Constrained Control of Multi-Vehicle Systems for Smart Cities and Industry 4.0: from Model Predictive Control to Reinforcement Learning

Part 2: A Receding-Horizon Collision Avoidance Strategy for Constrained UGV

Speaker: Dr. Walter Lucia

UNIVERSITÉ Concordia UNIVERSITY

2023 IEEE Intelligent Vehicle Symposium

Outline

1. Considered Scenario: vehicles moving in a shared environment
2. Problem Formulation: collision avoidance problem
3. Proposed Solution
4. Simulation Examples
5. Conclusions and Future Works
   - Some recent advancements and work in progress

Controlling autonomous vehicles using MPC

- MPC-based solutions have been proposed to solve control problems related to ground vehicles.
- MPC can handle nonlinear/linearized/feedback linearized vehicle models
  - **Nonlinear models** leads to non-convex MPC formulations. However, resulting strategies suffer from high computational complexity and local minima.
  - **Linearized models** leads to computationally appealing convex MPC formulations. However, the used model for prediction might be inaccurate and the resulting strategy suboptimal.
  - **Feedback linearized models** are more accurate than linearized ones. However, they lead to convex MPC only in the absence of constraints. Indeed, input constraints (e.g., velocity constraints) on the feedback linearized model are time-varying. However, invariant inner approximations can be found for convex MPC formulations.

Considered Robot Model

- In the sequel, we model each robot as a constrained discrete-time double integrator subject to bounded disturbances

$$x_i(t+1) = Ax_i(t) + Bu_i(t) + d_i(t)$$

$$A = \begin{bmatrix} I_2 & T_s I_2 \\ 0_2 & I_2 \end{bmatrix}, \ B = \begin{bmatrix} \frac{T_s^2 I_2}{2} \\ T_s I_2 \end{bmatrix}$$

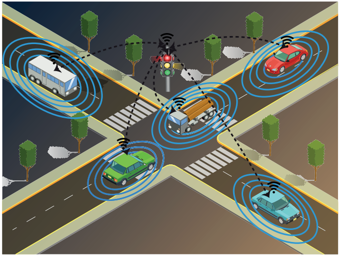
  where $x_i = [p_i^T, v_i^T]^T \in \mathbb{R}^4$ is the state-space vector
- Each robot is subject to a bounded disturbance

$$d_i(t) \in \mathcal{D}_i \subset \mathbb{R}^4$$

  and convex state (for the velocities) and input (for the accelerations) polyhedral constraints:

$$x_i(t) \in \mathcal{X} := \mathbb{R}^2 \times \mathcal{V}, \ \forall t \geq 0, \quad u_i(t) \in \mathcal{U}_i \subset \mathbb{R}^2$$

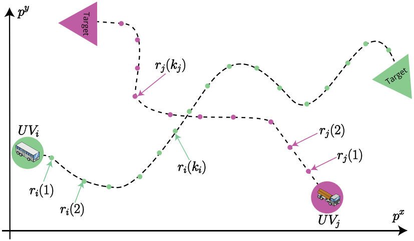
Scenario



[https://epthinktank.eu]

- A set of heterogeneous unmanned vehicles moving in a shared planar environment.
- Each vehicle follows its own independent trajectory. For privacy reasons, inter-vehicle communications are not possible
- Collisions are possible at the intersection points if a traffic manager (e.g. traffic lights) is not used.

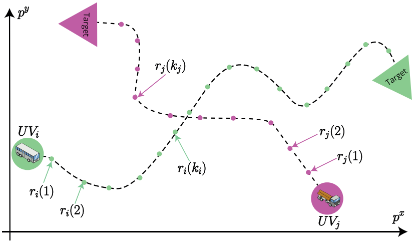
Assumption



- Each vehicle is equipped with a reference generator providing waypoints r_i such that $||r_i(k_i+1) - r_i(k_i)||_2 \leq \delta_i$
- Let $r_i(k_i)$ the current waypoint for the $i-th$ vehicle, a finite preview of waypoints is available (assuming a limited vision radius), i.e.,

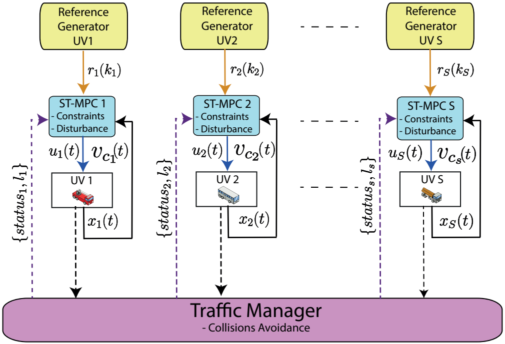
$$R(k_i, H_i) := \{r_i(k_i),\, r_i(k_i+1), \ldots, r_i(k_i+H_i)\}, \quad H_i > 0$$

Control Problem



- Regardless of **disturbance** realization, input/ state **constraints** and **reference trajectories**, the UVs must be able to:
  - **(O1)** Sequentially track the given **waypoints**
  - **(O2) Avoid collisions** while minimizing the number of stop occurrences along the path.

Proposed Control Architecture



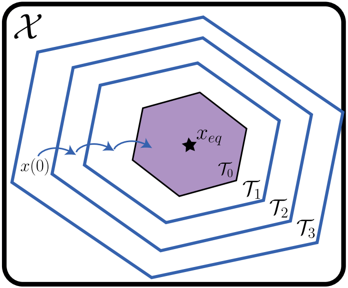
- A set of “S” decentralized waypoint reference tracking controllers
- A centralized Traffic Manager (**TM**) in charge of ensuring **absence of collisions** while **minimizing the number of vehicle’s stops**

ST-MPC [*Angeli et al., Aut., 08*] - single waypoint

Offline Operations

1. Design $u_i(t) = K_i^0(x_i(t) - x^{eq}_{r_i(0)})$ and find the associated smallest **RCI** region $\mathcal{T}_i^0(r_i(0))$ using, e.g., [*Rakovic et al., TAC, 05*].
2. Build a family of robust one-step controllable sets $\{\mathcal{T}_i^{n_i}(r_i(0))\}_{n_i=1}^{N_i}$ until the desired state-space region is covered

$$\mathcal{T}_{i+1} = \{x \in \mathcal{X} : \exists u \in \mathcal{U} \; s.t. \; x^+ \in \mathcal{T}_i, \forall d \in \mathcal{D}\}$$



Online Operations

1. At each t, **Find** the minimum $n_i(t)$ such that $x_i(t) \in \mathcal{T}_i^{n_i(t)}(r_i(0))$
2. **If** $n_i(t) > 0$ then

$$u(k) = \arg\min ||A_i x_i(t) + B_i u_i - x^{eq}_{r_i(0)}||_2^2, \; s.t.$$
$$A_i x_i(t) + B_i u \in \mathcal{T}_i^{n_i(t)-1}(r_i(0)), \; u_i \in \mathcal{U}_i$$

Else $u(t) = K_i^0(x(t) - x^{eq}_{r_i(0)})$

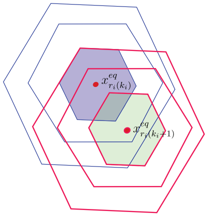
ST-MPC - tracking a sequence of waypoints

Remarks:

- Any point of the plane is an equilibrium point for zero acceleration and zero velocity.
- The state constraints are only on the velocity components.

- $\{\mathcal{T}_i^{n_i}(r_i(0))\}_{n_i=1}^{N_i}$ can be shifted (re-centered) to any equilibrium
- $x^{eq}_{r_i(k_i)} \rightarrow x^{eq}_{r_i(k_i+1)}$ is admissible from the terminal region of the current waypoint if [*Bagherzadeh et al., TAC, 21*]:

$$\bigcup_{n_i=0}^{N_i} \mathcal{T}_i^{n_i}(r_i(0)) \supseteq \mathcal{B}_{\delta_i}(r_i(0)) \oplus \mathcal{T}_i^0(0_2)$$



ST-MPC - waypoint tracking controller

[*Bagherzadeh et al., TAC, 21*]

1: Compute $n_i(t) := \min_{0 \le n_i \le N_i} n_i : x_i(t) \in \mathcal{T}_i^{n_i}(r_i(k_i))$

2: **if** $n_i(t) == 0$ and the successive waypoint exists and it is enabled, **then** $k_i \leftarrow k_i + 1$, goto Step 1

3: **end if**

4: $x^{eq} \leftarrow x^{eq}_{r_i(k_i)}$

5: **if** $n_i(t) == 0$ **then** $u_i(t) = K_i^0(x_i(t) - x^{eq}_{r_i(k_i)})$

6: **else** Find $u_i(t)$ by solving the following optimization problem

$$u_i(t) = \arg\min_{u_i} \|Ax_i(t) + Bu_i - x^{eq}_{p_i}\|_2^2 \quad s.t.$$
$$Ax_i(t) + Bu_i \in \tilde{\mathcal{T}}_i^{n_i(t)-1}(r_i(k_i)),\ u_i(t) \in \mathcal{U}_i$$

7: **end if**

8: $t \leftarrow t + 1$ and goto Step 1

- **Remark:** if necessary, a vehicle can be confined in the terminal region associated with the current waypoint $r_i(k_i)$, by disabling Step 2

Assumption 2 - ST-MPC feasibility under the TM operations

- Given the desired TM operations, we make the following assumption
  - **Assumption**: A finite set of $L > 1$ different velocity constraints can be imposed for each vehicle:

$$V = \{\mathcal{V}_1, \mathcal{V}_2 \ldots, \mathcal{V}_L\}, \text{ with } \mathcal{V}_1 \subset \mathcal{V}_2 \subset \ldots \subset \mathcal{V}_L \equiv \mathcal{V}$$

- Therefore, each vehicle might be subject to time-varying state constraints that will be imposed by the TM

$$x_i(k) \in \mathcal{X}_{l_i} := \mathbb{R}^2 \times \mathcal{V}_{l_i},\ l_i \in \{1, \ldots, L\}$$

The ST-MPC operations must be revised to ensure recursive feasibility

ST-MPC with time-varying velocity constraints

- Offline:
  1. The RCI region $\mathcal{T}_i^0(r_i(0))$ is built using the worst-case velocity constraint scenario $x_i(t) \in \mathcal{X}_1$.
  2. $\forall \mathcal{V}_{l_i} \in V$, a family of $N_{(i,l)}$ robust one-step controllable sets $\{\mathcal{T}_{(i,l)}^{n_i}(r_i(0))\}_{n_i=1}^{N_{(i,l)}}$ is built from the same $\mathcal{T}_i^0(r_i(0))$
- Online:
  1. constraint switches $\mathcal{X}_{l_1} \to \mathcal{X}_{l_2}$, $l_1, l_2 \leq L$ are enabled when $x_i(t)$ enters the domain of attraction of the controller using $\mathcal{X}_{l_2}$, i.e.,

$$x_i(t) \in \bigcup_{n=0}^{N} \{\mathcal{T}_{(i,l_2)}^{n}(r_i(0))\}_{n=1}^{N_{(i,l_2)}}$$

- **Remark** In the worst-case scenario, the switch happens when $\mathcal{T}_i^0(r_i(0))$ is reached

Data sent by the vehicle to the Traffic Manager

Offline:

- Each i^{th} UV sends $\{\mathcal{T}_i^{n_i}(r_i(0))\}_{n_i=0}^{N_i} = \{\mathcal{T}_{(i,L)}^{n_i}(r_i(0))\}_{n_i=0}^{N_i}$ and the associated controller domain $DoA^{N_i}(r_i(0))$.

Online:

- Each i^{th} UV sends the predicted waypoints $R(k_i, H_i)$ and the current set-membership index $n_i(t)$

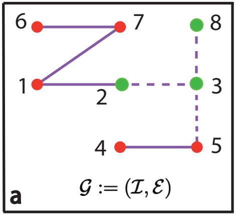
Traffic Manager Operations

- TM operations:
  1. Impose the vehicle's velocity constraints $\mathcal{V}_{l_i}$ to minimize the possibility of collisions in the future.
  2. Ensure the absence of collisions, by stopping, whenever strictly necessary, the minimal subset of vehicles.
- We model the TM operations as

$$\{status_i, l_i\}_{i=1}^{S} = TM\left(\{R(k_i, H_i), n_i(t)\}_{i=1}^{S}\right)$$

  - $status_i = \{go, stop\}$
  - l_i the velocity constraint $\mathcal{V}_{l_i} \in \{\mathcal{V}_1, \ldots, \mathcal{V}_L\}$ to be used by the vehicles

TM-Step 1: Collision Graph



- TM computes an undirected connectivity graph $\mathcal{G} := (\mathcal{I}, \mathcal{E})$, which characterizes:
  - Collisions between vehicles in the waypoint prediction horizons (solid lines)

$$DoA^{N_i}(r_i(k_i+\bar{h}_i)) \cap DoA^{N_j}(r_j(k_i+\bar{h}_j)) \neq 0$$

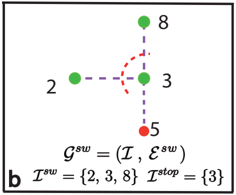
  - Potential collisions with the current waypoint (dashed lines)

$$\bigcup_{n_i=0}^{\max(n_i(t)-1,0)} \{\mathcal{T}_i^{n_i}(r_i(k_i))\} \bigcap \bigcup_{n_j=0}^{\max(n_j(t)-1,0)} \{\mathcal{T}_j^{n_j}(r_j(k_j))\} \neq 0$$

- In a receding horizon fashion, $\forall\, t > 0\ \mathcal{G}$, is partially updated [*Savehshemshaki et al., CDC, 22*]

TM-Step 2: Vehicles to be Stopped

- Let $\mathcal{G}^{sw} = (\mathcal{I}, \mathcal{E}^{sw})$ be the sub-graph containing imminent collisions, and $\mathcal{I}^{sw} \subset \mathcal{I}$ the vehicles that want to go to the next waypoint (green circles)



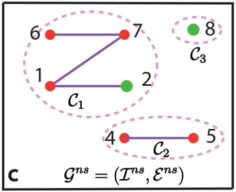
- **Remark:** Only the vehicles that just updated the desired waypoint can be stopped (i.e., vehicles in the terminal region of the current waypoint)

- TM stops the minimum number of UVs to avoid collisions. Iterative procedure [*Bagherzadeh et al., TAC, 21*]:[1]
  1. Find the vehicle $i \in \mathcal{I}^{sw}$ with the highest degree,
  2. Add i to $\mathcal{I}^{stop}$ and remove i from $\mathcal{G}^{sw}$
  3. If $\exists\, e^{sw}_{(i,j)}(t) \in \mathcal{E}^{sw}(t) \,:\, e^{sw}_{(ij)}(t) \neq 0$, then *goto* Step 1, else *stop* the procedure.

[1] This problem is also known as a vertex cover problem in a graph. An Integer Linear optimization problem can be formulated.

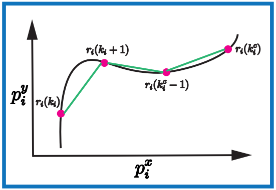
TM-Step 3: Future Collision Minimization - I

- Consider the graph of not stopped vehicles $\mathcal{G}^{ns} = (\mathcal{I}^{ns}, \mathcal{E}^{ns})$, with $\mathcal{I}^{ns} = \mathcal{I} \setminus \mathcal{I}^{stop}$



- Then, the TM:
  1. Finds all the connected components $\{\mathcal{C}_z\}_{z=1}^{Z}$, where $\mathcal{C}_z \subseteq \mathcal{I}^{ns}$,
  2. Computes, for each $i \in \mathcal{C}_z$, $|\mathcal{C}_z| > 1$, the distance to the closest collision waypoint k_i^c in the connected component

$$d_i^c = \sum_{h=k_i}^{k_i^c - 1} \|p_i(h+1) - p_i(h)\|_2, \quad i \in \mathcal{C}_z$$



TM-Step 3: Future Collision Minimization - II

3. Collects the computed distances d_i^c in an ordered vector

$$d^c = [d^c_{v_{c_1}}, \dots, d^c_{v_{c_z}}]$$

   - v_{c_1} is the vehicle at the maximum distance from a collision
   - v_{c_z} is the vehicle at the minimum distance from a collision

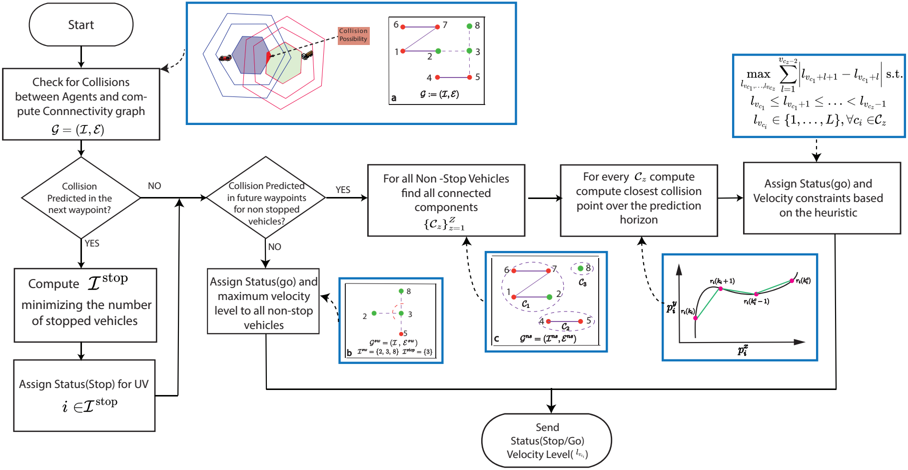
4. **Heuristic:** the vehicles closer to collision points use higher velocities

$$\begin{matrix} v_{c_1} \rightarrow l_{v_{c_1}} \\ \vdots \\ v_{c_z} \rightarrow l_{v_z} \end{matrix}, \; s.t. \; \begin{matrix} l_{v_{c_1}} \leq l_{v_{c_1}+1} \leq \dots < l_{v_{c_z}} \\ l_{v_{c_i}} \in \{1, \dots, L\}, \forall c_i \in \mathcal{C}_z \end{matrix}$$

5. An optimization is defined to maximize the difference in velocities between the vehicles in the same connected component

$$\max_{l_{v_{c_1}}, \dots, l_{v_{c_z}}} \sum_{l=1}^{v_{c_z-1}} \left| l_{v_{c_1}+l+1} - l_{v_{c_1}+l} \right| s.t.$$
$$l_{v_{c_1}} \leq l_{v_{c_1}+1} \leq \dots < l_{v_{c_z}}$$
$$l_{v_{c_i}} \in \{1, \dots, L\}, \forall c_i \in \mathcal{C}_z$$

Summary of the TM operations



Simulation Example 1: Setup - I

- 10 UVs, i.e. $\mathcal{I} = \{1, \ldots, 10\}$.
- The UVs dynamics are described by a discrete-time double integrator with $T_s = 0.1\,\mathrm{sec}$
- The disturbance set and the constraints are the followings:

$$|v_i^x| = |v_i^y| \leq \bar{v}_i, \;\; |u_i^x| = |u_i^y| \leq \bar{u}_i,$$
$$\mathcal{D}_i = \{d = [d_1, \ldots, d_4]^T : |d_j| \leq \bar{d}_i, \; j = 1, \ldots, 4\}$$

  where
  - $\forall i \rightarrow \bar{u}_i = 4$
  - $i \in \{1, 2, 3\} \rightarrow \bar{v}_i = 20, \; \bar{d}_i = 0.06$
  - $i \in \{4, 5\} \rightarrow \bar{v}_i = 25, \; \bar{d}_i = 0.085$
  - $i \in \{6, 9, 10\} \rightarrow \bar{v}_i = 8, \; \bar{d}_i = 0.07$
  - $i \in \{7, 8\} \rightarrow \bar{v}_i = 18, \; \bar{d}_i = 0.065$
- $L = 1$, i.e., no restriction on the velocity can be imposed by the TM
- The waypoint prediction horizon is $H_i = 1, \forall\, i$

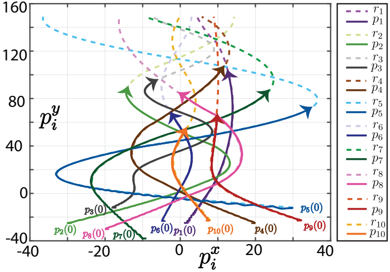
Simulation Example 1: Setup - II

- The waypoint max distances are:
  - $i \in \{1, 2, 3\} \rightarrow \delta_i = 4.02$
  - $i \in \{4, 5\} \rightarrow \delta_i = 5.62$
  - $i \in \{6, 9, 10\} \rightarrow \delta_i = 3.22$
  - $i \in \{7, 8\} \rightarrow \delta_i = 3.91$
- a family of N_i robust controllable sets $\{\mathcal{T}_i^l\}_{l=1}^{N_i}$ has been computed for each vehicle
  - $i \in \{1, 2, 3\} \rightarrow N_i = 21$
  - $i \in \{4, 5\} \rightarrow N_i = 19$
  - $i \in \{6, 9, 10\} \rightarrow N_i = 15$
  - $i \in \{7, 8\} \rightarrow N_i = 18$.

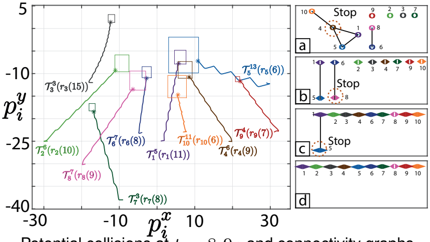
  to ensure that the waypoint switches are admissible

Simulation Example- 10 Vehicles

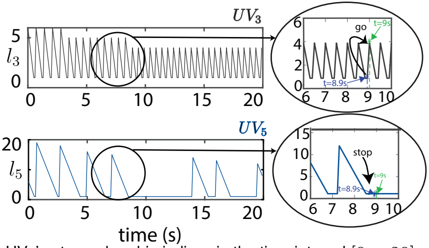
- At $t = 8.9s$:
  - $\mathcal{I}_{sw} = \{1, 2, 3, 4, 5, 8, 9\}$
  - $6, 7, 10$ cannot be stopped
  - $\mathcal{I}_{stop} = \{4, 5, 8\}$



UV Waypoints and trajectories for $t \in [0,\ 100]s$.



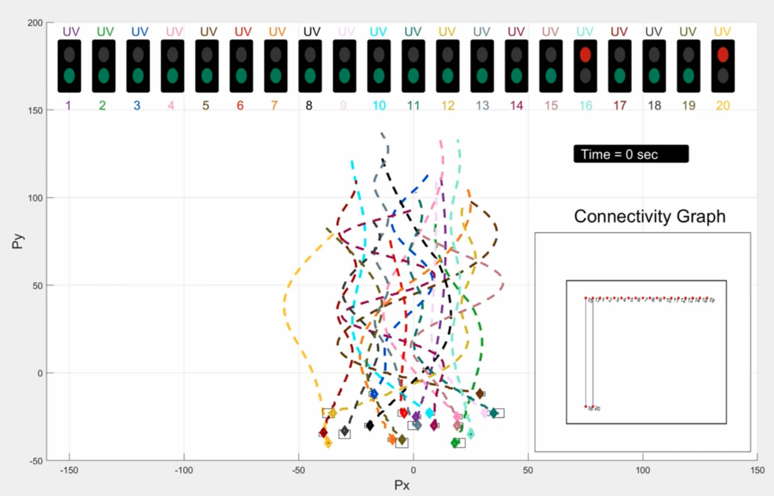
Potential collisions at $t = 8.9s$ and connectivity graphs.



UVs' set-membership indices in the time interval $[0 - 20]s$.

Simulation Example - 20 vehicles

- Performed demo for 20 vehicles: `https://youtu.be/SCeoJyle_VU`



Simulation Example 2: Setup

- Two UVs, i.e. $\mathcal{I} = \{1, 2\}$.
- The UVs dynamics are described by a discrete-time double integrator with $T_s = 0.1\,\text{sec}$ and subject to the disturbance sets

$$
\begin{aligned}
d_1(t) \in \mathcal{D}_1 = \{d \in \mathbb{R}^4 : |d(s)| \leq 7 \times 10^{-4},\ s = 1, \ldots, 4\} \\
d_2(t) \in \mathcal{D}_2 = \{d \in \mathbb{R}^4 : |d(s)| \leq 9 \times 10^{-4},\ s = 1, \ldots, 4\}
\end{aligned}
$$

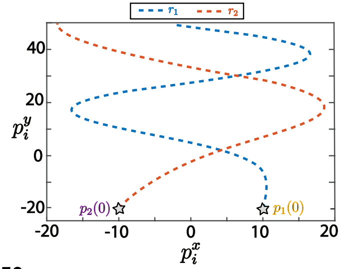
and constraints

$$
\begin{gathered}
\mathcal{V} = \{v \in \mathbb{R}^2 : |v(s)| \leq 5,\ s = 1,\ 2\} \\
u_1(t) \in \mathcal{U}_1 = \{u \in \mathbb{R}^2 : |u(s)| \leq 8,\ s = 1,\ 2\} \\
u_2(t) \in \mathcal{U}_2 = \{u \in \mathbb{R}^2 : |u(s)| \leq 7,\ s = 1,\ 2\}
\end{gathered}
$$

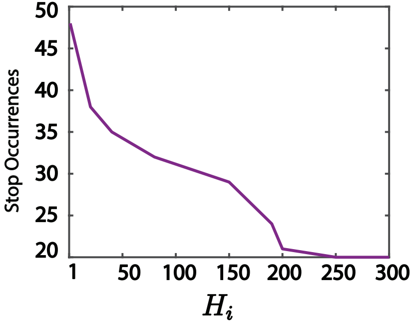
- $L = 3$ different velocity constraints levels are considered, i.e. $V = \{\mathcal{V}_1, \mathcal{V}_2, \mathcal{V}_3\}$:

$$
\begin{gathered}
\mathcal{V}_1 = \{v \in \mathbb{R}^2 : |v_i(s)| \leq 0.8,\ s = 1, 2\} \\
\mathcal{V}_2 = \{v \in \mathbb{R}^2 : |v_i(s)| \leq 2,\ s = 1, 2\},\ \ \mathcal{V}_3 \equiv \mathcal{V}
\end{gathered}
$$

1. Each vehicle follows an independent trajectory

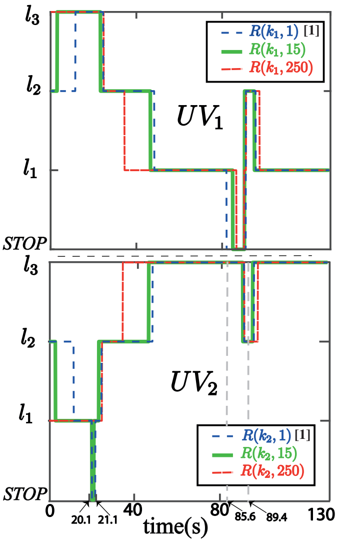


2. The number of stops imposed by the TM has been evaluated for different waypoint prediction horizons $H_i \in [1, 300]$



Simulation Results - II

1. Time-varying velocity constraints imposed by the TM for different waypoint horizon H_i for $t \in [0,\ 130]\ \mathrm{sec}$
2. In the figure, [1] refers to [*Bagherzadeh et al., TAC, 21*]



Conclusions and Future Works

Conclusions: control strategy addressing the collision avoidance problem for multi-unmanned vehicles moving in a shared environment

1. ST-MPC local controllers address reference tracking problem
2. Collision avoidance problem is solved using a centralized unit via “set- membership checks”.
3. The solution is capable of exploiting a prediction horizon on the waypoints to minimize the number of vehicles’ stops required to avoid collisions.

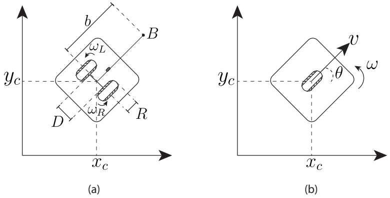
Future Works:

1. Solve a the collision avoidance in a distributed fashion.
2. Enhance the traffic manager using different collision risk indices (e.g., time to collisions) and priorities among vehicles.
3. Work with more complete (nonlinear) model of UVs such as differential-drive robot or car-like vehicles.

Some recent advancements and work in progress

Wheeled mobile robot - popular configurations - I

- Differential-Drive (DD) and unicycle (U) robots:



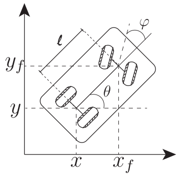
(a) differential-drive, *(b)* unicycle.

$$\begin{aligned} \dot{x}_c(t) &= \frac{R}{2}(\omega_R(t)+\omega_L(t))\cos\theta(t) \\ \dot{y}_c(t) &= \frac{R}{2}(\omega_R(t)+\omega_L(t))\sin\theta(t) \\ \dot{\theta}(t) &= \frac{R}{D}(\omega_R(t)-\omega_L(t)) \end{aligned} \quad , \quad \begin{aligned} \dot{x}_c(t) &= v(t)\cos(\theta(t)) \\ \dot{y}_c(t) &= v(t)\sin(\theta(t)) \\ \dot{\theta}(t) &= \omega(t) \end{aligned}$$

A DD robot model can be equivalently re-written as an unicycle one. Moreover, the U robot model is feedback linearizable [*De Luca et al., RMPC, 02*].

Wheeled mobile robot - popular configurations - II

- Car-like model (assuming rear driving wheels):

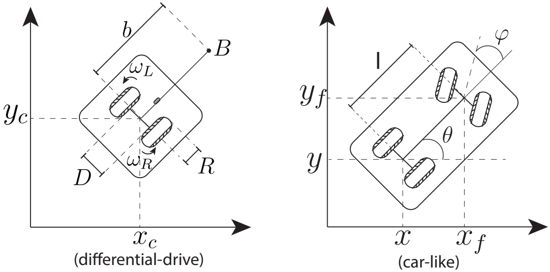


Car-like vehicle

$$\dot{q}_c(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \\ \dot{\varphi}(t) \end{bmatrix} = \begin{bmatrix} \cos\theta(t) \\ \sin\theta(t) \\ \frac{1}{l}\tan(\varphi(t)) \\ 0 \end{bmatrix} v(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \omega(t)$$

As shown in [*De Luca et al., RMPC, 05*], the car-like model can be linearized via an input-output feedback linearization.

Advancement 1: constraints dealing in feedback-linearized vehicle models - I

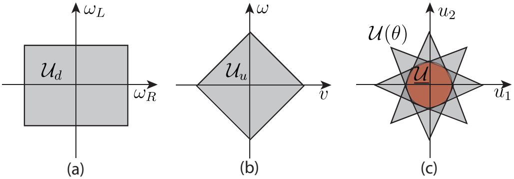


Using feedback-linearization, both the DD and Car-like models become linear. However, the input constraints become nonlinear and state-dependent, leading still to nonlinear MPC formulations

Advancement 1: constraints dealing in feedback-linearized vehicle models - II

- Considering the DD/U model. Under feedback linearization:

$$z(k+1) = Az(k) + Bu(k),\; A = I_{2\times 2},\; B = T_s I_{2\times 2},\; T_s > 0$$
$$u(k) \in \mathcal{U}(\theta) = \{[u_1, u_2]^T \in \mathbb{R}^2 :\; H(\theta)\,[u_1, u_2]^T \leq 1\}$$



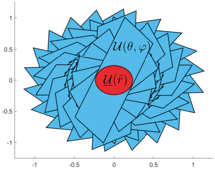
- In [*Tiriolo et al., TCST, 22*], it has been proved that $\mathcal{U}(\theta)$ admits the following worst-case circular approximation:

$$\underline{\mathcal{U}}(r_u) = \bigcap_{\forall\theta} \mathcal{U}(\theta) = \{u \in \mathbb{R}^2\, | u^T u \leq r_u^2\},\;\; r_u = \frac{2\bar{\Omega} R b}{\sqrt{4b^2 + D^2}}$$

Advancement 1: constraints dealing in feedback-linearized vehicle models - III

- Considering the car-like model. Under feedback linearization:

$$z(k+1) = Az(k) + Bu(k),\ A = I_{2\times 2},\ B = T_s I_{2\times 2},\ T_s > 0$$
$$u(k) \in \mathcal{U}(\theta,\varphi) = \{[u_1,u_2]^T \in \mathbb{R}^2 :\ L(\theta,\varphi)\,[u_1,u_2]^T \le 1\}$$



- In [*Tiriolo et al., L-CSS 23*], it has been proven that the time-varying polyhedron admits the following worst-case circular approximation:

$$\underline{\mathcal{U}}(\hat{r}) = \bigcap_{\forall\theta,\forall\varphi} \mathcal{U}(\theta,\varphi) = \{u \in \mathbb{R}^2\,|u^T u \le \hat{r}^2\},\ \hat{r} = \Delta l \overline{\omega}\sqrt{\frac{1}{\Delta^2 + l^2}}$$

Advancement 2: MPC for reference tracking for feedback-linearized vehicle models (I)

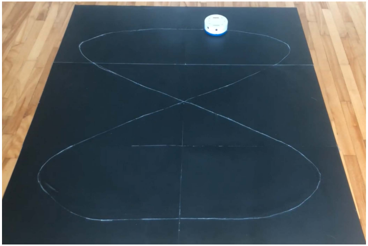
Remark: the worst-case time invariant inner approximations $\underline{\mathcal{U}}(r_u)$ and $\underline{\mathcal{U}}(\hat{r})$ are instrumental to define constraint-admissible control strategies.

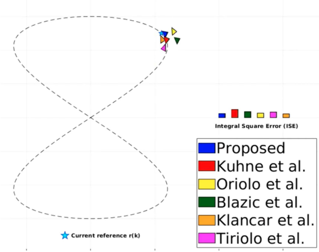
General procedure:

1. Use $\underline{\mathcal{U}}(r_u)$ or $\underline{\mathcal{U}}(\hat{r})$ to offline design a constraints-admissible but conservative control strategy for the vehicle.
2. Exploit the online measure of the vehicle parameters ($\theta(k)$ for differential-drive robots, $\theta(k), \varphi(k)$ for car-like vehicle) to online use the actual control input constraints $\mathcal{U}(\theta)$ or $\mathcal{U}(\theta, \varphi)$ and obtain non-conservative control strategies.

Advancement 2: MPC for reference tracking for feedback-linearized vehicle models (II)

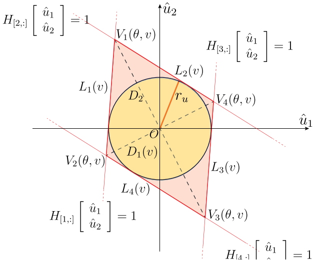
- In [*Tiriolo et al., TCST, 22*], we have developed RHC controller capable of solving a waypoint tracking problem for feedback-linearized DD robots
  - *Offline*, $\underline{\mathcal{U}}(r_u)$ is exploited to design a guaranteed RHC controller with associated invariant region $\mathcal{E}$. *Online*, the measure of $\theta(k)$ is used to relax the control problem using $\mathcal{U}(\theta)$ under an additional invariance condition for $\mathcal{E}$
  - Experiments on Khepera IV: `https://shorturl.at/demOW`

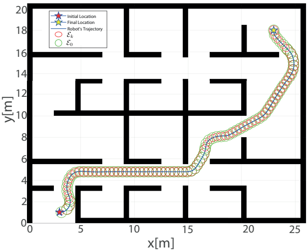




Advancement 2: MPC for reference tracking for feedback-linearized vehicle models (III)

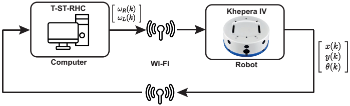
- In [*Tiriolo et al., ACC, 23*], we developed a RHC for solving the waypoint tracking with obstacle avoidance for DD robots.
    - The DD robot is linearized through dynamic-feedback linearization.
    - The linearization model is subject to a norm-bounded uncertainty and the worst-case approximation of the time-varying input constraint set is characterized.
    - The proposed RHC law ensures waypoint tracking and obstacle avoidance.

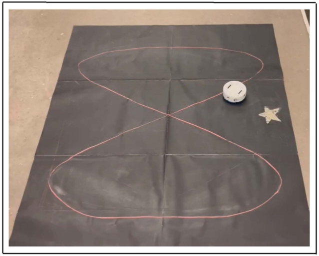




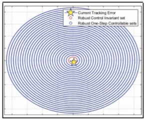
Advancement 2: MPC for reference tracking for feedback-linearized vehicle models (IV)

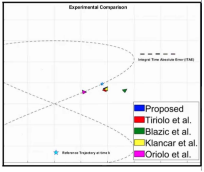
- In [*Tiriolo et al., L-CSS 23 (Accepted)*], we have developed a ST-MPC controller for solving the trajectory tracking problem for DD robots.
  - The reference trajectory is incorporated as a disturbance in the feedback-linearized error model.
  - *Offline*, the ST-MPC is developed considering $\underline{\mathcal{U}}(r_u)$. *Online*, $\mathcal{U}(\theta)$ is used to obtain a non-conservative control law
  - Experiments - Khepera IV: `https://youtu.be/A0Tlbgr08tY`





Khepera IV - 8-Shaped Trajectory Tracking



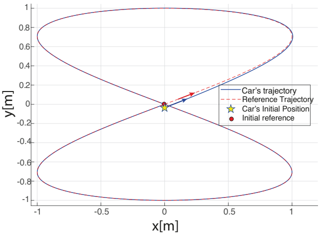


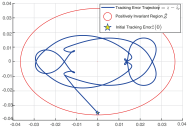
Advancement 2: MPC for reference tracking for feedback-linearized vehicle models (V)

- In [*Tiriolo et al., L-CSS 23*], considering the feedback linearized model of a car-like vehicle.
  - The feedback linearized tracking error dynamics are subject to a bounded disturbance depending on the reference trajectory:

    $$\tilde{z}(k+1) = A\tilde{z}(k) + Bu(k) + d_r(k)$$

  - A LQ controller $u(k) = -K\tilde{z}(k)$ is analytically designed to asymptotically stabilize the nominal model regardless of any input constraints.
  - Then, a RCI region Σ, associated to the computed LQ control law, is designed such that $\forall \tilde{z}(0) \in \Sigma$, then $\tilde{z}(k) \in \Sigma$, $\forall d_r(k) \in \mathcal{D}$, and $u(k) \in \underline{\mathcal{U}}(\hat{r}), \forall k > 0$





Work in progress/Future works

1. Design a complete dual-mode MPC controller for car-like vehicles
2. Design a centralized/distributed collision-avoidance strategies (e.g., Traffic Manager) for differential-drive and car-like vehicles.

End of Part 2 - Questions ?

Thank You!

References I

A. De Luca, G. Oriolo, C. Samson
Feedback control of a nonholonomic car-like robot
Robot Motion Planning and Control (RMPC), pp. 171–253, 2005

A. De Luca, G. Oriolo, M. Vendittelli
Control of wheeled mobile robots: An experimental overview
RAMSETE, pp. 181–226, 2002

S. Savehshemshaki, W. Lucia
A Robust Receding-Horizon Collision Avoidance Strategy for Constrained Unmanned Ground Vehicles Moving in Shared Planar Environments
IEEE Conference on Decision and Control (CDC) , pp. 5253–5258, 2022.

M. Bagherzadeh, S. Savehshemshaki, W. Lucia
Guaranteed Collision-Free Reference Tracking in Constrained Multi Unmanned Vehicle Systems
IEEE Transactions on Automatic Control, Vol. 67, No. 6, pp. 3083–3089, 2021

References II

M. Bagherzadeh, W. Lucia
Multi-Vehicle Reference Tracking with Guaranteed Collision Avoidance
IEEE European Control Conference (ECC), pp. 1574–1579, 2019

S. V. Rakovic, E. C. Kerrigan, K. I. Kouramas, D. Q. Mayne
Invariant approximations of the minimal robust positively invariant set
IEEE Transactions on Automatic Control, Vol. 50, No. 3, pp. 406–410, 2005

D. Angeli, A. Casavola, G. Franze', E. Mosca
An ellipsoidal off-line MPC scheme for uncertain polytopic discrete-time systems
Automatica, Vol. 44, No. 12, pp. 3113–3119, 2008

C. Tiriolo, G. Franze', W. Lucia
A Receding Horizon Trajectory Tracking Strategy for Input-Constrained Differential-Drive Robots via Feedback Linearization.
IEEE Transactions on Control Systems Technology, 2022

References III

C,Tiriolo, G. Franze', W. Lucia
An Obstacle-Avoidance Receding Horizon Control Scheme for Constrained Differential-Drive Robot via Dynamic Feedback Linearization.
IEEE American Control Conference (ACC), 2023 (to appear).

C. Tiriolo, W. Lucia
On the Design of Control Invariant Regions for Feed- back Linearized Car-Like Vehicles.
IEEE Control Systems Letters, vol. 7, pp. 739-744, 2022

C. Tiriolo, W. Lucia
A Set-Theoretic Control Approach to the Trajectory Tracking Problem for Input-Output Linearized Wheeled Mobile Robots.
IEEE Control Systems Letters, 2023 (Under review)