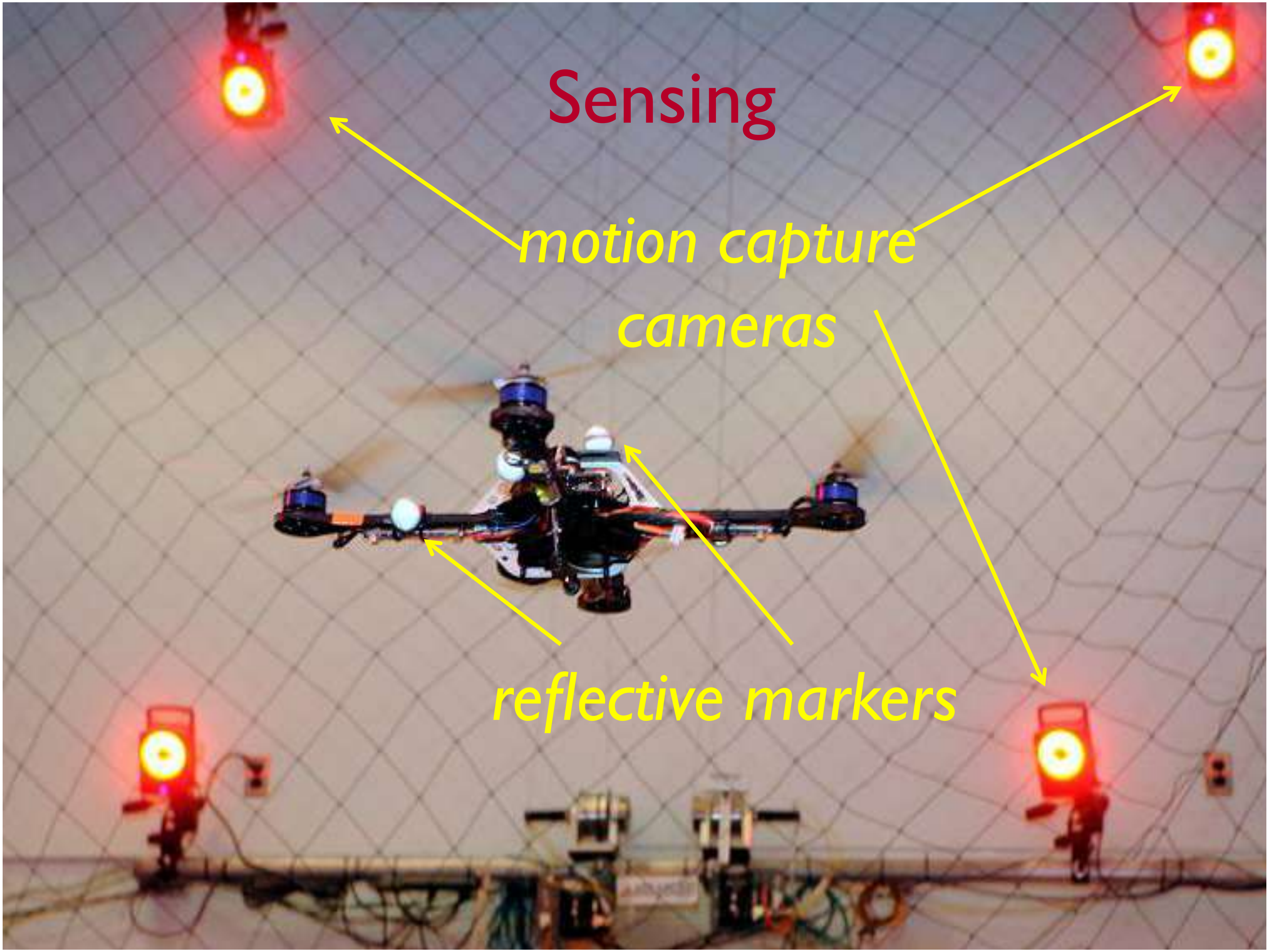
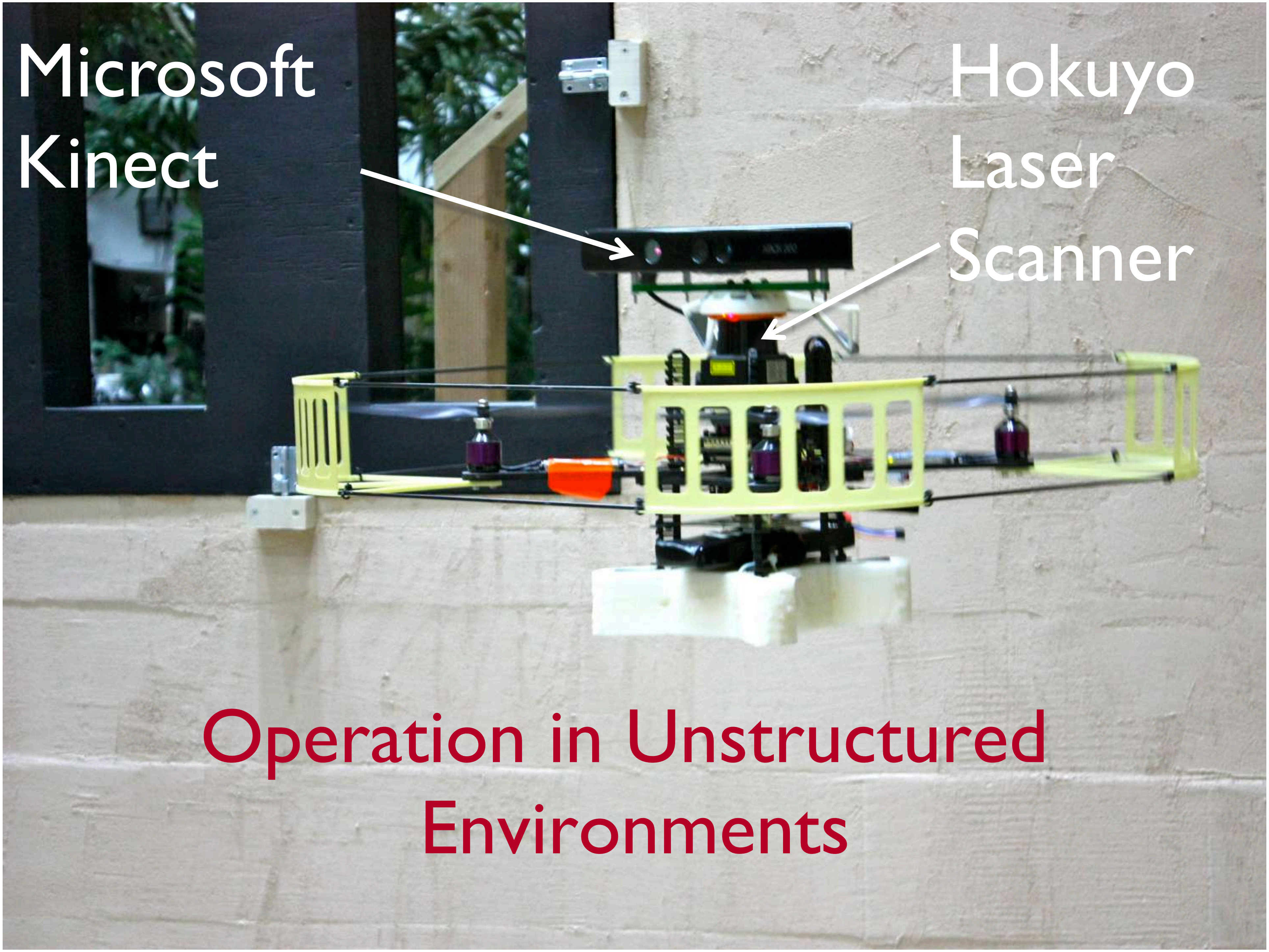
Sensing and Estimation

Sensing

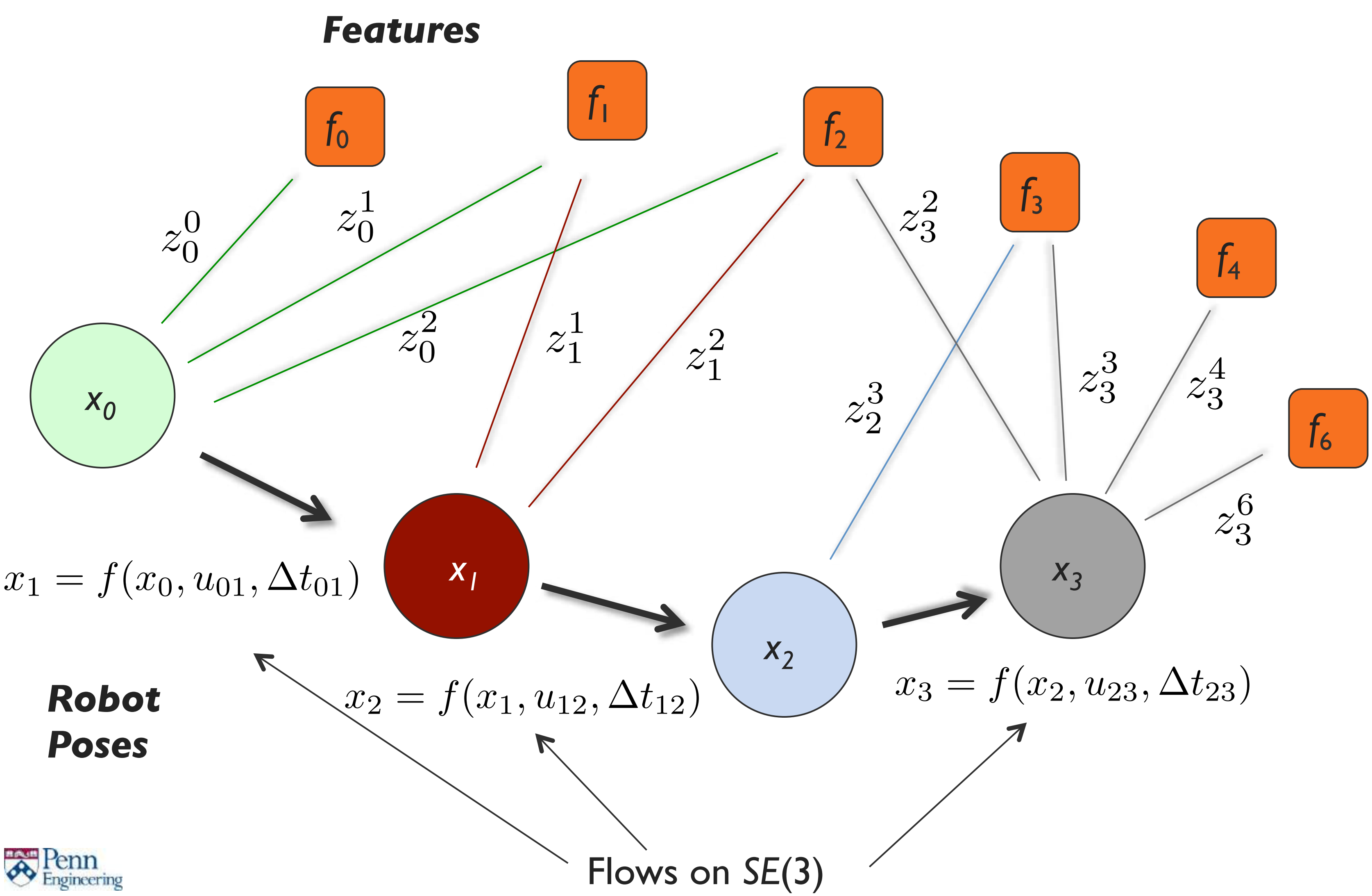


Onboard State Estimation

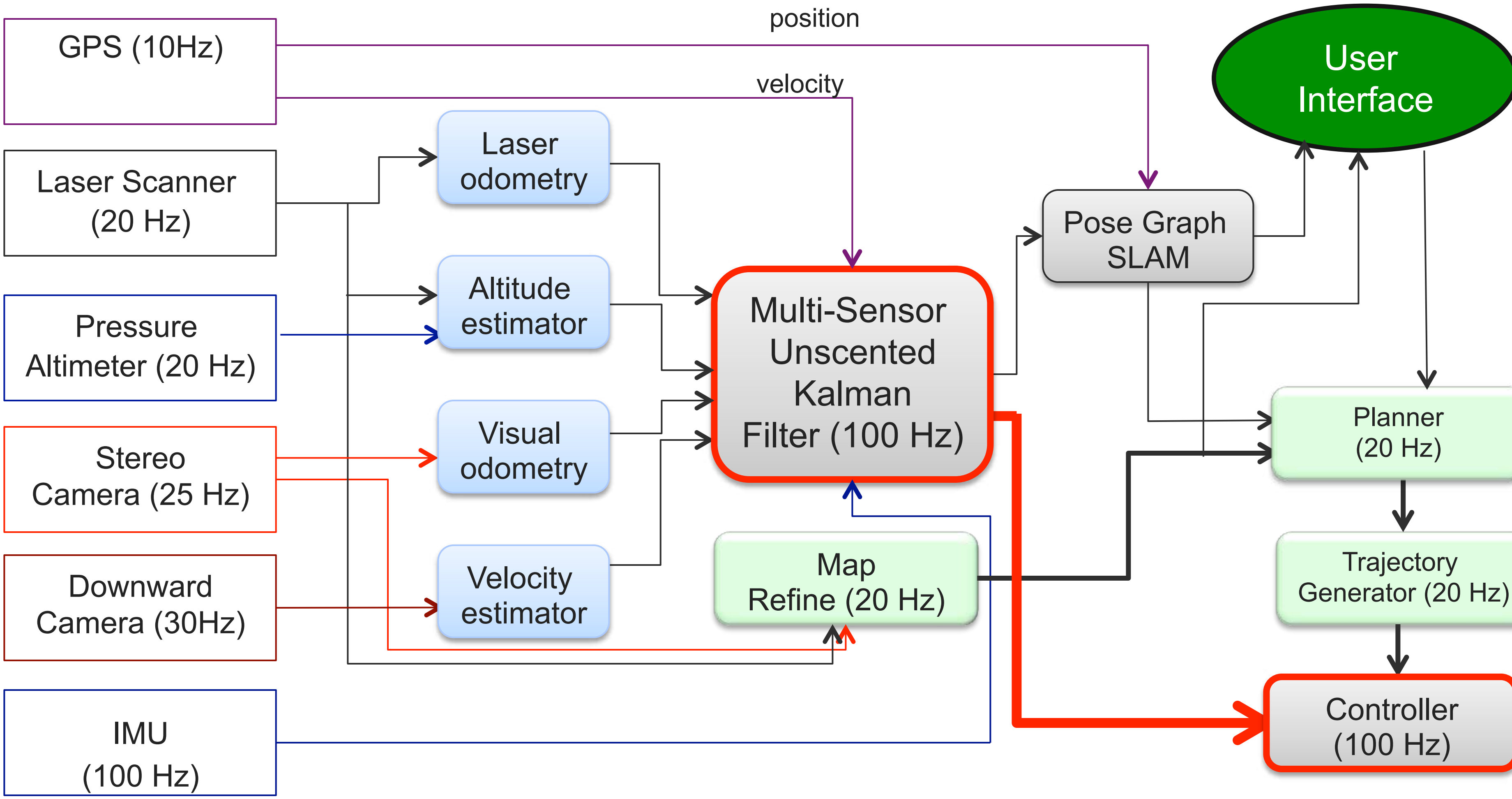

Microsoft
Kinect
Hokuyo
Laser
Scanner
Operation in Unstructured
Environments

Shaojie Shen, Yash Mulgaonkar, Nathan Michael and Vijay Kumar, "Multi-Sensor Fusion for Robust Autonomous Flight in Indoor and Outdoor Environments with a Rotorcraft MAV," *Proceedings of IEEE International Conference on Robotics and Automation* (ICRA), 2014.

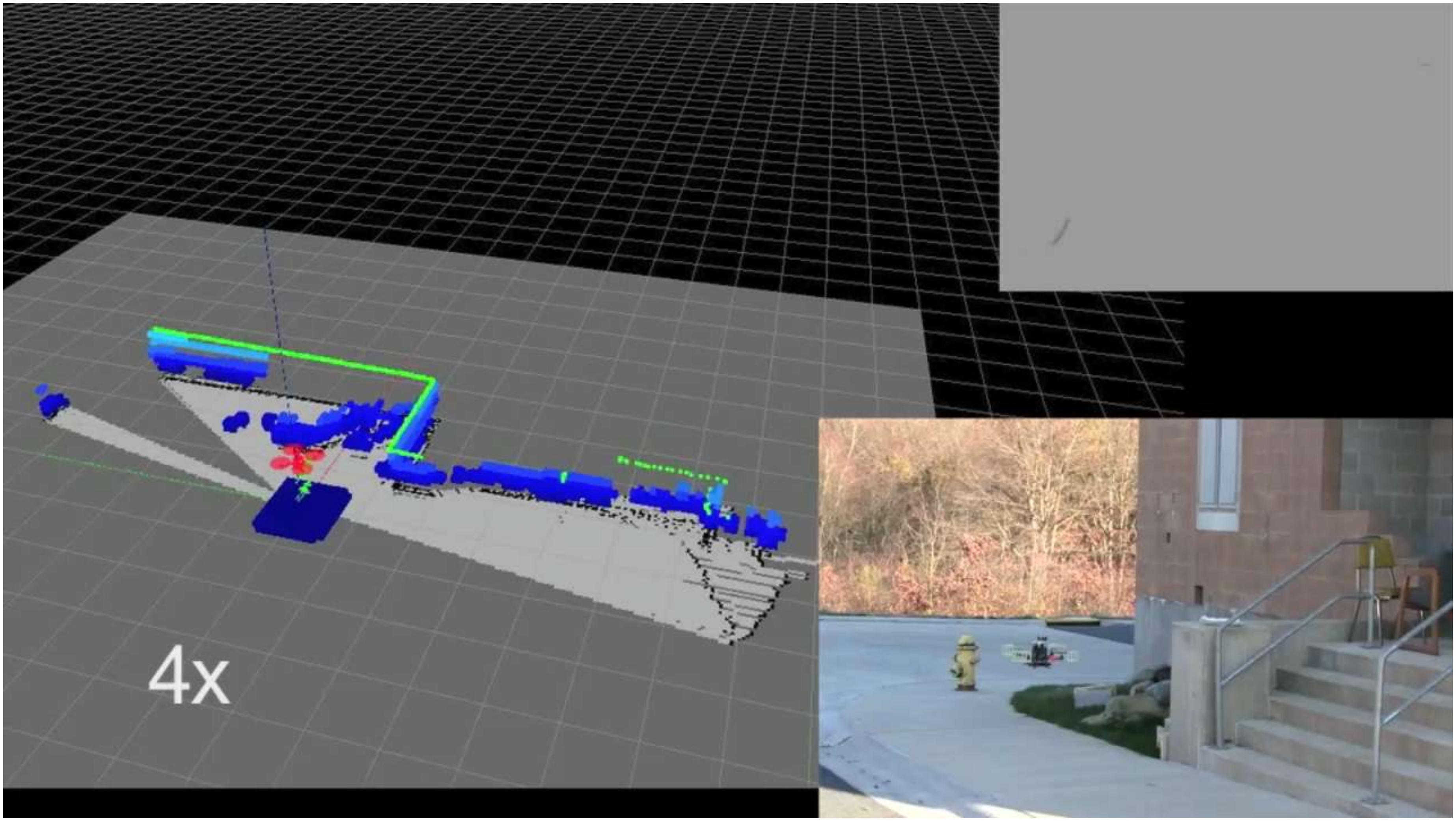
Simultaneous Localization and Mapping also Structure from Motion



Estimation and Control Architecture

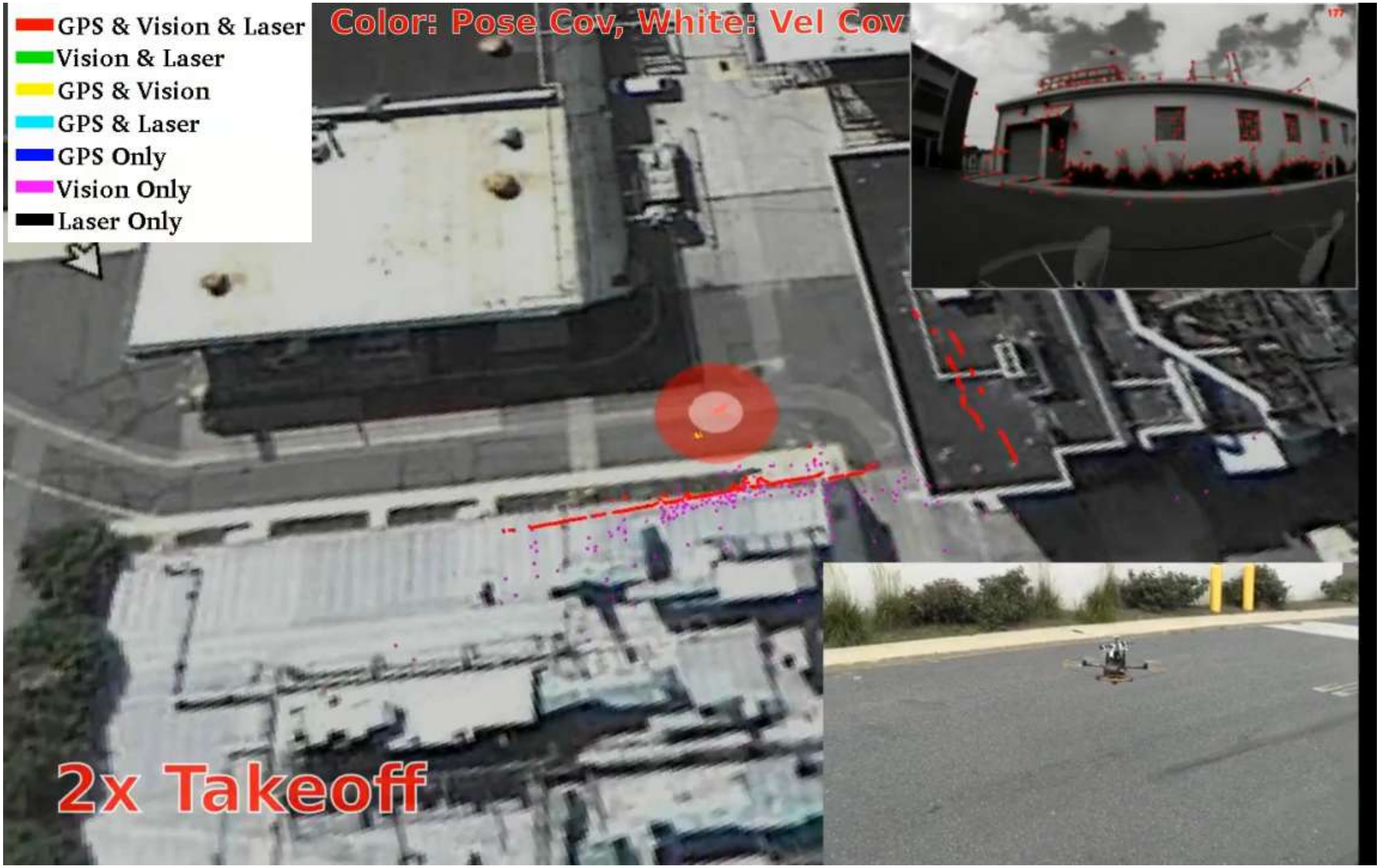


Onboard State Estimation



S. Shen, N. Michael and V. Kumar, "Autonomous navigation in confined indoor environments with a micro-aerial vehicle," *IEEE Robotics and Automation Magazine*, 2013

Penn Engineering

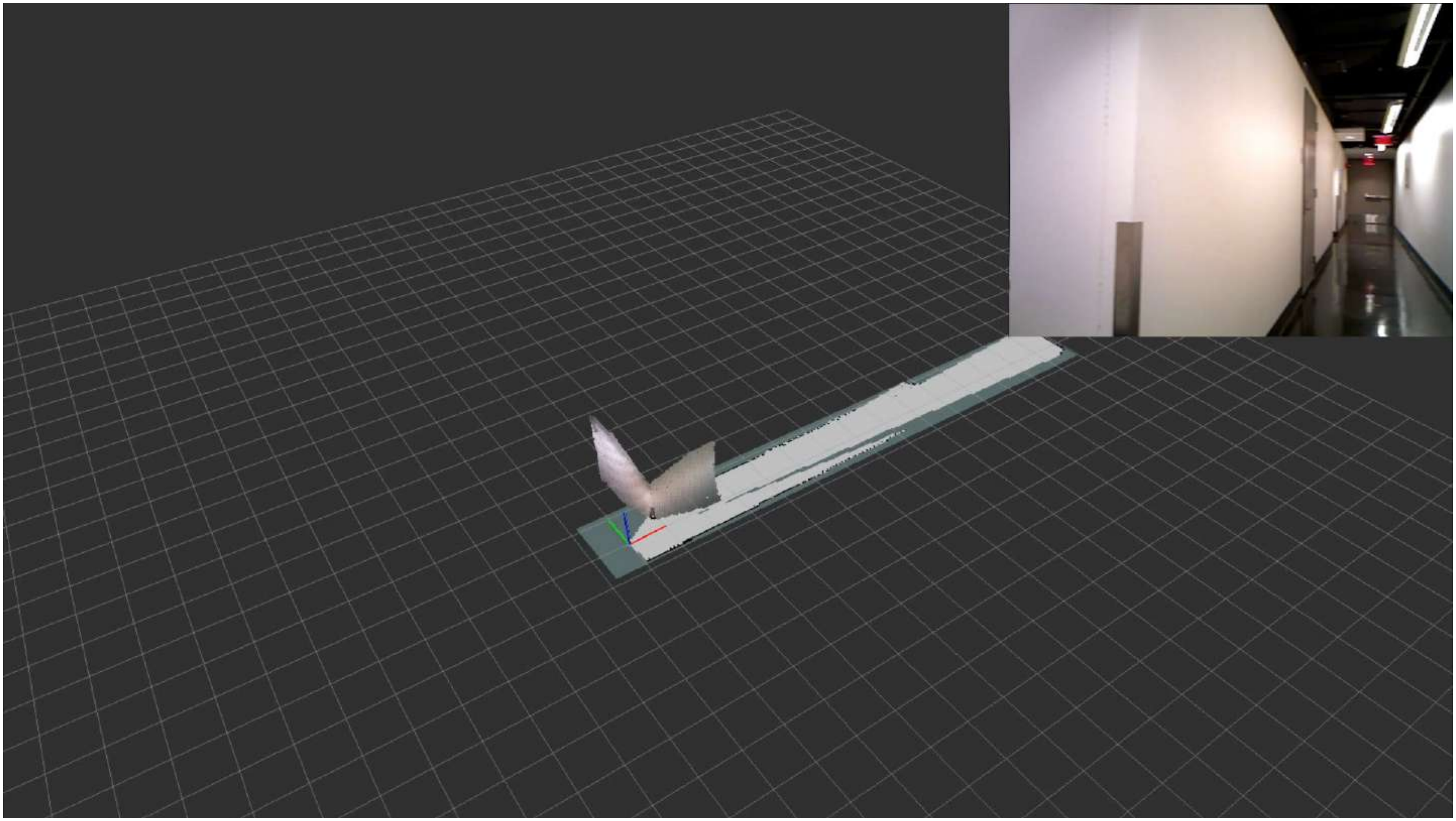


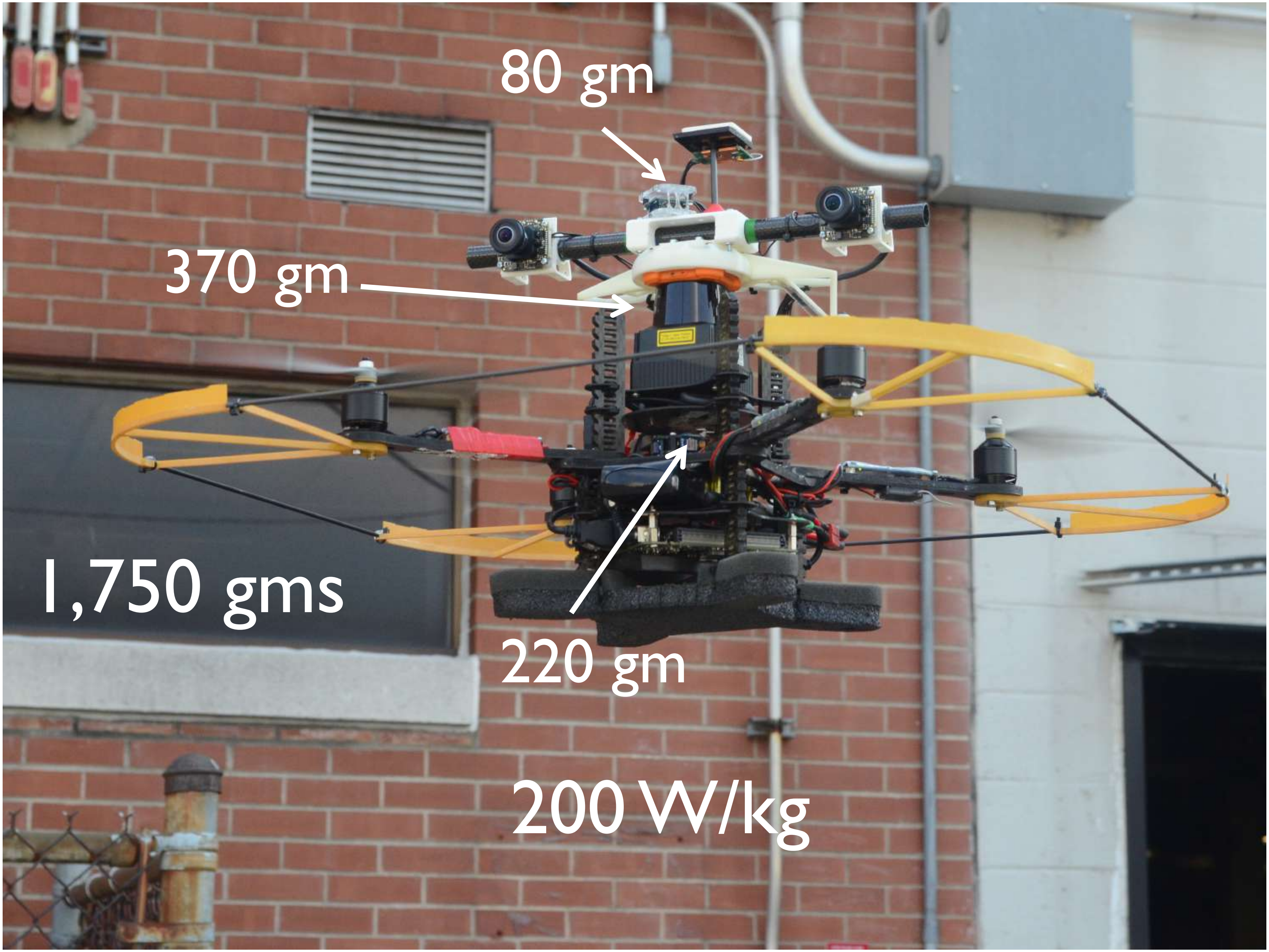
½ km, 1.5 m/s, indoor/outdoor

Shaojie Shen, Yash Mulgaonkar, Nathan Michael and Vijay Kumar, "Multi-Sensor Fusion for Robust Autonomous Flight in Indoor and Outdoor Environments with a Rotorcraft MAV," *Proceedings of IEEE International Conference on Robotics and Automation* (ICRA), 2014.

Penn Engineering

Indoor Navigation and Mapping




80 gm
370 gm
1,750 gms
220 gm
200 W/kg

Systems Design Considerations

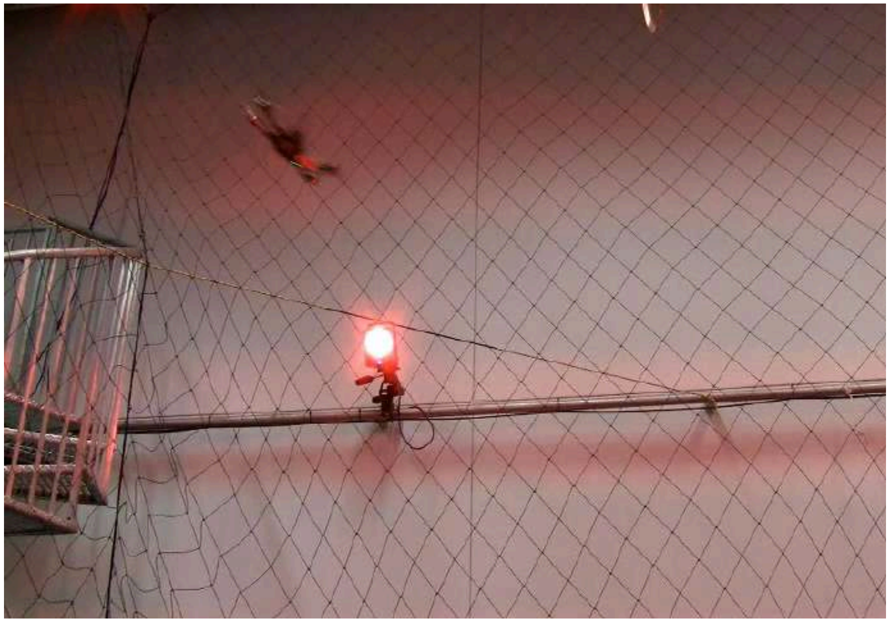
- Larger vehicles are more capable (better sensors, processors)
- Larger vehicles can exhibit longer missions (bigger batteries)

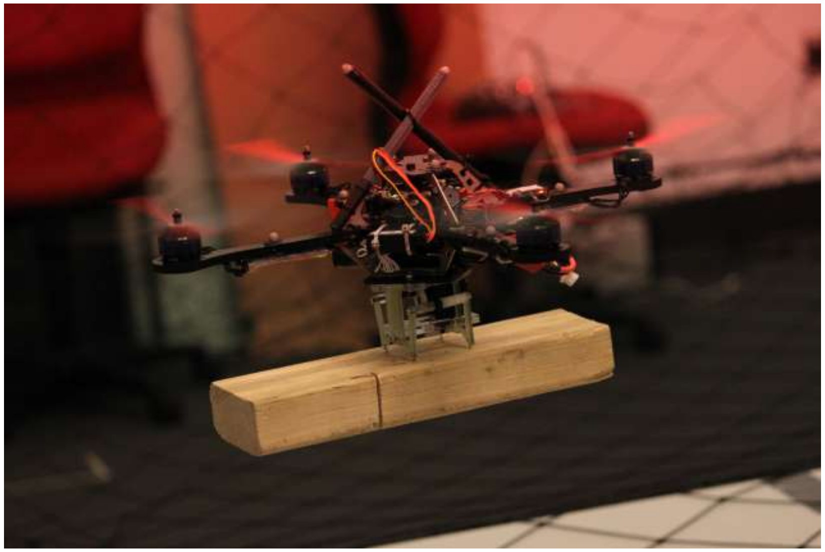
- Smaller vehicles can navigate in more constrained environments
- Smaller vehicles are more agile and maneuverable

Nonlinear Control

Limitations of Linear Control

- Assumption: roll and pitch angles, and all velocities are close to zero





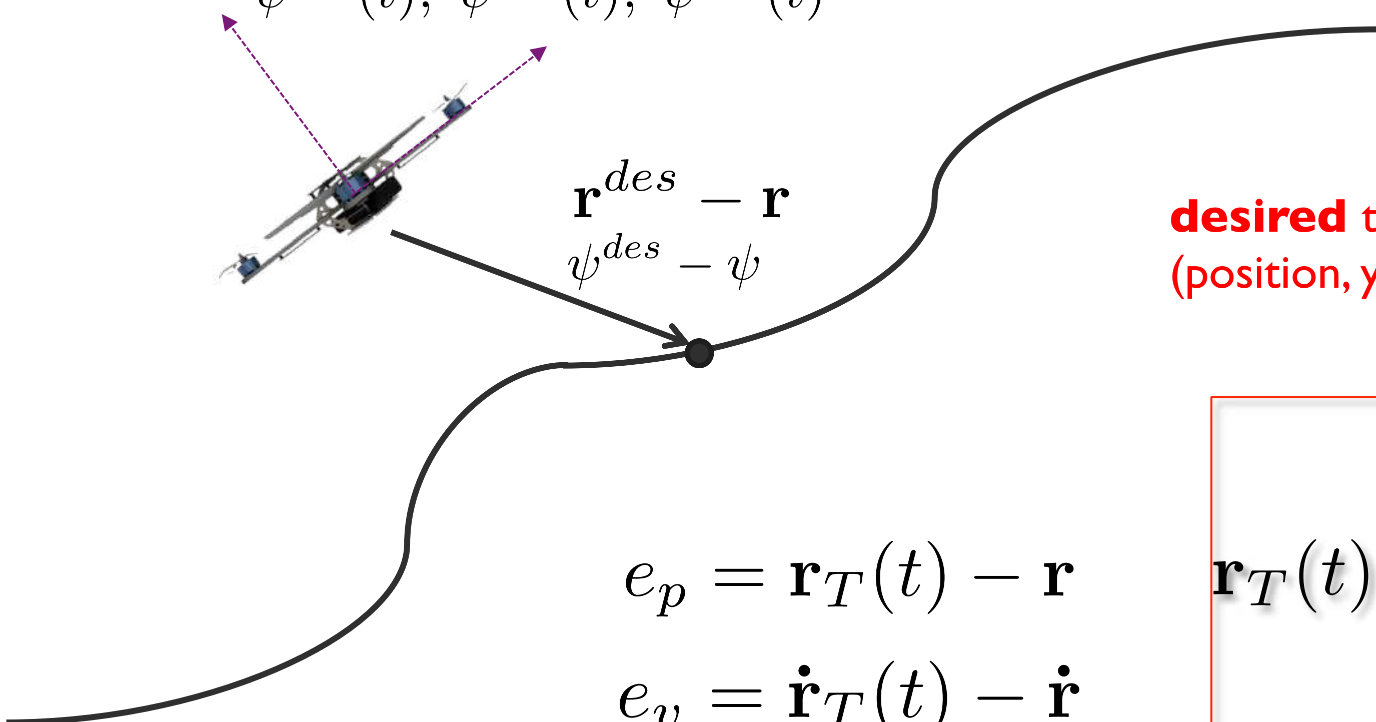
Nonlinear Control

Control the robot at states far away from the equilibrium (hover) state

Trajectory Tracking

Given $\mathbf{r}_T(t), \dot{\mathbf{r}}_T(t), \ddot{\mathbf{r}}_T(t)$

$$\mathbf{r}^{des}(t),\ \dot{\mathbf{r}}^{des}(t),\ \ddot{\mathbf{r}}^{des}(t)$$
$$\psi^{des}(t),\ \dot{\psi}^{des}(t),\ \ddot{\psi}^{des}(t)$$

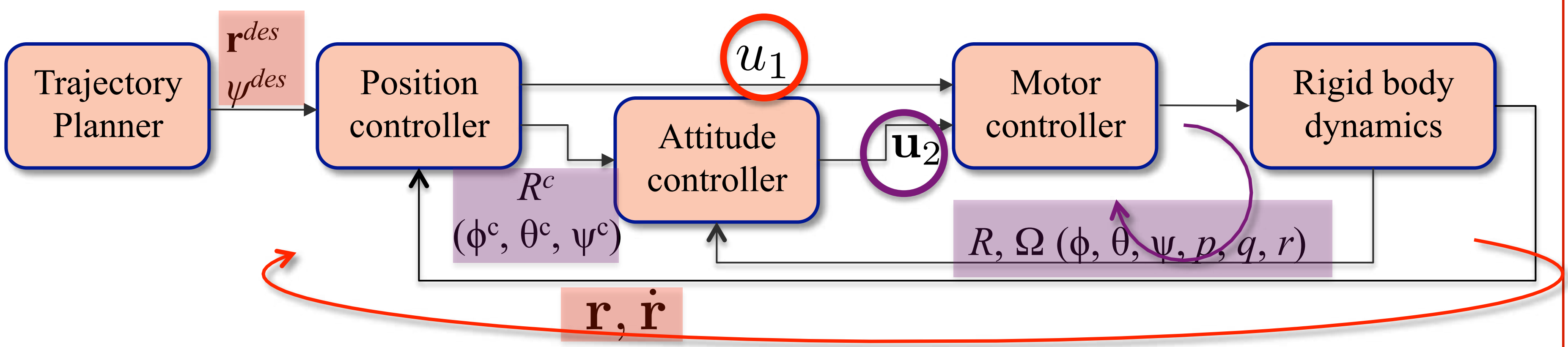


$$e_p = \mathbf{r}_T(t) - \mathbf{r}$$
$$e_v = \dot{\mathbf{r}}_T(t) - \dot{\mathbf{r}}$$

$$\boxed{\mathbf{r}_T(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \psi(t) \end{bmatrix}}$$

Want $(\ddot{\mathbf{r}}_T(t) - \ddot{\mathbf{r}}_c) + k_{d,x} e_v + k_{p,x} e_p = 0$

Commanded acceleration, calculated by the controller



$$m\ddot{\mathbf{r}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + R \begin{bmatrix} 0 \\ 0 \\ \boxed{F_1 + F_2 + F_3 + F_4} \end{bmatrix}$$

u_1

$$I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \boxed{\begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_1 - M_2 + M_3 - M_4 \end{bmatrix}} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

$\mathbf{u}_2$

Trajectory Tracking

$$r_T(t) = \begin{bmatrix} x^{des}(t) \\ y^{des}(t) \\ z^{des}(t) \\ \psi^{des}(t) \end{bmatrix}$$

$\mathbf{t}$

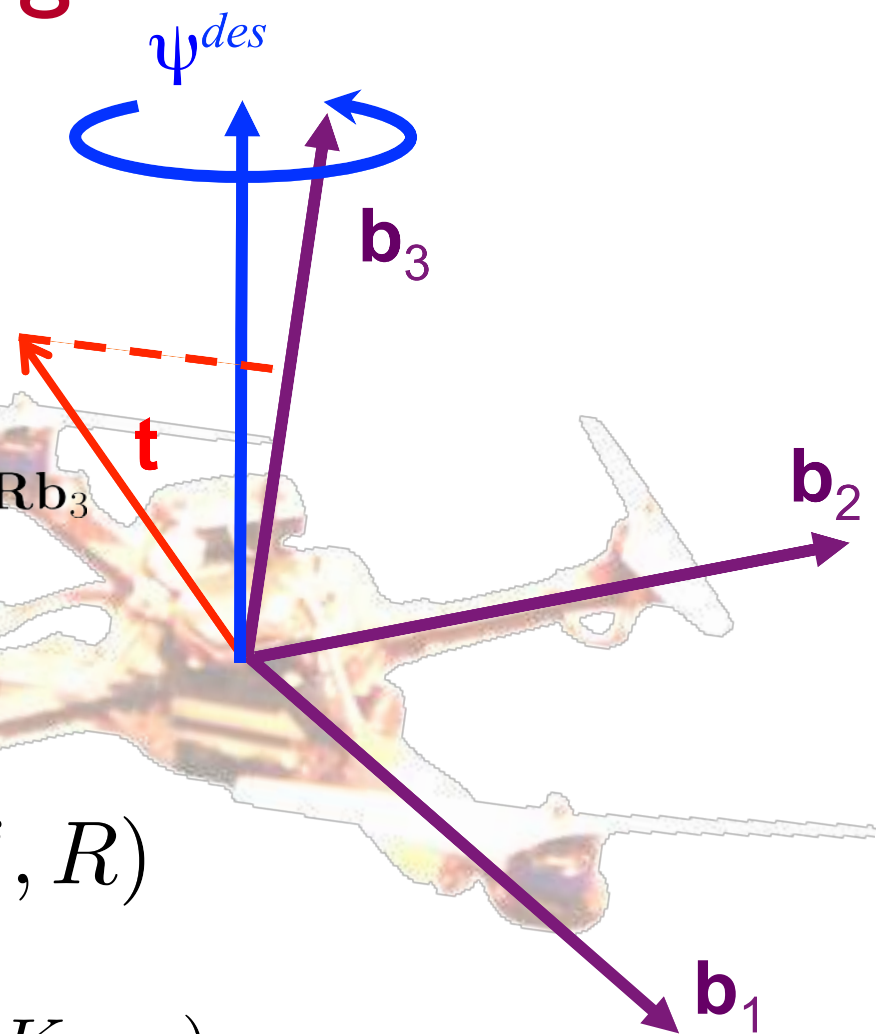
$$u_1 = \boxed{\left(\ddot{r}^{des} + K_v \mathbf{e}_{\dot{r}} + K_p \mathbf{e}_r + mg\mathbf{a}_3\right)} \cdot \mathbf{R}\mathbf{b}_3$$

$$\boxed{\begin{aligned} \mathbf{R}^{des}\mathbf{b}_3 &= \frac{\mathbf{t}}{\|\mathbf{t}\|} \\ \psi &= \psi^{des} \end{aligned}}$$

$$R^{des}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$e_R(R^{des}, R)$$

$$\mathbf{u}_2 = \omega \times \mathbf{I}\omega + \mathbf{I}\left(-K_R \mathbf{e}_R - K_\omega \mathbf{e}_\omega\right)$$



How to determine $\mathbf{R}^{des}$?

You are given two pieces of information

$$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \longrightarrow \mathbf{R}^{des}\mathbf{b}_3 = \frac{\mathbf{t}}{\|\mathbf{t}\|}$$

$$\psi = \psi^{des}$$

You know that the rotation matrix has the form

$$\mathbf{R} = \begin{bmatrix} c\psi c\theta - s\phi s\psi s\theta & -c\phi s\psi & c\psi s\theta + c\theta s\phi s\psi \\ c\theta s\psi + c\psi s\phi s\theta & c\phi c\psi & s\psi s\theta - c\theta s\phi c\psi \\ -c\phi s\theta & s\phi & c\phi c\theta \end{bmatrix}$$

You should be able to find the roll and pitch angles.

How to calculate the error $\mathbf{e}_R(\mathbf{R}^{des}, \mathbf{R})$?

- Cannot simply take the difference of two rotation matrices

What is the magnitude of the rotation required to go from the current orientation to the desired orientation?

$$\mathbf{R} \rightarrow \mathbf{R}^{des}$$

The required rotation is

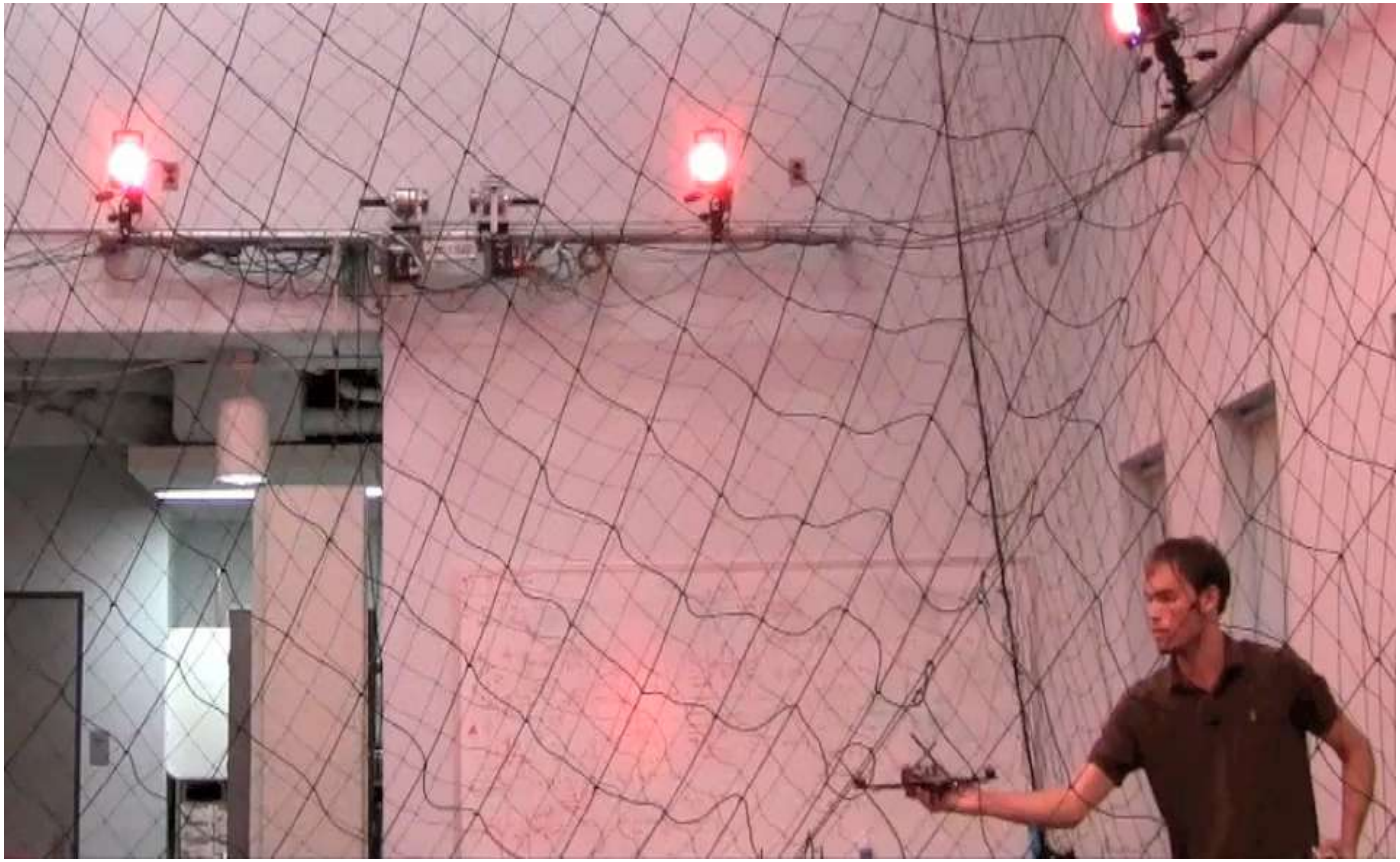
$$\Delta R = \mathbf{R}^T \mathbf{R}^{des}$$

The angle and axis of rotation can be determined using Rodrigues formula

Stability

Large basin of attraction*

$$tr[I - (R^{des})^T R] < 2 \qquad \|e_\omega(0)\|^2 \leq \frac{2}{\boxed{\lambda_{min}(I)}} k_R \left(1 - \frac{1}{2} tr\left[I - (R^{des})^T R\right]\right)$$

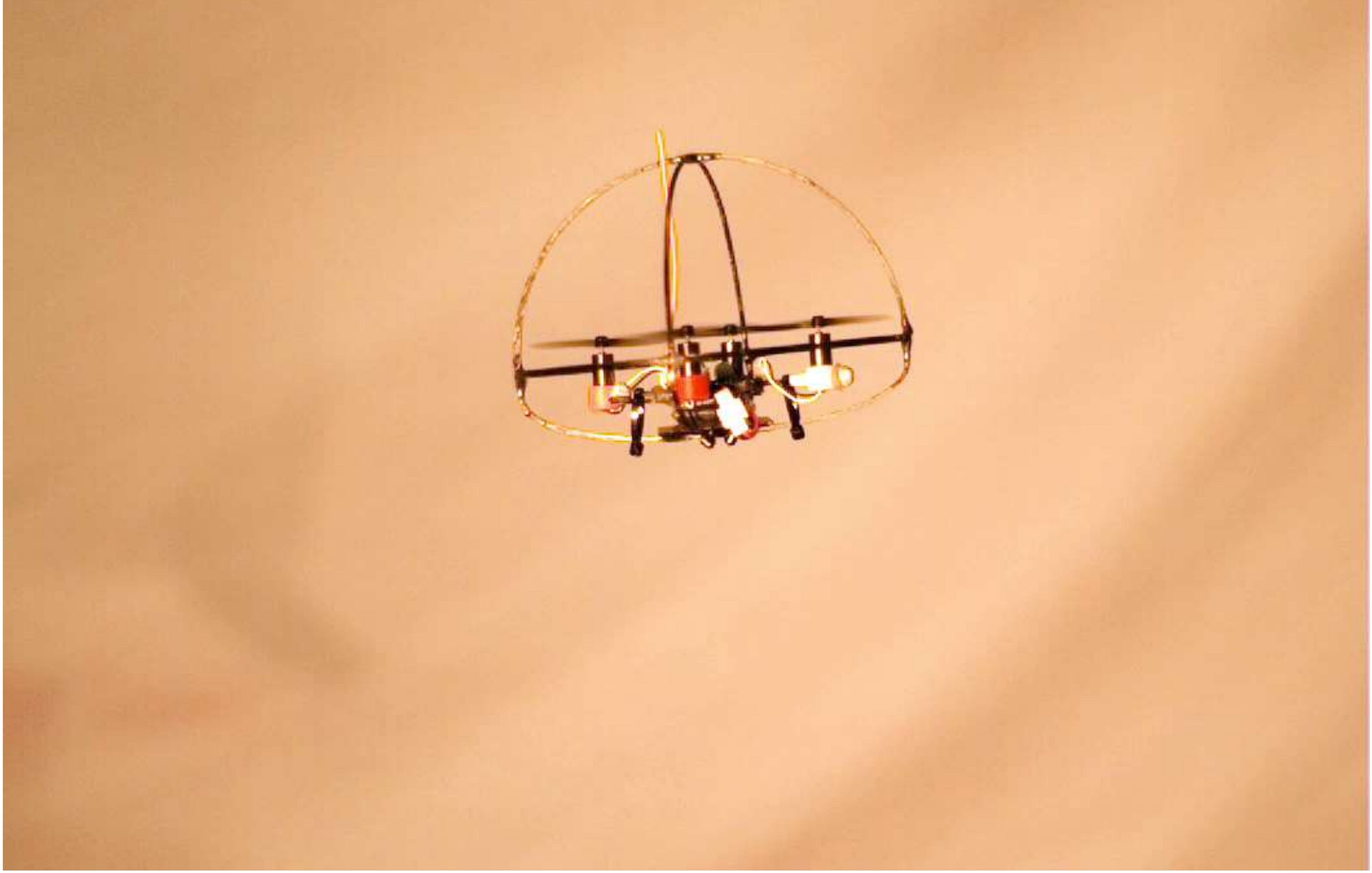


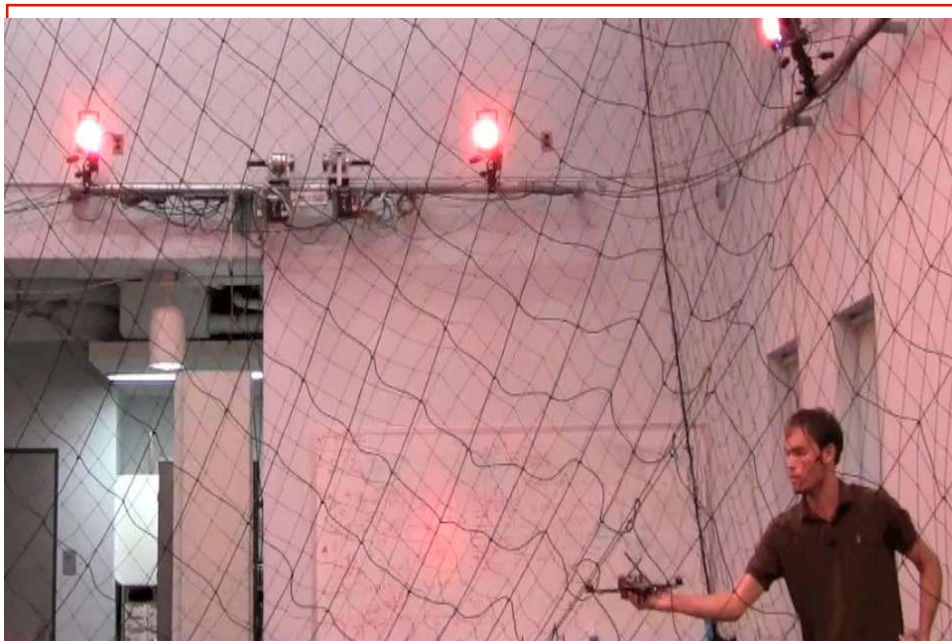
*T. Lee, M. Leoky, and N. H. McClamroch, Geometric tracking control of a quadrotor UAV on SE(3), *IEEE Conference on Decision and Control*, 2010.

Smaller, safer …


Pico Quadrotor
11 cm
20 g,
6.5 Watts
Max speed 6m/s
Safer
More maneuverable
Y. Mulgaonkar, G. Cross and V. Kumar, "Design of small, safe and robust quadrotor swarms," IEEE International Conference on Robotics and Automation (ICRA), Seattle WA, May 2015.

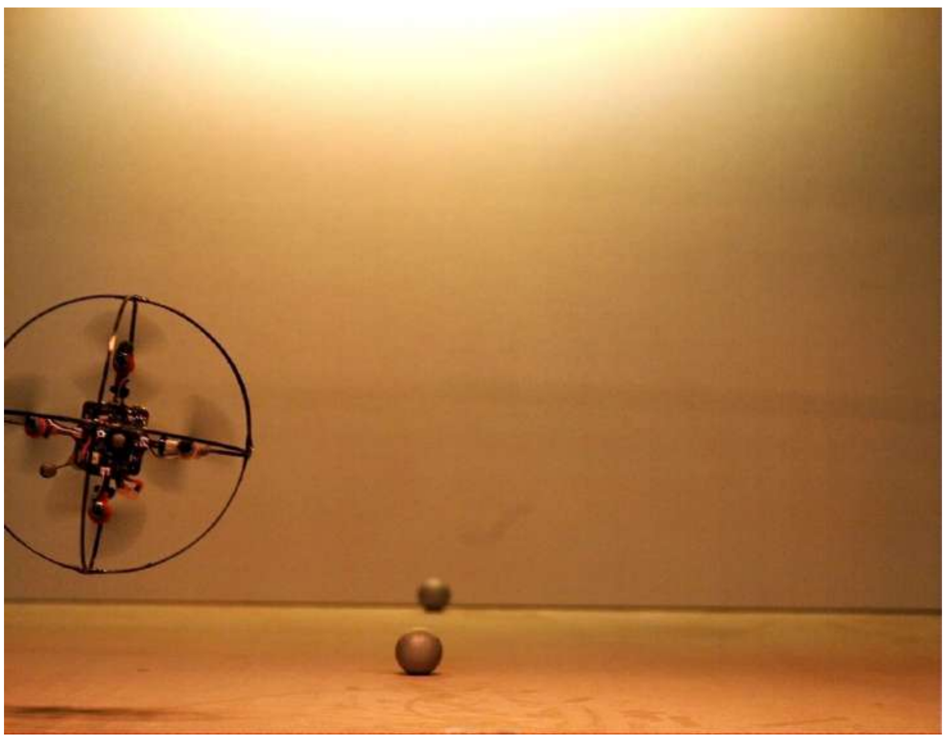
Recovery from mid air collisions





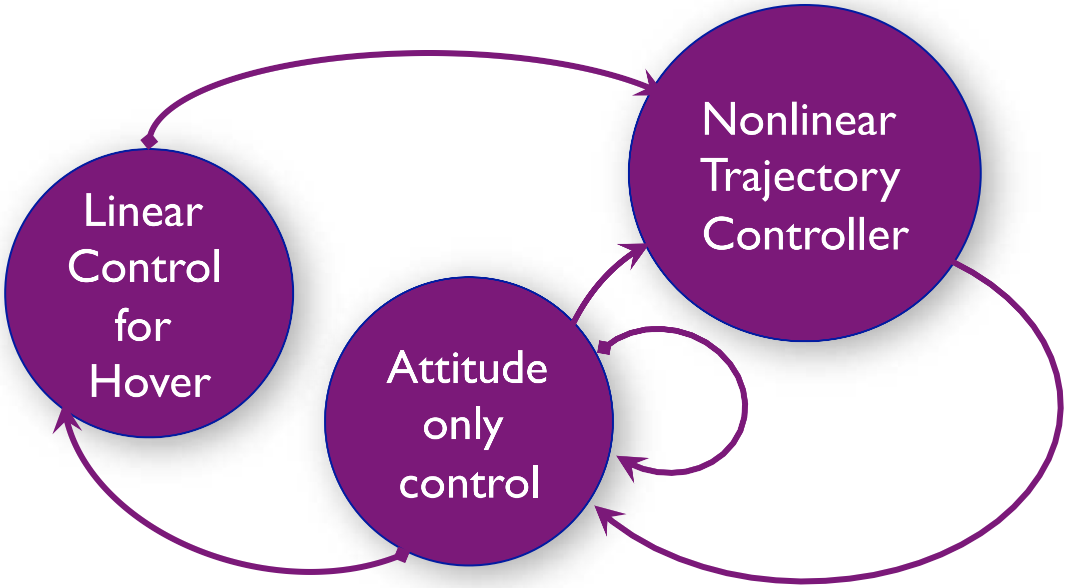
basin of attraction $\sim \frac{1}{L^{\frac{5}{2}}}$

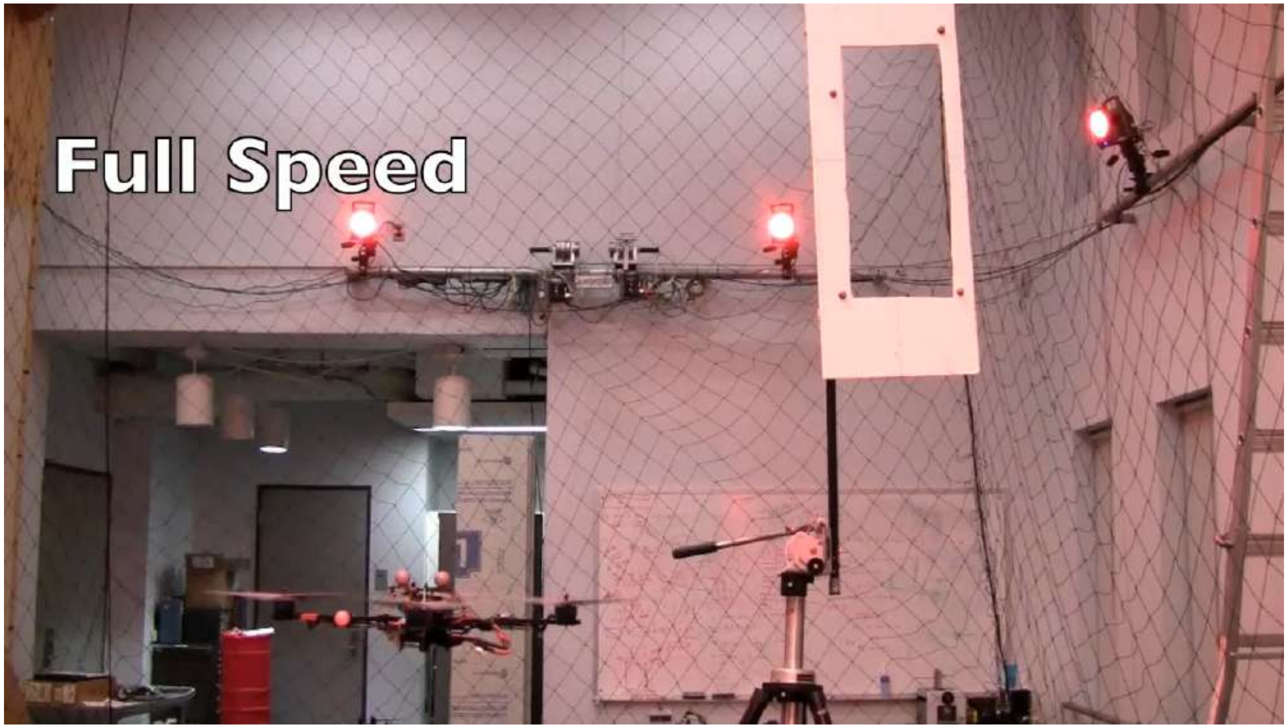
D. Mellinger and V. Kumar, "Minimum Snap Trajectory Generation and Control for Quadrotors," *Proc. IEEE International Conference on Robotics and Automation*. Shanghai, China, May, 2011.



Y. Mulgaonkar, G. Cross and V. Kumar, "Design of small, safe and robust quadrotor swarms," in IEEE International Conference on Robotics and Automation (ICRA), Seattle WA, May 2015.

Sequential Composition





Trajectory Planning

Inputs

$u_1, \mathbf{u}_2$

$$u_1 = \sum_{i=1}^{4} F_i \qquad \mathbf{u}_2 = L \begin{bmatrix} 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ \mu & -\mu & \mu & -\mu \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}$$

State

$(\mathbf{q}, \dot{\mathbf{q}})$

position
yaw

jerk
snap
yaw

$$\begin{bmatrix} \mathbf{r} \\ \mathbf{v} \\ \mathbf{a} \\ \mathbf{j} \\ \mathbf{s} \\ \psi \\ \dot{\psi} \\ \ddot{\psi} \end{bmatrix} \leftrightarrow \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \\ u_1 \\ \dot{u}_1 \\ \ddot{u}_1 \\ \mathbf{u_2} \end{bmatrix}$$

acceleration in the vertical direction

jerk in the vertical direction

$$u_1 = m(a_3 - \mathbf{g}.\mathbf{b}_3)$$

$$\dot{u}_1 = m j_3$$

$$\ddot{u}_1 = m s_3 + u_1(q^2 + rp)$$

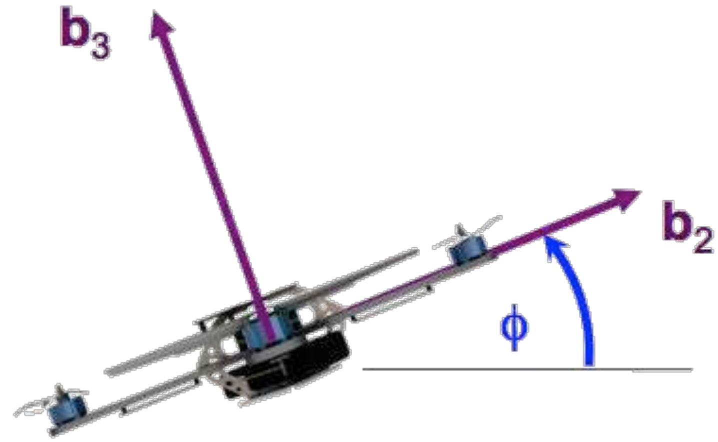
$$p = \frac{-m j_2}{u_1}$$

$$q = \frac{m j_1}{u_1}$$

snap in the vertical direction

$$\omega = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Planar Quadrotor



Inputs

u_1, u_2

$$u_1 = F_2 + F_4$$

$$u_2 = (F_2 - F_4)L$$

$$\mathbf{q} = \begin{bmatrix} y \\ z \\ \phi \end{bmatrix}$$

State

$(\mathbf{q}, \dot{\mathbf{q}})$

Equations of motion

$$\begin{bmatrix} \ddot{y} \\ \ddot{z} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 \\ -g \\ 0 \end{bmatrix} + \begin{bmatrix} -\frac{1}{m}\sin\phi & 0 \\ \frac{1}{m}\cos\phi & 0 \\ 0 & \frac{1}{I_{xx}} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

Differential Flatness

All state variables and the inputs can be written as smooth functions of *flat outputs* and their derivatives (and the other way around)

Planar Quadrotor

$$\begin{bmatrix} \boxed{\begin{matrix} y \\ z \end{matrix}} \\ \dot{y} \\ \dot{z} \\ \ddot{y} \\ \ddot{z} \\ \dddot{y} \\ \dddot{z} \\ y^{(iv)} \\ z^{(iv)} \end{bmatrix} \quad \rightleftarrows \quad \begin{bmatrix} y \\ z \\ \phi \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ u_1 \\ \dot{u}_1 \\ \ddot{u}_1 \\ u_2 \end{bmatrix}$$

Planar Quadrotor

The flat outputs and their derivatives can be written as a function of the state, the inputs, and their derivatives

Flat outputs

$$\begin{bmatrix} y \\ z \end{bmatrix}$$

State

$$\begin{bmatrix} y \\ z \\ \phi \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \end{bmatrix}$$

Input

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} -\frac{1}{m}\sin\phi \\ \frac{1}{m}\cos\phi \end{bmatrix} u_1$$

$$\begin{bmatrix} y^{(iii)} \\ z^{(iii)} \end{bmatrix} = \frac{1}{m}\begin{bmatrix} -u_1\dot{\phi}\cos\phi - \dot{u}_1\sin\phi \\ -u_1\dot{\phi}\sin\phi + \dot{u}_1\cos\phi \end{bmatrix}$$

$$\begin{bmatrix} y^{(iv)} \\ z^{(iv)} \end{bmatrix} = \frac{1}{m}\begin{bmatrix} -\sin\phi & -\frac{u_1}{I_{xx}}\cos\phi \\ \cos\phi & -\frac{u_1}{I_{xx}}\sin\phi \end{bmatrix}\begin{bmatrix} \ddot{u}_1 \\ u_2 \end{bmatrix} + \frac{1}{m}\begin{bmatrix} -2\dot{u}_1\dot{\phi}\cos\phi + u_1\dot{\phi}^2\sin\phi \\ -2\dot{u}_1\dot{\phi}\sin\phi - u_1\dot{\phi}^2\cos\phi \end{bmatrix}$$

Planar Quadrotor

The state, the inputs, and their derivatives can be written as a function of the flat outputs and their derivatives

Flat outputs	State	Input
$\begin{bmatrix} y \\ z \end{bmatrix}$	$\begin{bmatrix} y \\ z \\ \phi \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \end{bmatrix}$	$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$

$$u_1 = m\left(\ddot{y}^2 + \ddot{z}^2\right)$$

$$\phi = \mathrm{atan2}\left(-\frac{m\ddot{y}}{u_1}, \frac{m\ddot{z}}{u_1}\right)$$

$$\dot{u}_1 = m(-y^{(iii)}\sin\phi + z^{(iii)}\cos\phi)$$

$$\dot{\phi} = \frac{-m}{u_1}\left(y^{(iii)}\cos\phi + z^{(iii)}\sin\phi\right)$$

$$\ddot{u}_1 = \dots$$

$$\ddot{\phi} = \dots$$

$$u_2 = \dots$$

Differential Flatness

All state variables and the inputs can be written as smooth functions of *flat outputs* and their derivatives (and the other way around)

Planar Quadrotor

$$\begin{bmatrix} \boxed{\begin{matrix} y \\ z \end{matrix}} \\ \dot{y} \\ \dot{z} \\ \ddot{y} \\ \ddot{z} \\ \dddot{y} \\ \dddot{z} \\ y^{(iv)} \\ z^{(iv)} \end{bmatrix} \rightleftarrows \begin{bmatrix} y \\ z \\ \phi \\ \dot{y} \\ \dot{z} \\ \dot{\phi} \\ u_1 \\ \dot{u}_1 \\ \ddot{u}_1 \\ u_2 \end{bmatrix}$$

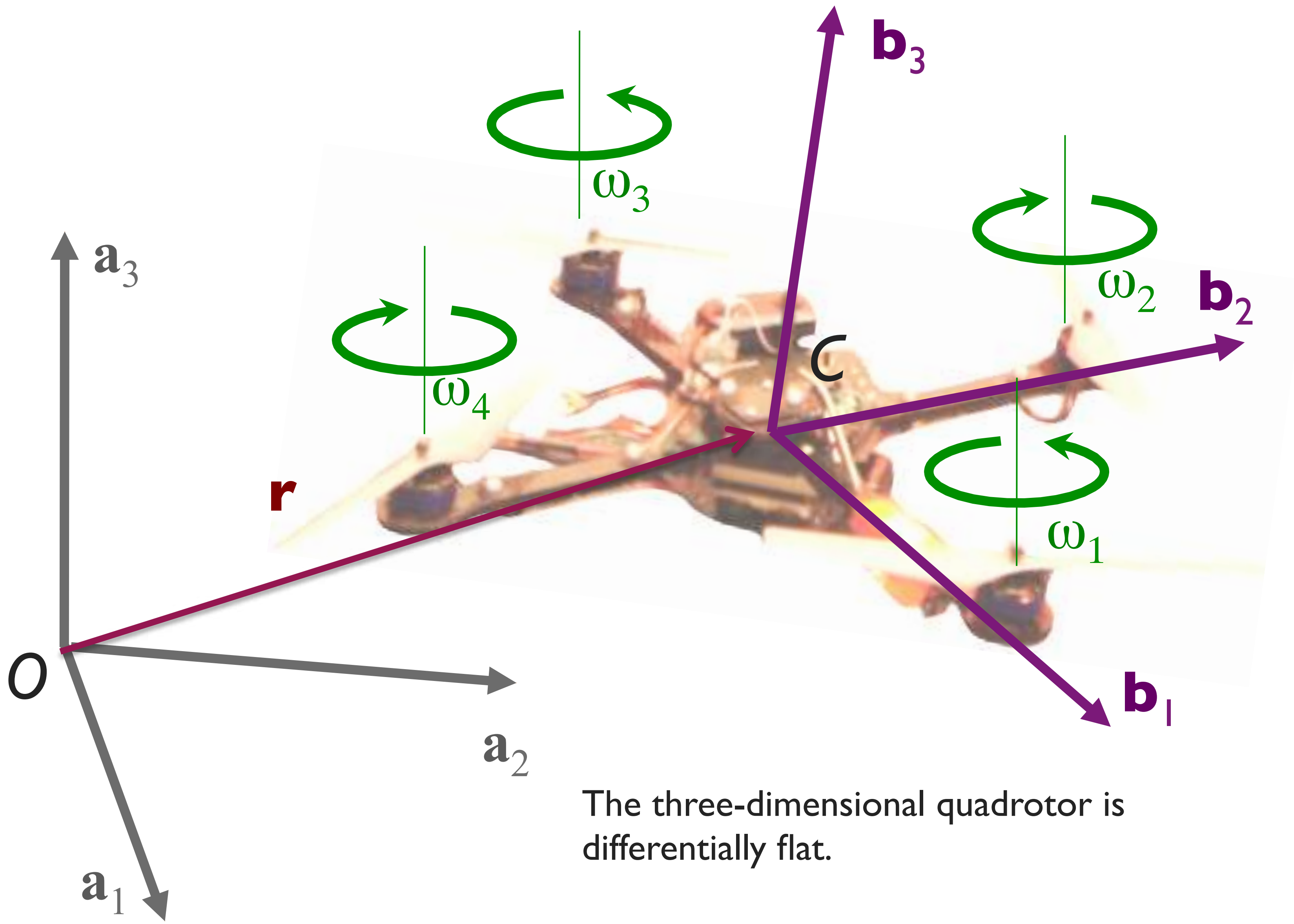
Diffeomorphism

Differential Flatness

All state variables and the inputs can be written as smooth functions of *flat outputs* and their derivatives

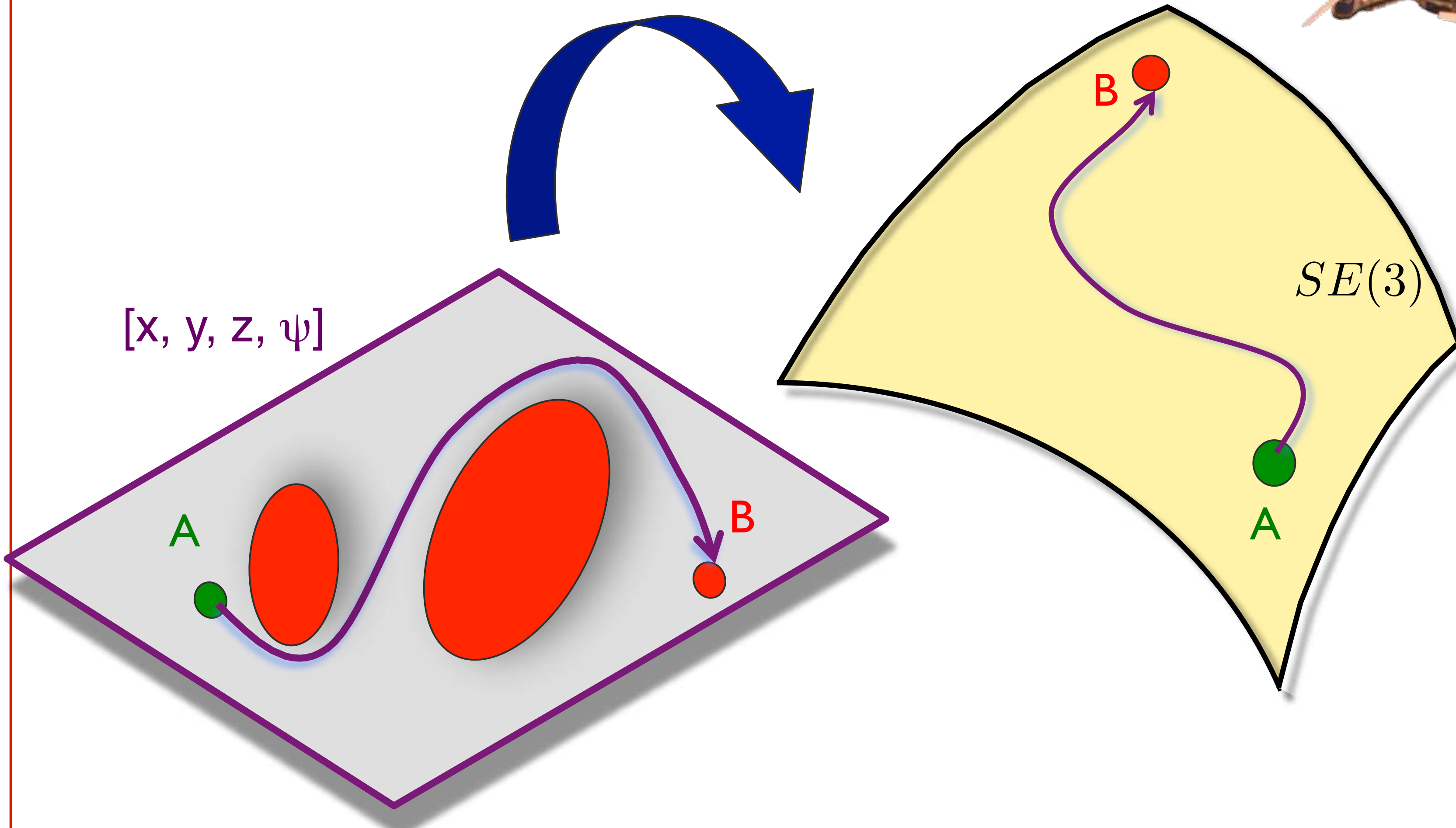
<u>3-D Quadrotor</u>

$$\begin{bmatrix} \boxed{\mathbf{r}} \\ \mathbf{v} \\ \mathbf{a} \\ \mathbf{j} \\ \mathbf{s} \\ \boxed{\psi} \\ \dot{\psi} \\ \ddot{\psi} \end{bmatrix} \leftrightarrow \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \\ u_1 \\ \dot{u}_1 \\ \ddot{u}_1 \\ \mathbf{u_2} \end{bmatrix}$$



The three-dimensional quadrotor is differentially flat.

Trajectory Planning



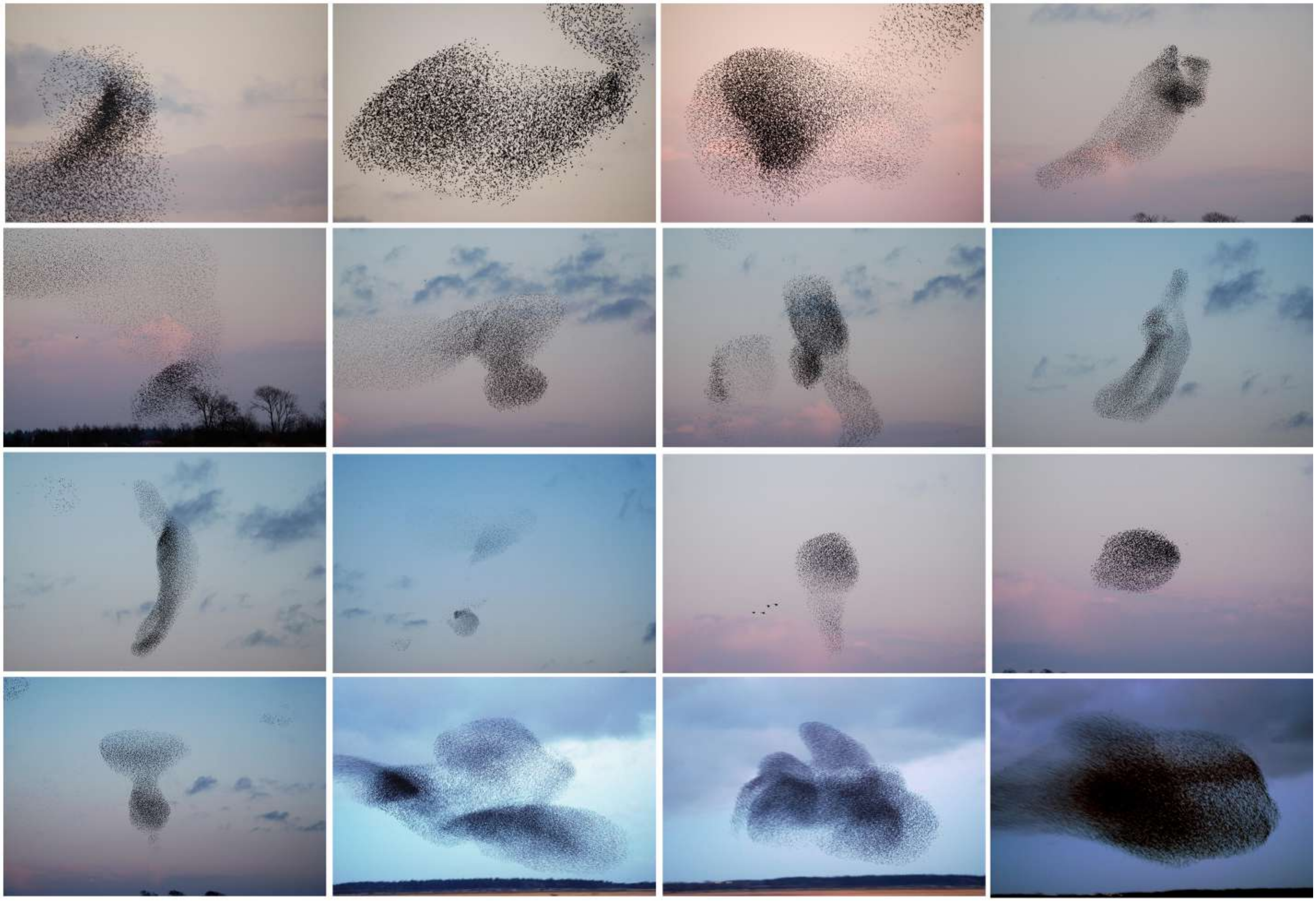
$$\begin{bmatrix} x \\ y \\ z \\ \theta \\ \phi \\ \psi \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{\phi} \end{bmatrix}$$

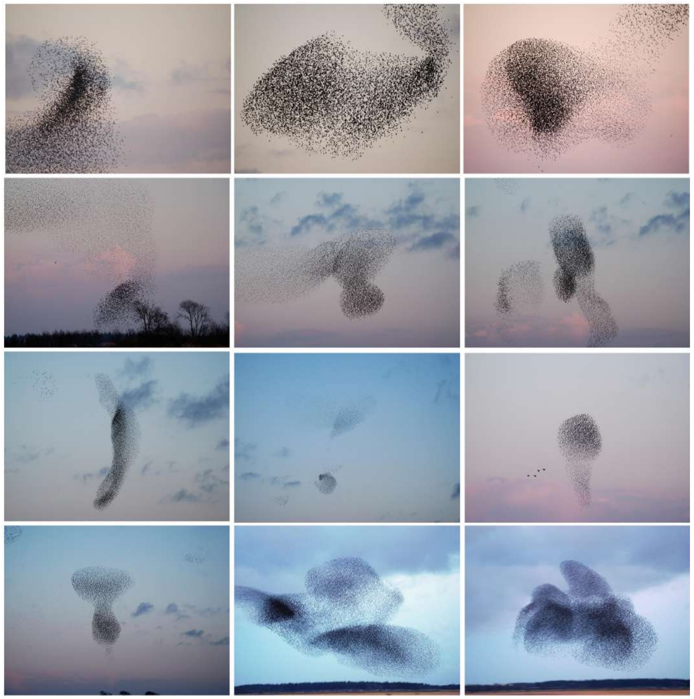
Minimum snap trajectory

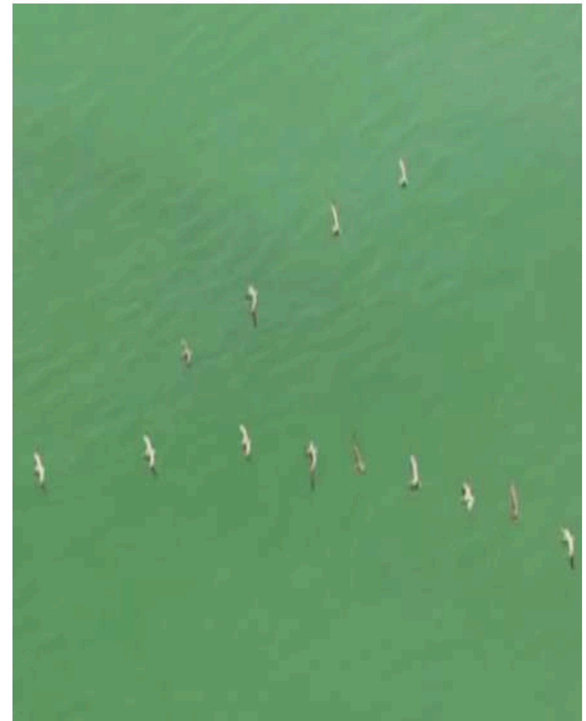
$$\min_{\sigma(t)} \int_0^T \alpha \| \ddddot{\mathbf{r}}(t) \|^2 + \beta \ddot{\psi}(t)^2 dt$$


Minimum Snap Trajectory


Robots for Emergency Response
Swarms







Three Organizing Principles for Collective Behavior

- Each individual acts independently
- Actions are based on local information
- Anonymity in coordination

Example: Transportation and Construction

Complexity

n robots, *m* obstacles

- Dimensionality of the state space increases linearly with *n* $O(n)$

- Number of potential interactions with neighbors increases as n^2 $O(mn+n^2)$
- Number of potential interactions with obstacles increases as mn

- Number of assignments of robots to goal positions $O(n!)$

Assignment of robots to goals

factorial

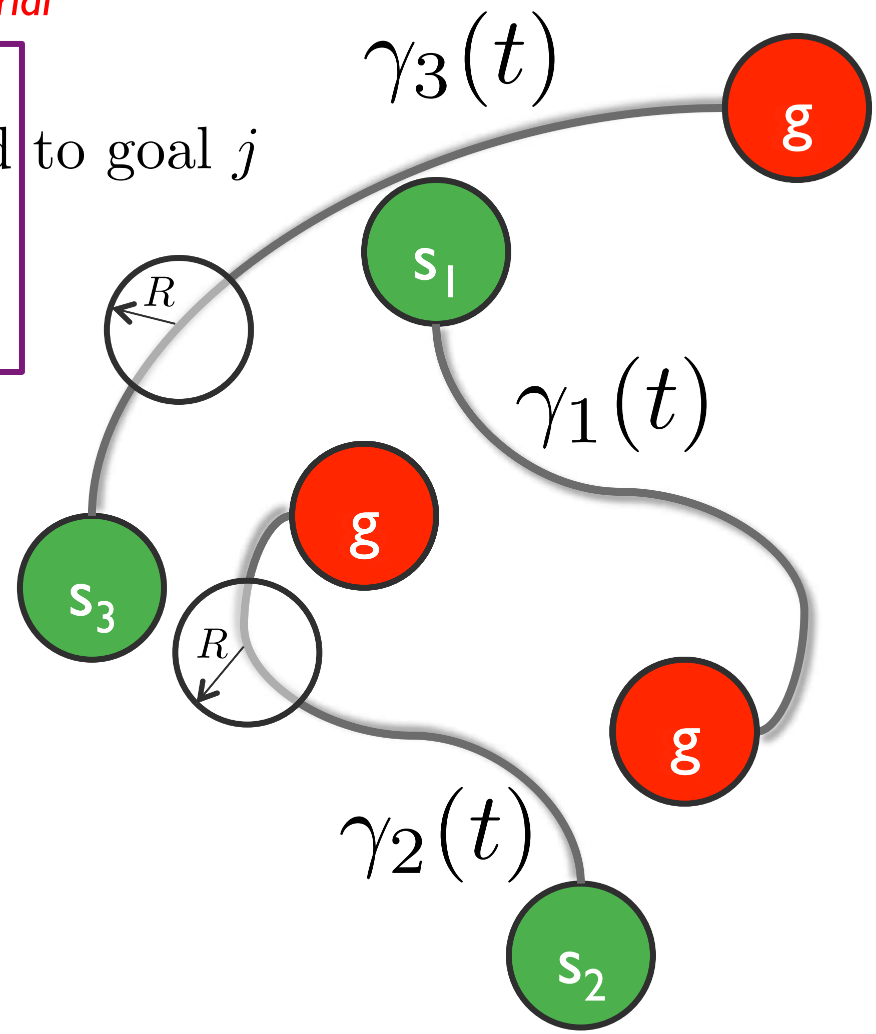
$$\phi_{i,j} = \begin{cases} 1 & \text{if robot } i \text{ is assigned to goal } j \\ 0 & \text{otherwise} \end{cases}$$

Planning trajectories

exponential

$$\mathbf{X}(t) = \begin{bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \\ \dots \\ \mathbf{x}_N(t) \end{bmatrix}$$

$$\gamma(t) : [t_0, t_f] \rightarrow \mathbf{X}(t)$$



Safety

$$\left[\inf_{i \neq j \in \mathcal{I}, t \in [t_0, t_f]} ||\mathbf{x}_i(t) - \mathbf{x}_j(t)|| - 2R \right] > 0$$

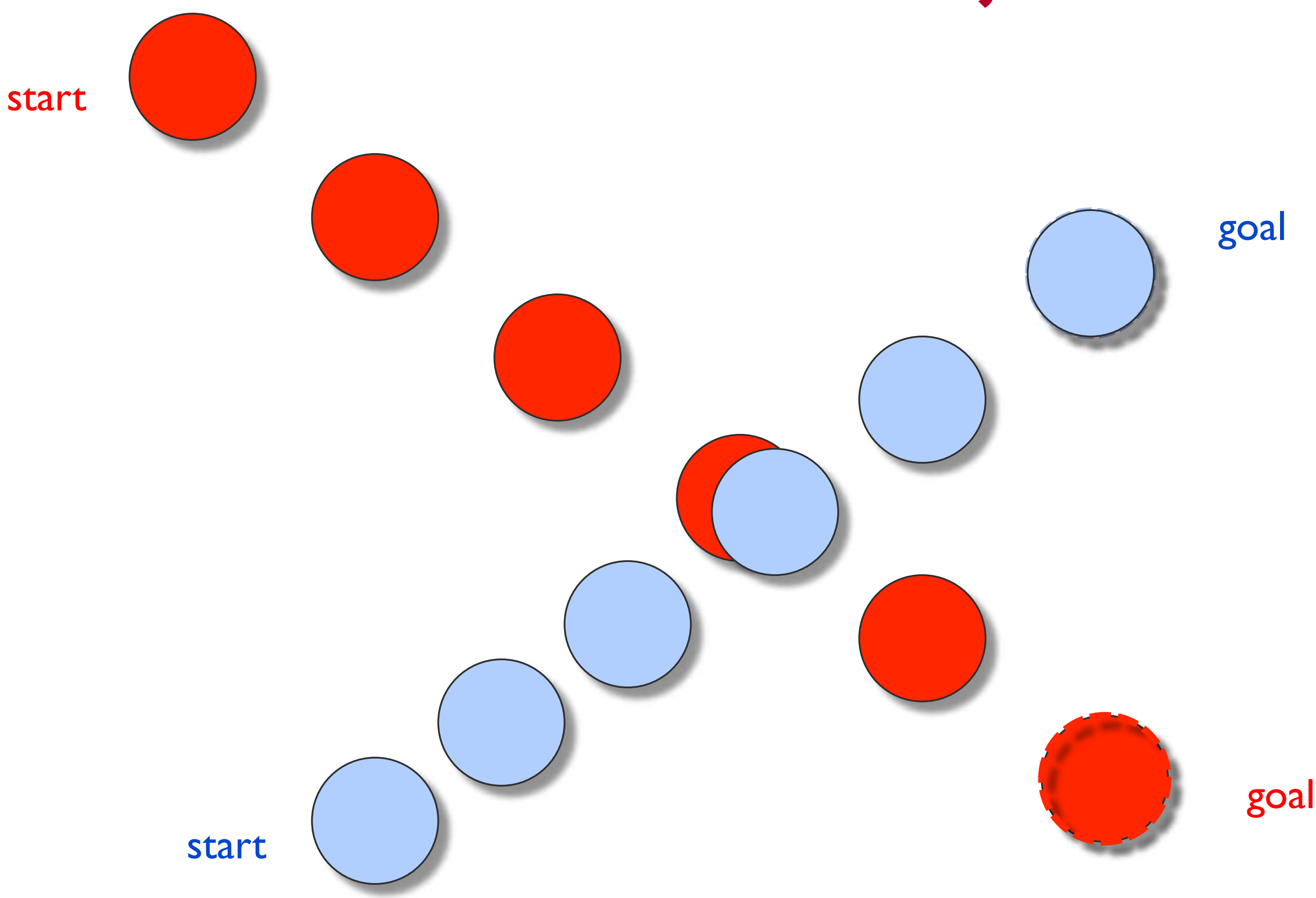
Optimality

$$\gamma^\star(t) = \underset{\gamma(t)}{\operatorname{argmin}} \int_{t_0}^{t_f} L(\gamma(t))dt$$

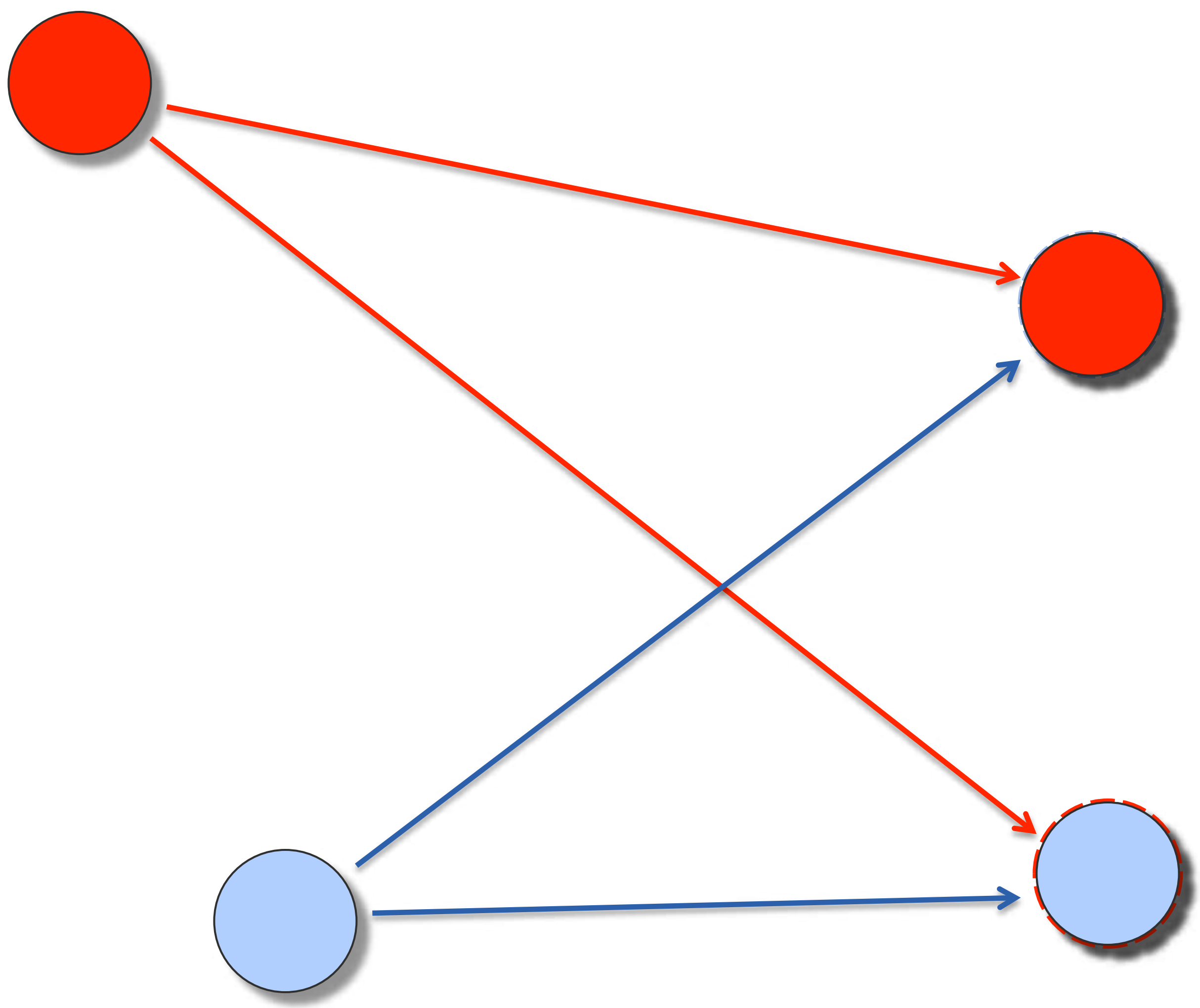
Four Key Ideas

- Concurrent assignment of goals and trajectories
- Leader-follower networks
- Anonymity
- Sharing information

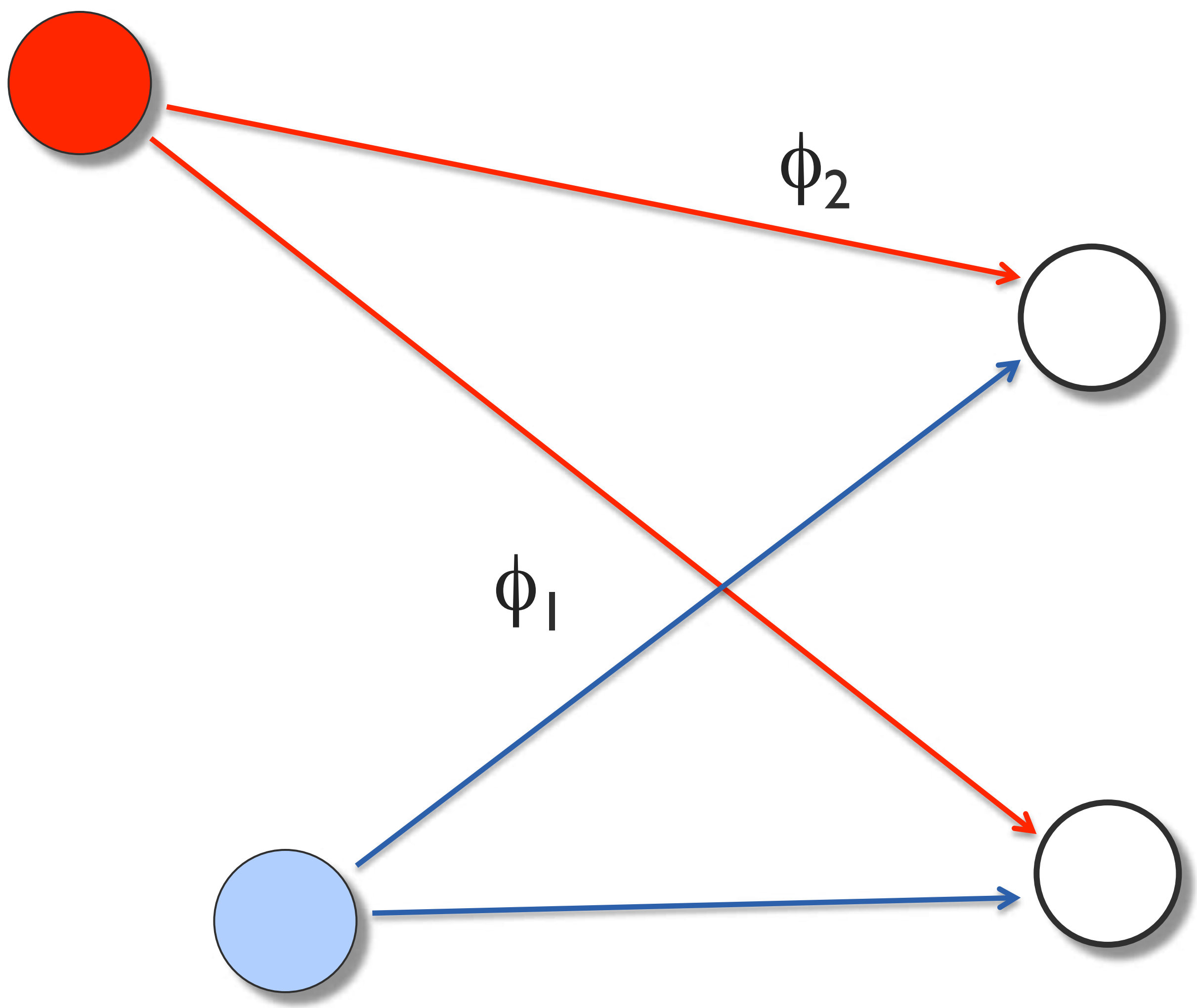
1. Assignment of Goals and Collision Free Trajectories



Concurrent Assignment and Planning of Trajectories: CAPT



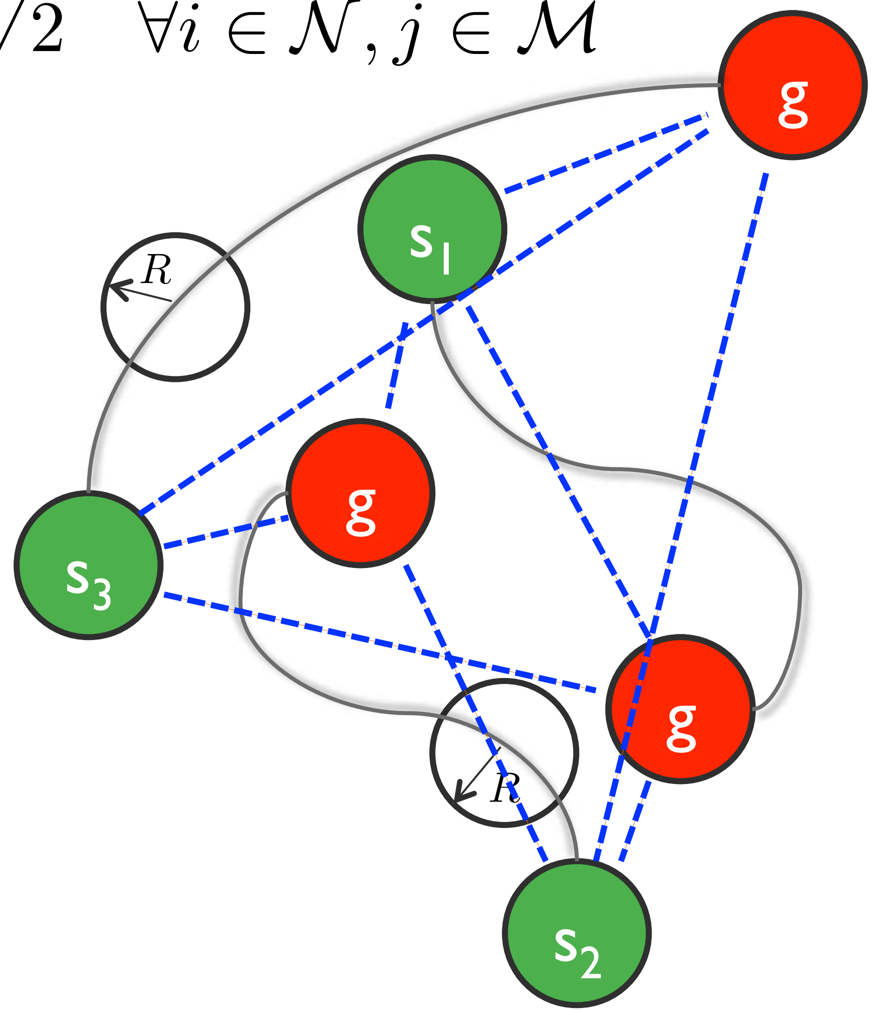
CAPT



Concurrent Assignment and Planning

Assumption

$$||\mathbf{s}_i - \mathbf{g}_j|| > 2R\sqrt{2} \quad \forall i \in \mathcal{N}, j \in \mathcal{M}$$



Theorem

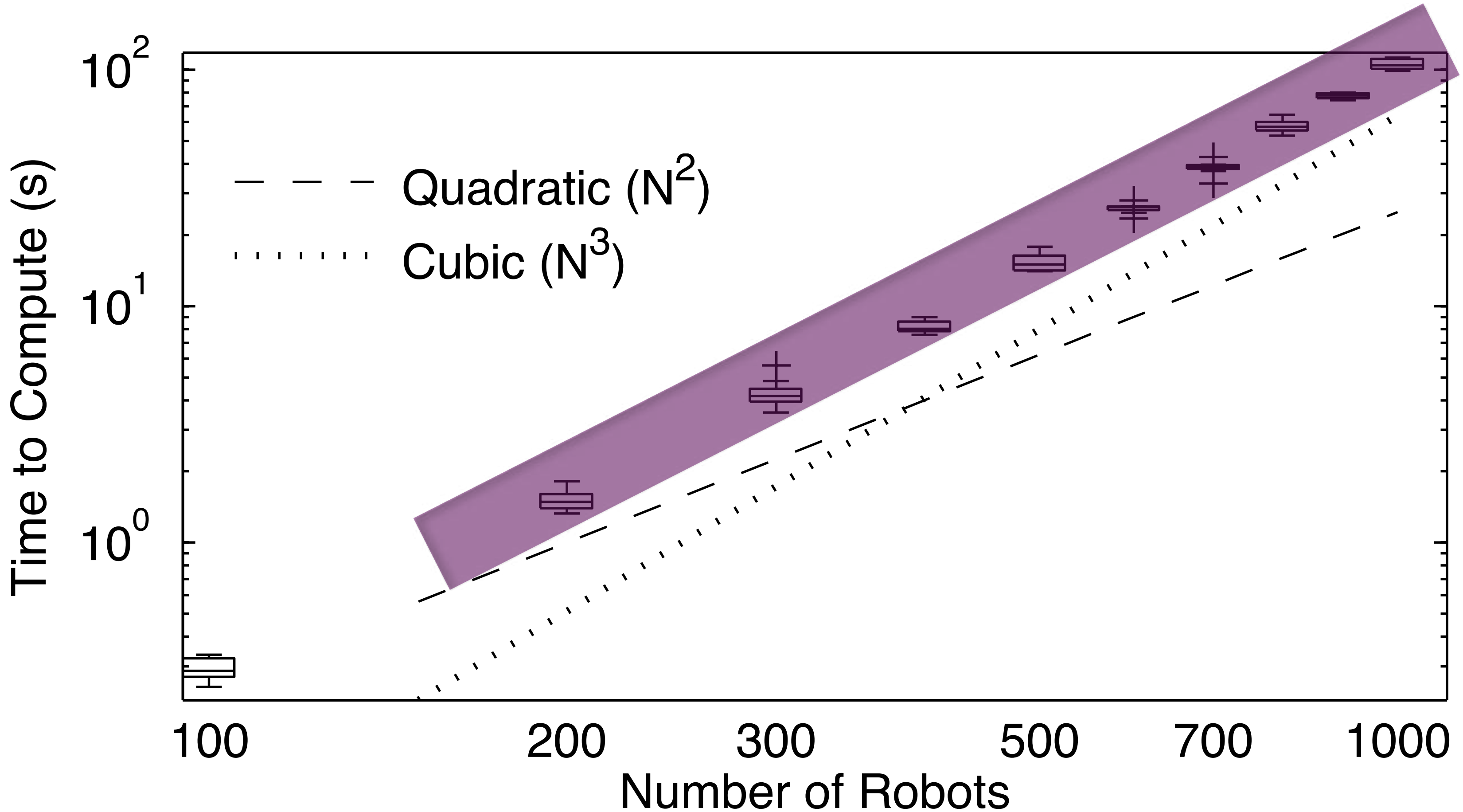
Assignments and trajectories that minimize the sum of square of distances

$$\underset{\phi, \gamma(t)}{\text{minimize}} \int_{t_0}^{t_f} \dot{X}(t)^T \dot{X}(t) dt$$

will be safe (no collisions)

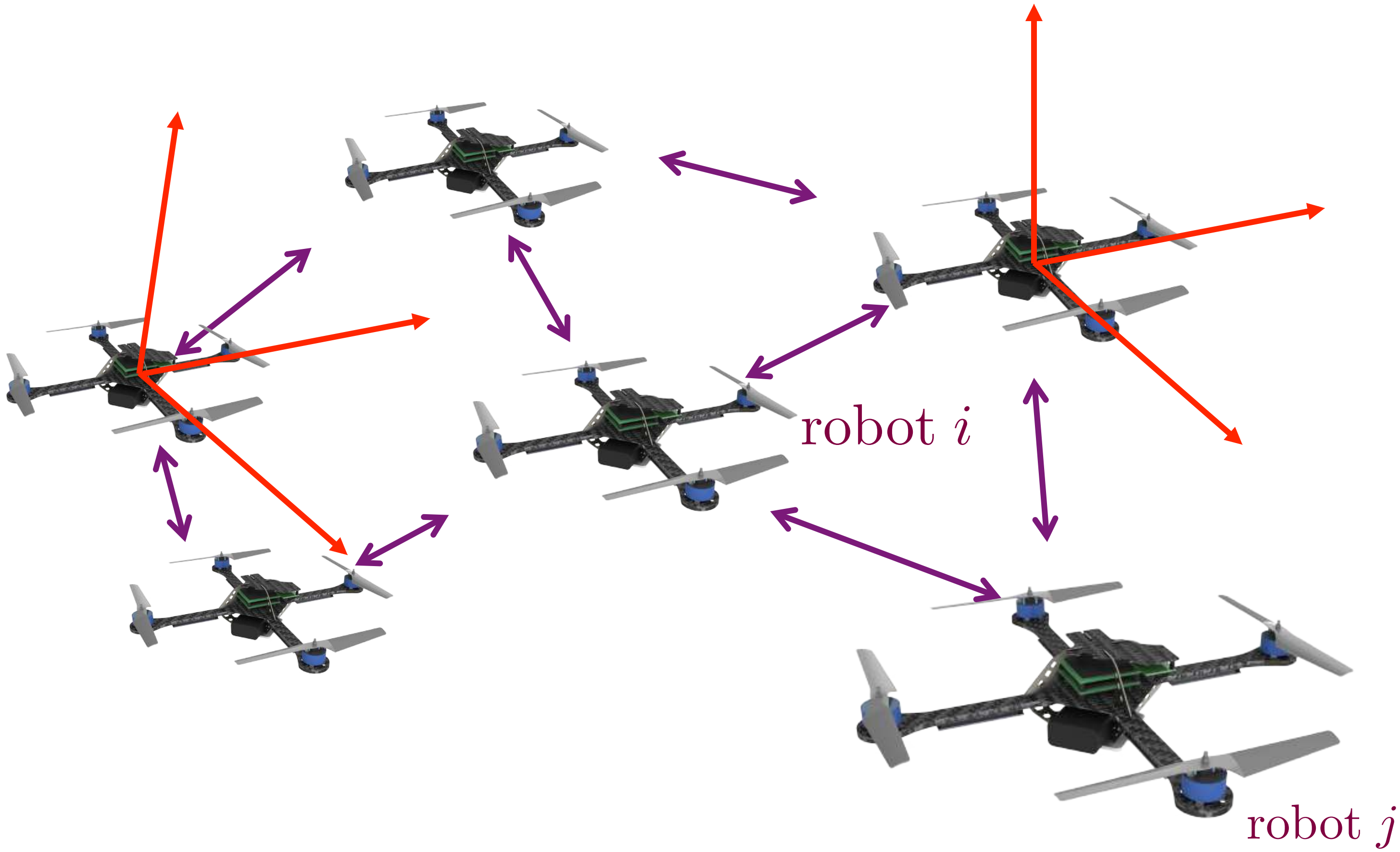
$$||\mathbf{x}_i(t) - \mathbf{x}_j(t)|| > 2R$$

CAPT



10^2
10^1
10^0
Time to Compute (s)
Quadratic (N^2)
Cubic (N^3)
100
200
300
500
700
1000
Number of Robots

2. Leader-Follower Networks



$$\mathbf{s}_{i,j}(t) = \mathbf{x}_j(t) - \mathbf{x}_i(t)$$

Leader-Follower Networks

PBS NOVA: Making Stuff Wilder (Hosted by David Pogue)

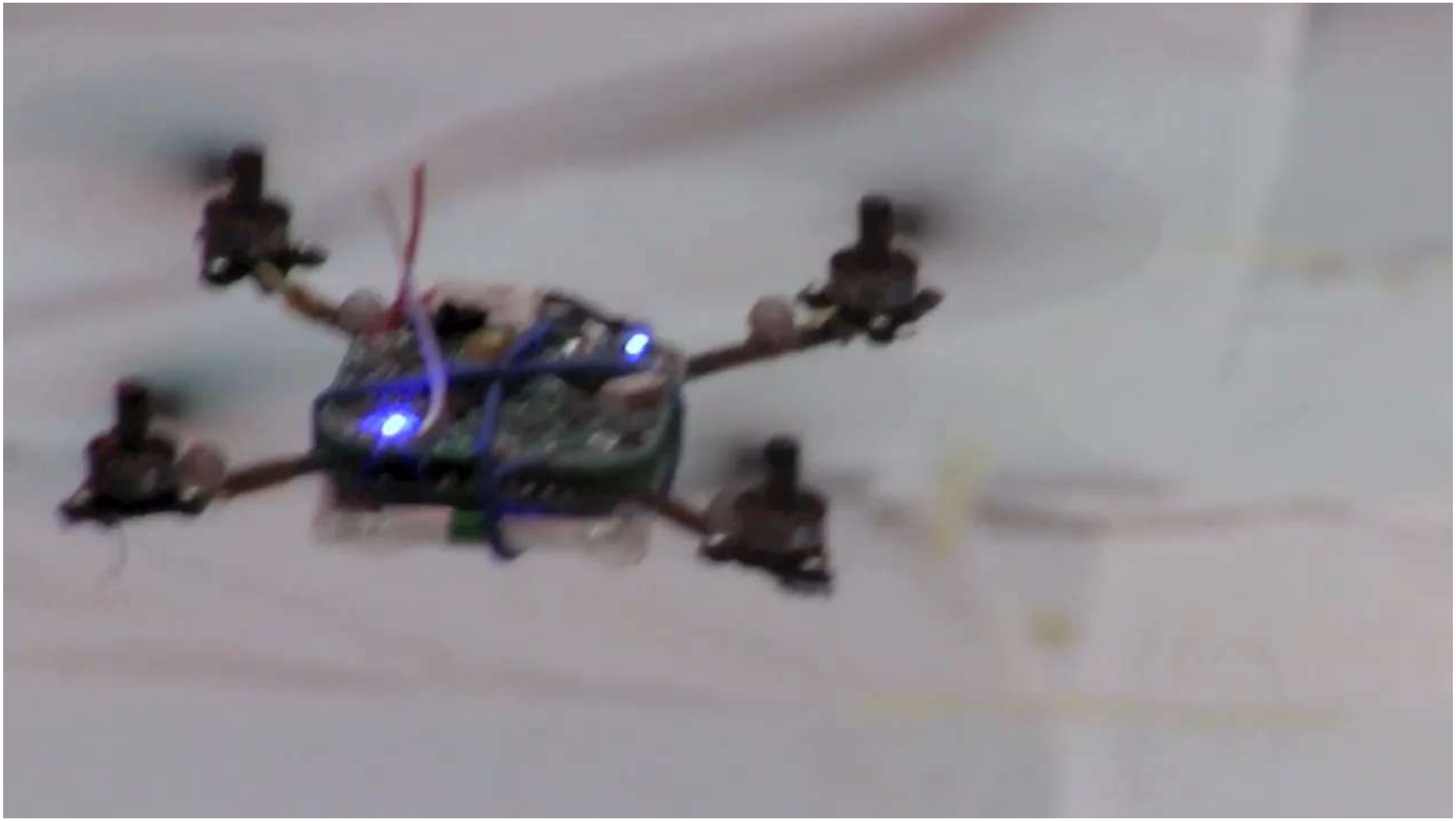
3. Anonymity

PBS NOVA: Making Stuff Wilder (Hosted by David Pogue)

Control of Formation Shape and Group Motion

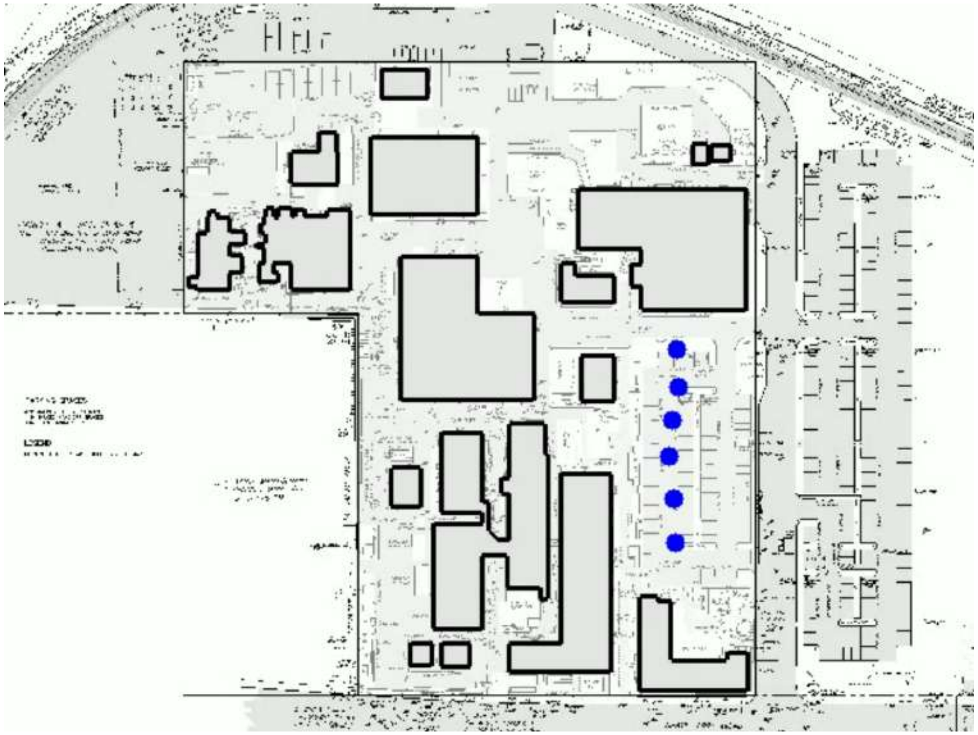
(Turpin, Michael, and Kumar, 2013)

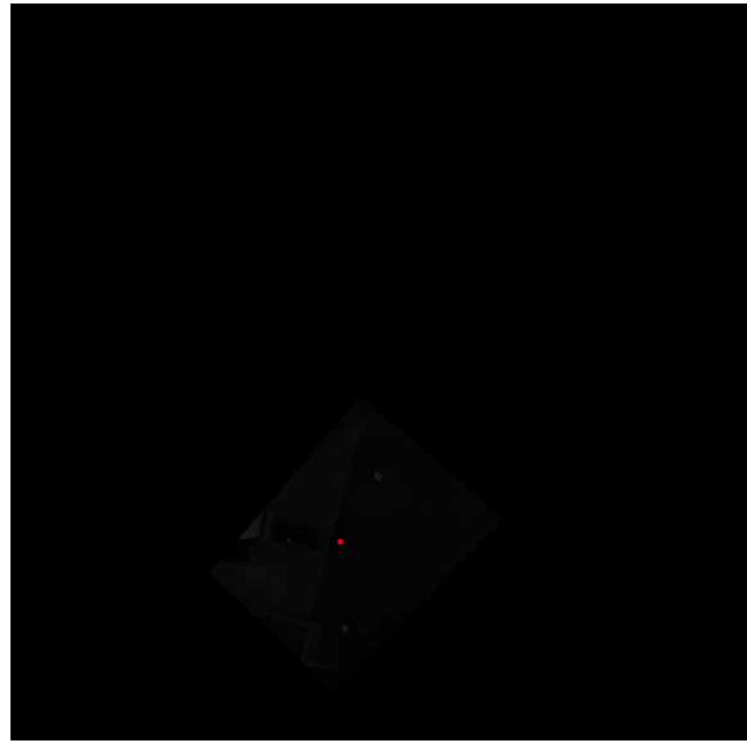
Control of Formation Shape and Group Motion

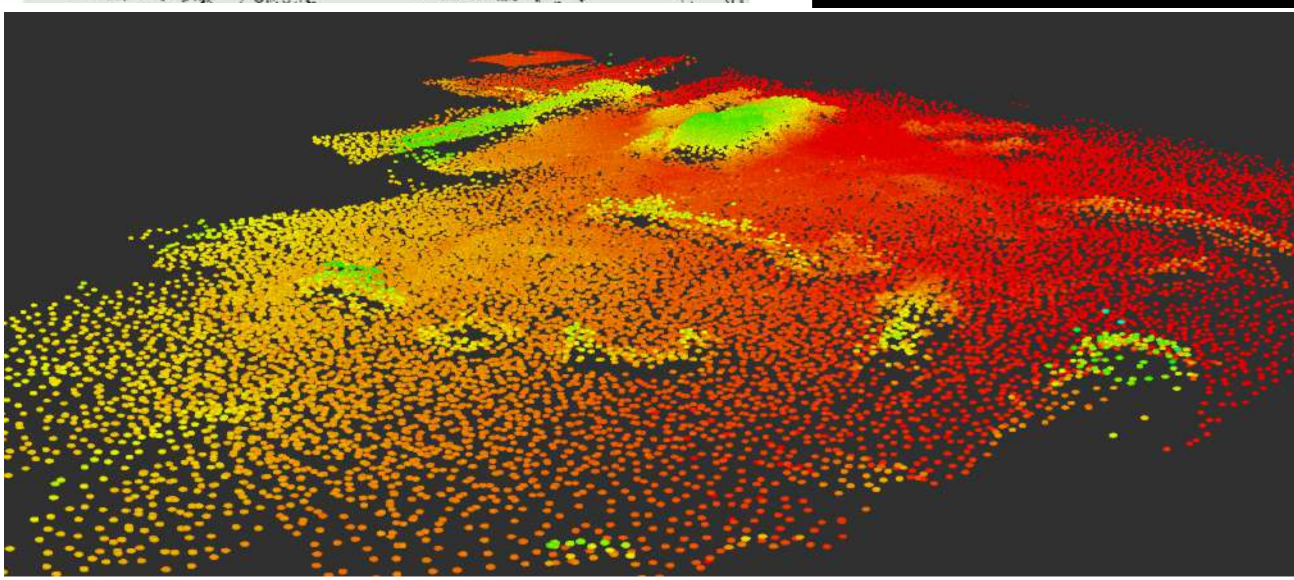


Robot First Responders

Kartik Mohta, Matthew Turpin, Alex Kushleyev, Daniel Mellinger, Nathan Michael, and Vijay Kumar, "QuadCloud: A Rapid Response Force with Quadrotor Teams," *Int. Symp. on Experimental Robotics* (ISER), 2014.





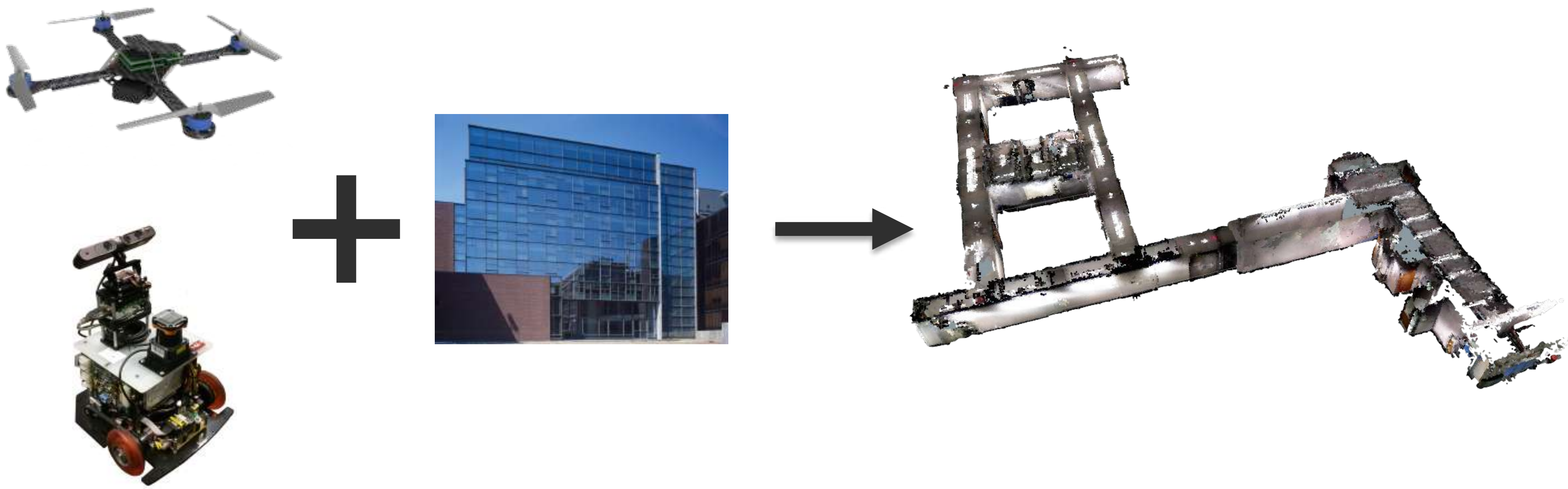


Kartik Mohta, Matthew Turpin, Alex Kushleyev, Daniel Mellinger, Nathan Michael, and Vijay Kumar, "QuadCloud: A Rapid Response Force with Quadrotor Teams," *Int. Symp. on Experimental Robotics* (ISER), 2014.

Enabling Cooperation

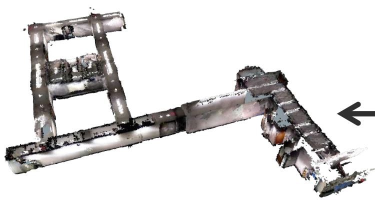
Active Mapping

Autonomously create 3D map of an unknown environment
with ground and aerial robots



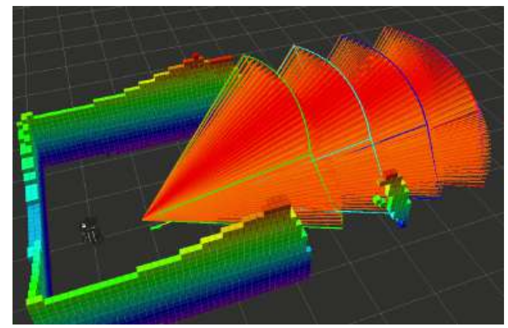
Control Policy

Reduce uncertainty of map by maximizing information gain



Map

Measurements

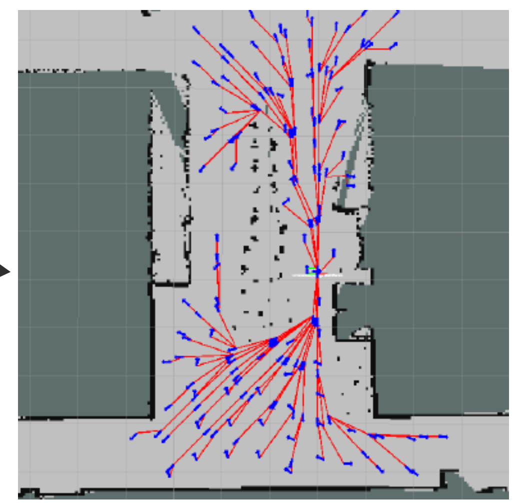


$$x^*_{\boldsymbol{\tau}} = \arg\max_{x_{\boldsymbol{\tau}}\in\mathcal{X}} \frac{\mathrm{I}_{\mathrm{CS}}[m; z_{\boldsymbol{\tau}} \mid x_{\boldsymbol{\tau}}]}{D(x_{\boldsymbol{\tau}})}$$

Information

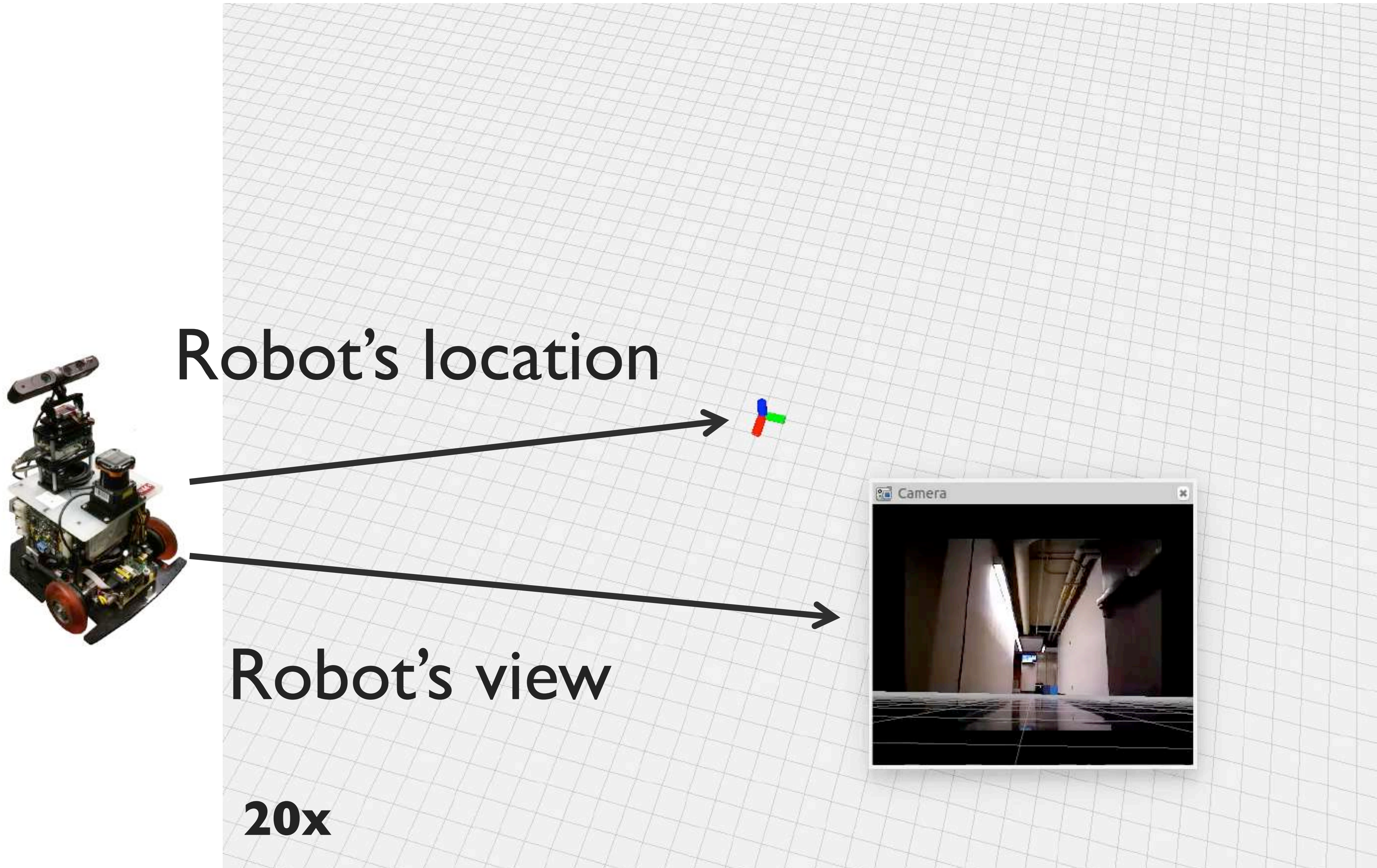
Duration

Paths



Change I_CS to IG (information gain)

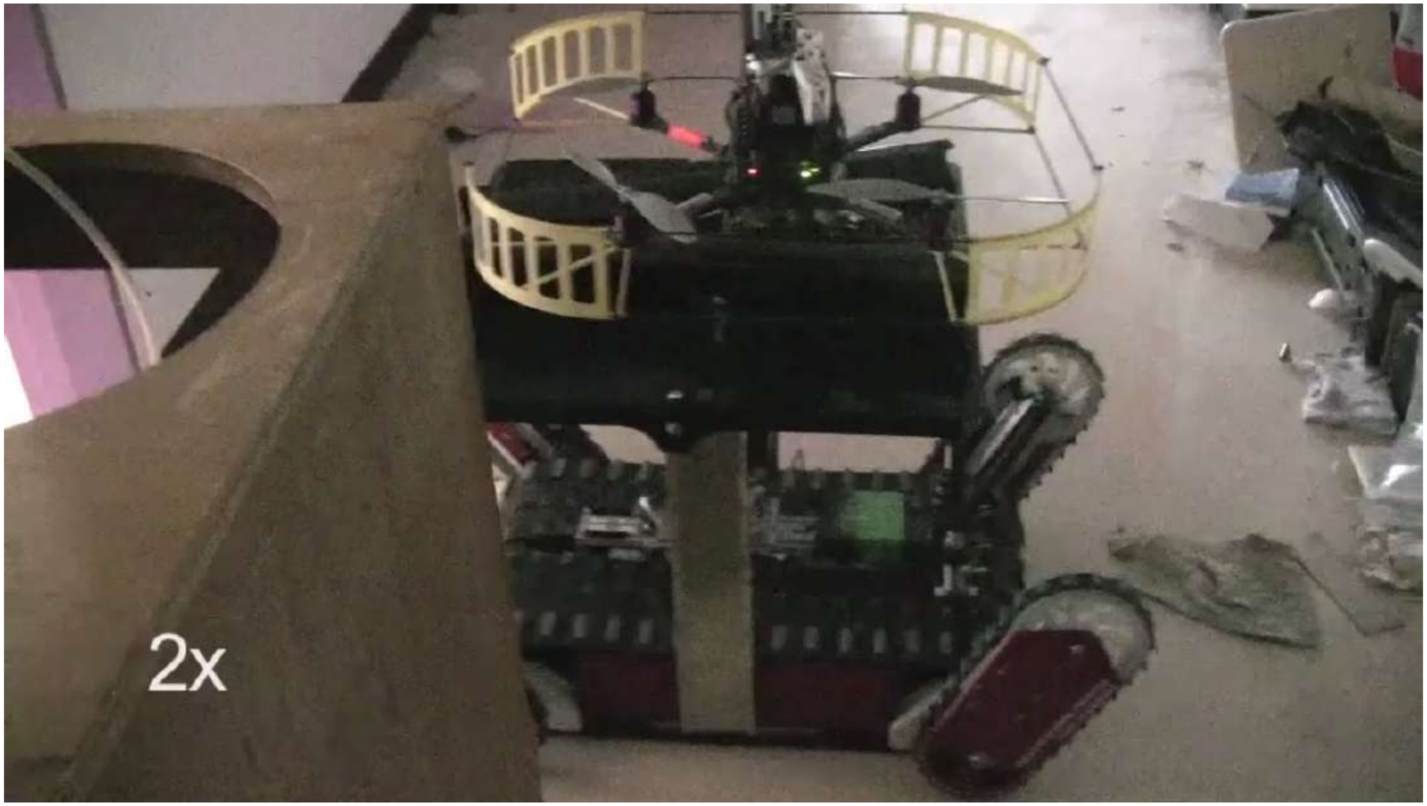
Active Mapping



Active Mapping

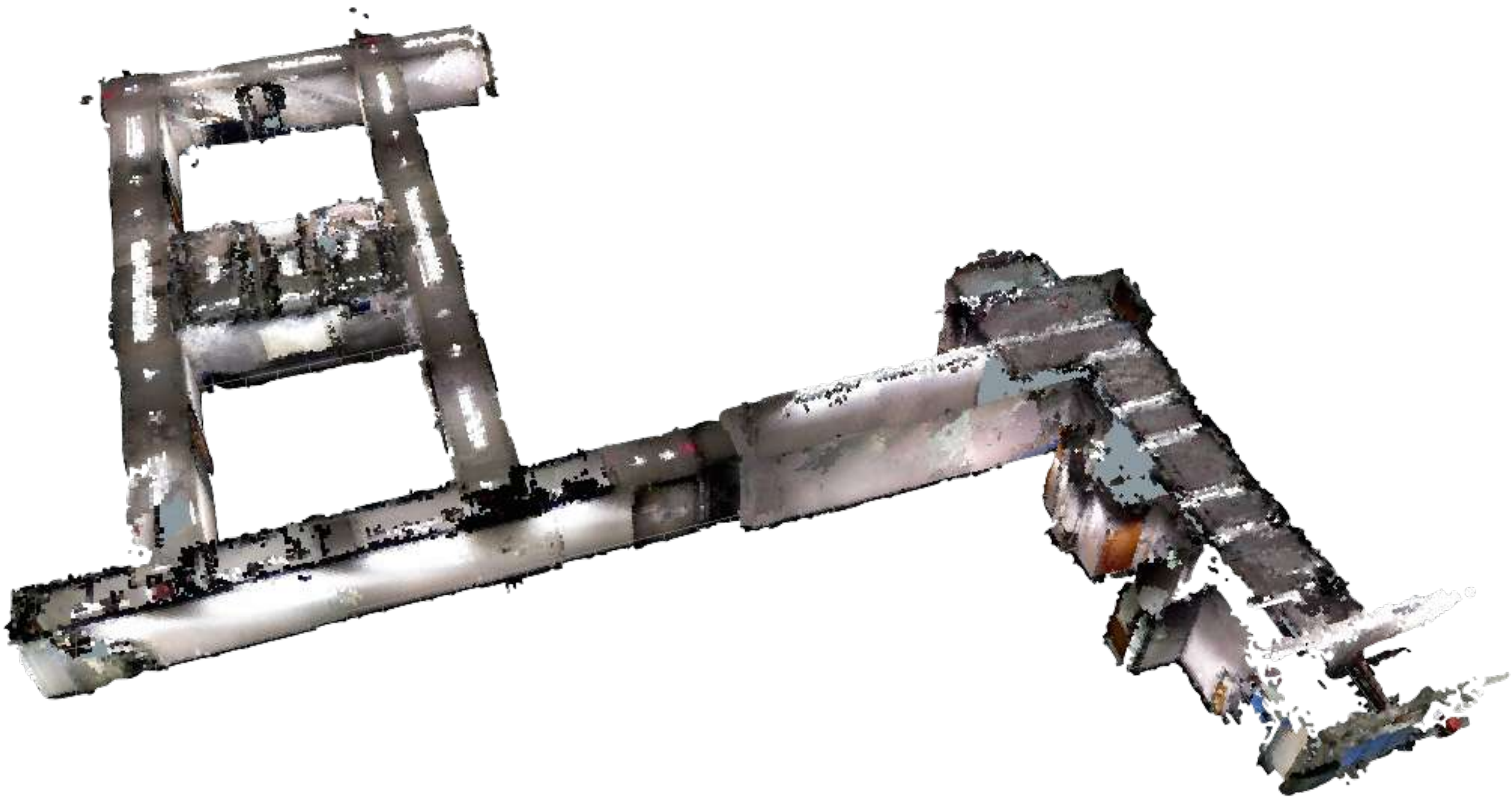
Search and Rescue

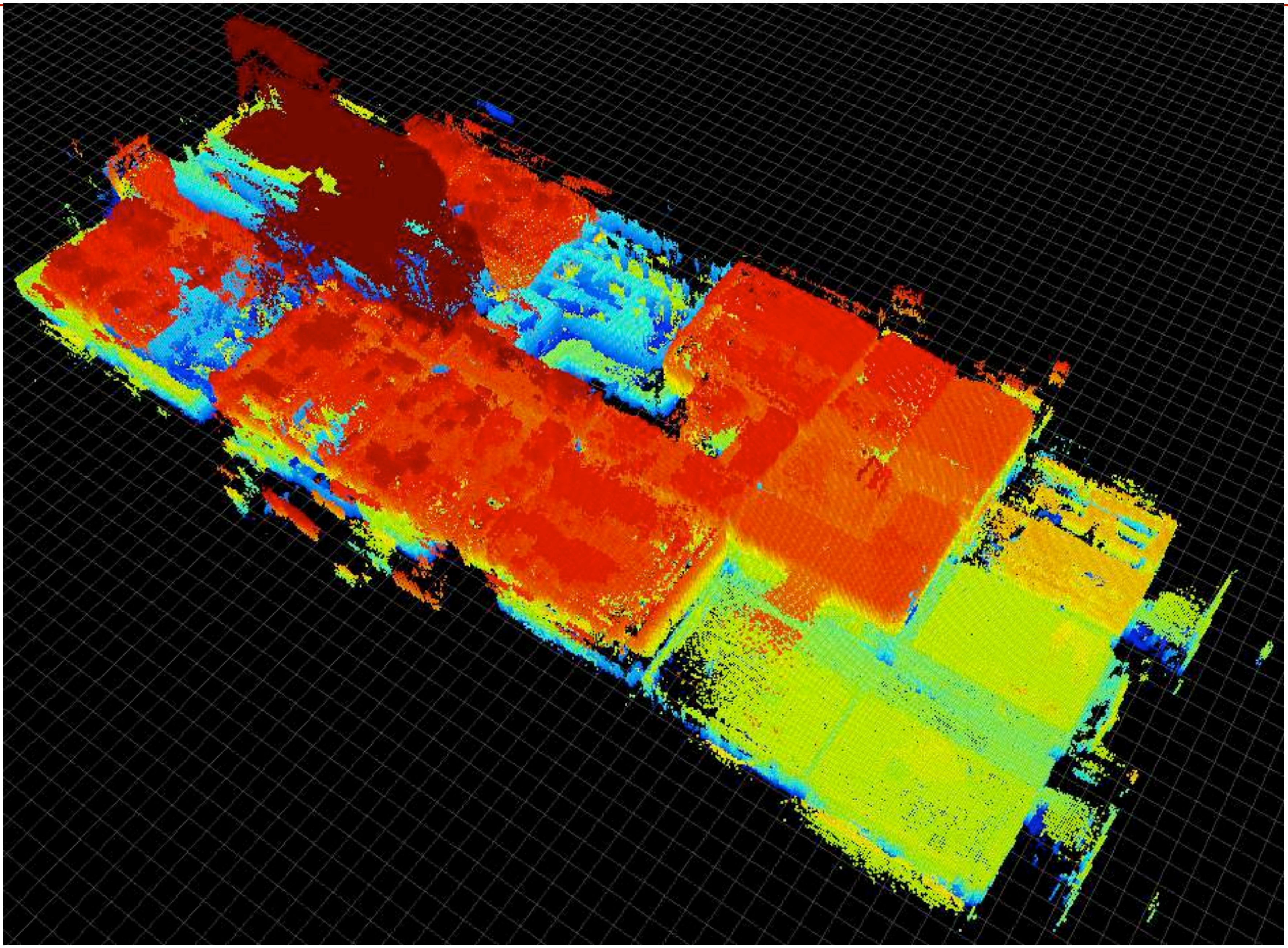
N. Michael, S. Shen, K. Mohta, Y. Mulgaonkar, V. Kumar, K. Nagatani, Y. Okada, S. Kiribayashi, K. Otake, K. Yoshida, K. Ohno, E. Takeuchi, and S. Tadokoro, "Collaborative mapping of an earthquake-damaged building via ground and aerial robots," J. Field Robotics, vol. 29, no. 5, pp. 832–841, 2012.



[*Michael et al*, 2012]

Final Map





3 floors of a 9 story building


Swarms!