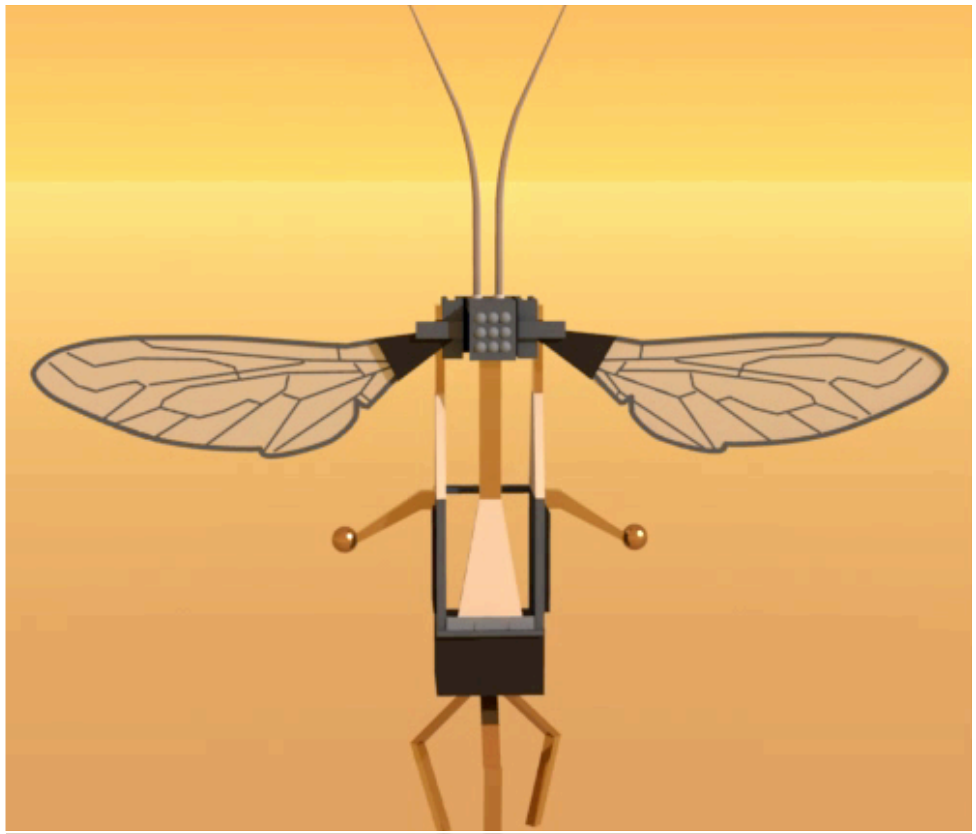
Programming Micro-Aerial Vehicle Swarms with *Karma*

Karthik Dantu
Bryan Kate
Jason Waterman
Harvard

Peter Bailis
UC Berkeley

Matt Welsh
Google Inc.

RoboBees Project



3cm length

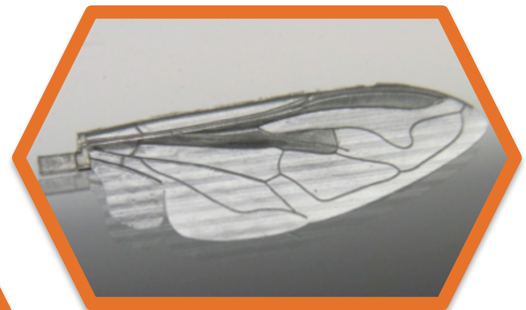
1g weight

500mW total power

5-yr $10M NSF grant to create flapping wing micro-aerial vehicles (MAV)

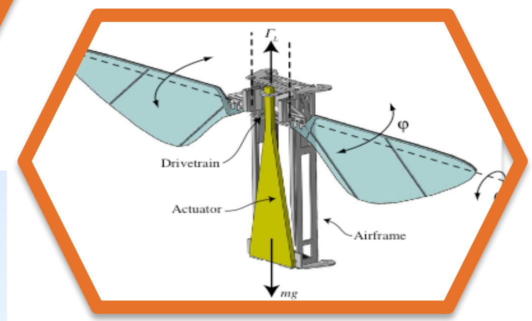
The RoboBees Project

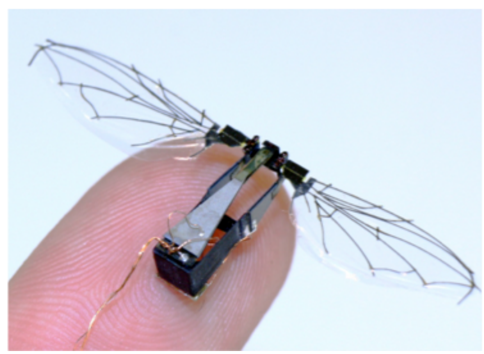
Biomimetic wing design

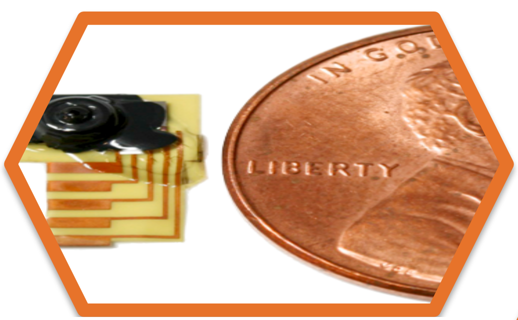


Coordination algorithms

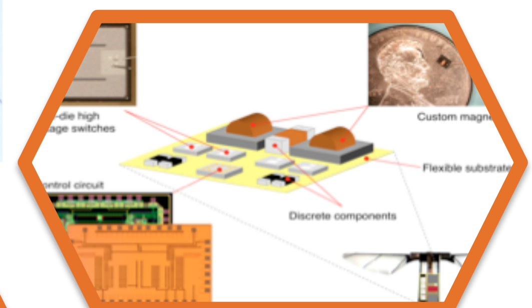
Novel actuation mechanisms



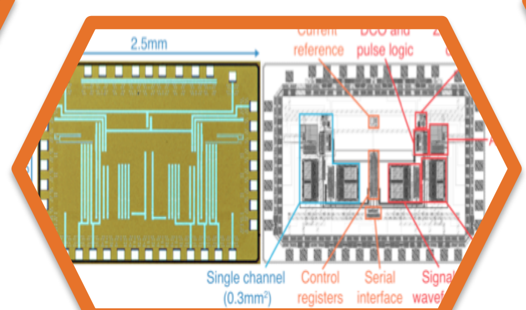




Low power vision sensors



Accelerator-based computing



Custom power electronics

Why MAV Swarms?

Micro-manipulation

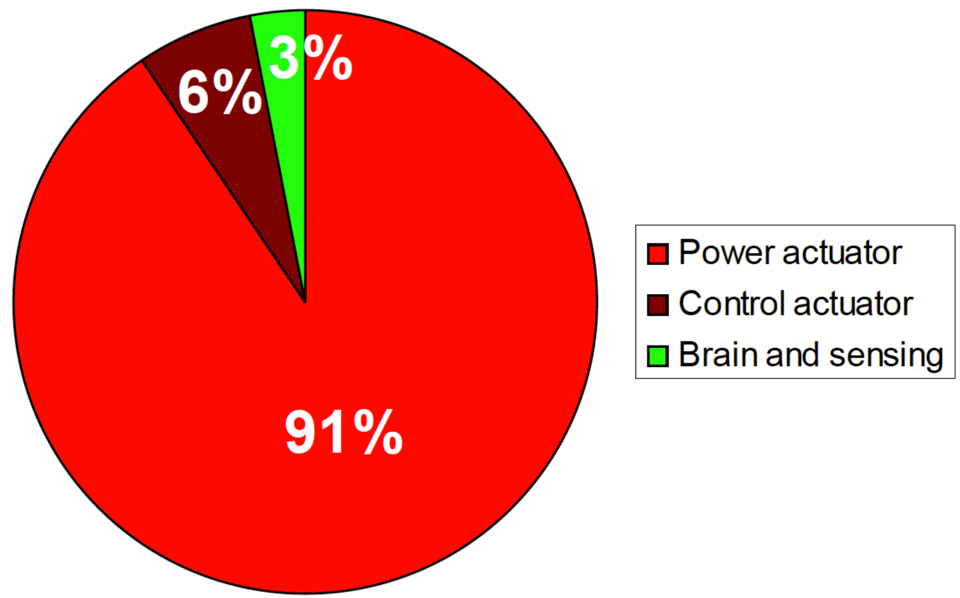
Covert Operation

Hazardous Missions

Tracking Dynamic Phenomena

Challenge #1: Limited Energy

- Actuation dominates energy budget
- Expected flight time **around 10 minutes**
- Sensing and computing have a **budget of 15 mW** from a total of about 500mW

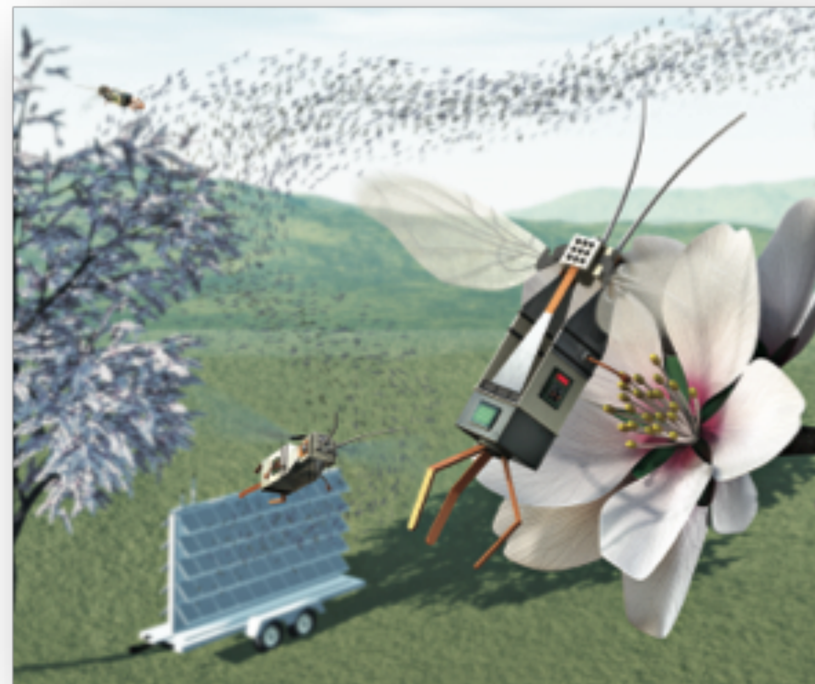


RoboBee Power Projection

Challenge #2: Programming The Swarm

- Hundreds of MAVs in a swarm
  - Programming individual MAVs is difficult
  - Inter-MAV coordination is complicated

- Individual MAVs are resource-constrained
  - Can only execute simple sensing/actuation commands



Challenge #3: Adapting To Changing Conditions

- Environmental conditions
- Individual MAV failure
- Error in sensing, actuation, localization

Contributions

Karma – An operating system for programming and coordinating MAV swarms

- Simplified programming model to decompose complex applications for the swarm
- Provides centralized coordination that efficiently manages swarm resources
- Adapts to changing environmental conditions and is resilient to individual MAV failures

Example Application: Crop Pollination

- 70% of cultivated produce is pollinated by insects
- Massively parallel micro-manipulation problem

Key Observation

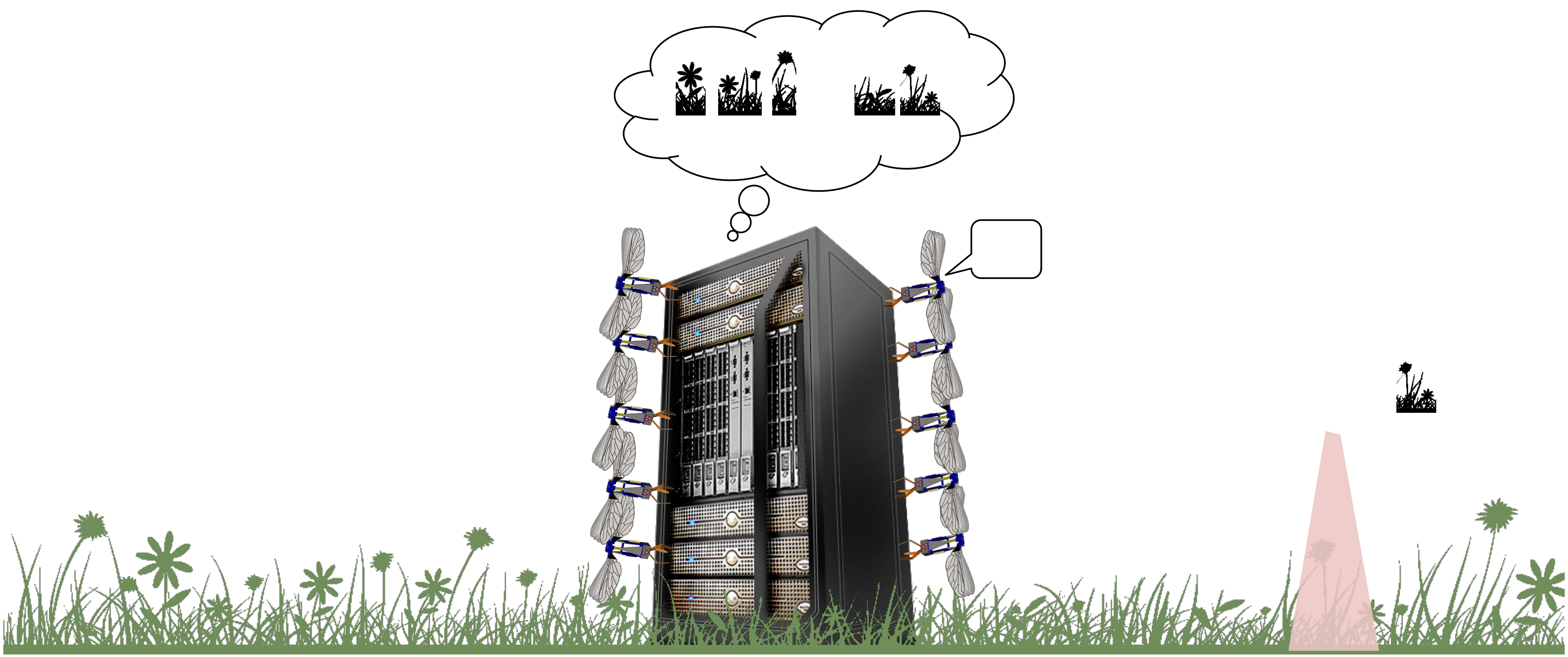
MAVs have limited flight time and need to frequently recharge

Idea: Use the recharge station for centralized coordination

The Hive-Drone Model

HIVE: uses information to dispatch new drones

DRONES: collect data from the field and return it to the hive



Programming the Swarm

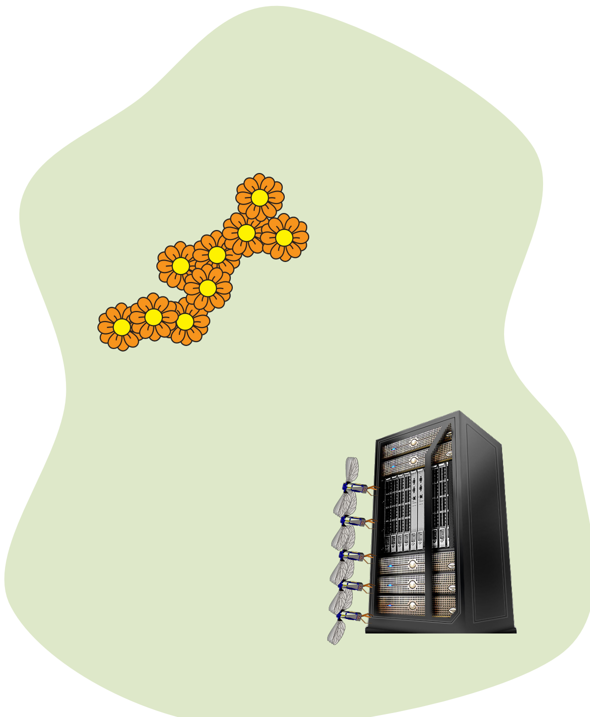
Idea : Hive can reprogram each drone on every sortie
Sortie: atomic unit of work

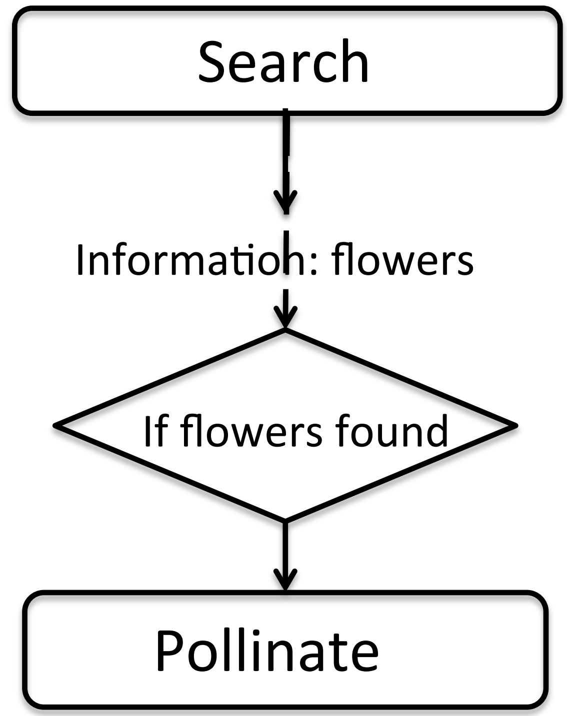
Behavior: Program run by a single drone on each sortie

- On execution, behaviors produce information

This information is used by the hive to drive future deployment decisions of behaviors to drones

Application Composition





Key Idea: Arbitrarily complex applications can be composed by indirectly wiring behaviors through information produced and consumed

Recap

Hive-Drone Model: due to limited MAV flight time, use centralized hive for coordination

Drones repeatedly run sorties to perform actions and collect information

Compose simple drone-level behaviors to build complex swarm-level applications

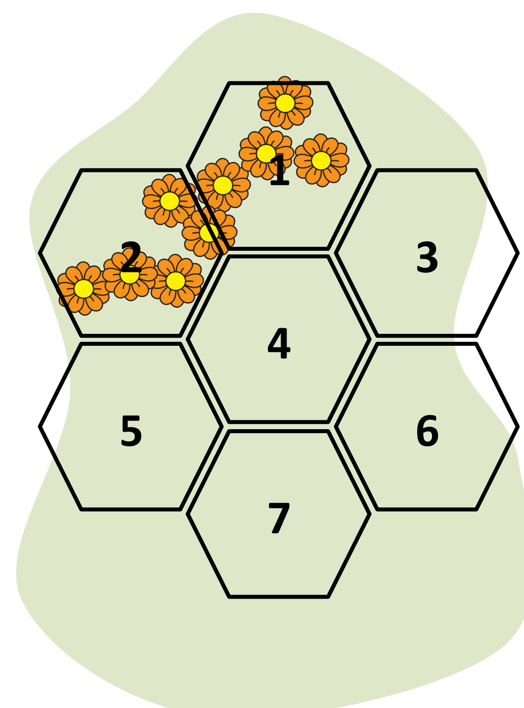
Reasoning About Space

Every swarm application is inherently spatial

Problem: Each MAV can only cover a limited area in one sortie

Solution:

- Divide space into Regions
- Reason about behavior-region pairs
- Turn coordination into a scheduling problem

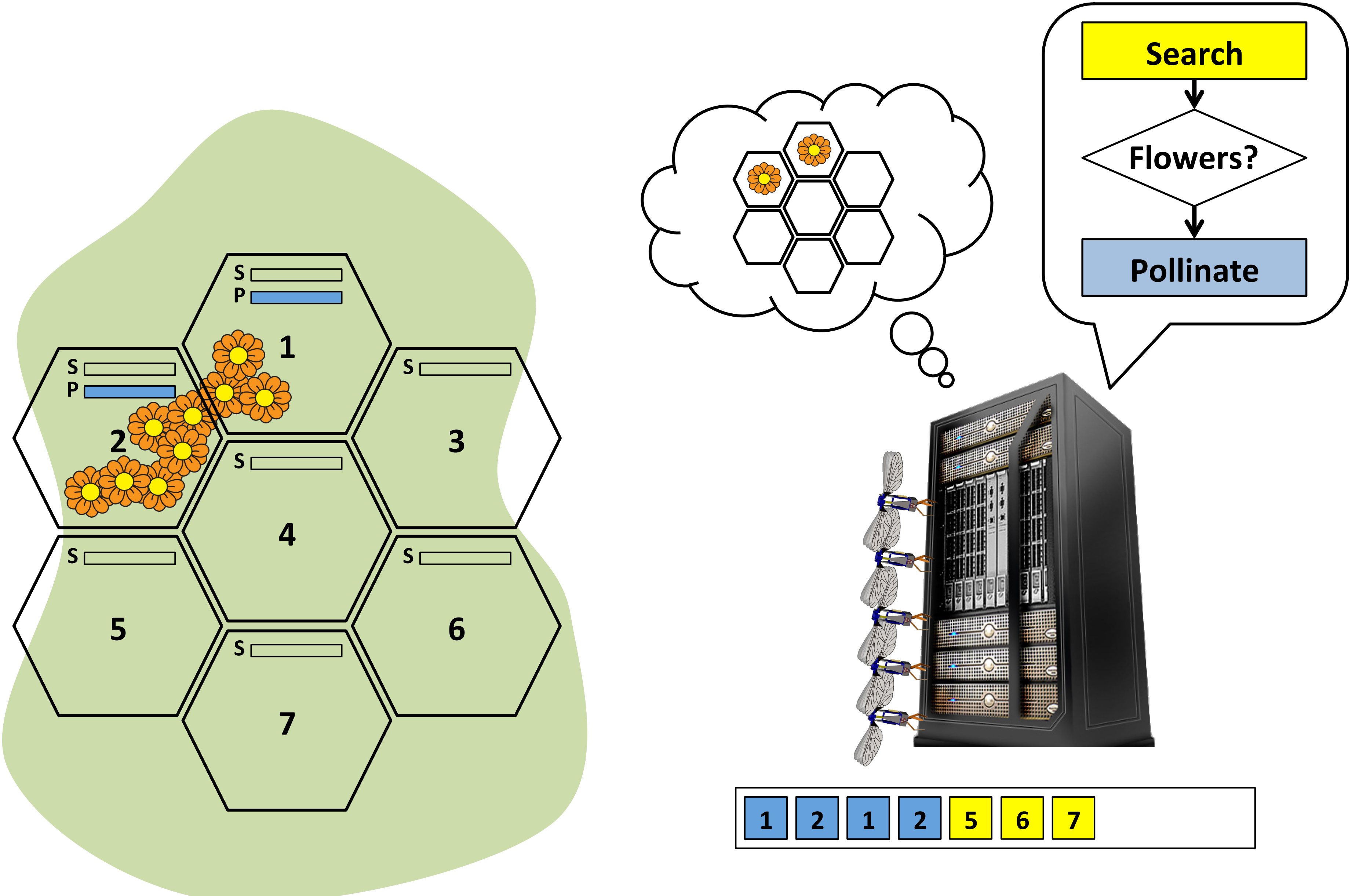


Scheduling in *Karma*

Problem: Allocate sorties to behavior-region pairs

Solution: Drones allocated to behavior-region pairs according to estimated work remaining

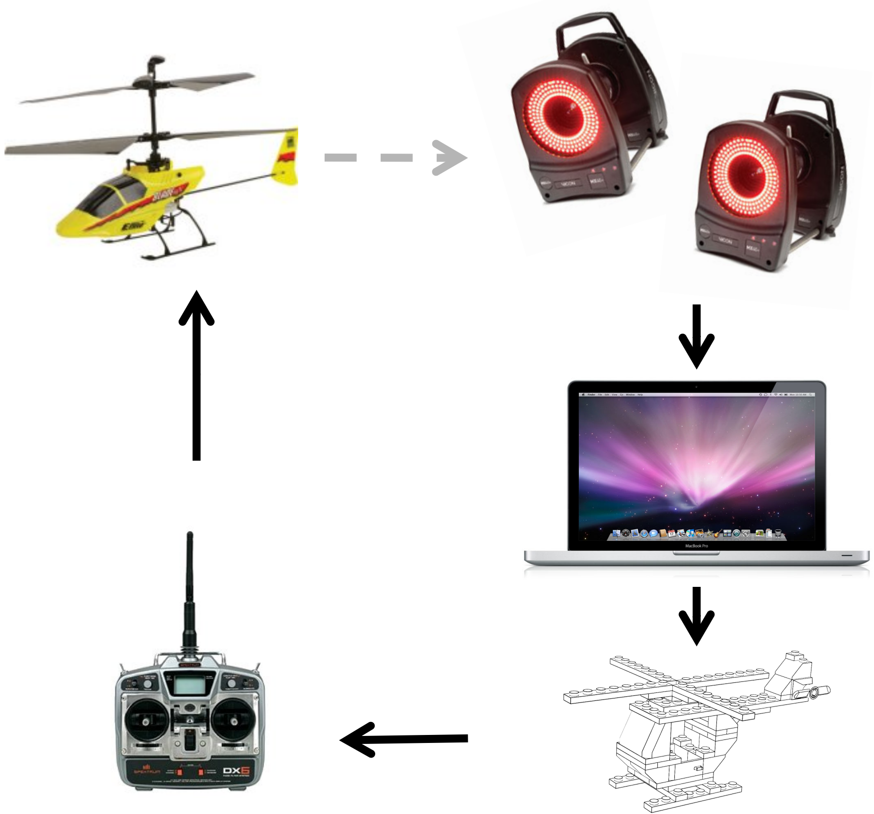
Policy: Minimize application completion time while balancing the rate of progress of each behavior



S
P
1
S
P
2
S
3
S
4
S
5
S
6
S
7
Search
Flowers?
Pollinate
1
2
1
2
5
6
7

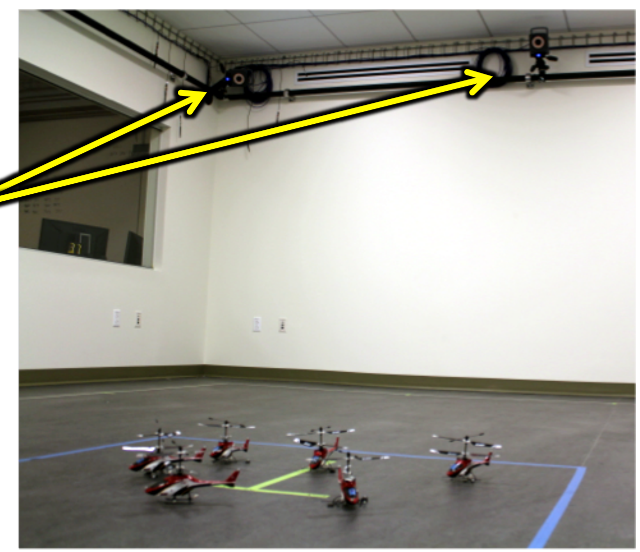
Testbed Overview

- E-flite MCX2 micro-helicopters
- Vicon motion capture system



Vicon markers

Vicon cameras

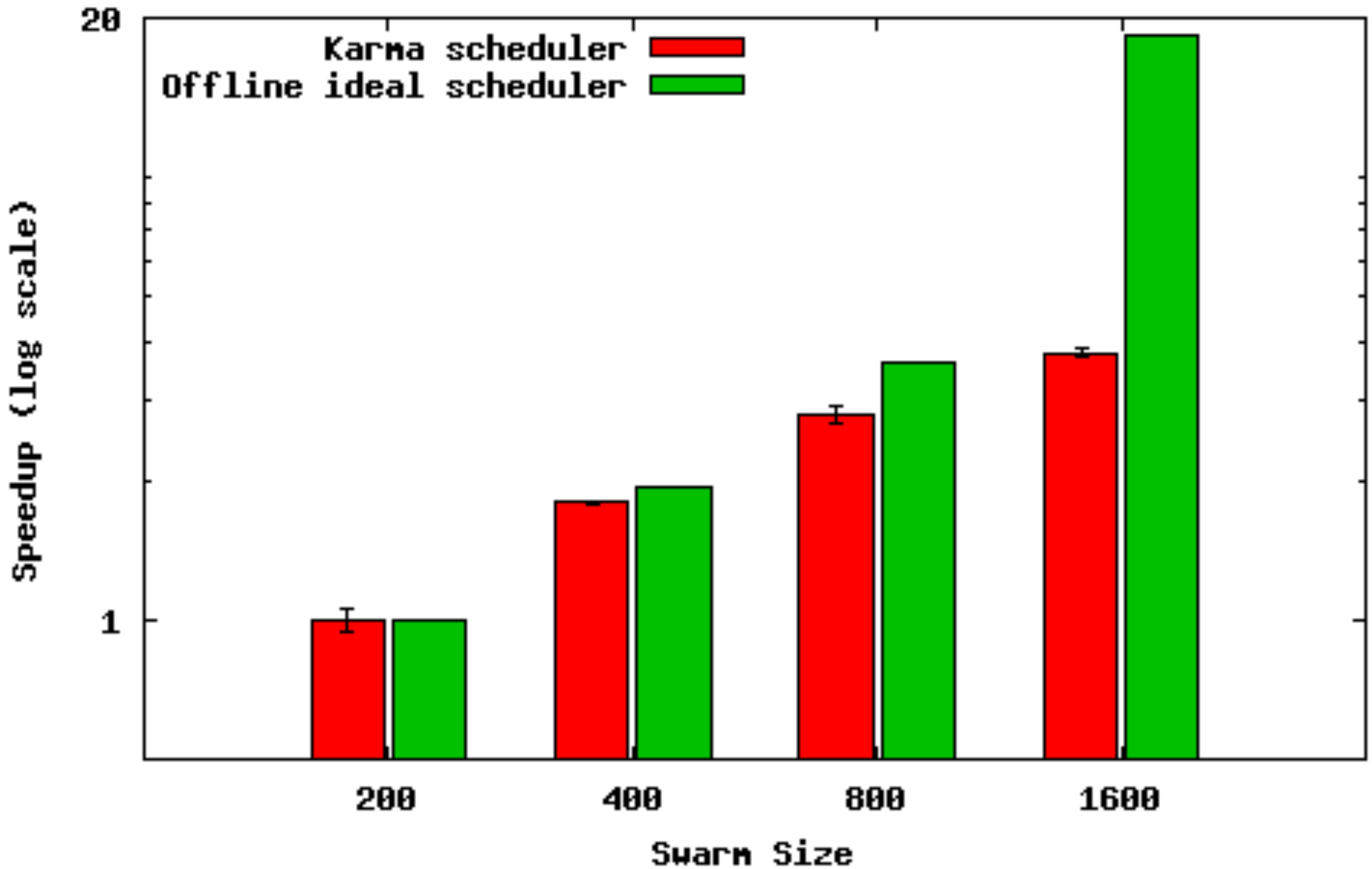


Evaluation Metrics

Efficiency: How efficiently does *Karma* manage its resources?

Adaptivity: Does *Karma* adapt to changing environmental conditions?

Evaluation: Efficiency



Scales well with increase in swarm size

Evaluation: Adaptivity

- Introduced wind in a third of the area of operation
- Decreased sortie times by 32%
- *Karma* deployed 12% more drones to that area of operation
- Scenario took 7% longer for completion

Karma is adaptive to changes in environmental conditions

Increased environmental dynamics results in shorter sorties and longer scenarios

Evaluation

Also in the paper ….

- Demonstrated other applications – Plume Tracking, Target Tracking
- Studied effect of individual drone failure on performance
- Measured information latency and introduced a continuous scheduler to minimize information latency

Conclusion

Karma – a system to program and coordinate swarms of Micro-Aerial Vehicles

- Provides a simple programming model to express complex applications
- Central coordination uses swarm resources efficiently
- System adapts to changing conditions and individual MAV failure

Karthik Dantu (kar@eecs.harvard.edu)

http://robobees.seas.harvard.edu

http://youtube.com/RobobeesColony

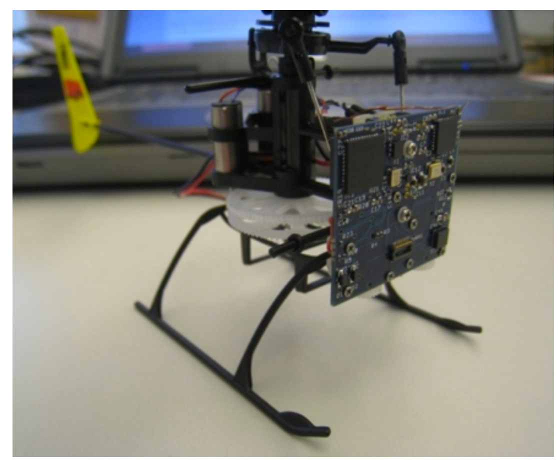
Backup

Localization

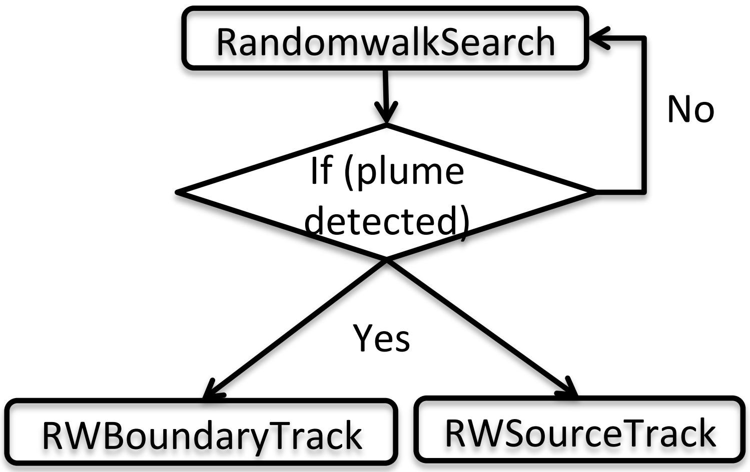
- Outdoor
  - Harmonic Radar
  - Spotlight
- Indoor
  - Dead-reckoning + beaconing
  - Infrastructure-based support
- Local feature-based
- Stigmergy
- Localization can be imprecise
- Near the hive vs. in-field
- Long-term navigation vs. close range detection

Potential Mechanisms

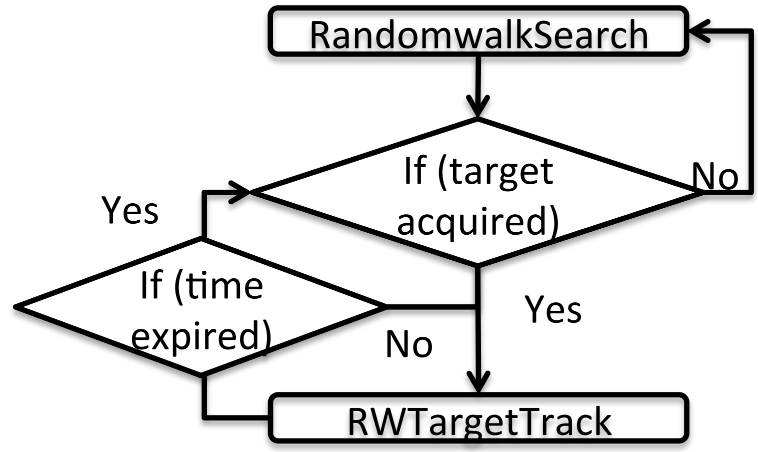
- Kinect Tracking
- Sun Compass
- Beaconing from the hive - Harmonic Radar
- Inertial sensing
- Template matching for target detection
- RF
  - Interferometry
  - Connectivity
- Compass and optic-flow odometry



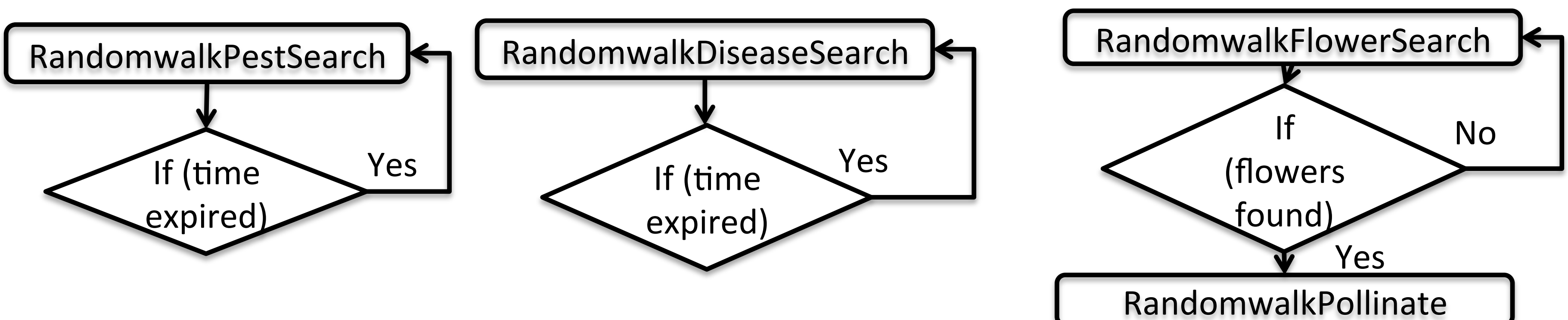
Sample Application Task Graphs



Plume Tracking



Target Tracking



Alfalfa Crop Monitoring

Evaluation: Resilience to Failure

Probability of Failure (% per sortie)	Average Time for Completion
0.277	81.1
0.55	84.1
1.11	86.16
2.22	102.76