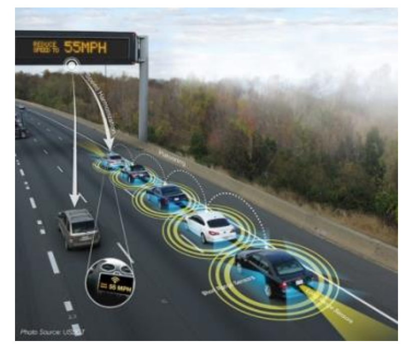
Constrained Optimization and Distributed Computation Based Car-Following Control of A Connected and Autonomous Vehicle Platoon

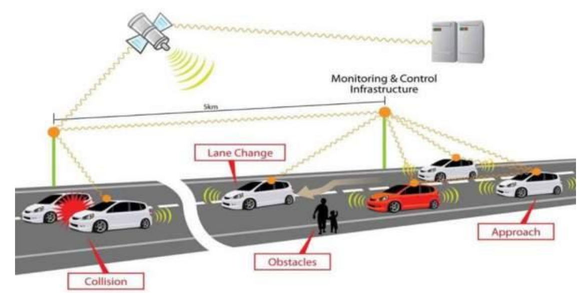


Siyuan Gong[a], Jinglai Shen[b], Lili Du[a]

ldu3@iit.edu

a: Illinois Institute Technology

b: University of Maryland Baltimore County



Background and motivation

Human-driven Car-following Behavior

- Relay on driver’ s perception and driving experience
- Safety problem. No system effect control.

Existing Adaptive Cruise Control

- Ensure individual vehicles’ mobility and safety.
- No system effect control

Connected and autonomous vehicle (CAV)

- V2V enables information exchange
- Local computation enables autonomous drive
- Enable traffic safety and efficiency of the entire platoon, sustaining individual vehicle’s mobility

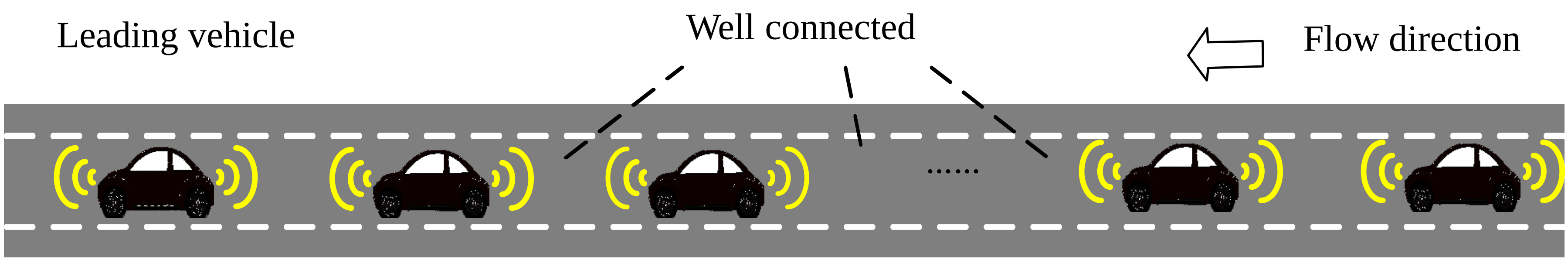
Cooperative Adaptive Cruise Control (CACC)

State of the Art:

- Study in transportation community: keeping safe or stable gap
  - Focused on neighborhood traffic safety and efficiency
  - No system effect control
- Study in control community
  - Focus on asymptotic string stability
  - Miss the consideration of the transient process which may affect traffic flow stability significantly.
- Data structure has been applied
  - Immediate preceding (IP) vehicle
  - Multiple preceding (MP) vehicles
  - Preceding (one or multiple) and one following (FP) vehicles
  - Not fully take advantage of the connectivity yet

ILLINOIS INSTITUTE OF TECHNOLOGY

Coordinated Platoon Car-following Control



Assumptions

- A pure CAV platoon, including a leading vehicle and several following vehicles
- Global information structure: A well-connected platoon thus enables a vehicle share information with all other vehicles in the platoon.

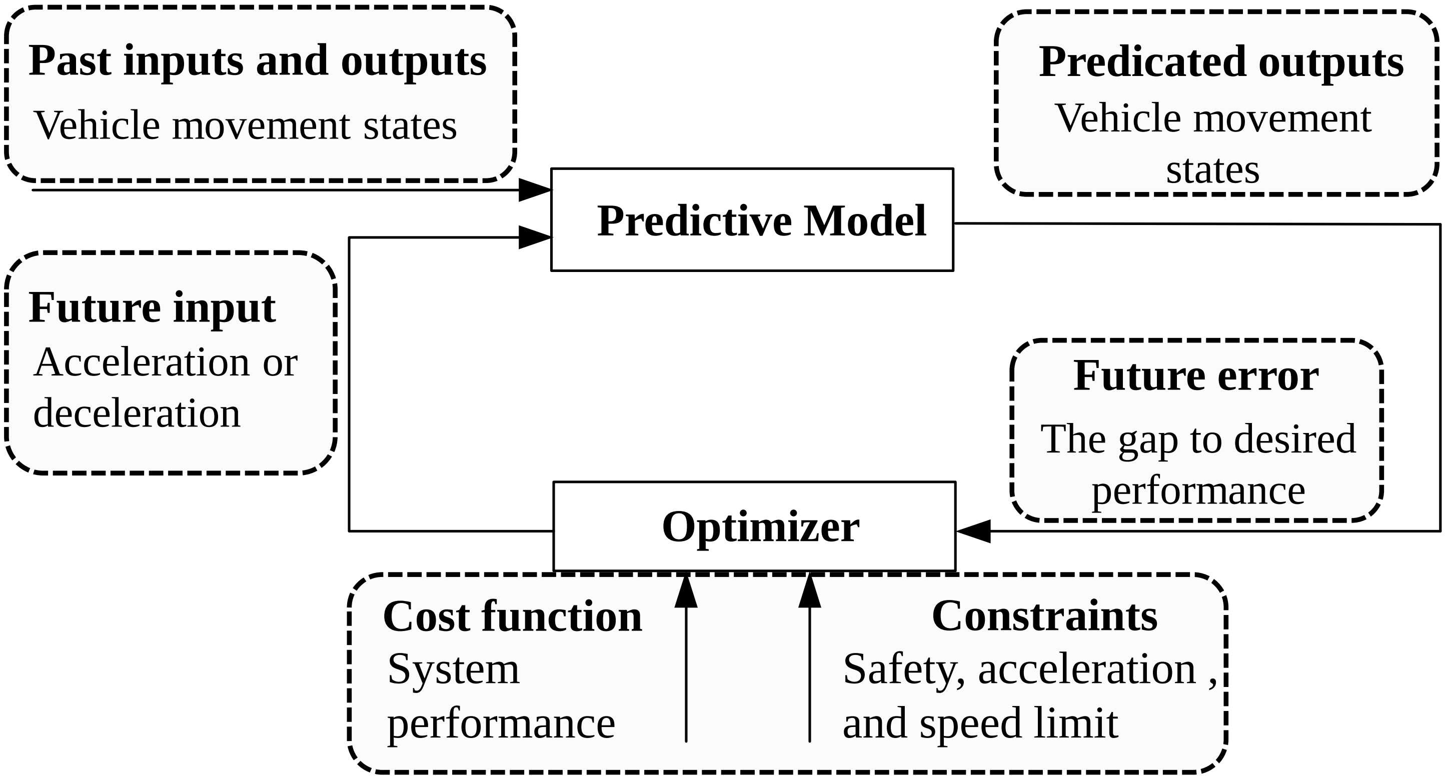
Objectives

- Design a closed loop control so that vehicles coordinately determine their movements to approach desired system performance
- Develop distributed computation to conduct the control algorithm

Close-Loop Control for CAV Platoon

Decision Variables: acceleration/deceleration of following vehicles at each time step

Procedure:



Prediction Model: Vehicle Dynamics

τ: the sample length, the control u_i is constant on each time interval [kτ, (k + 1)τ) for k ∈ Z^+:= { 0 , 1 , 2 , . . . },

The discrete-time longitudinal dynamics is described by the following double-integrator model

- The speed of vehicle *i* at next time step $k+1$ is

$$v_i(k+1) = v_i(k) + u_i(k)\tau$$

- The location of vehicle *i* at next time step $k+1$ is

$$x_i(k+1) = x_i(k) + v_i(k)\tau + \frac{u_i(k)}{2}\tau^2$$

- The spacing between vehicle *i-1* and vehicle *i* at next time step $k+1$ is

$$s_{\mathrm{i}-1,i}(k+1) = x_{i-1}(k+1) - x_i(k+1)$$

- The relative speed fluctuation of vehicle *i* at next time step $k+1$ is

$$\Delta v_{i-1,i}(k+1) = v_{i-1}(k+1) - v_i(k+1)$$

Optimizer in the Closed Loop Control

OPT-C	$W = \sum_{i=1}^{n} \left\{ \alpha \left[s_{i-1,i}(k+1) - \Delta \right]^2 + \beta \left[\Delta v_{i-1,i}(k+1) \right]^2 + [u_i(k)]^2 \right\}$	
s.t.		
	$a_{min} \leq u_i(k) \leq a_{max}, \forall \mathrm{i} = 1, \cdots, \mathrm{n}$	Acceleration limit
	$0 \leq v_i(k+1) \leq v_{max}, \forall \mathrm{i} = 1, \cdots, \mathrm{n}$	Speed limit
	$0 \geq L + v_i(k) r - (v_i(k) - v_{min})^2 / 2a_{min} - s_{i-1,i}(k+1), \forall\, \mathrm{i} = 1, \cdots, \mathrm{n}$	Safety

- Objective(strictly convex): minimize traffic oscillation using mild control
  - Penalty on the relative spacing variation
  - Penalty on speed variation and the magnitude of control
  - Penalty weights α and β affect closed-loop dynamics
  - They together ensure transient state and asymptotic stability
- Three constraints (convex and compact set)
  - Ensure safety distance, speed limit and acceleration limits

Properties of the Optimizer

- **Lemma 3.1. (Sequential feasibility)**
  Suppose $(x_s, v_s)_{s=0}^{n}$ and u_0 are initially feasible such that they satisfy speed & acceleration limits, and safety constraints for all vehicles in the platoon. Then, the constraint set is always nonempty.

- **Lemma 3.2.**
  The constraint set has nonempty interior when $v_0 > v_{min}$; thus satisfies Slater's constraint qualification (CQ).

- **Theorem 3.1.**
  The optimizer has a unique optimal solution
  - Strictly convex objective function
  - Constraints define a convex and compact set

Please refer to the paper for the proofs of the above lemmas and theory.
Gong, S., Shen, J., Du, L. (2016). Constrained Optimization and Distributed Computation Based Car-Following Control of A Connected and Autonomous Vehicle Platoon. Transportation Research Part B: Methodological, Volume 94, Pages 314-334.*

Distributed Algorithms: Reformulate the Optimizer

- The optimization problem can be rewritten in a compact format
- Focus on its mathematical structure

minimize $J(u) := \frac{1}{2} u^T H u + c^T u + \gamma$ ⇒ Quadratic function

(A)

s.t. $\begin{cases} u_i \in \chi_i, & \forall i = 1, \cdots, n \\ g_i(u) \leq 0, & \forall i = 1, \cdots, n \end{cases}$ ⇒ Quadratic function

  - Where χ_i is the intersection of speed and acceleration limits-box constraints (compact and convex).
  - $g_i(u)$ is the safety distance constraint- a coupled constraint.
- Motivated by the distributed algorithm developed in Koshal et.al (2011), we develop our distributed algorithm to solve (A)
  - Gradient projection algorithm to iteratively explore a feasible and better solution
  - Primal-dual theory ensures the convergence

Distributed Algorithms: Primal-Dual Problems

- The Lagrangian dual function of (A) is

$$\mathcal{L}(u,\lambda) = J(u) + \lambda^T g(u), \quad \text{where } \lambda \in \mathbb{R}_+^n \text{ is the multiplier vector.}$$

- The primal and associated dual problems:

Primal (P) $\quad inf_{u\in\chi} sup_{\lambda\in\mathbb{R}_+^n} \mathcal{L}(u,\lambda)$ (1)

Dual(D) $\quad sup_{\lambda\in\mathbb{R}_+^n} inf_{u\in\chi} \mathcal{L}(u,\lambda)$ (2)

  - Where $\chi := \chi_1 \times \cdots \times \chi_n$ are box constraints.
  - $\mathbb{R}_+^n$ is a not compact set, which causes issues of the algorithm convergence

- Following from Slater's CQ and convexity of (A)
  - The strong duality holds: exists a dual optimal solution λ_*; and the optimal values of P and D match at optimality, (u_*, λ_*).
  - The primal-dual optimal pair (u_*, λ_*) is a saddle point of the Lagrangian dual function $\mathcal{L}(\mu, \lambda)$.

Distributed Alg: Make Duel Constraints Compact

- According to the definition of *saddle point* of $\mathcal{L}$, we have

$$\mathcal{L}(u', \lambda_*) \geq \mathcal{L}(u_*, \lambda_*) \geq \mathcal{L}(u_*, \lambda), \text{ where } u' \text{ be an interior point} \quad (1)$$

- Next, make λ=0, we change the inequality to

$$\mathcal{L}(u', \lambda_*) \geq \mathcal{L}(u_*, \lambda_*) \geq \mathcal{L}(u_*, 0) = J(u_*) \geq \min_{u \in \mathbb{R}^n} J(u) \coloneqq \mu, \quad (2)$$

$$\mathcal{L}(u', \lambda_*) = \underline{J(u') + \sum_{i=1}^{n} \lambda_{*,i} g_i(u') \geq \mu}. \quad (3)$$

- Given $g_i(u) \leq 0$ in (A), we have

$$J(u') - \mu \geq \sum_{i=1}^{n} \lambda_{*,i}\big(-g_i(u')\big) \geq \lambda_{*,i}\big(-g_i(u')\big). \quad (4)$$

- The following (convex) box constraint for the dual optimal solutions $\lambda_{*,i}$

$$\mathbb{I}_i \coloneqq \left\{ \lambda_i \in \mathbb{R}_+^n \middle| 0 \leq \lambda_i \leq \frac{J(u') - \mu}{-g_i(u')} \right\}, \qquad \forall i = 1, \cdots, n \quad (5)$$

- $\lambda_i \in \mathbb{R}_+^n$ in the primal and dual problems can be replaced by $\lambda_i \in \mathbb{I}_i$.

Distributed Algorithms

- According to Koshal et.al (2011), a necessary and sufficient optimality condition for (A) is that (u_*, λ_*) gives the solution to the following system

$$u_{*,i} = \prod_{\chi_i}\left(u_{*,i} - \xi \nabla_{u_i}\mathcal{L}(u_*, \lambda_*)\right), \qquad \lambda_{*,i} = \prod_{\mathbb{I}_i}\left(\lambda_{*,i} + \theta g_i(u_*)\right), \qquad \forall i = 1, \cdots, n,$$

where

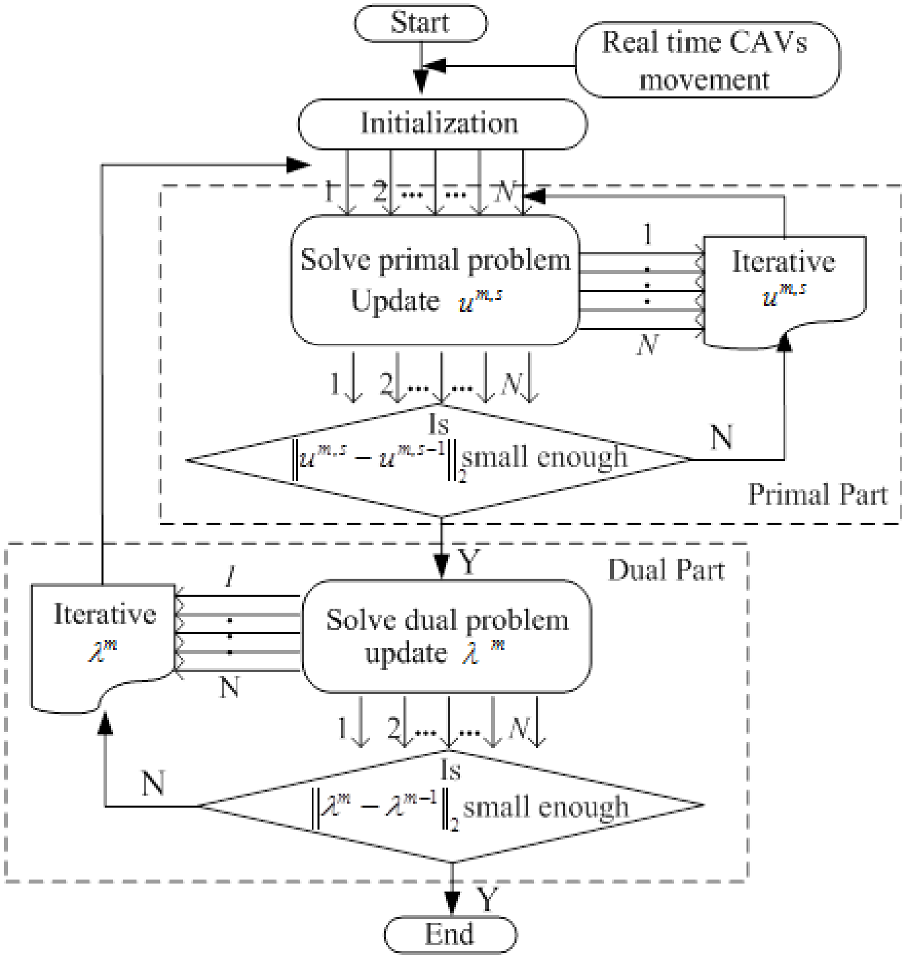
  - χ_i or $\mathbb{I}_i$ is an interval constraint of the form $[a_i, b_i]$
  - Then for any $z \in \mathbb{R}$, the Euclidean projection Π is shown as following

$$\prod_{[a_i, b_i]}(z) = \begin{cases} b_i, & if\ z \geq b_i \\ z, & if\ z \in [a_i, b_i] \\ a_i & if\ z \leq a_i \end{cases}$$

Iteratively solve λ and u until converge

- Euclidean project is decoupled; can be computed in a decentralized manner.
- Both dual and primal-dual based distributed algorithms are discussed for (A).
- The dual based regularized distributed algorithm was selected due to its better computational performance.

Dual based Regularized Distributed Algorithm



Repeat until converge

- Each CAV iteratively solves its own primal and dual variables
- Share its temporary decision

Perform two gradient projection algorithms in a distributed manner

- Given λ^m and u^m, update $u_i^{m,s+1}$

$$u_i^{m,s+1} = \prod_{\chi_i}\left(u_i^{m,s} - \xi\nabla_{u_i}\mathcal{L}(u^{m,s},\lambda^m)\right)$$

- Given u^{m+1} and λ^m, update λ^{m+1}

$$\lambda_i^{m+1} = \prod_{\mathbb{I}_i}(\lambda_i^m + \theta[g_i(u^{m+1}) - \varepsilon\lambda_i^m])$$

- *ξ and θ are the step lengths; their values affect the convergence.*
- *$\varepsilon\lambda_i^m$ is the regularization term to remove degenerate cases*
- *Please refer to our paper for technical details.*

Linear Stability Analysis

- Focus on the stability analysis where all the constraints are inactive.
- Control system under inactive constraints

$$z(k+1) = z(k) + \tau z'(k) + \frac{\tau^2}{2}\omega(k), \quad z'(k+1) = z'(k) + \tau\omega(k), \text{ where} \quad (1)$$

  - $z(k) = (x_0 - x_1 - \Delta, \ldots x_{n-1} - x_n - \Delta)^T$(k) : spacing error (output at k)
  - $z'(k) = (v_0 - v_1, \ldots v_{n-1} - v_n)^T$(k): relative speed (output at k)
  - $\omega(k) = (u_0 - u_1, \ldots u_{n-1} - u_n)^T$(k): interactive control decision (input at k)
- $\omega(k)$ is the optimal solution of (A) and it is linear in $(z(k);\ z'(k))$
  - A linear closed-loop dynamics is given below

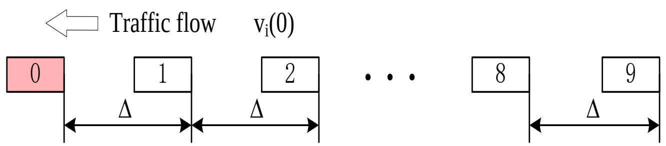
$$\begin{bmatrix} z(k+1) \\ z'(k+1) \end{bmatrix} = A(\alpha,\beta,\tau)\begin{bmatrix} z(k) \\ z'(k) \end{bmatrix} + \begin{bmatrix} \frac{\tau^2}{2} I_n \\ \tau I_n \end{bmatrix} W(\alpha;\beta;\tau)\mathbf{1}\, u_0(k), \text{ wherer } \mathbf{1} \text{ is a vector} \quad (2)$$

- **Linear Stability**: For any positive numbers τ, and α_i, β_i for each $i = 1, \ldots, n$, $A(\alpha,\beta,\tau)$ is Schur stable (i.e., each eigenvalue satisfies $|\mu| < 1$) such that the linear closed-loop system is asymptotically stable as $u_0(k) \rightarrow 0$.
- **Choice of Weights**: recommend formulations such that α_i, β_i to be the order of n^2, and $\beta_i \geq 4\tau^2/\alpha_i$ to ensure fast dynamic response and mild input.

Please refer our paper for the choice of penalty weights based on linear stability results.

Numerical Experiment

Test Platoon



- 10 autonomous vehicle platoon. One leading vehicle (n=0) and nine following vehicles (n=i,…,9).
- Input data: the desired spacing (50m), the acceleration (1.35m/s) and deceleration limits (-8m/s), speed limit, sample time (1s or 0.5s).

Three scenarios are tested:

- Scenario 1, leading vehicle performs instantaneous deceleration\acceleration and keeps a constant speed for a while.
- Scenario 2, leading vehicle performs periodical acceleration\deceleration.
- Scenario 3, using real world trajectory data from an oscillating traffic flow.

Objective:

- Test the computation performance of the distributed algorithm.
- Test the performance of the proposed control scheme.
- Compare the platoon car-following control to a CACC in literature.

Numerical Experiment

I: Examining the Computational Performance:

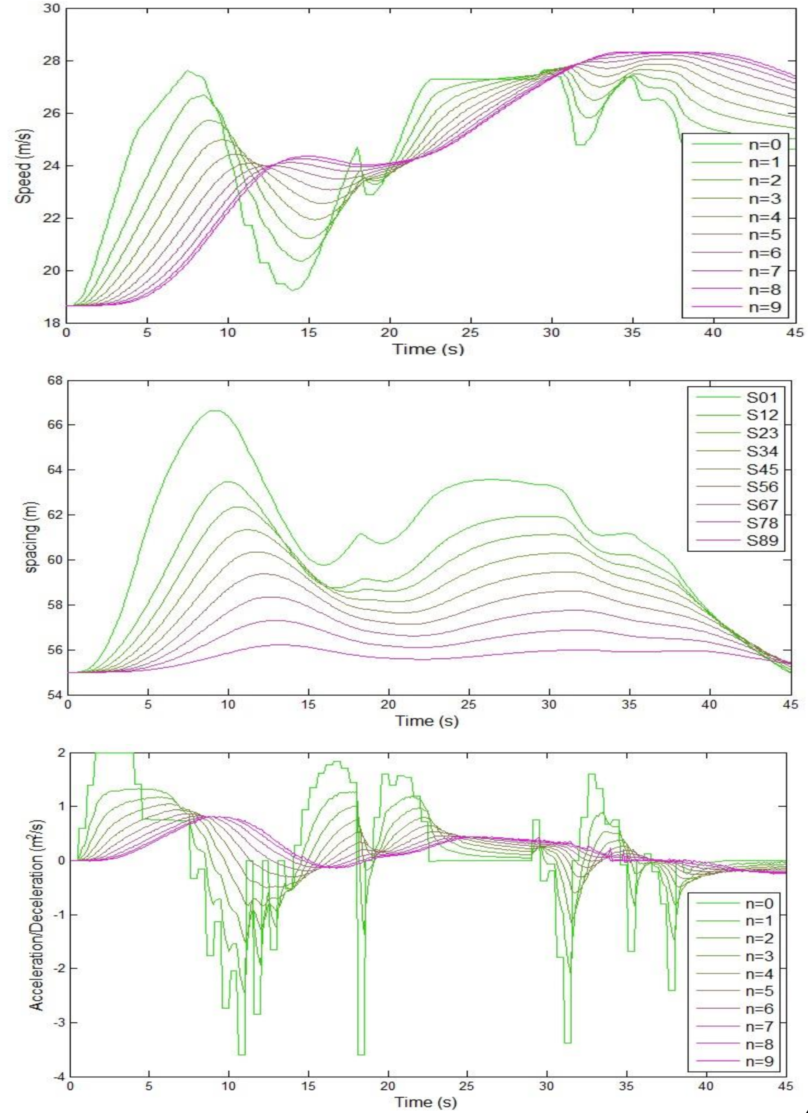
Scenarios	Computation time (s)		The number of iterations	
	Mean	Variance	Mean	Variance
1	0.0115	0.000388	297.91	0.9595
2	0.0114	0.000361	297.93	0.4540
3	0.0047	0.000390	109.34	1.1334

- The mean convergence time for each scenarios is very short with a small variance. The number of iterations showed the similar observations.
- The distributed algorithm converges quickly and it satisfies the online applications.

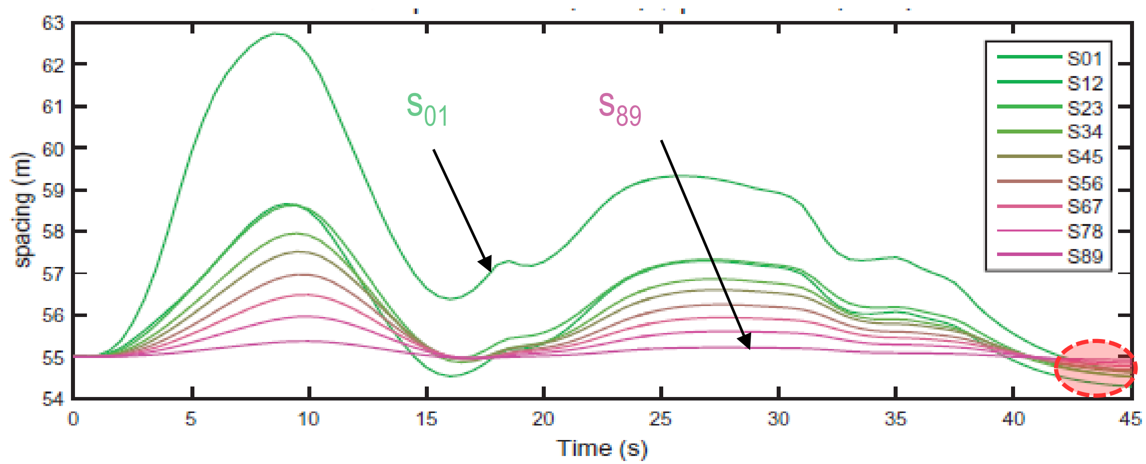
Numerical Experiment

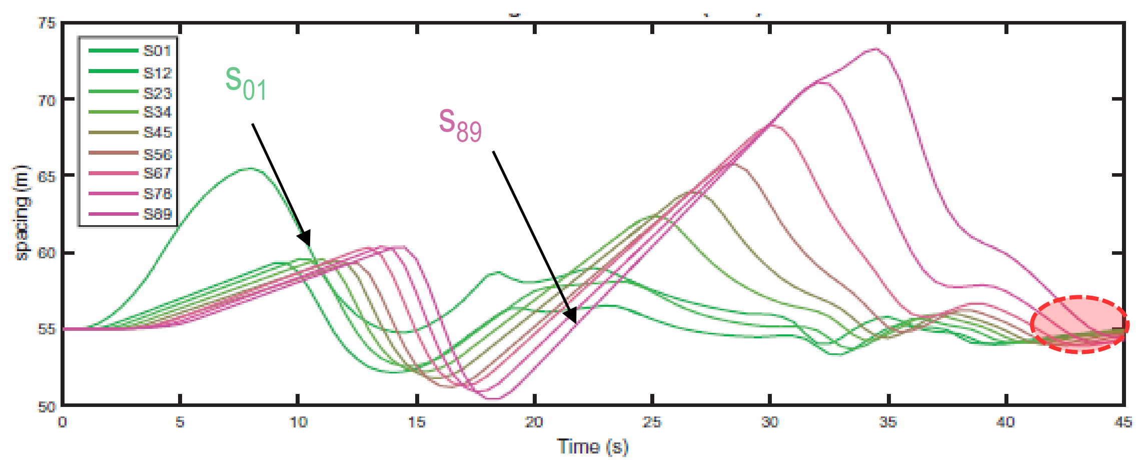
II. Key Observations for Scenario 3

- The movement of the leading vehicle shows a slow-and-fast traffic state
- The proposed car-following control help keep traffic stability and dampen traffic oscillation along a platoon.
  - Dampen the propagation of speed fluctuation along the platoon.
  - Decreases the propagation of spacing variation along the platoon.
  - Smoothen control inputs (acceleration/deceleration) along the platoon.



III. Comparing with a CACC

<table>
<tr>
<td>The platoon car-following control</td>
<td>


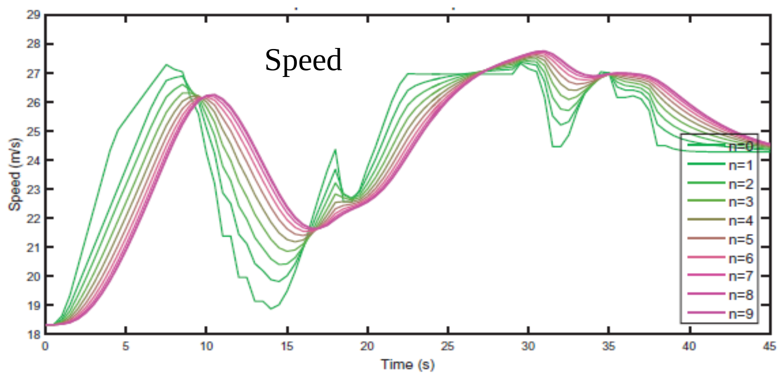
</td>
</tr>
<tr>
<td>CACC in (Schakel et al., 2010)</td>
<td>


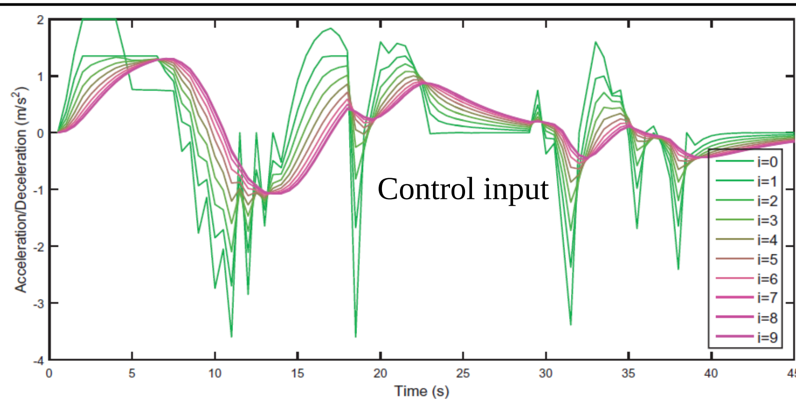
</td>
</tr>
</table>

- Both schemes render the vehicles back to the desired spacing eventually
- The transient dynamics under the platoon control is more stable

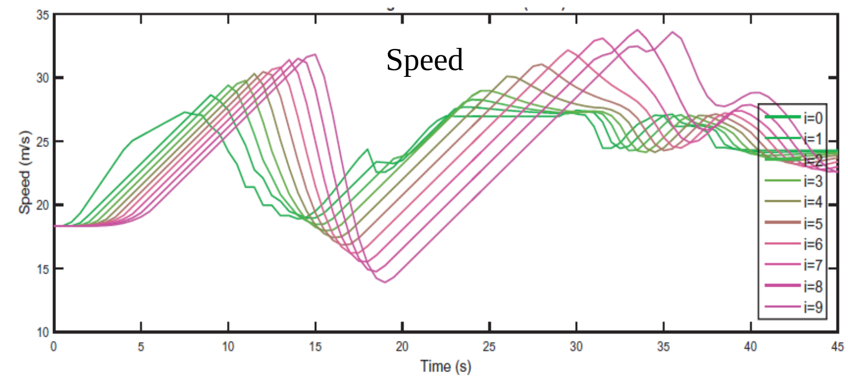
III. Comparing with a CACC mechanism

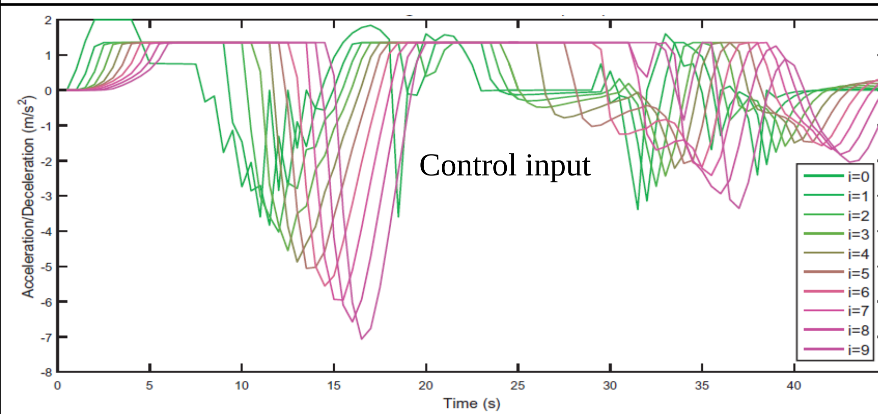
The platoon car-following control





CACC (Schakel et al., 2010)

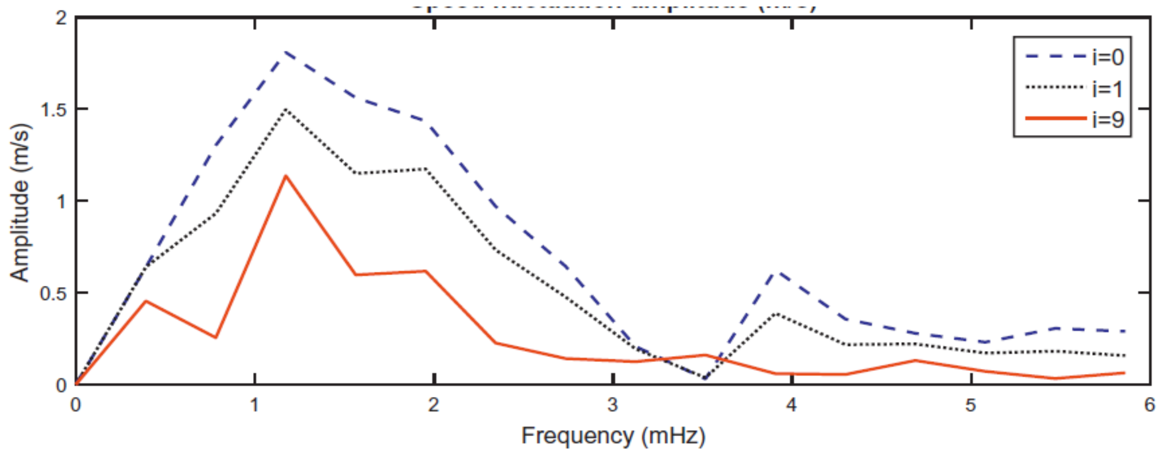




- Similar observation can be obtained from the speed and control input responses

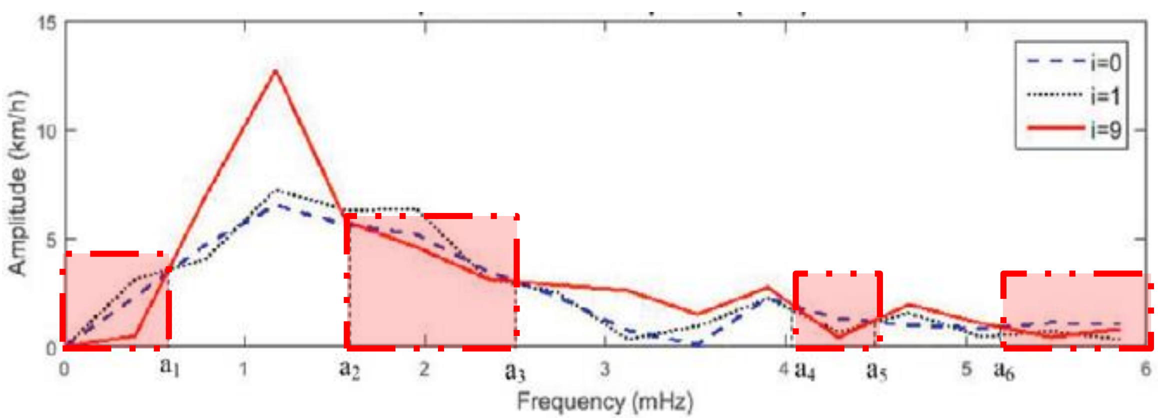
III. Comparing a CACC mechanism

(a) Platoon under the proposed control scheme



- The proposed control scheme reduces speed fluctuation at almost all frequencies

(b) Platoon under CACC (Schakel et al., 2010) scheme



- The CACC (Schakel et al., 2010) can reduce speed fluctuation under certain frequencies

Summary

- This paper develops a novel platoon car-following control scheme based on constrained optimization and distributed computation.
  - Consider a platoon of connected and autonomous vehicles
  - Model it as an interconnected dynamic system subject to acceleration, speed, and safety distance constraints, under the global information structure.
  - Develop a constrained optimization problem to achieve desired multiple platoon performance objectives arising from the transient and asymptotic dynamics
  - Develop dual or primal-dual based distributed algorithms to implement the control algorithm using the special properties and structure of the optimizer.
  - Study the stability of the proposed control scheme, particularly for the unconstrained linear closed-loop system which is shown to be asymptotically stable.
- This study conduct numerical experiments based on field data to demonstrate the proposed platoon control scheme.
  - It effectively reduces the propagation of traffic fluctuation/oscillation along a platoon
  - It outperforms the conventional cooperative cruise control.

Thank You Very Much!
Questions?

ldu3@iit.edu
Illinois Institute of Technology

Acknowledgement: this research is partially supported by NSF awards CMMI-1436786 and CMMI-1554559