Orchestrating sensors for global hybrid robust stabilization of unicycles

Philipp Braun

School of Engineering,
Australian National University, Canberra, Australia

IFAC World Congress (Busan, 2026)

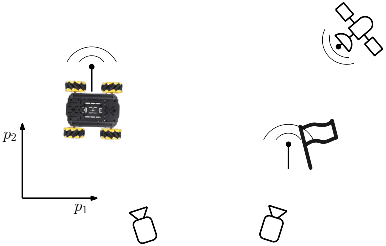
In Collaboration with:

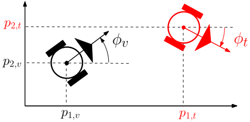
R. Ballaben:	Faculty of Engineering, University of Porto (Portugal)
A. Astolfi:	Applied Mathematics and Computational Science, King Abdullah University of Science and Technology (Saudi Arabia)
L. Zaccarian:	Dipartimento di Ingegneria Industriale, University of Trento (Italy); and Laboratoire d'Analyse et d'Architecture des Systèmes-Centre Nationnal de la Recherche Scientifique, Université de Toulouse, CNRS (France)

Australian National University

Ballaben, R., Astolfi, A., Braun, P., & Zaccarian, L. (2026b). *Orchestrating on-board sensors for global hybrid robust stabilization of unicycles* [Automatica]. https://hal.science/hal-05004452

Setting & Assumptions: What can we measure?

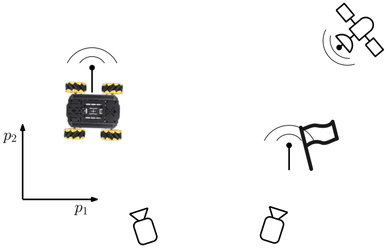


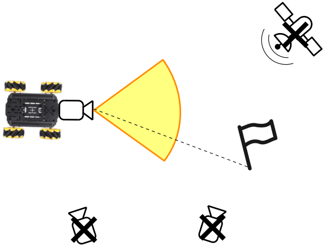


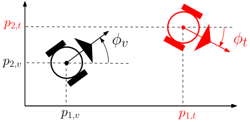
Cartesian coordinates

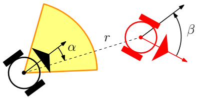
$$\begin{bmatrix} p_1 \\ p_2 \\ \phi \end{bmatrix} = \begin{bmatrix} p_{1,v} - p_{1,t} \\ p_{2,v} - p_{2,t} \\ \phi_v - \phi_t \end{bmatrix}$$

Setting & Assumptions: What can we measure?









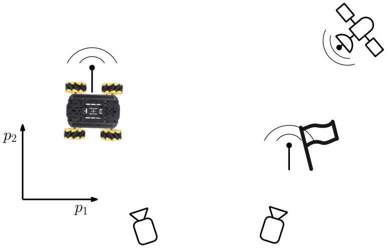
Cartesian coordinates

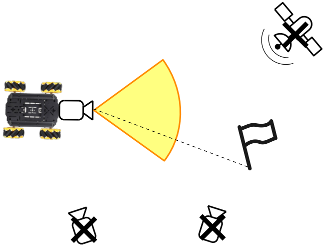
$$\begin{bmatrix} p_1 \\ p_2 \\ \phi \end{bmatrix} = \begin{bmatrix} p_{1,v} - p_{1,t} \\ p_{2,v} - p_{2,t} \\ \phi_v - \phi_t \end{bmatrix}$$

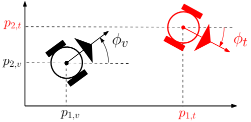
Camera-based coordinates

$$z = \begin{bmatrix} r \\ \beta \\ \alpha \end{bmatrix}, \quad \alpha \in [-\overline{\alpha}, \overline{\alpha}]$$

Setting & Assumptions: What can we measure?







Cartesian coordinates

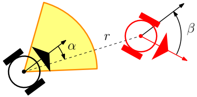
$$\begin{bmatrix} p_1 \\ p_2 \\ \phi \end{bmatrix} = \begin{bmatrix} p_{1,v} - p_{1,t} \\ p_{2,v} - p_{2,t} \\ \phi_v - \phi_t \end{bmatrix}$$

Coordinate transformation

$$\begin{bmatrix} p_1 \\ p_2 \\ \phi \end{bmatrix} = \gamma(z) := \begin{bmatrix} r\cos(\alpha - \beta - \pi) \\ r\sin(\alpha - \beta - \pi) \\ -\beta \end{bmatrix}$$

Stabilization using camera measurements

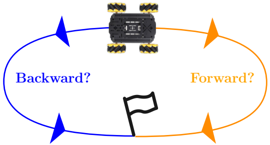
- If $(r, \alpha) \to 0$, then $(p_1, p_2) \to 0$
- Additionally, if $\beta \to 0$, then $\phi \to 0$

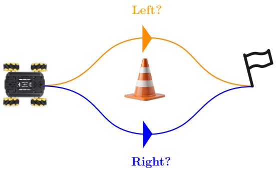


Camera-based coordinates

$$z = \begin{bmatrix} r \\ \beta \\ \alpha \end{bmatrix}, \quad \alpha \in [-\overline{\alpha}, \overline{\alpha}]$$

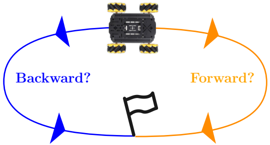
Decisions naturally arise when controlling a vehicle

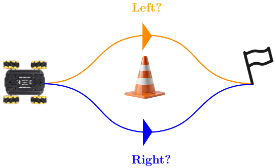




Decisions are discontinuous control actions...

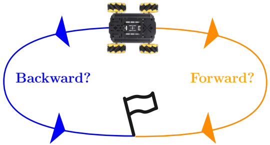
Decisions naturally arise when controlling a vehicle

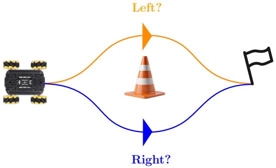




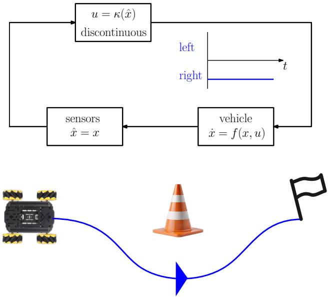
Decisions are discontinuous control actions... and they are necessary

Decisions naturally arise when controlling a vehicle



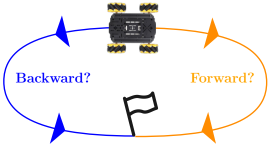


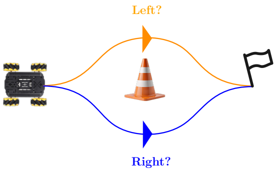
Decisions are discontinuous control actions... and they are necessary



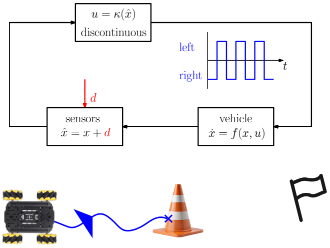
- Discontinuous controllers work well in theory

Decisions naturally arise when controlling a vehicle





Decisions are discontinuous control actions... and they are necessary



- Discontinuous controllers work well in theory but perturbations can be disruptive
- Discontinuous actions should be implemented in a **robust** way

Hybrid dynamical systems [Teel, Goebel, Sanfelice, 2012]: a tool for modeling discont. actions

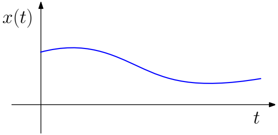
Continuous-time systems

$$\dot{x} = f(x), \; x \in \mathbb{R}^n$$

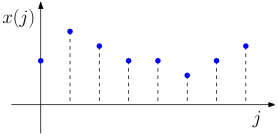
Discrete-time systems

$$x^+ = g(x), \; x \in \mathbb{R}^n$$

Model the dynamics of physical systems



Model instantaneous events

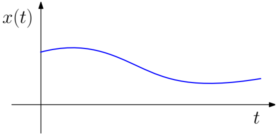


Hybrid dynamical systems [Teel, Goebel, Sanfelice, 2012]: a tool for modeling discont. actions

Continuous-time systems

$$\dot{x} = f(x), \ x \in \mathbb{R}^n$$

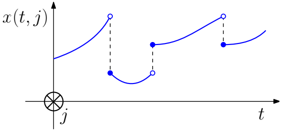
Model the dynamics of physical systems



Hybrid dynamical systems

$$\begin{cases} \dot{x} = f(x), & x \in \mathcal{C}, \\ x^+ = g(x), & x \in \mathcal{D}. \end{cases}$$

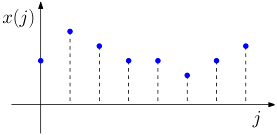
Combine both worlds



Discrete-time systems

$$x^+ = g(x), \ x \in \mathbb{R}^n$$

Model instantaneous events



Hybrid dynamical systems [Teel, Goebel, Sanfelice, 2012]: a tool for modeling discont. actions

Continuous-time systems

$\dot{x} = f(x), \; x \in \mathbb{R}^n$

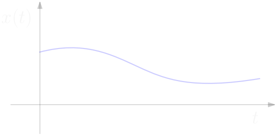
Hybrid dynamical systems

$$\begin{cases} \dot{x} &= f(x), \quad x \in \mathcal{C}, \\ x^+ &= g(x), \quad x \in \mathcal{D}. \end{cases}$$

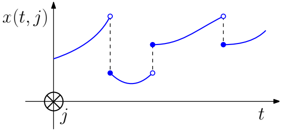
Discrete-time systems

$x^+ = g(x), \; x \in \mathbb{R}^n$

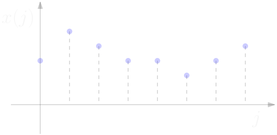
Model the dynamics of physical systems



Combine both worlds



Model instantaneous events



Terminology:

- f *flow map*
- $\mathcal{C}$ *flow set*
- g *jump map*
- $\mathcal{D}$ *jump set*

Can model:

- Instantaneous events

$$x^+ = g(x), \; x \in \mathcal{D}$$

- Discontinuous actions

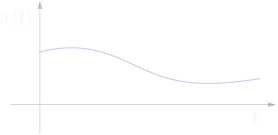
$q \in \{-1, 1\}$, q is a "logic" state variable

$$\dot{x} = f(x, q) = \begin{cases} f_+(x) & \text{if } q = 1, \\ f_-(x) & \text{if } q = -1. \end{cases}$$

Continuous-time systems

$\dot{x} = f(x), \; x \in \mathbb{R}^n$

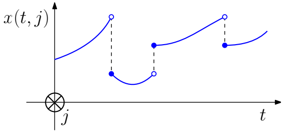
Model the dynamics of physical systems



Hybrid dynamical systems

$$\begin{cases} \dot{x} &= f(x), \quad x \in \mathcal{C}, \\ x^+ &= g(x), \quad x \in \mathcal{D}. \end{cases}$$

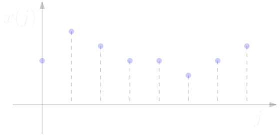
Combine both worlds



Discrete-time systems

$x^+ = g(x), \; x \in \mathbb{R}^n$

Model instantaneous events



Terminology:

- f *flow map*
- $\mathcal{C}$ *flow set*
- g *jump map*
- $\mathcal{D}$ *jump set*

Can model:

- Instantaneous events

$$x^+ = g(x), \; x \in \mathcal{D}$$

- Discontinuous actions

$q \in \{-1, 1\}$, q is a "logic" state variable

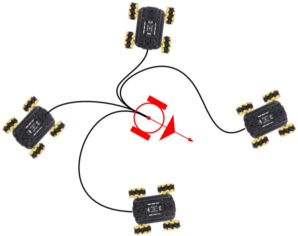
$$\dot{x} = f(x,q) = \begin{cases} f_+(x) & \text{if } q = 1, \\ f_-(x) & \text{if } q = -1. \end{cases}$$

Design guidelines (Hybrid Basic Conditions)

- $\mathcal{C}$, $\mathcal{D}$ closed
- f, g continuous

The HBC guarantees **robustness** of asymptotic stability to perturbations and uncertainties

Lyapunov functions: a tool for control design



- Goal: stabilize fw invariant compact set $\mathcal{A}$
- Given the equations of dynamics of vehicle

$$\dot{x} = f(x, u)$$

- Design hybrid control architecture s.t.

$$(\mathcal{H}) \begin{cases} \dot{x} &= f(x,u), & x \in \mathcal{C}, \\ u &= \kappa(x), & \\ x^{+} &= g(x), & x \in \mathcal{D}, \end{cases}$$

$\mathcal{A}$ is GAS for $(\mathcal{H})$

Lyapunov-based control design methodology

Choose

- $u = \kappa(x)$, $\mathcal{C}$ — The control law driving the vehicle
- $g(x), \mathcal{D}$ — The discontinuous action, when it happens
- $V(x)$ positive definite, radially unbounded — The Lyapunov function candidate

such that

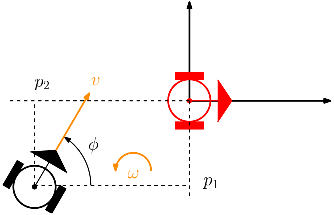
$$\dot{V}(x) := \nabla V(x)^{\top} f(x, \kappa(x)) \leq 0, \qquad \forall x \in \mathcal{C} \setminus \mathcal{A}$$
$$\Delta V(x) := V(g(x)) - V(x) \leq 0, \qquad \forall x \in \mathcal{D} \setminus \mathcal{A}$$

V is decreasing along solutions to $(\mathcal{H})$

+technical solution properties

then, $\mathcal{A}$ is Globally Asymptotically Stable (GAS)

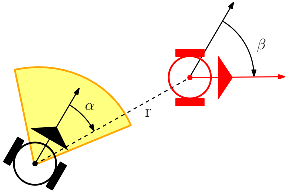
A kinematic unicycle model in the camera-based coordinates



- We consider, the motion of a UGV described by the kinematic equations

$$\begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} v\cos(\phi) \\ v\sin(\phi) \\ \omega \end{bmatrix},$$

$$u = [v \ \ \omega]^\top$$



- Exploiting the coordinate transformation $x = \gamma(z)$, the kinematic equations of a UGV in the camera-based coordinates are obtained

$$\begin{bmatrix} \dot{r} \\ \dot{\beta} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -\nu r\cos(\alpha) \\ -\omega \\ \nu\sin(\alpha) - \omega \end{bmatrix},$$

$$\alpha \in [-\overline{\alpha}, \overline{\alpha}], \quad u = [\nu \ \ \omega] = [\tfrac{v}{r} \ \ \omega]^\top$$

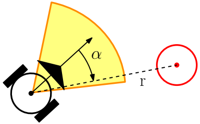
The measurements coming from one camera can be used to drive the vehicle to the target...

★ $\dot{z} = \begin{bmatrix} \dot{r} \\ \dot{\beta} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -\nu r \cos(\alpha) \\ -\omega \\ \nu \sin(\alpha) - \omega \end{bmatrix}, \qquad \alpha \in [-\overline{\alpha}, \overline{\alpha}], \;\; u = [\nu \;\; \omega] = [\tfrac{v}{r} \;\; \omega]$

The measurements coming from one camera can be used to drive the vehicle to the target...

$$\bigstar \quad \dot{z} = \begin{bmatrix} \dot{r} \\ \dot{\beta} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -\nu r \cos(\alpha) \\ -\omega \\ \nu \sin(\alpha) - \omega \end{bmatrix}, \qquad \alpha \in [-\overline{\alpha}, \overline{\alpha}], \ u = [\nu \ \omega] = [\tfrac{v}{r} \ \omega]$$

Camera measures (r, α)



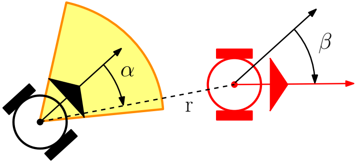
$$\kappa_0(r, \alpha) := \begin{bmatrix} k_r \cos(\alpha) \\ k_r \cos(\alpha)\sin(\alpha) + k_\alpha \alpha \end{bmatrix}$$

Theorem: Stabilization of target position

Let $k_r, k_\alpha \in \mathbb{R}_{>0}$ be arbitrary and $u = \kappa_0(r, \alpha)$. The set $\mathcal{A}_\phi = \{r = \alpha = 0\}$ is pre-asymptotically stable (pAS) for $\bigstar$.

$$V(z) = \frac{1}{2}(r^2 + \alpha^2)$$

Camera measures $z = (r, \beta, \alpha)$



$$\kappa(z) = \begin{bmatrix} k_r \cos(\alpha) \\ k_r \cos(\alpha)\sin(\alpha) + k_\alpha \alpha + k_\beta (\alpha - \beta) \frac{\cos(\alpha)\sin(\alpha)}{\alpha} \end{bmatrix}$$

Theorem: Stabilization of the target pose

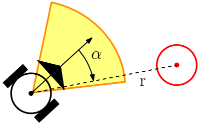
Let $k_r, k_\beta, k_\alpha \in \mathbb{R}_{>0}$ be arbitrary and $u = \kappa(z)$. The origin $\mathcal{A} = (0, 0, 0)$ is pAS for $\bigstar$.

$$V(z) = \frac{1}{2}\left(r^2 + \frac{k_\beta}{k_r}(\alpha - \beta)^2 + \alpha^2\right)$$

The measurements coming from one camera can be used to drive the vehicle to the target...

$$\bigstar \quad \dot{z} = \begin{bmatrix} \dot{r} \\ \dot{\beta} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -\nu r \cos(\alpha) \\ -\omega \\ \nu \sin(\alpha) - \omega \end{bmatrix}, \qquad \alpha \in [-\overline{\alpha}, \overline{\alpha}], \;\; u = [\nu \;\, \omega] = [\tfrac{v}{r} \;\, \omega]$$

Camera measures (r, α)



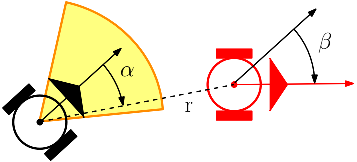
$$\kappa_0(r, \alpha) := \begin{bmatrix} k_r \cos(\alpha) \\ k_r \cos(\alpha)\sin(\alpha) + k_\alpha \alpha \end{bmatrix}$$

Theorem: Stabilization of target position

Let $k_r, k_\alpha \in \mathbb{R}_{>0}$ be arbitrary and $u = \kappa_0(r, \alpha)$. The set $\mathcal{A}_\phi = \{r = \alpha = 0\}$ is pre-asymptotically stable (pAS) for $\bigstar$.

$$V(z) = \frac{1}{2}(r^2 + \alpha^2)$$

Camera measures $z = (r, \beta, \alpha)$



$$\kappa(z) = \begin{bmatrix} k_r \cos(\alpha) \\ k_r \cos(\alpha)\sin(\alpha) + k_\alpha \alpha + k_\beta (\alpha - \beta) \frac{\cos(\alpha)\sin(\alpha)}{\alpha} \end{bmatrix}$$

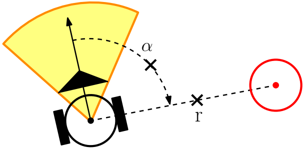
Theorem: Stabilization of the target pose

Let $k_r, k_\beta, k_\alpha \in \mathbb{R}_{>0}$ be arbitrary and $u = \kappa(z)$. The origin $\mathcal{A} = (0, 0, 0)$ is pAS for $\bigstar$.

$$V(z) = \frac{1}{2}\left(r^2 + \frac{k_\beta}{k_r}(\alpha - \beta)^2 + \alpha^2\right)$$

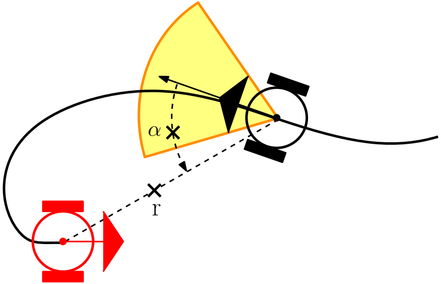
Solutions converge to the target if the target remains inside the field of view ($\alpha(t) \in [-\overline{\alpha}, \overline{\alpha}]$ for all t)

...but multiple cameras are required to achieve global stabilization



- The camera does not see the target for all possible configurations of the vehicle

$$\gamma(\mathcal{C} \cup \mathcal{D}) \neq \mathcal{X}$$



- The camera may lose vision of the target during the vehicle motion
- Not every solution to

$$\bigstar \; \dot{z} = f(z, u) := \begin{bmatrix} \dot{r} \\ \dot{\beta} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -\nu r \cos(\alpha) \\ -\omega \\ \nu \sin(\alpha) - \omega \end{bmatrix},$$

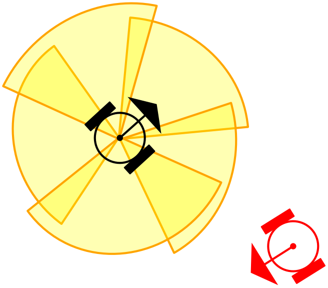
$$u = \kappa(z),$$

$$\alpha \in [-\overline{\alpha}, \overline{\alpha}]$$

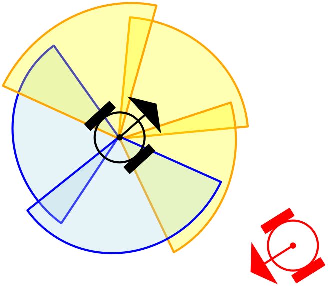
is complete ($t \mapsto z(t)$ defined $\forall t \geq 0$)

Measurements from multiple cameras must be combined to globally stabilize the target

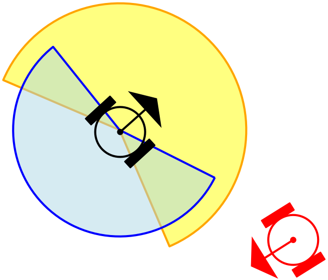
Multiple cameras can be combined into virtual front and rear camera, leading to forward and reverse motion



Multiple cameras can be combined into virtual front and rear camera, leading to forward and reverse motion



Multiple cameras can be combined into virtual front and rear camera, leading to forward and reverse motion

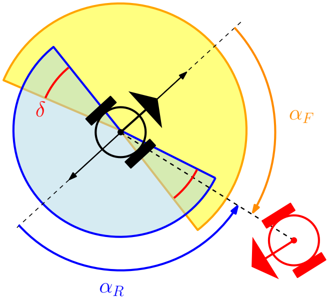


Multiple cameras can be combined into virtual front and rear camera, leading to forward and reverse motion

Two-cameras setup

Group cameras into a virtual front and rear camera

- Front camera measures $(r, \beta, \alpha_\mathsf{F})$ if $\alpha_\mathsf{F} \in [-\frac{\pi}{2} - \delta, \frac{\pi}{2} + \delta]$
- Rear camera measures $(r, \beta, \alpha_\mathsf{R})$ if $\alpha_\mathsf{R} \in [-\frac{\pi}{2} - \delta, \frac{\pi}{2} + \delta]$
- $\delta \in (0, \frac{\pi}{2})$ defines overlapping field of view of cameras

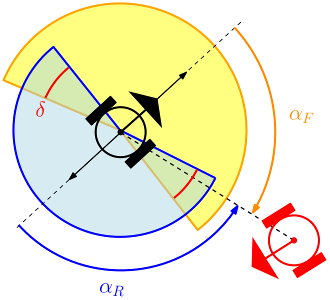


Multiple cameras can be combined into virtual front and rear camera, leading to forward and reverse motion

Two-cameras setup

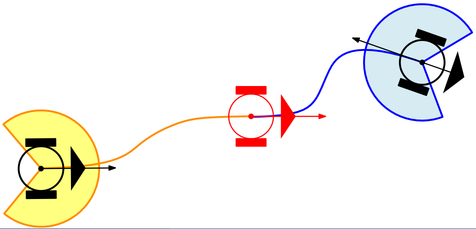
Group cameras into a virtual front and rear camera

- Front camera measures $(r, \beta, \alpha_{\mathrm{F}})$ if $\alpha_{\mathrm{F}} \in [-\frac{\pi}{2} - \delta, \frac{\pi}{2} + \delta]$
- Rear camera measures $(r, \beta, \alpha_{\mathrm{R}})$ if $\alpha_{\mathrm{R}} \in [-\frac{\pi}{2} - \delta, \frac{\pi}{2} + \delta]$
- $\delta \in (0, \frac{\pi}{2})$ defines overlapping field of view of cameras

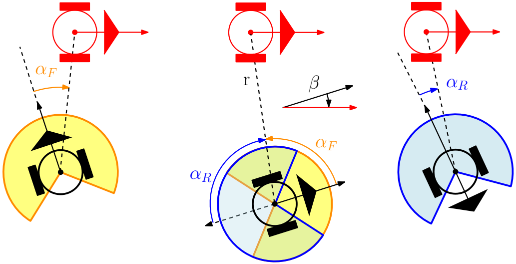


Discontinuous control action

- Front camera sees the target $\Rightarrow \alpha_{\mathrm{F}} \to 0 \Rightarrow$ forward motion $(v_1 > 0)$
- Rear camera sees the target $\Rightarrow \alpha_{\mathrm{R}} \to 0 \Rightarrow$ reverse motion $(v_1 < 0)$



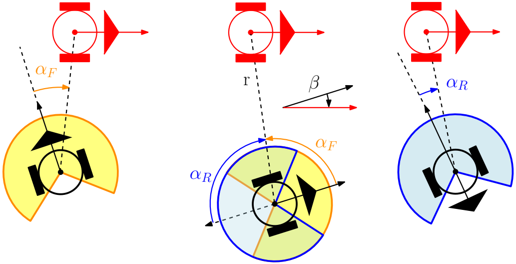
The target can be stabilized using front and rear camera measurements



- Two sets of coordinates $z_F = (r, \beta, \alpha_F)$ and $z_R = (r, \beta, \alpha_R)$

$$\alpha_i \in [-\tfrac{\pi}{2} - \delta, \tfrac{\pi}{2} + \delta], \; i \in \{F, R\}, \quad \delta \in (0, \tfrac{\pi}{2})$$

The target can be stabilized using front and rear camera measurements



- Two sets of coordinates $z_F = (r, \beta, \alpha_F)$ and $z_R = (r, \beta, \alpha_R)$

$$\alpha_i \in [-\tfrac{\pi}{2} - \delta, \tfrac{\pi}{2} + \delta], \; i \in \{F, R\}, \quad \delta \in (0, \tfrac{\pi}{2})$$

- Two sets of kinematic equations

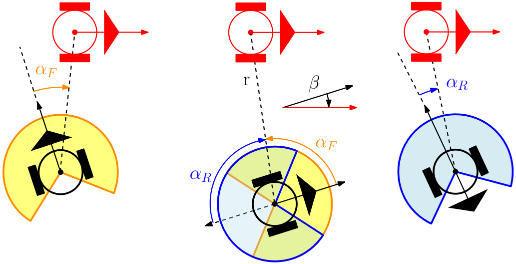
★ $\dot{z}_F = \begin{bmatrix} \dot{r} \\ \dot{\beta}_F \\ \dot{\alpha}_F \end{bmatrix} = \begin{bmatrix} -\nu r \cos(\alpha_F) \\ -\omega \\ +\nu \sin(\alpha_F) - \omega \end{bmatrix}$

$\alpha_F \in [-\frac{\pi}{2} - \delta, \frac{\pi}{2} + \delta]$

★ $\dot{z}_R = \begin{bmatrix} \dot{r} \\ \dot{\beta}_R \\ \dot{\alpha}_R \end{bmatrix} = \begin{bmatrix} +\nu r \cos(\alpha_R) \\ -\omega \\ -\nu \sin(\alpha_R) - \omega \end{bmatrix}$

$\alpha_R \in [-\frac{\pi}{2} - \delta, \frac{\pi}{2} + \delta]$

The target can be stabilized using front and rear camera measurements



- Two sets of coordinates $z_F = (r, \beta, \alpha_F)$ and $z_R = (r, \beta, \alpha_R)$

$$\alpha_i \in [-\tfrac{\pi}{2} - \delta, \tfrac{\pi}{2} + \delta], \; i \in \{F, R\}, \quad \delta \in (0, \tfrac{\pi}{2})$$

- Two sets of kinematic equations

$$\star \; \dot{z}_F = \begin{bmatrix} \dot{r} \\ \dot{\beta}_F \\ \dot{\alpha}_F \end{bmatrix} = \begin{bmatrix} -\nu r \cos(\alpha_F) \\ -\omega \\ +\nu \sin(\alpha_F) - \omega \end{bmatrix} \qquad \star \; \dot{z}_R = \begin{bmatrix} \dot{r} \\ \dot{\beta}_R \\ \dot{\alpha}_R \end{bmatrix} = \begin{bmatrix} +\nu r \cos(\alpha_R) \\ -\omega \\ -\nu \sin(\alpha_R) - \omega \end{bmatrix}$$

$$\alpha_F \in [-\tfrac{\pi}{2} - \delta, \tfrac{\pi}{2} + \delta] \qquad \alpha_R \in [-\tfrac{\pi}{2} - \delta, \tfrac{\pi}{2} + \delta]$$

- Two stabilizers $i \in \{F, R\}$

Target position stabilizer

$$\kappa_0(z_F / z_R) = \begin{bmatrix} +/- k_r \cos(\alpha_i) \\ k_r \cos(\alpha_i) \sin(\alpha_i) + k_\alpha \alpha_i \end{bmatrix}$$

Target pose stabilizer

$$\kappa(z_F / z_R) = \begin{bmatrix} +/- k_r \cos(\alpha_i) \\ k_r \cos(\alpha_i) \sin(\alpha_i) + k_\alpha \alpha_i + k_\beta (\alpha_i - \beta) \frac{\cos(\alpha_i) \sin(\alpha_i)}{\alpha_i} \end{bmatrix}$$

A hybrid control architecture can combine front and rear camera measurements

- A logic variable $q \in \{-1, 1\}$ can be used to describe the camera used
- Hybrid state combines camera measurements with logic variable

$$\xi = (r, \beta, \alpha, q)$$

$$q = 1 \Rightarrow (r, \beta, \alpha) = z_{\mathrm{F}} \qquad q = -1 \Rightarrow (r, \beta, \alpha) = z_{\mathrm{R}}$$

A hybrid control architecture can combine front and rear camera measurements

- A logic variable $q \in \{-1, 1\}$ can be used to describe the camera used
- Hybrid state combines camera measurements with logic variable

$$\xi = (r, \beta, \alpha, q)$$

$$q = 1 \Rightarrow (r, \beta, \alpha) = z_{\mathrm{F}} \qquad\qquad q = -1 \Rightarrow (r, \beta, \alpha) = z_{\mathrm{R}}$$

- Logic variable q unifies equations of motion and feedback selections into **flow dynamics**

$$\dot{\xi} = \begin{bmatrix} \dot{r} \\ \dot{\beta} \\ \dot{\alpha} \\ \dot{q} \end{bmatrix} = f(\xi, u) := \begin{bmatrix} -q\nu r \cos(\alpha) \\ -\omega \\ q\nu \sin(\alpha) - \omega \\ 0 \end{bmatrix}, \xi \in \mathcal{C}, \quad u = \kappa(\xi) = \begin{bmatrix} qk_r \cos(\alpha) \\ k_r \cos(\alpha)\sin(\alpha) + k_\alpha \alpha + k_\beta (\alpha - \beta) \frac{\cos(\alpha)\sin(\alpha)}{\alpha} \end{bmatrix},$$

$$\mathcal{C} = \{\xi : |\alpha| \leq \tfrac{\pi}{2} + \delta\}$$

A hybrid control architecture can combine front and rear camera measurements

- A logic variable $q \in \{-1, 1\}$ can be used to describe the camera used
- Hybrid state combines camera measurements with logic variable

$$\xi = (r, \beta, \alpha, q)$$

$$q = 1 \Rightarrow (r, \beta, \alpha) = z_{\mathrm{F}} \qquad\qquad q = -1 \Rightarrow (r, \beta, \alpha) = z_{\mathrm{R}}$$

- Logic variable q unifies equations of motion and feedback selections into **flow dynamics**

$$\dot{\xi} = \begin{bmatrix} \dot{r} \\ \dot{\beta} \\ \dot{\alpha} \\ \dot{q} \end{bmatrix} = f(\xi, u) := \begin{bmatrix} -q\nu r \cos(\alpha) \\ -\omega \\ q\nu \sin(\alpha) - \omega \\ 0 \end{bmatrix}, \xi \in \mathcal{C}, \quad u = \kappa(\xi) = \begin{bmatrix} qk_r \cos(\alpha) \\ k_r \cos(\alpha)\sin(\alpha) + k_\alpha \alpha + k_\beta(\alpha - \beta)\frac{\cos(\alpha)\sin(\alpha)}{\alpha} \end{bmatrix},$$

$$\mathcal{C} = \{\xi : |\alpha| \leq \tfrac{\pi}{2} + \delta\}$$

- **Jump dynamics** defines the switch between cameras (the discontinuous action)

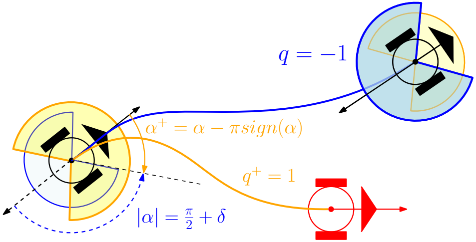
When cameras are switched

$$\xi \in \mathcal{D} := \{\xi : |\alpha| \geq \tfrac{\pi}{2} + \delta\}$$

How measurements change

$$\xi^+ = g(\xi) := \begin{bmatrix} r \\ \beta \\ \alpha - \pi \operatorname{sign}(\alpha) \\ -q \end{bmatrix}$$

$\delta \in (0, \frac{\pi}{2})$ ensures **hysteresis** in the camera switch



Reset of vehicle orientation

- Hybrid state combines camera measurements with logic variable

$$\xi = (r, \beta, \alpha, q), \quad \alpha \in [-\tfrac{\pi}{2} - \delta, \tfrac{\pi}{2} + \delta], \quad \beta \in [-\tfrac{3}{2}\pi - \delta, \tfrac{3}{2}\pi + \delta]$$

$$q = 1 \Rightarrow (r, \beta, \alpha) = z_F \qquad\qquad q = -1 \Rightarrow (r, \beta, \alpha) = z_R$$

- Logic variable q unifies equations of motion and feedback selections into flow dynamics

$$\dot{\xi} = \begin{bmatrix} \dot{r} \\ \dot{\beta} \\ \dot{\alpha} \\ \dot{q} \end{bmatrix} = f(\xi, u) := \begin{bmatrix} -q\nu r\cos(\alpha) \\ -\omega \\ q\nu\sin(\alpha) - \omega \\ 0 \end{bmatrix}, \xi \in \mathcal{C}, \quad u = \kappa(\xi) = \begin{bmatrix} qk_r\cos(\alpha) \\ k_r\cos(\alpha)\sin(\alpha) + k_\alpha\alpha + k_\beta(\alpha - \beta)\frac{\cos(\alpha)\sin(\alpha)}{\alpha} \end{bmatrix},$$

$$\mathcal{C} = \{\xi : |\alpha| \leq \tfrac{\pi}{2} + \delta, \beta \in [-\tfrac{3}{2}\pi - \delta, \tfrac{3}{2}\pi + \delta]\}$$

- Jump dynamics

Camera switch

$$\mathcal{D}_\alpha := \{\xi \in \Xi : |\alpha| \geq \tfrac{\pi}{2} + \delta\}$$

$$g_\alpha(\xi) := \begin{bmatrix} r \\ \beta \\ \alpha - \pi\,\mathrm{sign}(\alpha) \\ -q \end{bmatrix}$$

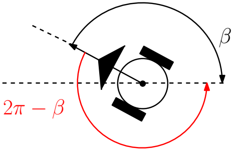
Orientation reset

$$\mathcal{D}_\beta := \{\xi \in \Xi : |\beta| \geq \tfrac{3}{2}\pi + \delta\}$$

$$g_\beta(\xi) := \begin{bmatrix} r \\ \beta - 2\pi\,\mathrm{sign}(\beta) \\ \alpha \\ q \end{bmatrix}$$

$$\xi^+ \in G(\xi) = \begin{cases} \{g_\beta(\xi)\}, & \text{if } \xi \in \mathcal{D}_\beta \backslash \mathcal{D}_\alpha, \\ \{g_\alpha(\xi)\}, & \text{if } \xi \in \mathcal{D}_\alpha \backslash \mathcal{D}_\beta, \\ \{g_\alpha(\xi)\} \cup \{g_\beta(\xi)\}, & \text{if } \xi \in \mathcal{D}_\alpha \cap \mathcal{D}_\beta, \end{cases}$$

$$\mathcal{D} := \mathcal{D}_\alpha \cup \mathcal{D}_\beta$$



Main result: global stabilization of target using camera measurements

- Closed-loop hybrid system

$$(\mathcal{H}) \begin{cases} \dot{\xi} = f(\xi, \kappa(\xi)), & \xi \in \mathcal{C}, \\ \xi^+ \in g(\xi), & \xi \in \mathcal{D}. \end{cases}$$

- The Lyapunov function candidate

$$V(\xi) = \frac{1}{2}\left(r^2 + \frac{k_\beta}{k_r}(\alpha - \beta)^2 + \alpha^2\right)$$

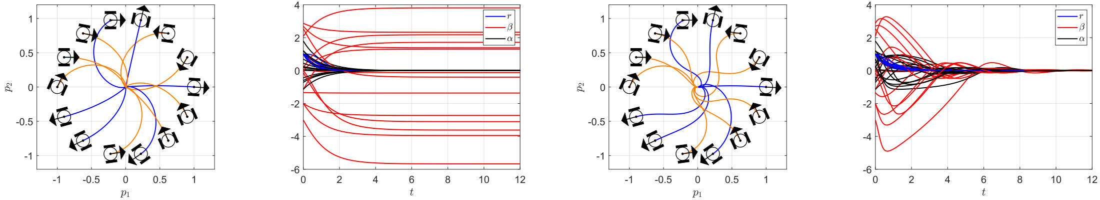
Lemma: camera invariance

All solutions to $(\mathcal{H})$ jump from $\mathcal{D}_\alpha$ at most once.

Theorem: Target stabilization via two-cameras architecture

For any selection of control gains $k_r, k_\alpha \in \mathbb{R}_{>0}, k_\beta \in \mathbb{R}_{\geq 0}$ and $\delta \in (0, \frac{\pi}{2})$

- $\mathcal{A}_\phi = \{r = \alpha = 0\}$ is GAS for $(\mathcal{H})$
- If $k_\beta > 0$, $\mathcal{A} = \{r = \alpha = \beta = 0\}$ is GAS for $(\mathcal{H})$
- All solutions are complete



Robustness of GAS is guaranteed by hybrid well-posedness

$$(\mathcal{H}) \begin{cases} \dot{\xi} &= f(\xi, \kappa(\xi)), \quad \xi \in \mathcal{C}, \\ \xi^+ &\in g(\xi), \qquad\quad \xi \in \mathcal{D}. \end{cases}$$

$(\mathcal{H})$ satisfies the HBC

HBC guarantee robustness of GAS

Theorem: Robustness to perturbations, measurement noise, and switch uncertainties

There exists a non-trivial perturbation function $\xi \mapsto \mu(\xi)$ such that $\mathcal{A}$ is GAS for the perturbed system $(\mathcal{H}_\mu)$

$$(\mathcal{H}_\mu) \begin{cases} \dot{\xi} &= f(\xi, \kappa(\xi)), \quad \xi \in \mathcal{C}, \\ \xi^+ &\in g(\xi), \qquad\quad \xi \in \mathcal{D}. \end{cases}$$

Robustness of GAS is guaranteed by hybrid well-posedness

$$(\mathcal{H}) \begin{cases} \dot{\xi} = f(\xi, \kappa(\xi)), & \xi \in \mathcal{C}, \\ \xi^+ \in g(\xi), & \xi \in \mathcal{D}. \end{cases}$$

$(\mathcal{H})$ satisfies the HBC

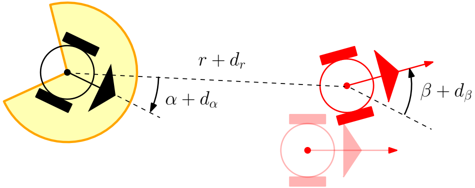
HBC guarantee robustness of GAS

Theorem: Robustness to perturbations, measurement noise, and switch uncertainties

There exists a non-trivial perturbation function $\xi \mapsto \mu(\xi)$ such that $\mathcal{A}$ is GAS for the perturbed system $(\mathcal{H}_\mu)$

$$(\mathcal{H}_\mu) \begin{cases} \dot{\xi} \in f(\xi, \kappa(\xi+\mu(\xi)\mathbb{B})), & \xi \in \mathcal{C}, \\ \xi^+ \in g(\xi), & \xi \in \mathcal{D}. \end{cases}$$

- **Measurement errors in the feedback**



Robustness of GAS is guaranteed by hybrid well-posedness

$$(\mathcal{H}) \begin{cases} \dot{\xi} & = & f(\xi, \kappa(\xi)), & \xi \in \mathcal{C}, \\ \xi^+ & \in & g(\xi), & \xi \in \mathcal{D}. \end{cases}$$

$(\mathcal{H})$ satisfies the HBC

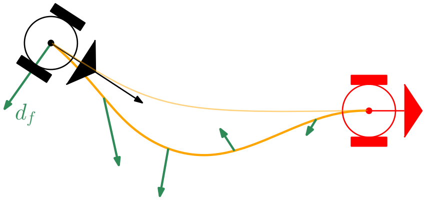
HBC guarantee robustness of GAS

Theorem: Robustness to perturbations, measurement noise, and switch uncertainties

There exists a non-trivial perturbation function $\xi \mapsto \mu(\xi)$ such that $\mathcal{A}$ is GAS for the perturbed system $(\mathcal{H}_\mu)$

$$(\mathcal{H}_\mu) \begin{cases} \dot{\xi} & \in & f(\xi, \kappa(\xi+\mu(\xi)\mathbb{B}))+\mu(\xi)\mathbb{B}, & \xi \in \mathcal{C}, \\ \xi^+ & \in & g(\xi)+\mu(\xi)\mathbb{B}, & \xi \in \mathcal{D}. \end{cases}$$

- Measurement errors in the feedback
- **Arbitrary external perturbations**



Robustness of GAS is guaranteed by hybrid well-posedness

$$(\mathcal{H}) \begin{cases} \dot{\xi} &= f(\xi, \kappa(\xi)), & \xi \in \mathcal{C}, \\ \xi^+ &\in g(\xi), & \xi \in \mathcal{D}. \end{cases}$$

$(\mathcal{H})$ satisfies the HBC

HBC guarantee robustness of GAS

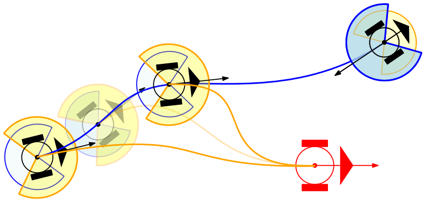
Theorem: Robustness to perturbations, measurement noise, and switch uncertainties

There exists a non-trivial perturbation function $\xi \mapsto \mu(\xi)$ such that $\mathcal{A}$ is GAS for the perturbed system $(\mathcal{H}_\mu)$

$$(\mathcal{H}_\mu) \begin{cases} \dot{\xi} &\in f(\xi, \kappa(\xi+\mu(\xi)\mathbb{B}))+\mu(\xi)\mathbb{B}, & \xi \in \mathcal{C}_\mu, \\ \xi^+ &\in g(\xi)+\mu(\xi)\mathbb{B}, & \xi \in \mathcal{D}_\mu. \end{cases}$$

- Measurement errors in the feedback
- Arbitrary external perturbations
- **Uncertainties affecting the switch time**

$$\mathcal{C}_\mu = \{\xi : |\alpha| \leq \tfrac{\pi}{2} + \delta + \mu(\xi)\},$$
$$\mathcal{D}_\mu = \{\xi : |\alpha| \leq \tfrac{\pi}{2} + \delta - \mu(\xi)\}.$$



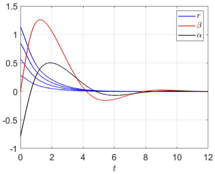
Stability properties in the Cartesian coordinates

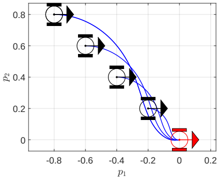
AS in the polar coordinates $z = (r, \beta, \alpha)$ $\Rightarrow$? AS in the Cartesian coordinates $x = (p_1, p_2, \phi)$
The Cartesian state x can be seen as an output of the system

$$\bigstar \quad \begin{cases} \dot{z} & = f(z, \kappa(z)) \\ x & = \gamma(z) \end{cases}$$

- $|x|^2 = p_1^2 + p_2^2 + \phi^2 \overset{x=\gamma(z)}{=} r^2 + \beta^2 \leq |z|^2$
  $\Rightarrow |z(t)| \to 0$ implies $|x(t)| \to 0$ $\Rightarrow$ **attractivity**

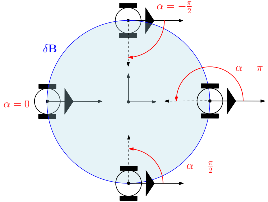
- $|x(0)| \leq \delta_x$ does **NOT** imply $|z(0)| \leq \delta_z$!
  $\Rightarrow$ large $|z(0)|$ can lead to $\beta(t) = -\phi(t)$ overshoot!





- The coordinate transformation $x = \gamma(z)$ is not a **homeomorphism**!

$$(r, \beta, \alpha) = \gamma^{-1}(\delta\mathbb{B}) = [0, \delta] \times [0, \delta] \times (-\pi, \pi]$$



Stabilization in polar coordinates guarantees convergence in Cartesian coordinates

Recent Extensions

Ballaben, R., Astolfi, A., Braun, P., & Zaccarian, L. (2025). *Towards global stabilization of a hovercraft model using hybrid systems and discontinuous feedback laws* [CDC].

Ballaben, R., Astolfi, A., Braun, P., & Zaccarian, L. (2026a). *Hybrid orchestration of onboard cameras for global stabilization of a hovercraft* [Conditionally accepted: IFAC Nonlinear Analysis: Hybrid Systems].