Networking Swarm

Members: Rachel Chong
         Zhilong Liu
         James Lam Yi
         Sarah Hung

Instructor: Prof. Edward Lee
           Prof. Alberto L. Sangiovanni-Vincentelli
Agenda

- Objective
- Hardware
- RSSI
- Software Architecture
  - Gradient Ascend
  - Communication
- Video
- Application
- Future Work
- Acknowledgement
- Reference
Objective

The objective of this project is to implement a swarm of robots that collaborate via network communication as well as low powered radio RSSI (Received Signal Strength Indicator) to coordinate their movements.
Objective

Gradient Descend
Objective

Bot 1
Bot 2
Bot 3
Bot 4

Gradient Ascend
Objective

Gradient Descend
Objective

Gradient Ascend
Objective

Bot 1
Bot 2
Bot 3
Bot 4

Gradient Ascend
Objective

Gradient Descend
Objective

Bot 1

Bot 2

Bot 3

Bot 4

Gradient Ascend
Objective

Gradient Ascend
Objective

Gradient Ascend
Objective

Gradient Ascend
Objective

Gradient Ascend
Objective

Gradient Ascend
Objective

Gradient Ascend
Objective

Gradient Ascend
Objective

Gradient Ascend
Objective

Gradient Ascend
Objective

Gradient Descend
Objective

Gradient Descend
Hardware

Biomimetic Millisystems Laboratory
Hardware

Zumy Robot

- Battery
- Power Module
- Odroid
- WiFi
- XBee Module
- Motors
- Mbed

Odroid U3

transmit
receive
Hardware
Hardware

- XBee Series 1
  - 2.4 GHz
  - (802.15.4)
Hardware

Zumy Robot

- Battery
- Power Module
- Odroid
- WiFi
- XBee Module
- Motors
- Mbed

transmit  receive

WiFi
Hardware

Zumo Robot Kit

Pololu DRV8833 Dual Motor Driver Carrier

Zumo Robot Kit
Hardware

Pololu Step-Down Voltage Regulator
Received Signal Strength Indicator

- experiments investigating RSSI vs. distance
- fluctuate randomly over large distance but approx. linear within 1.5 m
- within this range, gradient ascend & descend is reasonably robust
Received Signal Strength Indicator

- experiments investigating RSSI vs. distance
- fluctuate randomly over large distance but approx. linear within 1.5 m
- within this range, gradient ascend & descend within this range is reasonably robust
Software Architecture

- Xbee Wrapper Library
- LCM Motor Driver
- Bot Chaining Algorithm
Bot Chaining Overall State Machines
Gradient Ascend Algorithm

- Basic algorithm
- Ending RSSI: -38dB
- Drive time function
- Stage benefit
Gradient Ascend State Machine

input: signal, buffer
output: transmit_state

ARRIVED

COLLECT

DRIVE FORWARD

done/

COLLECT & COMPARE

done/

TURN 90

ARRIVED

better_RSSI/

-worse_RSSI/

-ARRIVAL

Gradient Ascend
Gradient Descend Algorithm

- Basic algorithm
- Multiple Zumy Bots
Communication Threads

- Transmit
- Receive

ZUMY 1

ZUMY 2

ZUMY 3
Communication Threads

START_ASCEND
ACK
STOP_ACK
REGULAR
COMMAND-ACKNOWLEDGEMENT CYCLE

ZUMY 1
transmit
receive

ZUMY 2
transmit
receive

ZUMY 3
transmit
receive
Communication Threads

START_ASCEND

ACK

STOP_ACK

REGULAR

COMMAND-ACKNOWLEDGEMENT CYCLE

ZUMY 1

transmit

receive

ZUMY 2

transmit

receive

ZUMY 3

transmit

receive

COMMAND-ACKNOWLEDGEMENT CYCLE

ACK

ARRIVAL

STOP_ACK

REGULAR
Communication Packet

start gradient ascend
Communication

robot ID
(unique for each robot)

command

start gradient ascend
Communication

acknowledge gradient ascend
Communication

intended recipient

acknowledge gradient ascend
Communication

stop acknowledgement
Communication

go back to normal mode
Communication

go back to normal mode
Video

https://www.youtube.com/watch?v=NIDYIy6H6Tw
Final Touch

- Analyze runtime of
  - gradient ascend/descend
  - command-acknowledgement cycle
Application

- Exploration in hazardous environment
- Environment mapping
- Monitoring system
Future Work

- Append with Laser Range Finder
- Measuring RSSI in motion
- Automatic bot chaining
- Automatic motion coordination by neighbor bots
- Localization
Acknowledgement

Special thanks to:

Prof. Edward Lee
Prof. Alberto L. Sangiovanni-Vincentelli
Prof. Rob Fearing
Matt Weber
Ben Zhang
Antonio Ianopollo
Austin Buchan
Andrew Pullin
References

XBee Python Library by Paul Malmsten, Greg Rapp, and Brian.  
https://github.com/blalor/python-xbee

Zumy Library by Andrew Chen.  
https://github.com/andrewjchen/zumy