Multi-Agent Control for Safety-Critical Systems

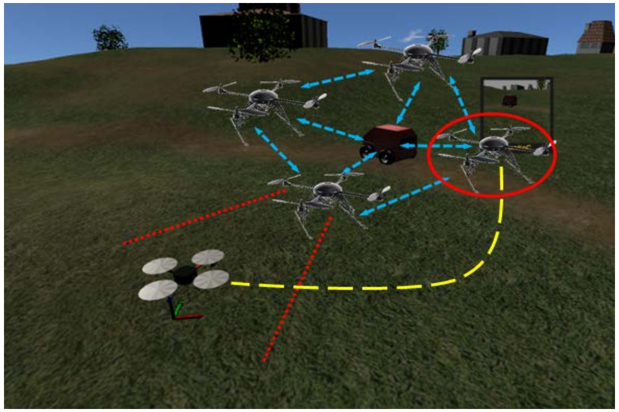
Dimitra Panagou

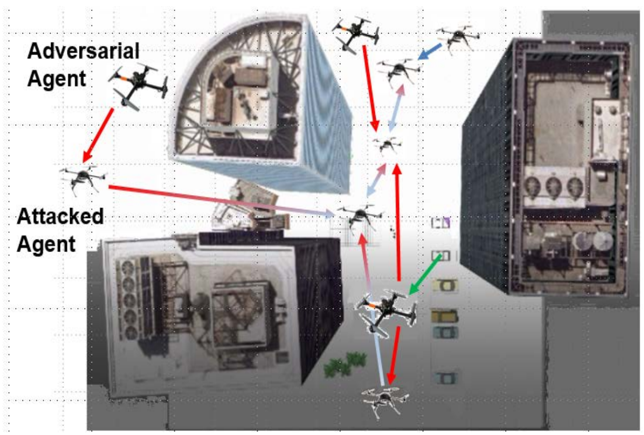
Assistant Professor, Aerospace Engineering
University of Michigan, Ann Arbor

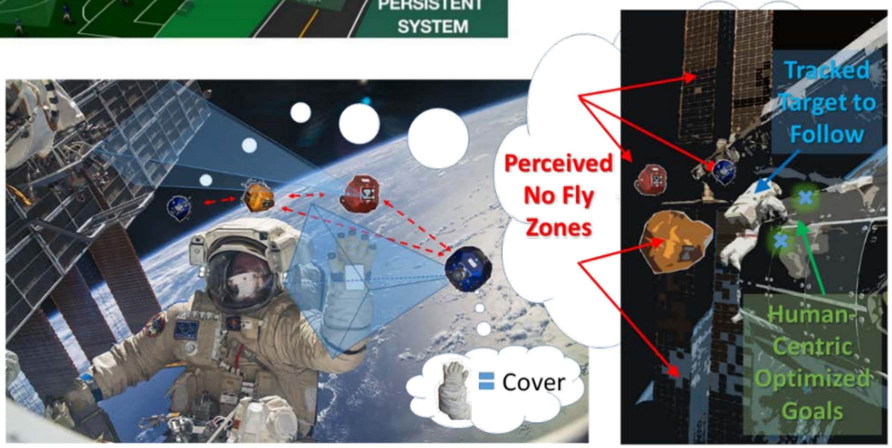
14th Young Researcher Workshop on Geometry, Mechanics and Control
December 17-18, 2019

AEROSPACE ENGINEERING
UNIVERSITY of MICHIGAN ■ COLLEGE of ENGINEERING

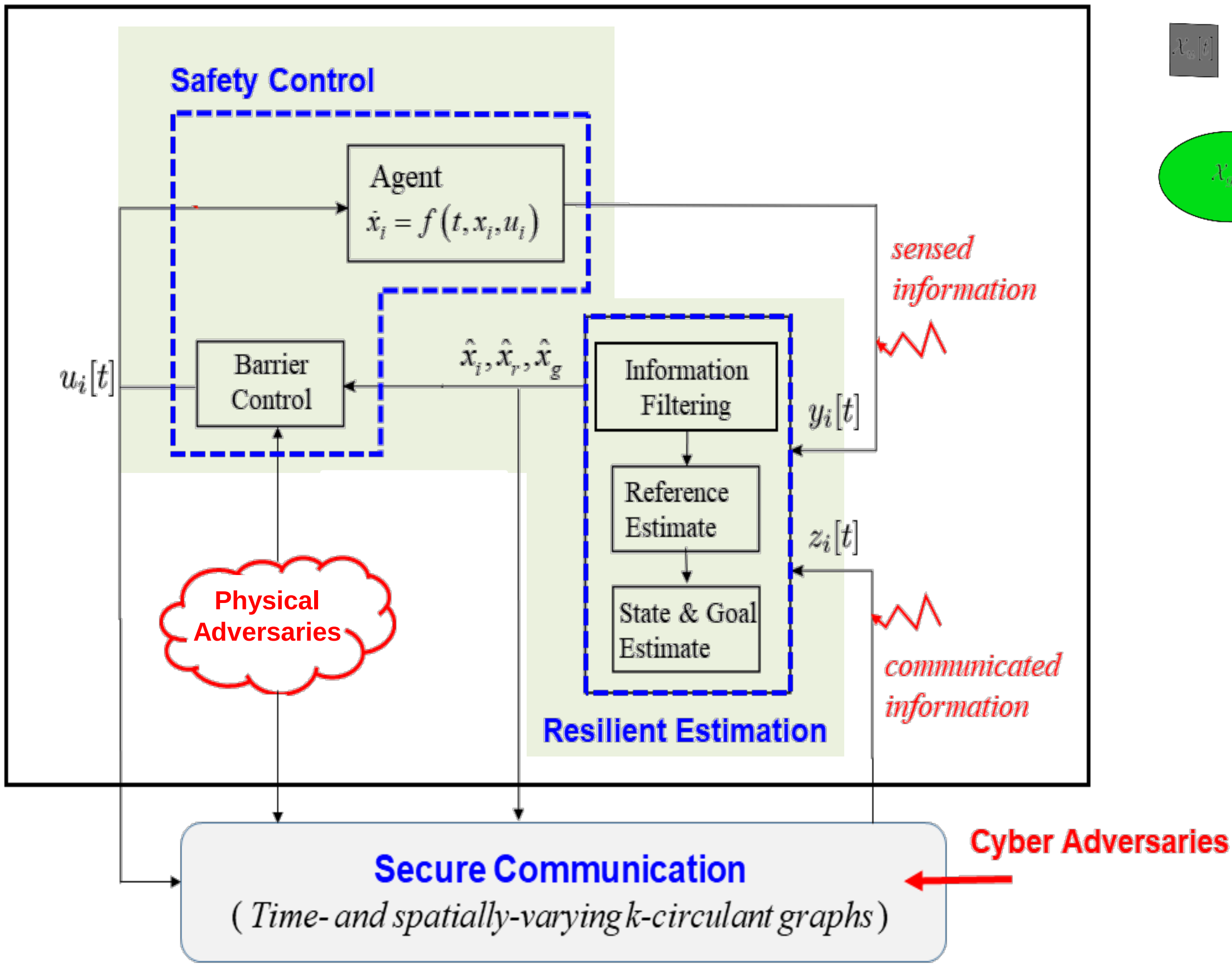
(Some) Motivation for Multi-Agent/Multi-Robot Control

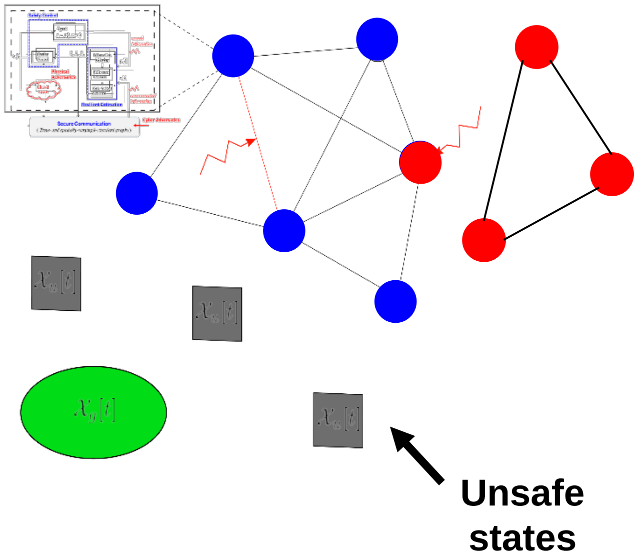






Safety and Resilience in Multi-Agent Systems



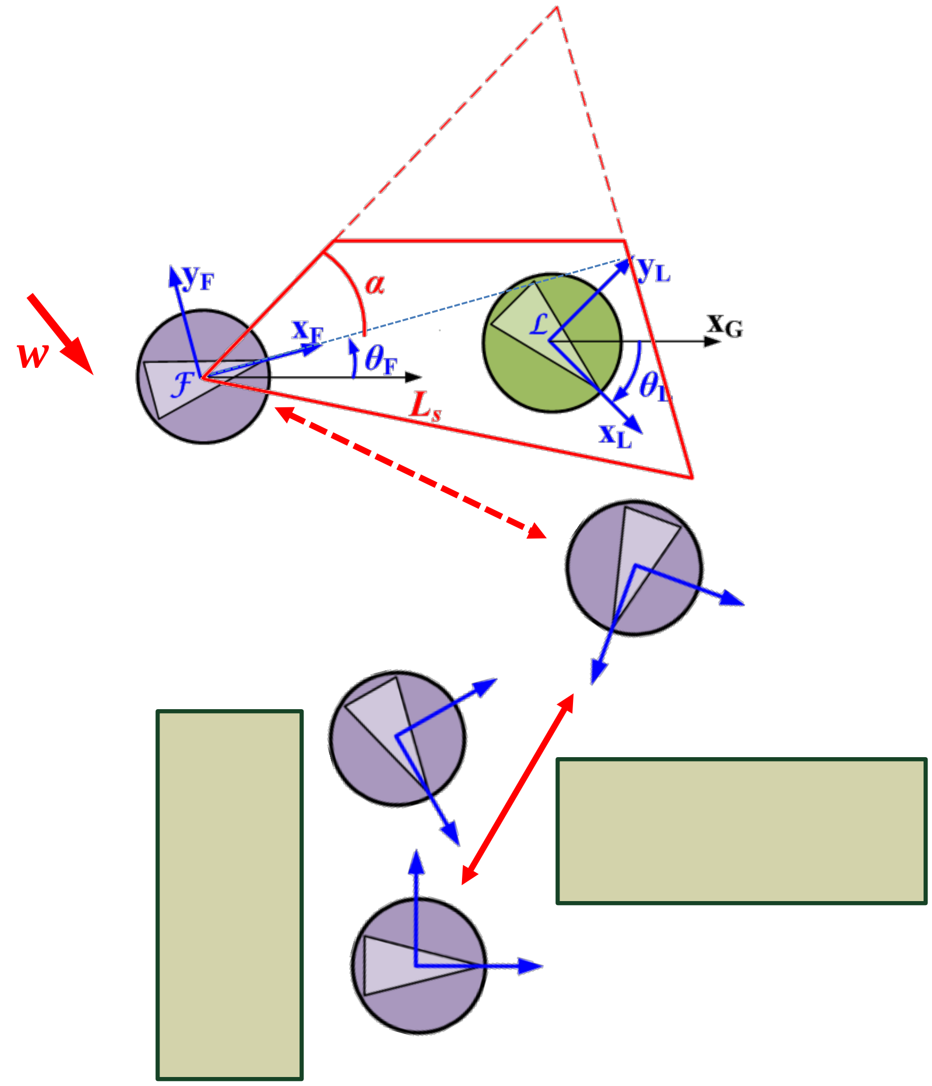


Course Outline

- Introduction to Multi-Agent Control: Motivating Examples.
- **Review on Lyapunov Stability Theory** and Switched Systems Theory
- **Control Barrier Functions (Lyapunov-like, Zeroing, Reciprocal)**
- **Introduction to Finite Time Stability (FTS) and Fixed Time Stability (FxTS) Theory**
- **Synthesis of Safety-Critical and Time-Critical Controllers**
- Introduction to Graph Theory
- Graph-Theoretical Representations of Networked Systems
- Fundamental network properties: Connectivity, r-robustness, Strong r-robustness
- Networked Control and Estimation for Security
- Synthesis of Secure Controllers with Safety Guarantees

Challenges for Multi-Agent Systems in Safety-Critical Environments

- **Nonlinear dynamical systems**
  - Non-negligible dynamics
- **Under-actuation**
  - Fewer controls than d.o.f.
  - Nonholonomic constraints
- **Perturbations**
  - Environmental disturbances
  - Model uncertainty
  - Sensing and communication errors
- **Constraints**
  - Physical obstacles
  - Sensing/communication limitations
  - Input saturations
- **Scalability with the number of agents**
  - Computationally-efficient solutions



ON THE LIGHTER SIDE «

Lyapunov Road Trip

So what if you don't like the route! We're making very rapid progress with this Lyapunov function. Just relax! You know it's stable and we'll slow down when we get near our destination.

Digital Object Identifier 10.1109/MCS.2015.2512100
Date of publication: 17 March 2016

In the *IEEE Control Systems Magazine*
April 2016

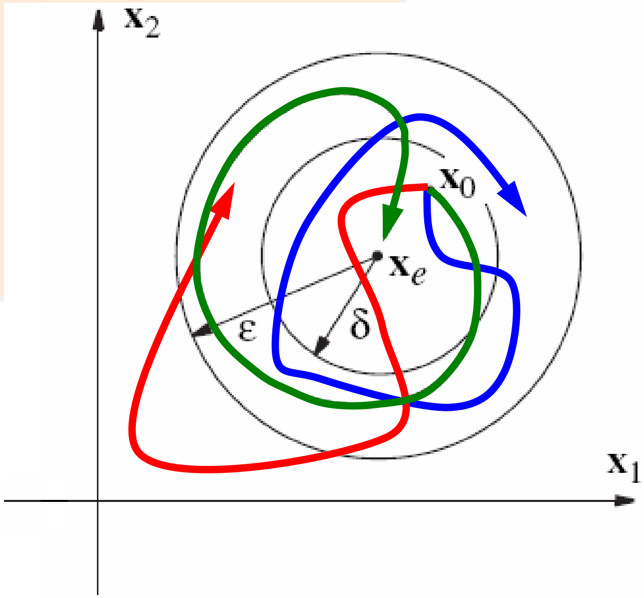
Review of Lyapunov Theory

Let $\dot{x} = f(x)$ with $f(x_e) = 0$.

Suppose $x(t, x_0)$ exists and is unique for each $x_0 = x(0), t \in [0, \infty]$.

Then, the equilibrium point x_e is

- ***Stable*** if $\forall \varepsilon > 0, \ \exists \delta(\varepsilon) > 0,$ such that
  $\|x_0 - x_e\| < \delta \ \Rightarrow \ \|x(t, x_0) - x_e\| < \varepsilon, \forall t \geq 0$
- ***Asymptotically stable*** if it is stable and
  $\lim_{t \to \infty} \|x(t, x_0) - x_e\| = 0$
- ***Unstable*** if it is not stable.



Exercise: Negate the definition of stable equilibrium to obtain the mathematical definition of the unstable equilibrium.

Theorem 4.1 [Khalil] Let $\dot{x} = f(x)$ with equilibrium point $x_e = 0$. Assume that:

- The function $f : D \to \mathbb{R}^n$ is locally Lipschitz.
- There exists a continuously differentiable function $V : D \to \mathbb{R}$ such that:

$$i) V(0) = 0$$

$$ii) V(x) > 0, \text{ for } x \in D, x \neq 0$$

$$iii) \dot{V}(x) \leq 0, \text{ for } x \in D, \text{where } \dot{V}(x) = \frac{\partial V}{\partial x}(x) f(x)$$

❑ Then, the equilibrium point is ***stable***.

- Moreover, if *i*) and *ii*) hold, and *iii*) is replaced by

$$iv) \dot{V}(x) < 0, \text{ for } x \in D, x \neq 0,$$

❑ Then the equilibrium point is ***asymptotically stable***.

***Remark*:** A function $V : D \to \mathbb{R}$ that satisfies the conditions of the Theorem is called a **Lyapunov function**.

Theorem 4.2 [Khalil] Let $\dot{x} = f(x)$ with equilibrium point $x_e = 0$. Assume that:

- The function f is locally Lipschitz on $\mathbb{R}^n (i.e., D = \mathbb{R}^n)$
- There exists a continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}^n$ such that

$$i) V(0) = 0 \text{ and } V(x) > 0, \forall x \neq 0$$

$$ii) \|x\| \to \infty \Rightarrow V(x) \to \infty$$

$$iii) \dot{V}(x) < 0, \ \forall x \neq 0$$

❑ Then, the equilibrium point is ***globally asymptotically stable***.

Corollary 4.1 [Khalil] Let $\dot{x} = f(x)$ with equilibrium point $x_e = 0$. Assume that:

- The function $f : D \rightarrow \mathbb{R}^n$ is locally Lipschitz.
- There exists a continuously differentiable function $V : D \rightarrow \mathbb{R}$ such that:

$$i) V(0) = 0$$

$$ii) V(x) > 0, \text{ for } x \in D, x \neq 0$$

$$iii) \dot{V}(x) \leq 0, \text{ for } x \in D$$

Let $S = \{x \in D \mid \dot{V}(x) = 0\}$ and suppose that no other solution can stay identically in S other than the trivial solution $x(t) \equiv 0$.

❑ Then the equilibrium point is ***asymptotically stable***.

Corollary 4.2 [Khalil] Let $\dot{x} = f(x)$ with equilibrium point $x_e = 0$. Assume that:

- The function f is locally Lipschitz on $\mathbb{R}^n (i.e., D = \mathbb{R}^n)$
- There exists a continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}^n$ such that

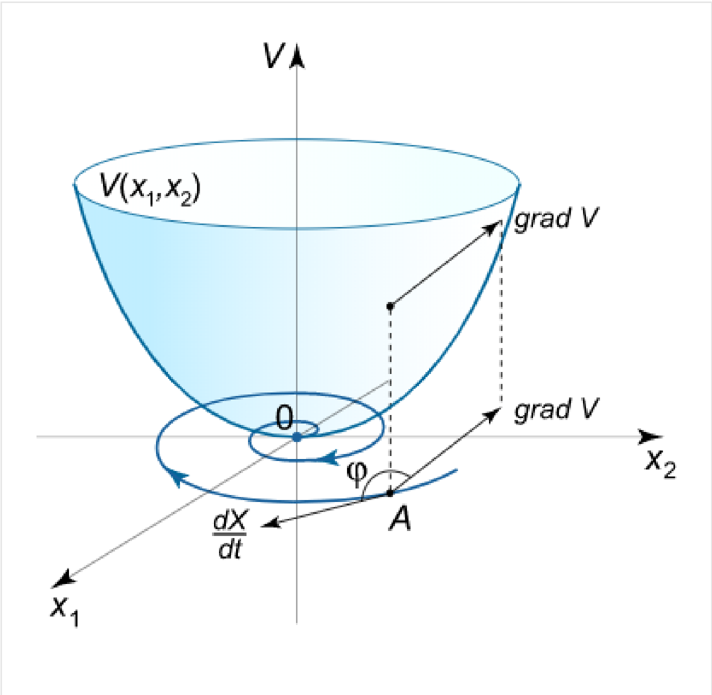
$$i) V(0) = 0 \text{ and } V(x) > 0, \forall x \neq 0$$
$$ii) \|x\| \to \infty \Rightarrow V(x) \to \infty$$
$$iii) \dot{V}(x) \leq 0, \ \forall x \in \mathbb{R}^n$$

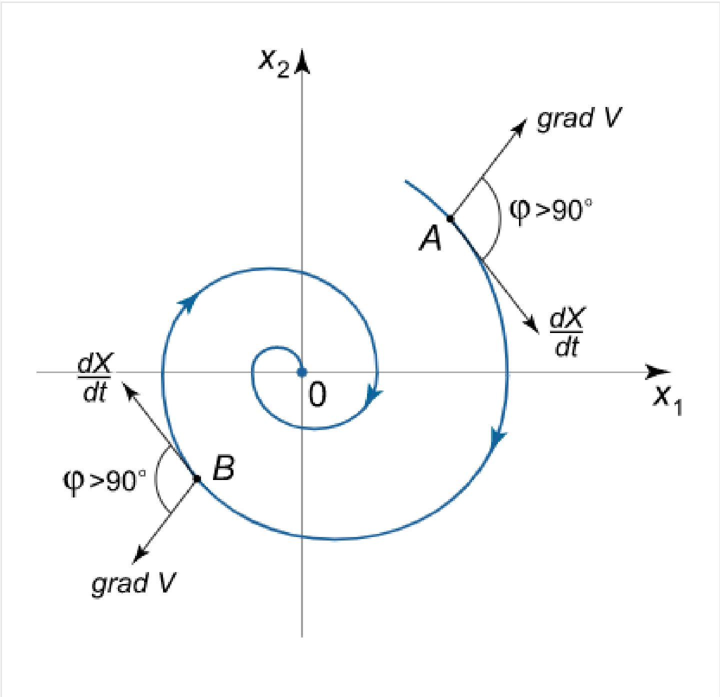
Let $S = \{x \in \mathbb{R}^n \mid \dot{V}(x) = 0\}$ and suppose that no other solution can stay identically in S other than the trivial solution $x(t) \equiv 0$.

❑ Then the equilibrium point is ***globally asymptotically stable***.

Examples

Geometric Representation of Asymptotic Stability





$$\dot{V}(x) = \left\langle grad\ V(x), \frac{dx}{dt} \right\rangle = \left\langle \frac{\partial V(x)}{\partial x}, f(x) \right\rangle < 0, \quad x \neq 0$$

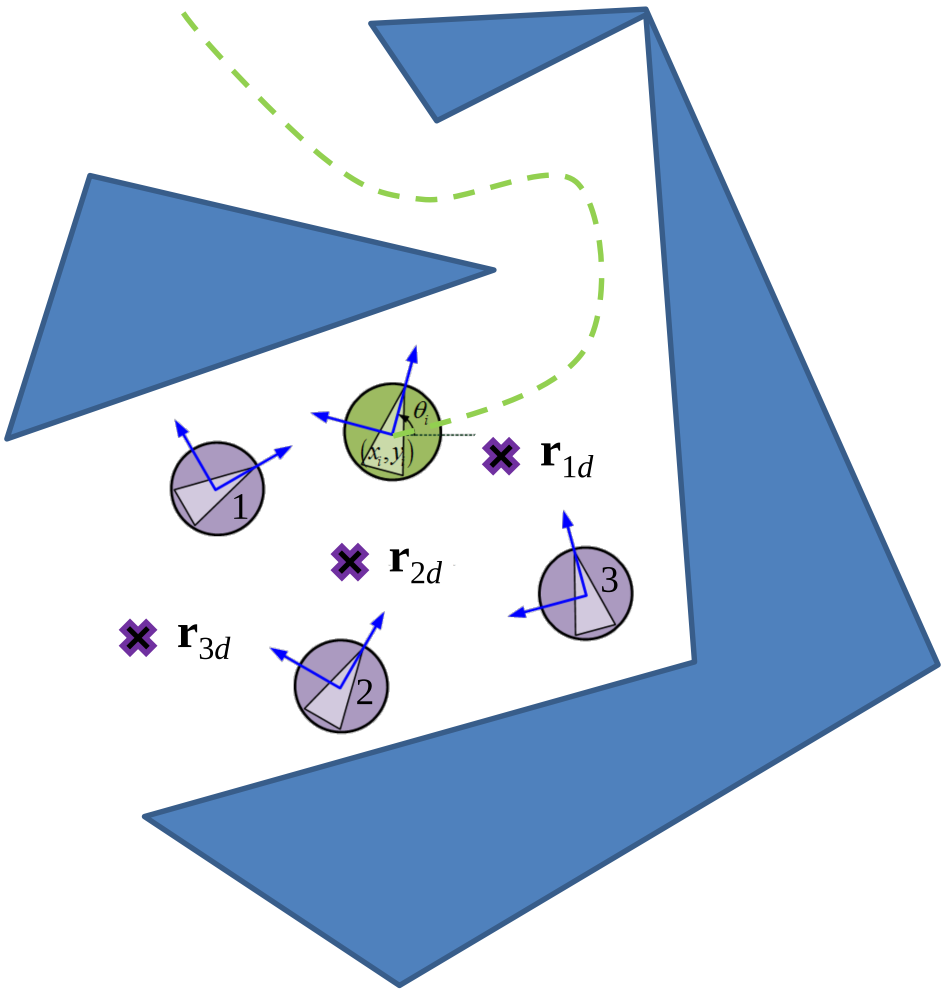
- Tracking the negated gradient of a Lyapunov function $V(x)$ provides a convergent trajectory $x(t)$ to the equilibrium $x = 0$.
- How can we construct functions that penalize trajectories from entering (or exiting) certain sets of the state space?

Multi-Agent Coordination

via

Lyapunov-like Barrier Functions

Multi-Robot Coordination: Modeling and Assumptions



❑ Agent Models: N unicycle robots

$$\dot{x}_i = u_i \cos\theta_i$$
$$\dot{y}_i = u_i \sin\theta_i \qquad i \in \{1, 2, \ldots, N\}$$
$$\dot{\theta}_i = \omega_i$$

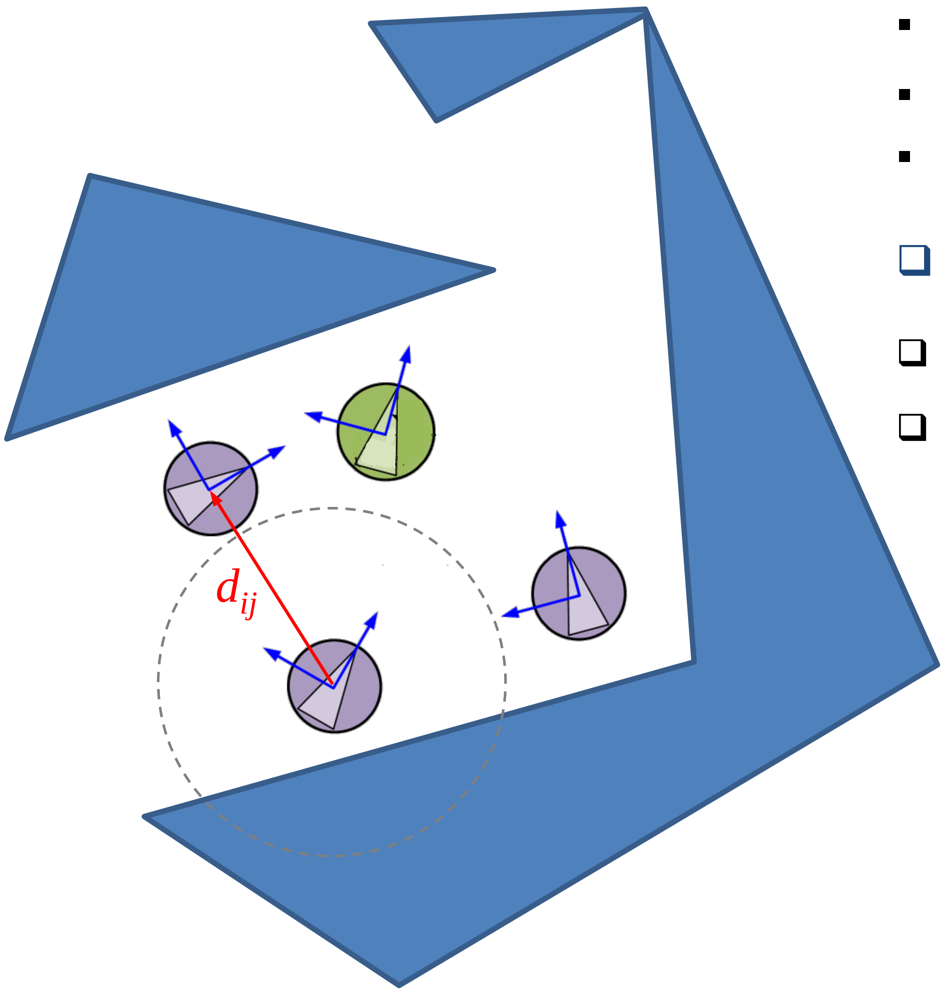
➢ **Assumption**

- 1 Leader, N-1 Followers
- Leader is the agent of **highest** priority

❑ **Assumption: Leader objectives**

- Track a nominal motion plan
- Communicate goal points to followers
- Re-plan if necessary to guide the group

Multi-Robot Coordination: Sensing and Communication Models

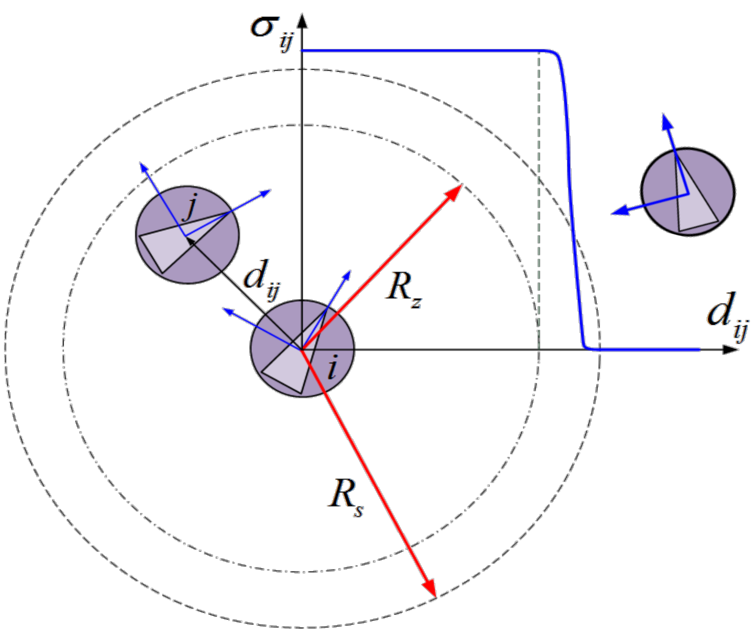


❑ Leader-to-Follower broadcast model

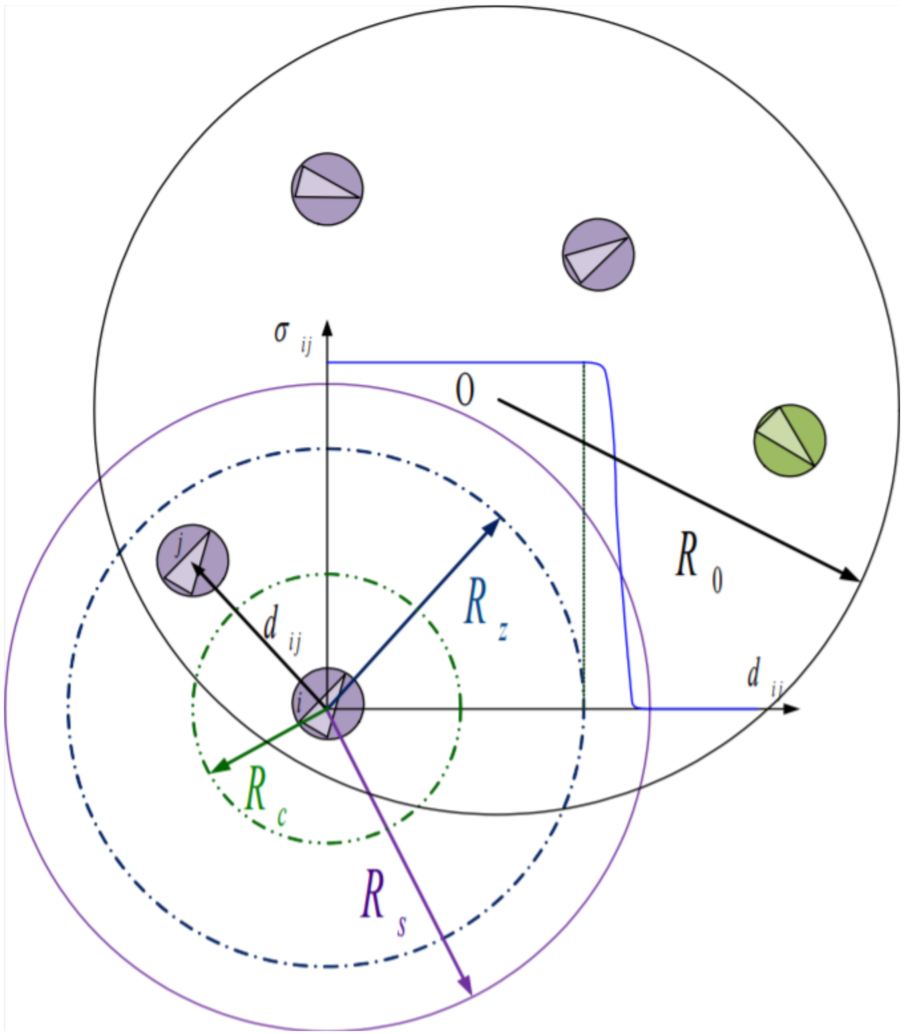
- Leader only broadcasts information
- Broadcast is reliable within a region of radius R_0
- Imposes that inter-agent distance $d_{ij} < 2\, R_0$

❑ Followers' sensing & communication model

- ❑ Followers sense agents within distance R_s
- ❑ Followers communicate with agents within $R_c < R_s$

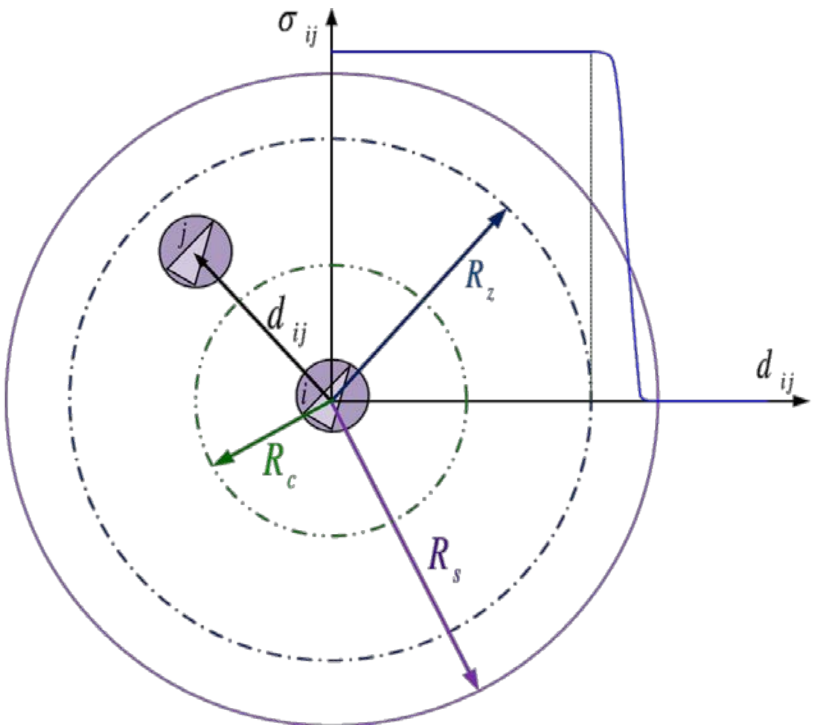


Multi-Robot Coordination: Sensing and Communication Models



- **The leader agent:**
  - Reliably broadcasts information to agents j lying within distance $d_{Lj} \leq 2R_0$
  - This is realized by forcing all agents to move in a circular connectivity region O of radius R_0 centered at some point $\mathbf{r}_0$.
- **Each follower agent:**
  - Measures the position $\mathbf{r}_j$ of agents j lying within distance $d_{ij} \leq R_s$
  - Receives the orientation θ_j and linear velocity u_j of every agent j lying within distance $d_{ij} \leq R_c$
  - Seeks to avoid collisions with every agent lying within distance $d_{ij} \leq R_z$

Multi-Robot Coordination: Sensing and Communication Models



- **Each follower (continued):**

- To encode that each follower aims to avoid only its local neighbors (i.e., the agents within its sensing zone), we define the function:

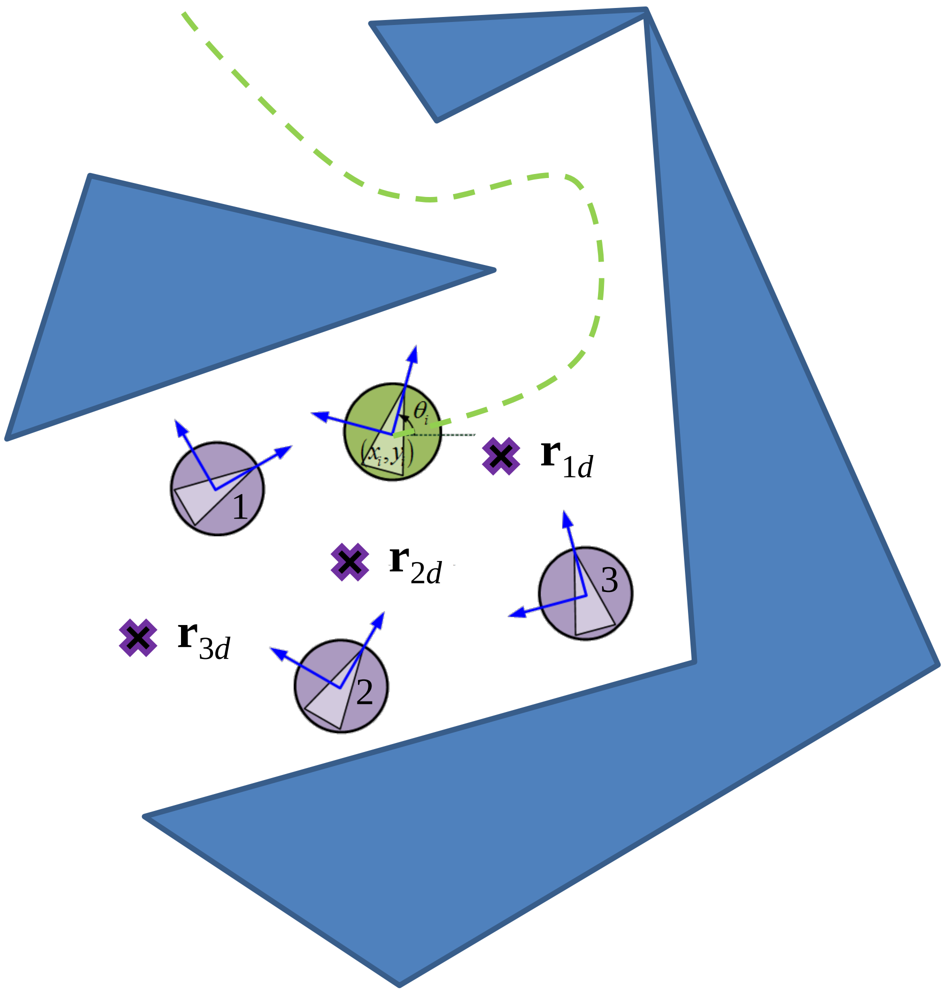
$$\sigma_{ij} = \begin{cases} 1, & \text{if } d_s \le d_{ij} \le R_z; \\ A{d_{ij}}^3 + B{d_{ij}}^2 + Cd_{ij} + D, & \text{if } R_z < d_{ij} < R_s; \\ 0, & \text{if } d_{ij} \ge R_s, \end{cases}$$

$$A = -\frac{2}{(R_z - R_s)^3} \quad B = \frac{3(R_z + R_s)}{(R_z - R_s)^3},$$

$$C = -\frac{6R_zR_s}{(R_z - R_s)^3}, \; D = \frac{{R_s}^2(3R_z - R_s)}{(R_z - R_s)^3}$$

where the coefficients have been computed such that $\sigma_{ij}(\cdot)$ is a C^2 function w.r.t. d_{ij}

Multi-Robot Coordination: Problem Formulation



- ❑ Agent Models: N unicycle robots

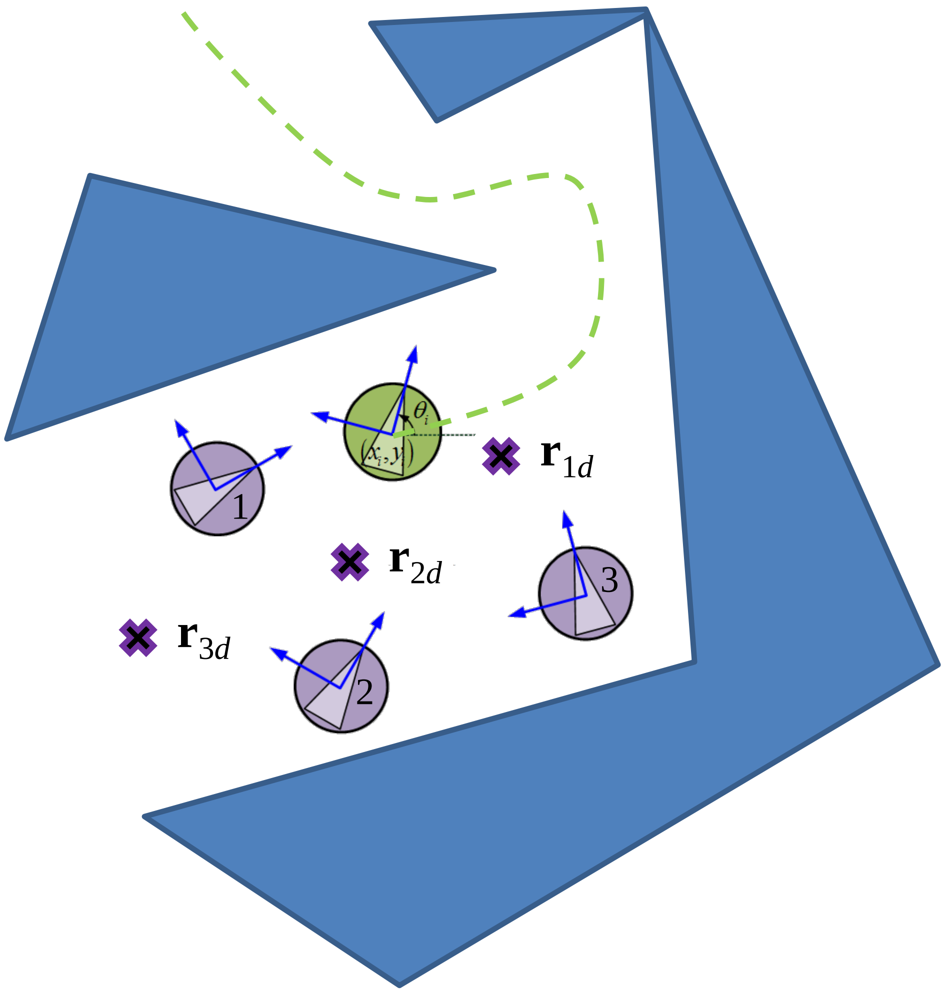
$$\dot{x}_i = u_i \cos\theta_i$$
$$\dot{y}_i = u_i \sin\theta_i \qquad i \in \{1, 2, \ldots, N\}$$
$$\dot{\theta}_i = \omega_i$$

- ➢ **Assumptions**
  - Sensing and communication as described earlier

- ❑ **Given: Leader objectives**
  - Track a nominal motion plan
  - Communicate goal points to followers
  - Re-plan if necessary to guide the group

- ❑ **Find: [Followers' objectives]**
  - Converge to, or track, the goal points
  - Avoid collisions
  - Stay close enough (for communication)

Multi-Robot Coordination: Technical Approach



- Agent Models: N unicycle robots

$$\dot{x}_i = u_i \cos\theta_i$$
$$\dot{y}_i = u_i \sin\theta_i \qquad i \in \{1, 2, \ldots, N\}$$
$$\dot{\theta}_i = \omega_i$$

- **Assumptions**

- Sensing and communication as described earlier

- **Approach**

- Encode the followers' objectives via Lyapunov-like Barrier Functions
- Design Lyapunov-based controllers
- Lyapunov-like functions penalize the violation of the objectives
- Similar in concept to potential functions (robotics), penalty functions (optimization).

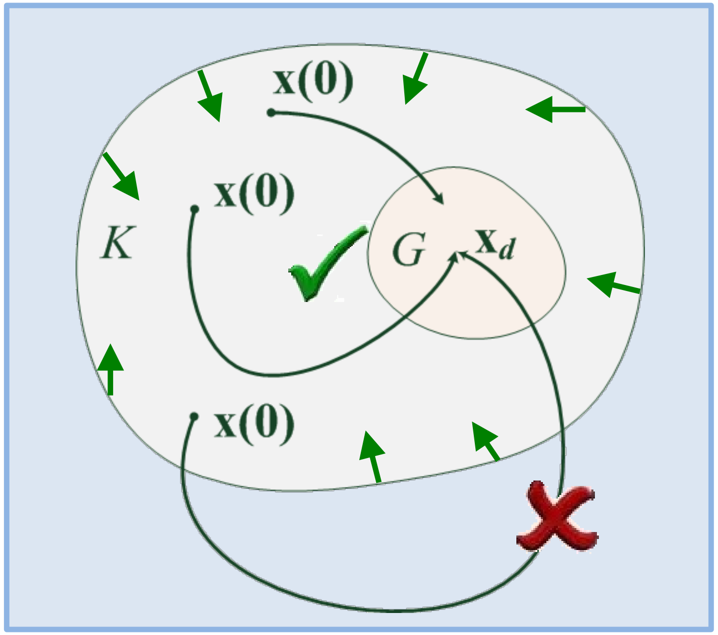
Multi-Robot Coordination: Problem Formulation Revisited

❑ **Set-theoretic representation of objectives**

- Collision avoidance:
  - o **Constrained set** K_i
- Convergence (close) to destination:
  - o **Goal set** G_i in K_i

❑ **Problem statement**

1. **Construct a function** $V_i(\mathbf{x})$ **to:**
   - Encode the sets K_i, G_i
2. **Control** so that system trajectories $\mathbf{x}_i(t)$:
   - Always remain in K_i
   - Converge to G_i

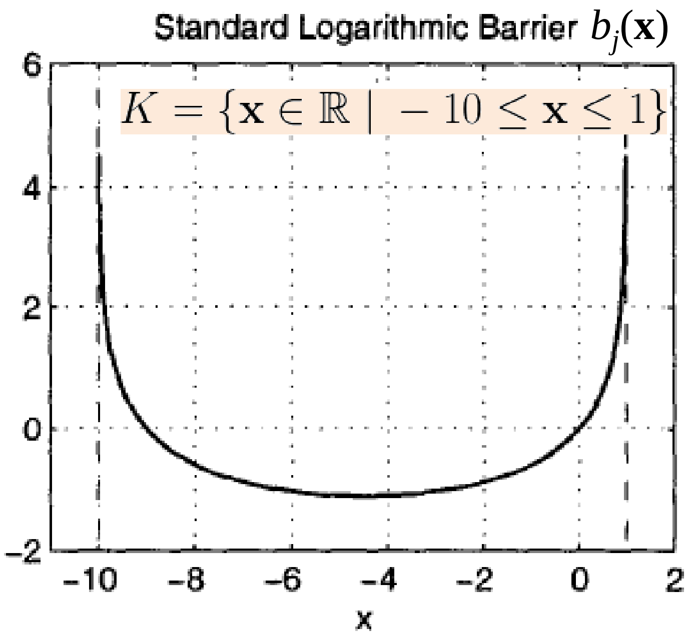


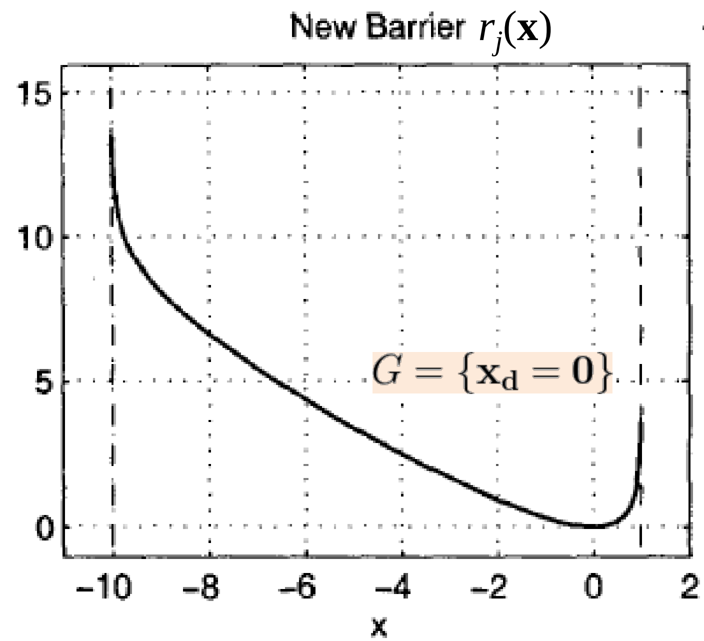
Abstract representation of the collision-free set K_i and the goal set G_i for each agent i. The agent trajectories should converge to G_i while always remaining in K_i

Definition of Lyapunov-like Barrier Functions

- **Logarithmic Barrier Function:** $b_j(\mathbf{x}) = -\ln(c_j(\mathbf{x}))$, where $c_j(\mathbf{x}) \geq 0$ a state-dependent constraint
- **Recentered Barrier Function*:** $r_j(\mathbf{x}) = b_j(\mathbf{x}) - b_j(\mathbf{x_d}) - \nabla b_j(\mathbf{x_d})^\top (\mathbf{x} - \mathbf{x_d})$

- Shapes a barrier function $b_j(\mathbf{x})$ so that
  - It tends to infinity on the boundary of the constrained set
  - It vanishes at a desired point $\mathbf{x_d}$
- Encodes that trajectories $\mathbf{x}(t)$ shall lie **in the constrained set** and **converge to the set point** $\mathbf{x_d}$

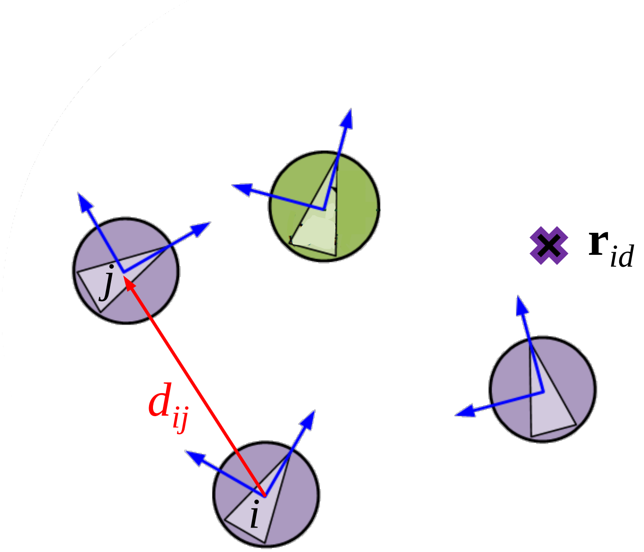




* **A. G. Wills and W. P. Heath**, "A recentred barrier for constrained receding horizon control" (ACC 2002)

Construction of Lyapunov-like Barrier Functions

Encoding Collision Avoidance and Convergence to Destination



- **Collision avoidance** of agent i w.r.t. agent j

$$c_{ij}(\cdot) = d_{ij}^{\ 2} - d_s^{\ 2} = \left\|\mathbf{r}_i - \mathbf{r}_j\right\|^2 - d_s^{\ 2} \geq 0$$

- **Barrier Function** : $b_{ij}(\mathbf{r}_i, \mathbf{r}_j) = -\ln\left(c_{ij}(\mathbf{r}_i, \mathbf{r}_j)\right)$
  - o Tends to $+\infty$ as $c_{ij} \to 0$
- **Recentered** Barrier Function **w.r.t. Destination**:

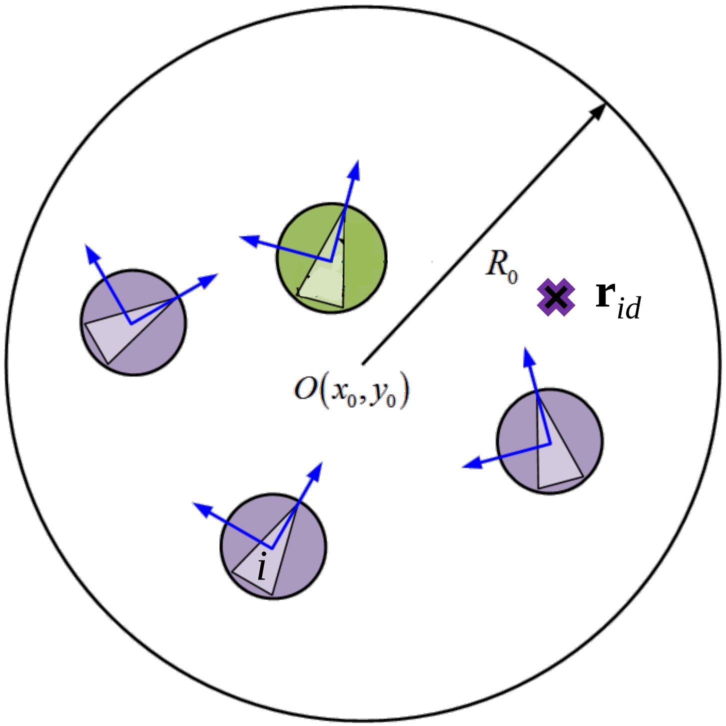
$$r_{ij}(\cdot) = b_{ij}(\mathbf{r}_i, \mathbf{r}_j) - b_{ij}(\mathbf{r}_{id}, \mathbf{r}_j) - \nabla b_{ij}(\mathbf{r}_{id}, \mathbf{r}_j)^{\top}(\mathbf{r}_i - \mathbf{r}_{id})$$

  - o Vanishes at the destination $\mathbf{r}_{id}$
- **Lyapunov-like** R. B. F. w.r.t. agent j

$$V_{ij}(\cdot) = \sigma_{ij}\left(r_{ij}(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_{id})\right)^2$$

- w.r.t. all agents: $$V_i^{\text{col}} = \sum_{j \neq i} V_{ij}(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_{id})$$

Construction of Lyapunov-like Barrier Functions



Encoding Proximity and Convergence to Destination

- **Proximity** constraint

$$c_{i0}(\cdot) = R_0^{\ 2} - \|\mathbf{r}_i - \mathbf{r}_0\|^2 = R_0^{\ 2} - (x_i - x_0)^2 - (y_i - y_0)^2 \geq 0$$

- **Barrier Function**: $b_{i0}(\mathbf{r}_i, \mathbf{r}_0) = -\ln\left(c_{i0}(\mathbf{r}_i, \mathbf{r}_0)\right)$
  - Tends to $+\infty$ as $c_{i0} \to 0$

- **Recentered** Barrier Function **w.r.t. Destination**:

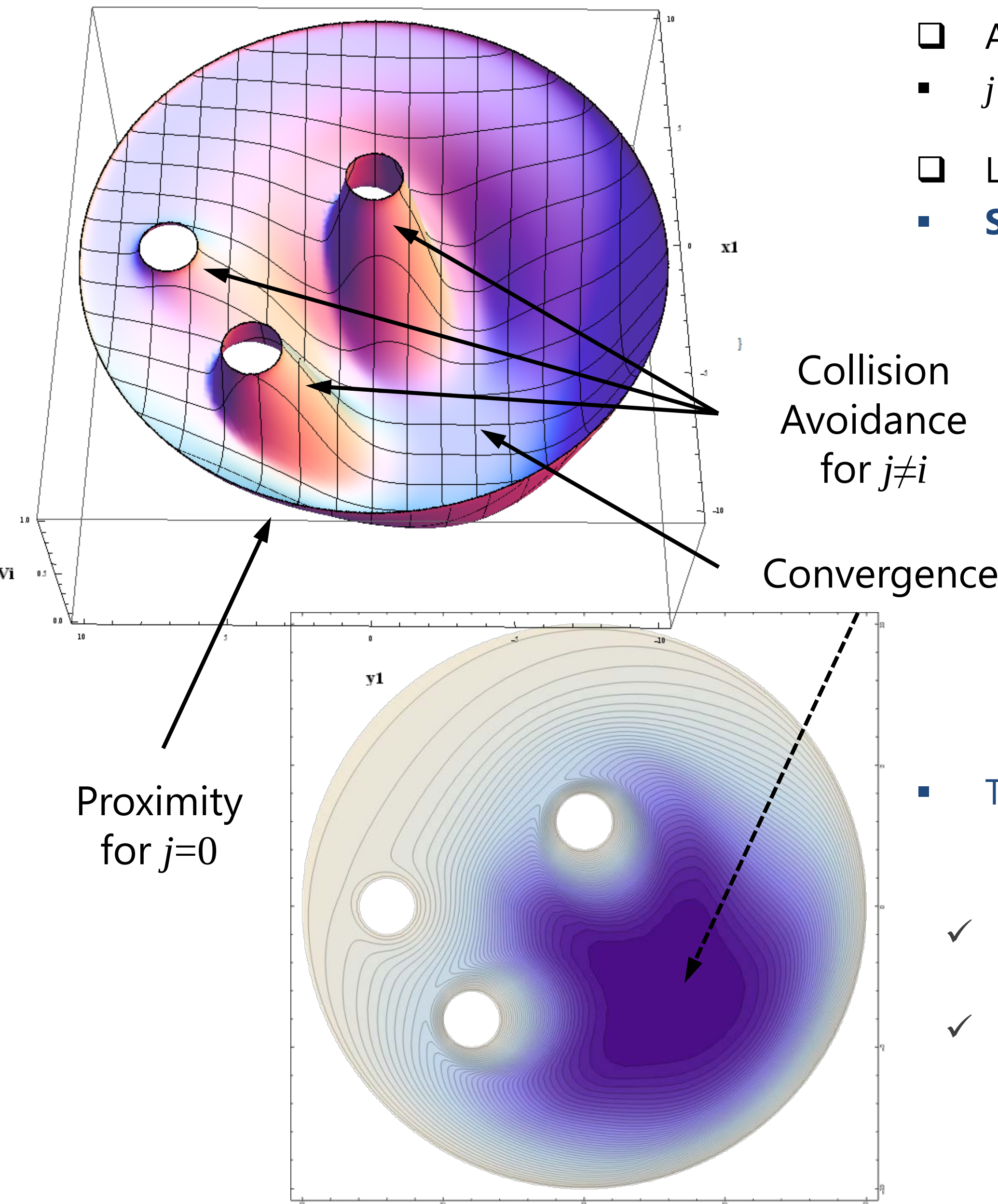
$$r_{i0}(\cdot) = b_{i0}(\mathbf{r}_i, \mathbf{r}_0) - b_{i0}(\mathbf{r}_{id}, \mathbf{r}_0) - \nabla b_{i0}(\mathbf{r}_{id}, \mathbf{r}_0)^\top (\mathbf{r}_i - \mathbf{r}_{id})$$

  - Vanishes at the destination $\mathbf{r}_{id}$

- **Lyapunov-like** R. B. F.

$$V_i^{\text{pro}} = V_{i0}(\mathbf{r}_i, \mathbf{r}_0, \mathbf{r}_{id}) = \left(r_{i0}(\mathbf{r}_i, \mathbf{r}_0, \mathbf{r}_{id})\right)^2$$

Construction of Lyapunov-like Barrier Functions



- ❑ Agent i has j **neighbors**
  - ▪ j Lyapunov-like barrier functions V_{ij}
- ❑ Lyapunov-like barrier function per agent i
  - ▪ **Scaled approximation of the maximum function**

$$V_i = \frac{\left(\sum_{j=0,\, j\neq i}^{N} \left(V_{ij}{}^{\delta} \right) \right)^{\frac{1}{\delta}}}{1 + \left(\sum_{j=0,\, j\neq i}^{N} \left(V_{ij}{}^{\delta} \right) \right)^{\frac{1}{\delta}}}$$

- ▪ The scaling is so that $V_i \in [0, 1]$

- ✓ **Encodes penalization of constraint violation and convergence to goal state**
- ✓ **Gives rise to gradient-based controllers**

Control Design and Coordination Protocol

Theorem Each agent j converges almost surely to its goal destination, avoids collisions and remains in the connectivity region under the control law:

$$u_j = \begin{cases} \min\limits_{k \in \mathcal{I}_j \mid J_k < 0} u_{j|k}, & d_s \le d_{jk} \le R_c, \\ u_{jc}, & R_c < d_{jk}; \end{cases}$$

$$\omega_j = -\lambda_j(\theta_j - \phi_j) + \dot{\phi}_j \qquad \phi_j \triangleq \arctan\left(-\frac{\partial V_j}{\partial y_j}, -\frac{\partial V_j}{\partial x_j}\right)$$

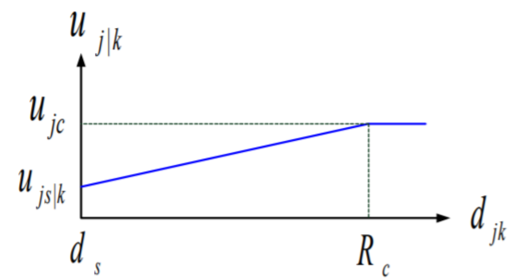
where: $\mathcal{I}_j$ is the set of neighbors of agent j,

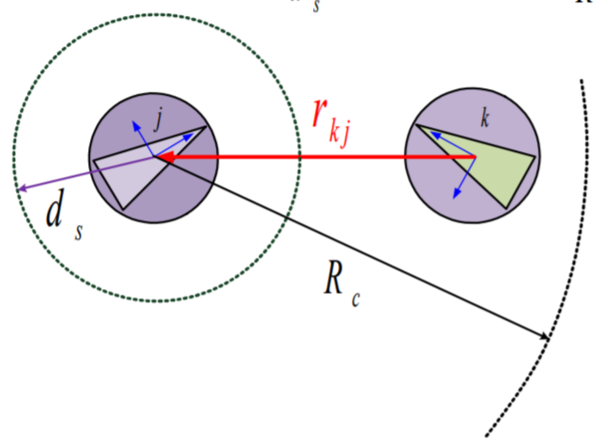
$$u_{j|k} = u_{jc}\frac{d_{ik} - d_s}{R_c - d_s} + u_{js|k}\frac{R_c - d_{ij}}{R_c - d_s}$$

$$u_{jc} = k_j \tanh\left(\left\|\mathrm{r}_j - \mathrm{r}_{jd}\right\|\right), \quad u_{js|k} = u_k \frac{\mathrm{r}_{kj}^{\;T}\eta_k}{\mathrm{r}_{kj}^{\;T}\eta_j}$$

$$J_k = \mathrm{r}_{kj}^{\;T}\begin{bmatrix}\cos\phi_j \\ \sin\phi_j\end{bmatrix}, \mathrm{r}_{kj} = \mathrm{r}_j - \mathrm{r}_k, d_{kj} = \left\|\mathrm{r}_{kj}\right\|,$$

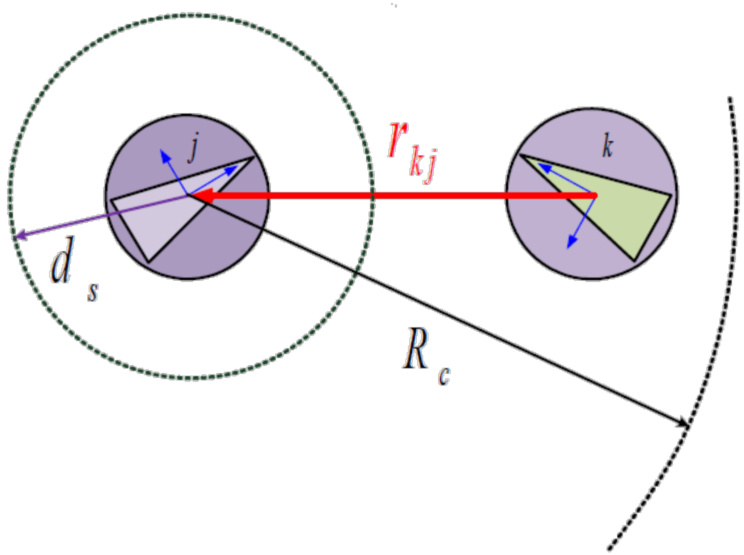
$$\eta_j = \begin{bmatrix}\cos\phi_j \\ \sin\phi_j\end{bmatrix}, \quad \delta \ge 1, \quad k_j, \lambda_j > 0$$





D. P., D. M. Stipanovic and P. G. Voulgaris "Distributed coordination control for multi-robot networks using Lyapunov-like barrier functions", IEEE Transactions on Automatic Control

Control Design and Coordination Protocol



- We control the speed of agent *j* w.r.t. an agent *k* out of the set of neighbor agents that satisfy $J_k < 0$. What is the justification for this choice?

- Recall the collision avoidance constraint:

$$c_{jk}(t) = \left(x_j(t) - x_k(t)\right)^2 + \left(y_j(t) - y_k(t)\right)^2 - d_s^2 \geq 0$$

- From Nagumo's Theorem, we have:

$$\frac{d}{dt} c_{jk} = \underbrace{2\, u_j\, \mathbf{r}_{kj}^T \begin{bmatrix} \cos\phi_j \\ \sin\phi_j \end{bmatrix}}_{J} \underbrace{- 2u_k\, \mathbf{r}_{kj}^T \begin{bmatrix} \cos\phi_k \\ \sin\phi_k \end{bmatrix}}_{K} \geq 0$$

- We then can define "semi-cooperative" interactions as:

Def. 1: If $J \geq 0$ and $K \geq 0$: "*fully cooperative avoidance*"

Def. 2: If $J \geq 0$ and $K \leq 0$ and $J + K \geq 0$: "*semi-cooperative avoidance*" by agent *j*

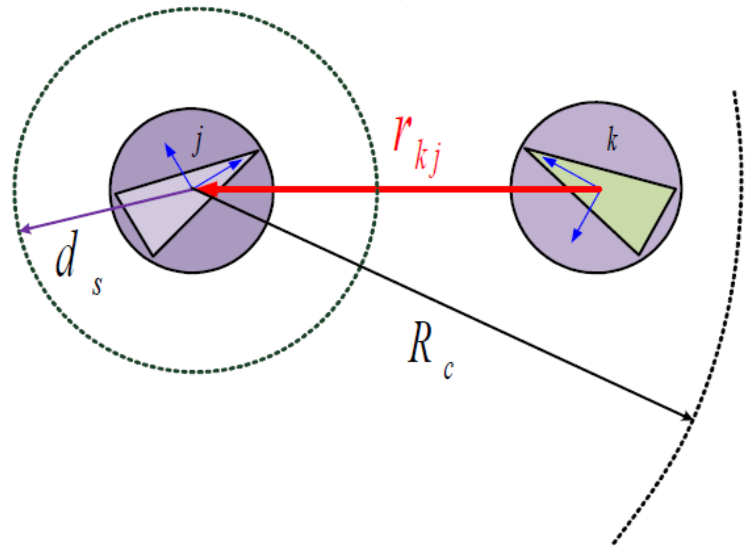
Def. 3: If $J \leq 0$ and $K \geq 0$ and $J + K \geq 0$: "*semi-cooperative avoidance*" by agent *k*

Def. 4: If $J \leq 0$ and $K \leq 0$, *or*

If $J \geq 0$ and $K \leq 0$ and $J + K \leq 0$, *or*

If $J \leq 0$ and $K \geq 0$ and $J + K \leq 0$, then we have: "*collision*"

Control Design and Coordination Protocol



- How do we choose the "worst-case" agent k?
- Recall collision avoidance constraint:

$$c_{jk}(t) = \left(x_j(t) - x_k(t)\right)^2 + \left(y_j(t) - y_k(t)\right)^2 - d_s^{\,2} \geq 0$$

- From Nagumo's Theorem, we have:

$$\frac{d}{dt} c_{jk} = \underbrace{2\, u_j\, \mathbf{r}_{kj}^{\ T} \begin{bmatrix} \cos\phi_j \\ \sin\phi_j \end{bmatrix}}_{J} \underbrace{- 2u_k\, \mathbf{r}_{kj}^{\ T} \begin{bmatrix} \cos\phi_k \\ \sin\phi_k \end{bmatrix}}_{K} \geq 0$$

- And we impose that the one who jeopardizes safety ($J < 0$) needs to resolve the conflict
- Hence the developed controller is "semi-cooperative" in the sense that each agent j adjusts its term J to ensure that J+K>0.
- As a result:
- Not **all** agents participate in conflict resolution
- Not **all** agents need to deviate from their nominal plan
- Computational demands are reduced

Guarantees for objectives' accomplishment – Proof Sketch

1) **Time-scale decomposition** into position and orientation subsystems

- Position trajectories $\mathbf{r}_i(t)$ are the reduced system ("slow" time scale)
- Orientation trajectories $\theta_i(t)$ are the boundary-layer system ("fast" time scale)

2) **Nagumo's theorem** for collision avoidance and connectivity maintenance

- System trajectories $\mathbf{r}_i(t)$ starting in the set K_i always remain in K_i

$$\frac{d}{dt} c_{ij}\left(\mathbf{r}_i,\ \mathbf{r}_j\right) \geq 0, \quad \forall\ \mathbf{r}_i \in \partial\mathcal{K}_i$$

- The constrained set K_i is rendered a **positively invariant set**

3) **Input-to-State Stability** for (almost global) convergence to G_i

$$\dot{V}_i \leq -\mu_1 \nu_1 k_i \tanh^2\left(\left\|\mathbf{r}_i - \mathbf{r}_{id}\right\|\right) + \max_{k \neq i}\{u_k\}\mu_2$$

- Perturbation signal u_k vanishes except for trajectories that may get stuck at critical points
- $\nabla V_i = 0 \Rightarrow \sum \nabla r_{ij} = 0$ **, i.e. at least $\mathrm{N_c}$ critical points away from the goal destination**
- **At best:** Tune parameter δ so that the critical points are saddles (unstable equilibria)

Control Barrier Functions

Safety verification via Barrier Certificates

- Safety verification via Barrier Certificates (Prajna et. al., HSCC 2004)

$$\text{Let } \dot{x} = f(x, d), \quad x(0) \in \mathcal{X}_0$$

$x \in \mathcal{X}$ is the state vector
$d \in \mathcal{D}$ is a disturbance vector
$\mathcal{X}_0$ is the set of initial states
$\mathcal{X}_u$ is the set of unsafe states

Theorem: Let $\dot{x} = f(x, d)$ and the sets $\mathcal{X}, \mathcal{D}, \mathcal{X}_0, \mathcal{X}_u$ be given.

- Suppose there exists a continuously differentiable function $B : \mathcal{X} \to \mathbb{R}$ such that

$$i) B(x) > 0, \forall x \in \mathcal{X}_u$$

$$ii) B(x) \leq 0, \forall x \in \mathcal{X}_0$$

$$iii) \frac{\partial B}{\partial x}(x) f(x, d) \leq 0, \quad \forall (x, d) \in \mathcal{X} \times \mathcal{D} \text{ such that } B(x) = 0$$

❑ Then, there exists no trajectory of the system contained in $\mathcal{X}$ that starts from an initial state in $\mathcal{X}_0$ and reaches another state in $\mathcal{X}_u$ (safety is guaranteed).

Constructive safety via Control Barrier Functions

- Constructive safety via Control Barrier Functions (Wieland and Allgöwer, IFAC 2007)

Definition: Let $\dot{x} = f(x) + g(x)u, \quad x \in \mathcal{X} \subset \mathbb{R}^n, u \in \mathbb{R}^p,$ and a set of unsafe states $\mathcal{X}_u \subset \mathcal{X}$.

A continuously differentiable function $B : \mathcal{X} \to \mathbb{R}$ satisfying

$$i) x \in \mathcal{X}_u \Rightarrow B(x) > 0$$

$$ii) \frac{\partial B}{\partial x}(x) g(x) = 0 \Rightarrow \frac{\partial B}{\partial x}(x) f(x) < 0$$

$$iii) \{x \in \mathcal{X} \mid B(x) \leq 0\} \neq \emptyset$$

is called a Control Barrier Function (CBF).

Constructive safety via Control Barrier Functions

- Constructive safety via Control Barrier Functions (Wieland and Allgöwer, IFAC 2007)

Theorem: Let $\dot{x} = f(x) + g(x)u, \quad x \in \mathcal{X} \subset \mathbb{R}^n, u \in \mathbb{R}^p,$
a set of unsafe states $\mathcal{X}_u \subset \mathcal{X}$.
and a CBF $B(x)$ for the system.

Define

$$k_0(x) = \begin{cases} -\frac{a + \sqrt{a^2 + k^2 b^T b}}{b^T b}, & \text{if } b \neq 0, \\ 0, & \text{if } b = 0. \end{cases}$$

where $a(x) = \frac{\partial B}{\partial x}(x) f(x),\ b^T(x) = \frac{\partial B}{\partial x}(x) g(x),\ k > 0.$

Then: the set of initial states can be taken as $\mathcal{X}_0 = \{x \in \mathcal{X} \mid B(x) \leq 0\}$
the control law $u = k_0(x)$ is continuous in x,
and ensures safety for the closed-loop system $\dot{x} = f(x) + g(x)k_0(x)$

Set Invariance via Reciprocal Barrier Functions

- Definition of Reciprocal Barrier Functions (Ames et al, TAC 2017)

Definition: Let $\dot{x} = f(x)$ where f is locally Lipschitz, and a closet set defined as

$$\mathcal{C} = \{x \in \mathbb{R}^n : h(x) \geq 0\}$$
$$\partial\mathcal{C} = \{x \in \mathbb{R}^n : h(x) = 0\}$$
$$\text{Int}(\mathcal{C}) = \{x \in \mathbb{R}^n : h(x) > 0\}$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable.

A continuously differentiable function $B : \text{Int}(\mathcal{C}) \to \mathbb{R}$ is called a Reciprocal Barrier Function (RBF) for the set C if there exist class K functions $\alpha_1, \alpha_2, \alpha_3$ such that for all $x \in \text{Int}(\mathcal{C})$

$$\frac{1}{\alpha_1(h(x))} \leq B(x) \leq \frac{1}{\alpha_2(h(x))}$$

$$L_f B(x) \leq \alpha_3(h(x))$$

Set Invariance via Reciprocal Barrier Functions

- Set Invariance using Reciprocal Barrier Functions (Ames et al, TAC 2017)

Theorem: Given the set defined as

$$\mathcal{C} = \{x \in \mathbb{R}^n : h(x) \geq 0\}$$
$$\partial\mathcal{C} = \{x \in \mathbb{R}^n : h(x) = 0\}$$
$$\text{Int}(\mathcal{C}) = \{x \in \mathbb{R}^n : h(x) > 0\}$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable, if there exists a Reciprocal Barrier Function $B : \text{Int}(\mathcal{C}) \to \mathbb{R}$, then $\text{Int}(\mathcal{C})$ is forward invariant.

Set Invariance via Zeroing Barrier Functions

- Definition of Zeroing Barrier Functions (Ames et al, TAC 2017)

Definition: Let $\dot{x} = f(x)$ where f is locally Lipschitz, and a closet set defined as

$$\mathcal{C} = \{x \in \mathbb{R}^n : h(x) \geq 0\}$$
$$\partial\mathcal{C} = \{x \in \mathbb{R}^n : h(x) = 0\}$$
$$\mathrm{Int}(\mathcal{C}) = \{x \in \mathbb{R}^n : h(x) > 0\}$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable.

The function $h : \mathbb{R}^n \to \mathbb{R}$ is called a Zeroing Barrier Function (ZBF) for the set C if there exists an extended class K function α, and a set $\mathcal{D}$ such that $\mathcal{C} \subseteq \mathcal{D} \subset \mathbb{R}^n$ such that for all $x \in \mathcal{D}$

$$L_f h(x) \geq -\alpha(h(x))$$

Proposition: If h is a ZBF on the set $\mathcal{D}$, then the set C is forward invariant.

Reciprocal Control Barrier Functions

- Reciprocal Control Barrier Functions (Ames et al, TAC 2017)

Definition: Let $\dot{x} = f(x) + g(x)u$, where $f(x), g(x)$ are locally Lipschitz

$x \in \mathbb{R}^n, u \in U \subset \mathbb{R}^m$

A continuously differentiable function $B : \text{Int}(\mathcal{C}) \to \mathbb{R}$ is called a Reciprocal Control Barrier Function (RCBF) for the set C if there exist class K functions $\alpha_1, \alpha_2, \alpha_3$ such that for all $x \in \text{Int}(\mathcal{C})$

$$\frac{1}{\alpha_1(h(x))} \leq B(x) \leq \frac{1}{\alpha_2(h(x))}$$

$$\inf_{u \in U} [L_f B(x) + L_g B(x)u - a_3(h(x))] \leq 0$$

Let the set $K_{rcbf}(x) = \{u \in U : L_f B(x) + L_g B(x)u - a_3(h(x)) \leq 0\}$
Then any locally Lipschitz $u : \text{Int}(\mathcal{C}) \to U$ such that $u(x) \in K_{rcbf}(x)$ will render Int(C) a forward invariant set.

Zeroing Control Barrier Functions

- Zeroing Control Barrier Functions (Ames et al, TAC 2017)

Definition: Let $\dot{x} = f(x) + g(x)u$, where $f(x), g(x)$ are locally Lipschitz

$$x \in \mathbb{R}^n, u \in U \subset \mathbb{R}^m$$

A continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ is called a Zeroing Control Barrier Function (ZCBF) for the set defined as $\mathcal{C} = \{x \in \mathbb{R}^n : h(x) \geq 0\}$

$$\partial\mathcal{C} = \{x \in \mathbb{R}^n : h(x) = 0\} \quad \mathrm{Int}(\mathcal{C}) = \{x \in \mathbb{R}^n : h(x) > 0\}$$

if there exists an extended class K function α, and a set $\mathcal{D}$ such that $\mathcal{C} \subseteq \mathcal{D} \subset \mathbb{R}^n$ such that for all $x \in \mathcal{D}$

$$\sup_{u \in U} [L_f h(x) + L_g h(x) u + \alpha(h(x))] \geq 0, \quad \forall x \in \mathcal{D}$$

Combining Performance (CLFs) and Safety (CBFs) via QPs

- Let the following CBL-CBF QP

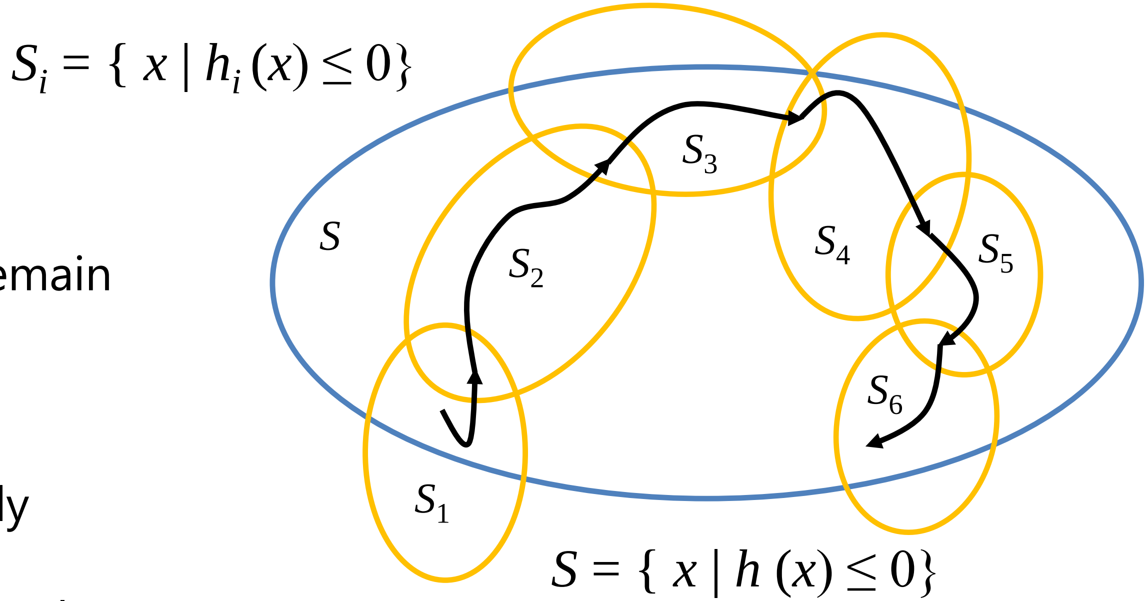
$$\mathbf{u}^\star(x) = \arg \min_{\mathbf{u}=(u,\delta)\in\mathbb{R}^m\times\mathbb{R}} \frac{1}{2}\mathbf{u}^T H(x)\mathbf{u} + F(x)^T\mathbf{u}$$

s.t.

$$L_f V(x) + L_g V(x)u + c_3 V(x) - \delta \leq 0$$
$$L_f B(x) + L_g B(x)u - \alpha(h(x)) \leq 0$$

- *Theorem 3 [Ames et al, TAC 2017]:* Suppose that the following functions are all locally Lipschitz: the vector fields *f* and *g* in the control system (21), the gradients of the RCBF *B* and CLF *V* , as well as the cost function terms *H*(*x*) and *F*(*x*) in (CLF-CBF QP). Suppose furthermore that the relative degree one condition, *LgB*(*x*) = 0 for all *x* ∈ Int(*C*), holds. Then the solution, **u**∗(*x*), of (CLF-CBF QP) is locally Lipschitz continuous for *x* ∈ Int(*C*). Moreover, a closed-form expression can be given for **u**∗(*x*).

Spatiotemporal (Safety- and Time-Critical) Control Synthesis via QPs

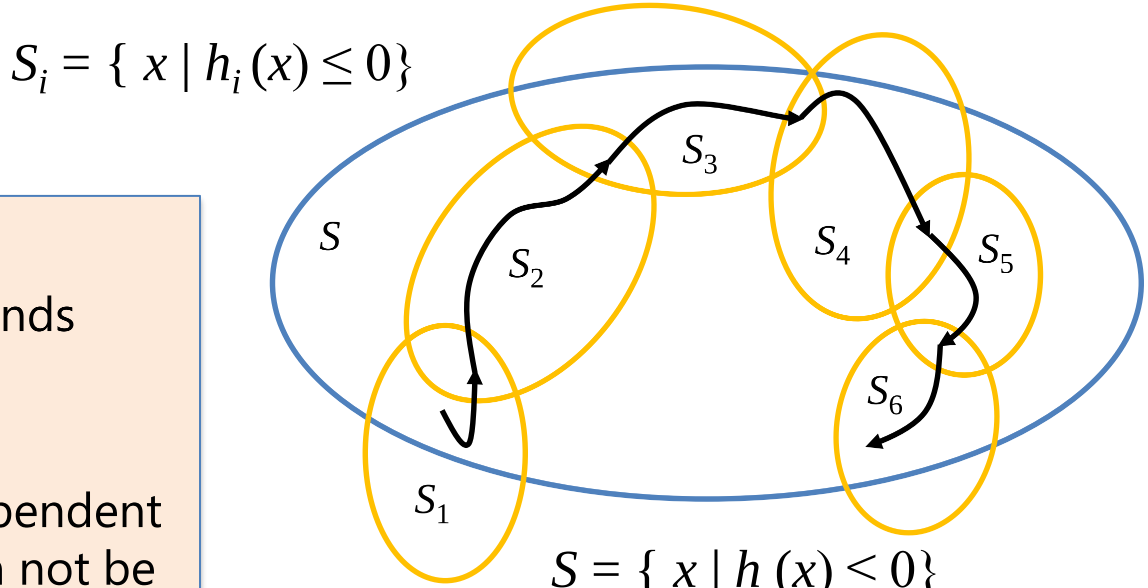


- Safety (invariance)
  - Trajectories must always remain in a safe set
- Performance (attractivity)
  - Trajectories must eventually reach desired sets, within **given/specified** time intervals
- Constraints
  - Input, state, dynamics

Question: How to synthesize CBFs for spatiotemporal specifications?

Approach: Quadratic Program (QP) that encodes **safety and FTS/FxTS**

Spatiotemporal (Safety- and Time-Critical) Control Synthesis via QPs



- **Finite-time** (FTS)
  - Time of convergence depends upon initial condition
- **Fixed-time** (FxTS)
  - Time of convergence independent of initial condition, but can not be predefined by the user
- **Prescribed-time** (PTS)
  - Time of convergence can be predefined by the user

Question: How to synthesize CBFs for spatiotemporal specifications?

Approach: Quadratic Program (QP) that encodes **safety and FTS/FxTS**

Finite-Time Stability (FTS) and Fixed-Time Stability (FxTS)

Finite-time Stability (FTS)

Theorem 1. *Suppose there exists a positive definite function V for system (1) such that*

$$\dot{V}(x) \leq -cV(x)^{\beta},$$

with $c > 0$ and $0 < \beta < 1$. Then, the origin of (1) is FTS with settling time function

$$T(x(0)) \leq \frac{V(x(0))^{1-\beta}}{c(1-\beta)}.$$

[1] (Bhat et al, 2000)

Let $\dot{x} = f(x)$
where f is continuous, $f(0) = 0$

Prescribed-time Stability (PTS)

Time of convergence T **can be chosen *a priori*** by the user. Also called predetermined-time or predefined-time.

Fixed-time Stability (FxTS)

Theorem 2 [2] *Suppose there exists a continuously differentiable, positive definite function $V : \mathcal{D} \to \mathbb{R}$, where $\mathcal{D} \subset \mathbb{R}^n$ is a neighborhood of the origin, for system (1) such that*

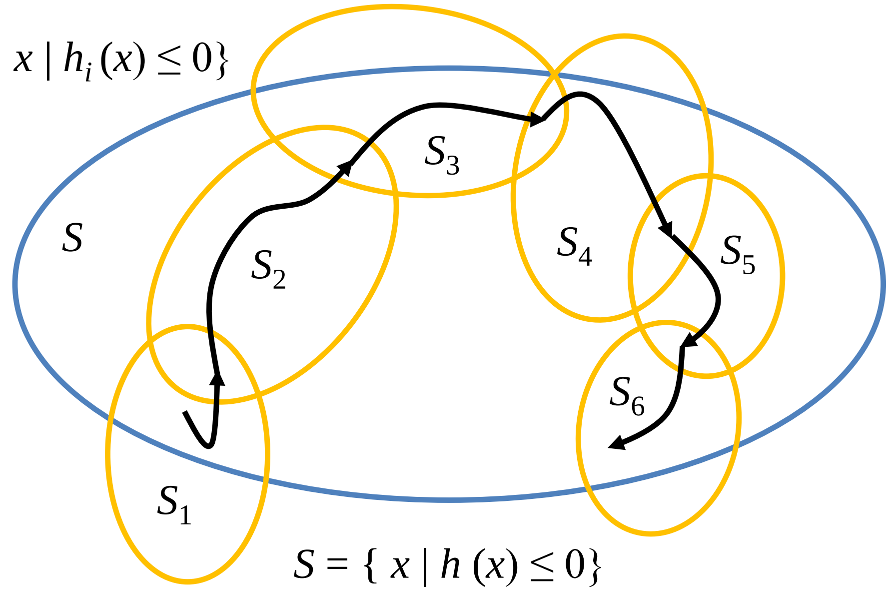
$$\dot{V}(x) \leq -(aV(x)^{\frac{\alpha}{\kappa}} + bV(x)^{\frac{\beta}{\kappa}})^{\kappa}, \quad (3)$$

with $a, b, \alpha, \beta > 0$, $\kappa\alpha < 1$ and $\kappa\beta > 1$. Then, the origin of (1) is FxTS with settling time function

$$T \leq \frac{1}{a^{\kappa}(1-\kappa\alpha)} + \frac{1}{b^{\kappa}(\kappa\beta - 1)}. \quad (4)$$

[2] (Polyakov et al, 2012)

Control Synthesis for Spatiotemporal Specifications



System dynamics:

$$\dot{x} = f(x) + g(x)u, \qquad x \in \mathbf{R}^n, u \in \mathbf{R}^m$$

Problem setup:

- Safe set $S_s = \{x \mid h(x) \le 0\}$, where $h(x)$ is C^1
- Sets $S_i = \{x \mid h_i(x) \le 0\}, i \in \Sigma = \{0, 1, 2, \cdots, N\}$, where $h_i(x)$ are C^1 functions
- $S_s \cap S_0 \neq \emptyset, S_i \cap S_{i+1} \neq \emptyset$, for $0 \le i \le N-1$
- Time intervals $[t_i, t_{i+1})$ such that $t_{i+1} - t_i \ge \bar{T} > 0$

Problem 1: Statement

Design the control input $u(t) \in U = \{A_u u \le b_u\}$, so that for $x(0) \in S_s \cap S_0$,

- $x(t) \in S_s$ for all $t \ge 0$
- $x(t) \in S_i$ for all $t \in [t_i, t_{i+1})$

Control Synthesis for Spatiotemporal Specifications

Theorem

If there exist $a_{i1}, a_{i2}, \lambda, \lambda_i > 0, \gamma_{i1} > 1, 0 < \gamma_{i2} < 1$ and control input u such that

$$\bar{T} \geq \max_{i\in\Sigma}\left\{\frac{1}{a_{i1}(\gamma_{i1}-1)} + \frac{1}{a_{i2}(1-\gamma_{i1})}\right\} \qquad (C_0)$$

$$\inf_{u\in U}\{L_f h + L_g h u + \lambda h\} \leq 0 \qquad (C_1)$$

$$\inf_{u\in U}\{L_f h_i + L_g h_i u + \lambda_i h_i\} \leq 0 \qquad (C_2)$$

$$\inf_{u\in U}\{L_f h_{i+1} + L_g h_{i+1} u\} \leq -a_{i1}\max\{0, h_{i+1}\}^{\gamma_{i1}} - a_{i2}\max\{0, h_{i+1}\}^{\gamma_{i2}} \quad (C_3)$$

hold for $t \in [t_i, t_{i+1})$, then, the control input $u(t)$ solves Problem 1.

- C_0 ensures exact convergence before $t = t_{i+1}$ (PTS)
- C_1 results into $h(x) = 0 \Rightarrow \dot{h}(x) \leq 0 \Rightarrow$ forward invariance of set S_s
- C_2 results into $h_i(x) = 0 \Rightarrow \dot{h}_i(x) \leq 0 \Rightarrow$ forward invariance of set S_i
- C_3 results into $\dot{h}_{i+1} \leq -a_{i1}h_{i+1}^{\gamma_{i1}} - a_{i2}h_{i+1}^{\gamma_{i2}} \Rightarrow$ FxTS to set S_{i+1}
- C_3 also results into forward invariance of S_{i+1} once $x(t) \in S_{i+1}$

K. Garg, D. Panagou. "Control-Lyapunov and Control-Barrier Functions based Quadratic Program for Spatio-temporal Specifications." 2019 IEEE Conference on Decision and Control, December 11-13, 2019, Nice, France

QP for Min-Norm Control and Spatiotemporal Specifications

Theorem

Let the solution to the following QP defined for $t \in [t_i, t_{i+1})$:

$$\min_{v, a_{i1}, a_{i2}, \lambda_i, \delta} \frac{1}{2} v^2$$

$$\text{s.t.} \qquad L_f h_i + L_g h_i v + \lambda_i h_i \leq 0,$$

$$L_f h_{i+1} + L_g h_{i+1} v \leq \delta h_{i+1} - a_{i1} \max\{0, h_{i+1}\}^{\gamma_{i1}} - a_{i2} \max\{0, h_{i+1}\}^{\gamma_{i2}},$$

$$A_u v \leq b_u,$$

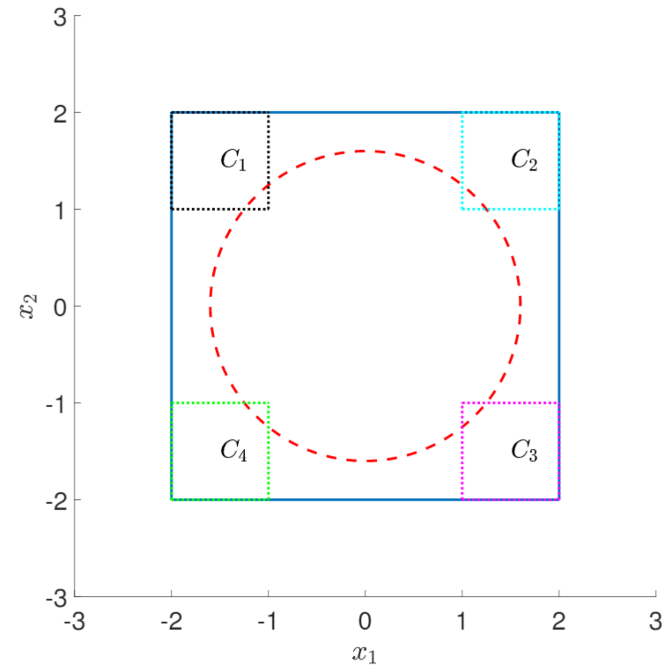
$$\frac{2}{\bar{T}} \leq a_{i1}(\gamma_{i1} - 1) \leq a_{i2}(1 - \gamma_{i2}),$$

be denoted as $[\bar{v}_i(t), a_{i1}, a_{i2}, \lambda_h, \lambda_i]$. Then, $u(t) = \bar{v}_i(t)$ for $t \in [t_i, t_{i+1})$ solves the considered problem.

Note: QPs can be solved very efficiently, can be used for real-time implementation

K. Garg, E. Arabi, D. Panagou. "Prescribed-time control under spatiotemporal and input constraints: A QP based approach," submitted to IEEE TAC, under review.

Example



System Dynamics:

$$\dot{x}_i = u_i$$

Objective encoded in STL:

$$(x_1, t) \models \; G_{[0,T_4]}\phi_s \wedge \; F_{[0,T_1]}\phi_2 \wedge F_{[T_1,T_2]}\phi_3 \wedge F_{[T_2,T_3]}\phi_4 \wedge F_{[T_3,T_4]}\phi_1$$
$$(x_2, t) \models \; G_{[0,T_4]}\phi_s \wedge \; F_{[0,T_1]}\phi_2 \wedge F_{[T_1,T_2]}\phi_1 \wedge F_{[T_2,T_3]}\phi_4 \wedge F_{[T_3,T_4]}\phi_3$$

Equivalently,

- Trajectories $x_1(t), x_2(t) \in S_s = \left\{x_i\,(t) \middle|\, \left|\left|x_i\right|\right|_{\infty} \leq 2, \left|\left|x_i\right|\right|_2 \geq 1.5\right\}$ for all $t \geq 0$, and
- $\left|\left|x_1\,(t) - x_2(t)\right|\right|_2 \geq d_s$ for all $t \geq 0$, and
- within a given $T_1 < \infty$, agents 1 and 2 should reach the square C_2,
- within a given $T_2 < \infty$, agents 1 and 2 should reach the square C_3,
- And so on

Example – Results