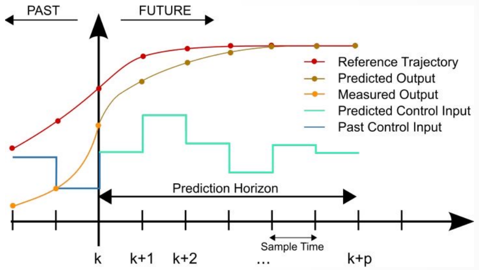
$$\min_{U_t,\xi_{t+1},\xi_{t+2},\dots,\xi_{t+Np,t}} J_{Nc}(\xi_t,U_t)$$

$$s.t.\quad \xi_{k+1,t}=f(\xi_t,u_{k,t}),\ k=t,\dots,N-1$$
$$\xi_{k,t}\in X$$
$$u_{k,t}\in U$$

Control ***Nc*** steps (perform the first), predict ***Np*** steps

- **Tracking control**

$$J(\xi(t),u(t-1),\Delta U(t))=\sum_{i=1}^{Np}\left\|\Psi_t\xi(t)+\Theta_t\Delta U(t)\right\|_Q^2+\sum_{i=1}^{Nc}\left\|\Delta U(t+i|t)\right\|_R^2+\rho\varepsilon^2$$
$$u_{\min}(t+k)\le u(t+k)\le u_{\max}(t+k),$$
$$\Delta u_{\min}(t+k)\le \Delta u(t+k)\le \Delta u_{\max}(t+k)$$



Model Predictive Control

Iterative Learning Control

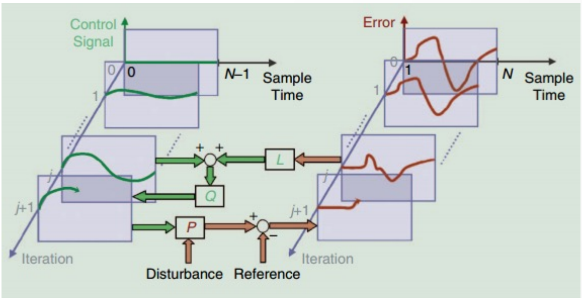
$$x^l{}_{k+1}=A^l x^l{}_k+B^l u^l{}_k,\quad y^l{}_k=C^l x^l{}_k$$

Define q-coefficient matrix $x^l{}_{k+1}=q^l x^l{}_k$, obtain $x^l{}_{k+1}=\left(q^l I-A^l\right)^{-1}u^l{}_k$

$$y^l{}_{jk}=P^l(q^l)\,u^l{}_{jk}+C^l(A^l)^k x^l{}_0,\ P^l(q^l)\equiv C^l(q^l I-A^l)^{-1}B^l$$

Initial state 0

$$\begin{bmatrix} y^l{}_{j1}\\ y^l{}_{j2}\\ \dots\\ y^l{}_{jN}\end{bmatrix}=\begin{bmatrix} P^l{}_{1,1} & 0 & \dots & 0\\ P^l{}_{2,1} & P^l{}_{1,1} & \dots & 0\\ \dots & \dots & \dots & 0\\ P^l{}_{N,1} & P^l{}_{N-1,1} & \dots & P_{1,1}\end{bmatrix}\begin{bmatrix} u^l{}_{j0}\\ u^l{}_{j1}\\ \dots\\ u^l{}_{jN-1}\end{bmatrix}\qquad p_{mk}=\begin{cases}0, m<k\\ C^l B^l, m=k\\ C^l A^l{}_{m-1}\dots A^l{}_k B^l{}_k, m>k\end{cases}$$



Iterative Learning Control

- Control law: $u^l{}_{j+1,k}=Q^l(u^l{}_{j,k}+L^l e^l_{j,k+1})$

$$Q^l{}_{opt}=\left((P^l)^T T_{LG}P^l+R_{LG}+S_{LG}\right)^T u^l+\left((P^l)^T T_{LG}P^l+S_{LG}\right)$$
$$L^l{}_{opt}=\left((P^l)^T T_{LG}P^l+S_{LG}\right)^T(P^l)^T T_{LG}$$

1 Introduction

2 Preliminaries

3 Double-layer data-driven parking

4 Comparison of simulation

5 Summary

Overview

- **Two parts: upper level + lower level**

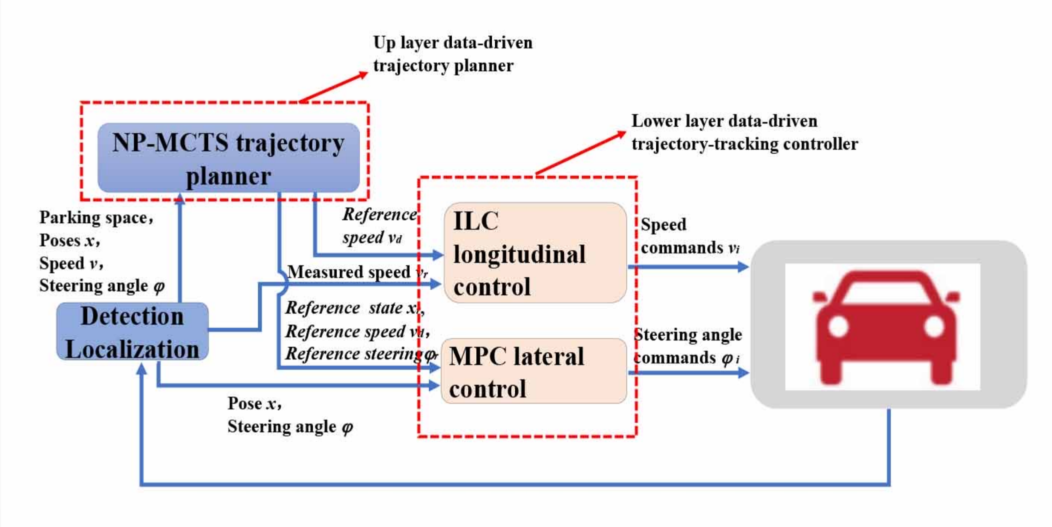


Figure 2: Overall diagram of the parking motion planning and trajectory following control system

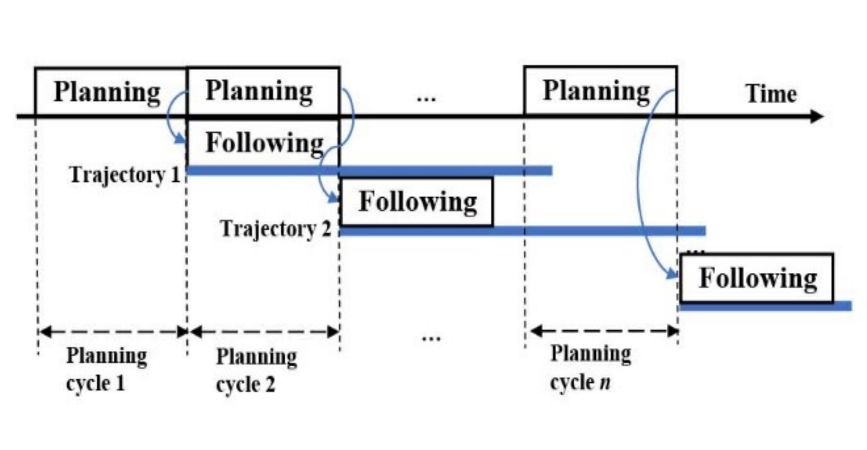


Figure 3: Iterative cycle, the vehicle and parking space information is our root model that is different from [18]

Lower-level data-driven learning control

- **Online trajectory search+ trajectory following control**

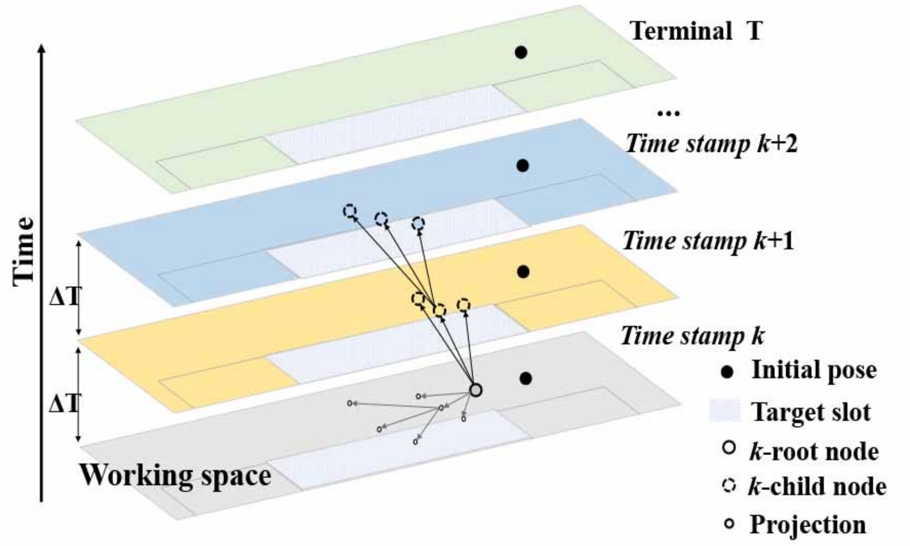


Figure 4: Search in spatiotemporal space in MCTS

Speed compensating

- **Step 1: learn**
  - **Obtain matrix Q and L**
  - **Collect error data**

$$\begin{bmatrix} e^l_{j,1} \\ e^l_{j,2} \\ \cdots \\ e^l_{j,N} \end{bmatrix} = \begin{bmatrix} y^l_{d,1} \\ y^l_{d,2} \\ \cdots \\ y^l_{d,N} \end{bmatrix} - \begin{bmatrix} y^l_{j,1} \\ y^l_{j,2} \\ \cdots \\ y^l_{j,N} \end{bmatrix}$$

  - **Update control law**

$$u^l{}_{j+1,k} = Q^l(u^l{}_{j,k} + L^l e^l_{j,k+1})$$

  - **Collect error data**

…

- **Step 2: online test**
  - The length of the commands N is different in different parking position

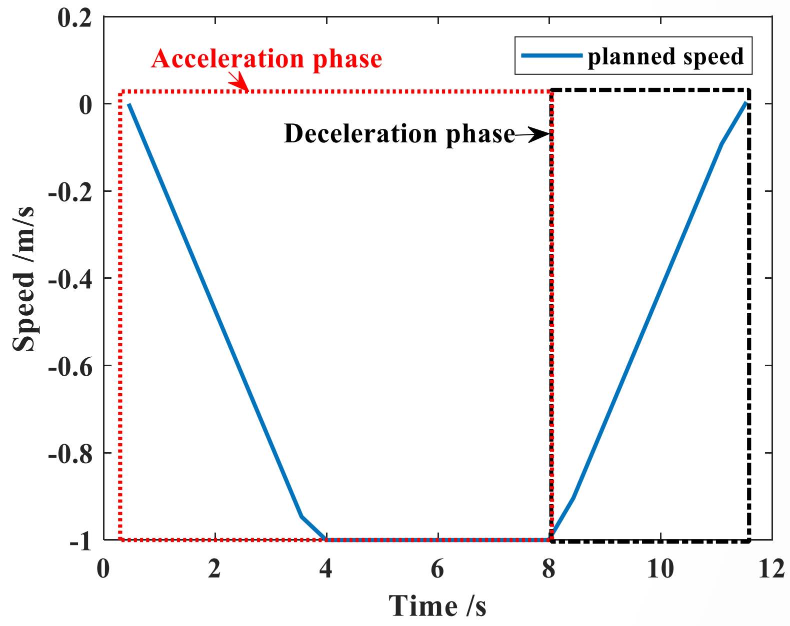


Figure 5: Phases of the speed during the parking

1 Introduction

2 Preliminaries

3 Double-layer data-driven parking

4 Comparison of simulation

5 Summary

Test conditions

Table 1: Vehicle and controller parameters

Item	Value	Item	Value
Vehicle length	3.569 m	Front overhang	0.72 m
Vehicle width	1.551 m	Rear overhang	0.54 m
Wheelbase	2.305 m	Trans. ratio	16.68
Vehicle length	3.569 m	Front overhang	0.72 m
Steering -MPC	400 °/s	Q weight -MPC	diag(1, 0.1, 2)
Prediction -MPC	20 steps	R weight-MPC	$1.5\times I_{Nc}$
Control -MPC	15 steps	T_{LG} -LQR	$1 \times I_{N\times N}$
Relaxation -MPC	5	R_{LG} -LQR	$0.1\times I_{N\times N}$
		S_{LG} -LQR	$0.01\times I_{N\times N}$

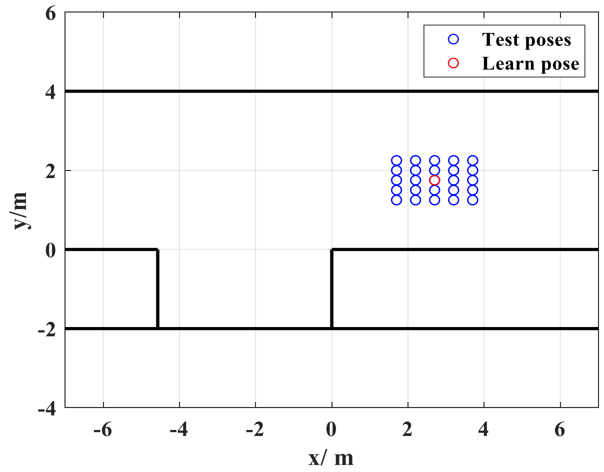


Figure 6: Initial positions of training in the double-layers data driven method, where the data in circles are used in motion planning layers and red circle is used to learn speed compensate

Verification of lower layer

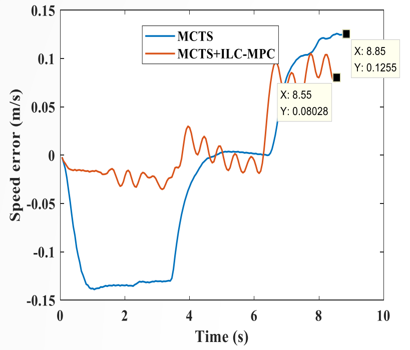


Figure 7: Comparsion of speed following errors at (1.7 m, 1.25 m, 0°)

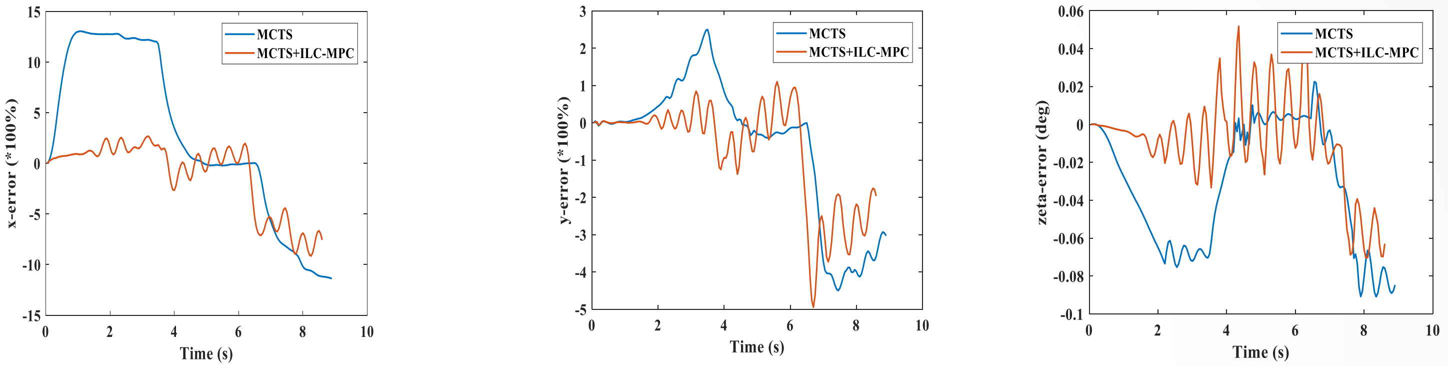


Figure 8: Comparsion of trajectory-following control errors at (1.7 m, 1.25 m, 0°): (a) *x*-coordinate; (b) *y*-coordinate, and (c) θ

Verification of overall system

Table 2: Statistical results of parking process in the training poses, 25 trials

	Item	Open-loop	With tracking
Y errors/ m	Mean	0.010	0.016
	Max	0.055	0.114
	Min	9.79e-05	0.003
	Std.	0.015	0.023
θ errors/ °	Mean	1.137	1.185
	Max	1.318	1.327
	Min	0.984	1.013
	Std.	0.091	0.090

Gear Changes times	Mean	2.00	2.00
	Max	2	2
	Min	2	2
	Std.	0.00	0.00
Time/ s	Mean	16.904	16.9
	Std.	0.715	1.11
	Effective rate	72%	

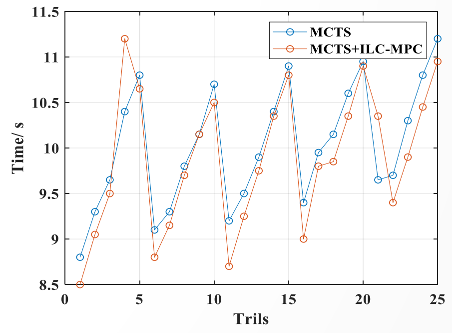


Figure 9: Motion time of parking in different initial positions

- ✓ Final parking errors of open-loop control is more stable and smaller
- ✓ Tracking control is beneficial to the **time** and **control errors during the parking**

Adaptability to different initial positions

changing the initial vehicle angle from 0 to [−8°, 8°]

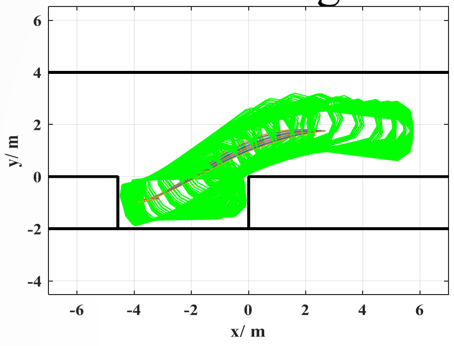


Figure 11: Parking trajectories when change the initial angle of vehicle from 0° to [−8°, 8°] at (2.7 m, 1.75 m)

Adaptability to different sizes of slots

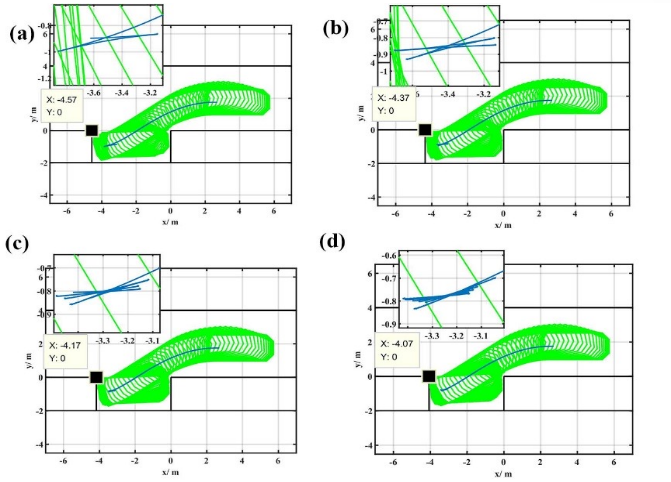


Figure 12: Parking trajectories using same model and parameters with different parking slot length: (a) 4.57 m; (b) 4.37 m; (c) 4.17 m; (d) 4.07 m

Adaptability to different road conditions

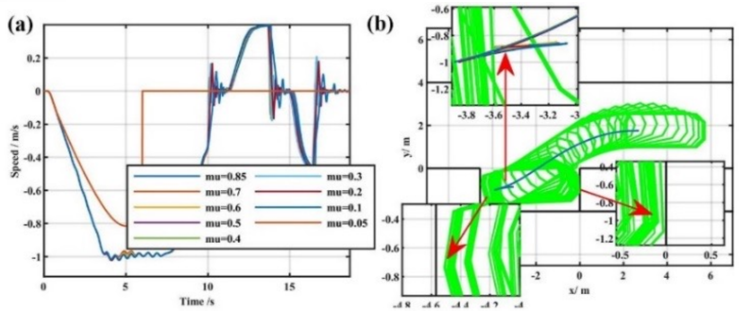


Figure 13: (a) Speed response, and (b) parking trajectories with different road friction 0.1-0.85, without 0.05

✓ Adaptability to different initial positions/ sizes of slots/ road friction has been confirmed

1 Introduction
2 Preliminaries
3 Double-layer data-driven parking
4 Comparison of simulation
5 Summary

Summary

1. The proposed method achieved high precision in position. The mean precisions of parking position in the *y*-axis for one-layer open-loop control and double-layers control are 0.010 m and 0.016 m, respectively.
2. Speed compensating is beneficial to the parking time performance and avoids sudden deceleration at the expense of higher *y*-axis errors.
3. Generalization ability of the proposed method is confirmed.

A double-layer data-driven motion planning and control method for parallel parking

Tongji university

Shaoyu Song PhD student

1810554

Tel：18621284659
Email：s.song@tongji.edu.cn
WOS ResearcherID：AAR-2372-2021
Orcid： https://orcid.org/0000-0001-7607-3167

1

Thank you