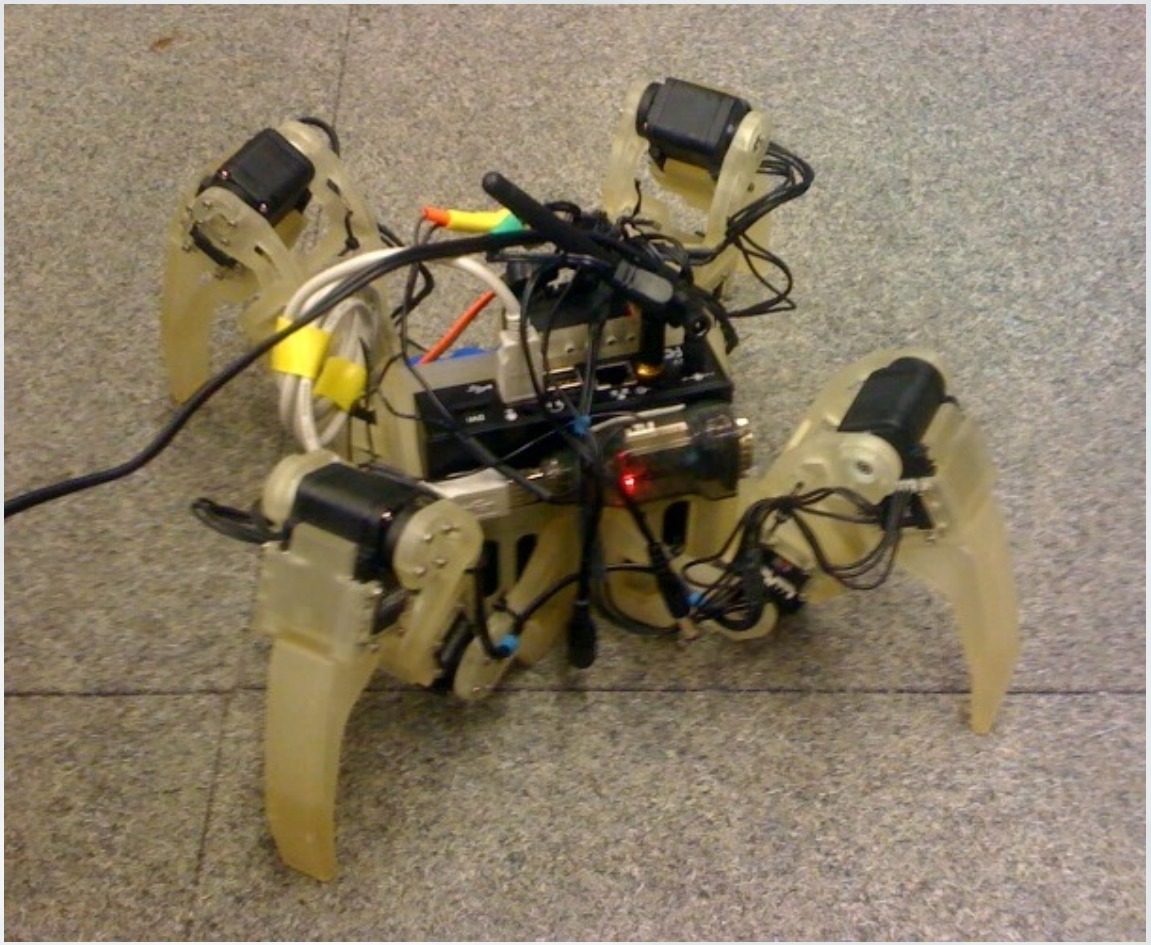
QuadraTot Bot

CS 4701: Practicum in A.I.
Code Review #1

Sarah Nguyen (smn64),
Jason Yosinski (jy495),
Diana Hidalgo (djh283)

Problem Statement and Motivation

- **Goal** – teach a quadruped robot how to walk using learning algorithms
- **Competition** – beat current preprogrammed gait



Schedule

	Milestones	**Deadlines**
Week 1 (9/13-9/19)	Read papers, get lab access, talk to relevant other researchers	9/17 Final proposals due
Week 2 (9/20-9/26)	Continue reading, get robot to move	
Week 3 (9/27-10/03)	Implement parametrized gait and determine proposed coding schedule for more advanced algorithms in time for Code Review #1.	
Week 4 (10/04-10/10)	Begin main algorithm dev/testing effort	10/5 Code Review #1
Week 5 (10/11-10/17)	Algorithm dev/testing	
Week 6 (10/18-10/24)	Algorithm dev/testing	
Week 7 (10/25-10/31)	Algorithm dev/testing, quantify/solidify current results for Code Review #2	
Week 8 (11/1-11/7)	Finish collecting results, begin writing	11/2 Code RReview #2
Week 9 (11/8-11/14)	Finish collecting results, writing	
Week 10 (11/15-11/21)	Finish collecting results, writing, get final demo ready	
Week 11 (11/22-11/28)	Finish collecting results, writing, get final demo ready	
Week 12 (11/29-11/30)	Final demo	11/30 Final presentation

Schedule

	Milestones	Deadlines
Week 1 (9/13-9/19)	~~Read papers, get lab access, talk to relevant other researchers~~	9/17 Final proposals due
Week 2 (9/20-9/26)	~~Continue reading, get robot to move~~	
Week 3 (9/27-10/03)	~~Implement parametrized gait and determine proposed coding schedule for more advanced algorithms in time for Code Review #1.~~	
Week 4 (10/04-10/10)	~~Begin main algorithm dev/testing effort~~	10/5 Code Review #1
Week 5 (10/11-10/17)	Algorithm dev/testing	
Week 6 (10/18-10/24)	Algorithm dev/testing	
Week 7 (10/25-10/31)	Algorithm dev/testing, quantify/solidify current results for Code Review #2	
Week 8 (11/1-11/7)	Finish collecting results, begin writing	11/2 Code Review #2
Week 9 (11/8-11/14)	Finish collecting results, writing	
Week 10 (11/15-11/21)	Finish collecting results, writing, get final demo ready	
Week 11 (11/22-11/28)	Finish collecting results, writing, get final demo ready	
Week 12 (11/29-11/30)	Final demo	11/30 Final presentation

Implementation Status

- Robot Class
- Optimization
- Parameterized Models
  - SineModel
- Camera Feedback

Implementation Status

Robot Class

- Current Implementation: Nearly done.
  - Commands robot positions
  - Accepts any motion model function: time -> R^9
  - Prevents legs colliding with self
  - Interpolates between fixed and moving positions for smooth motion and accurate measurements
- Future Improvements: None
- Status: 90% completed

Optimization

- Current Implementation: Random hill climbing algorithm that creates random neighbors by randomly choosing one parameter to adjust, and changing it either completely randomly or slightly.
- Future Improvements: Gradient Descent
- Status: Initial algorithms coded. Gradient descent algorithm to be completed by Sarah.

Camera Feedback

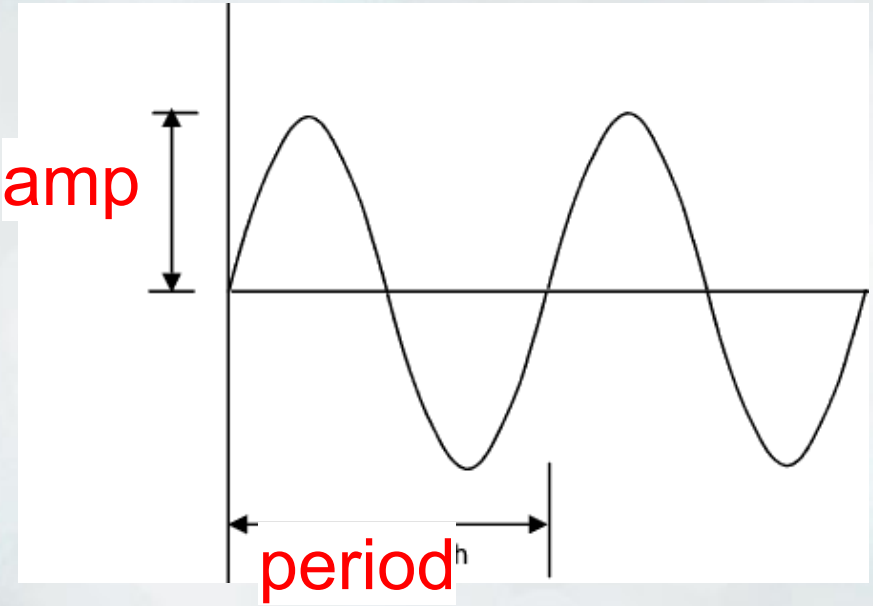
- Current Implementation: None
- Future Improvements: Get positional information from the camera system to the program
- Status: Cameras working by October 8, positional feedback incorporated into program by October 20. To be completed by Diana.

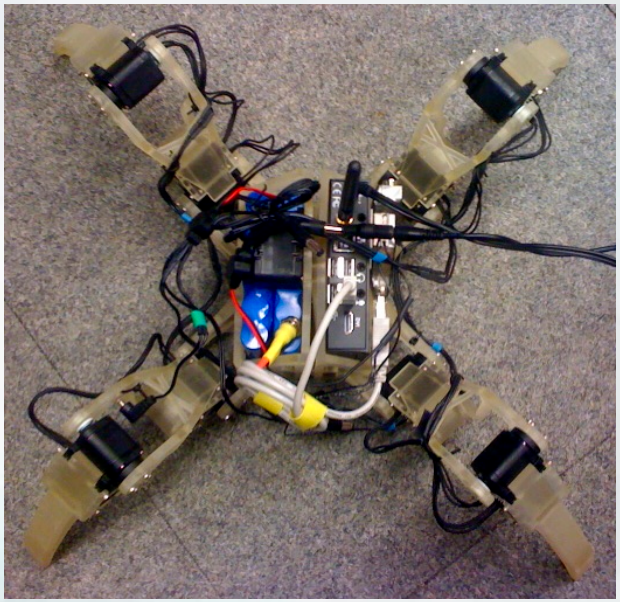
Parameterized Models

- Current Implementation:
  - SineModel class, several flavors:
    - 5 parameters (3 continuous, 2 binary)
    - 5 parameters (all continuous)
    - 7 parameters (all continuous)
- Future Improvements: Many.
- Status: We'll be working on this the whole term.

Parameterized Models

- Current Implementation:
  - SineModel class, several flavors:
    - 5 parameters (3 continuous, 2 binary)
    - 5 parameters (all continuous)
    - 7 parameters (all continuous)
- Future Improvements: Many.
- Status: We'll be working on this the whole term.



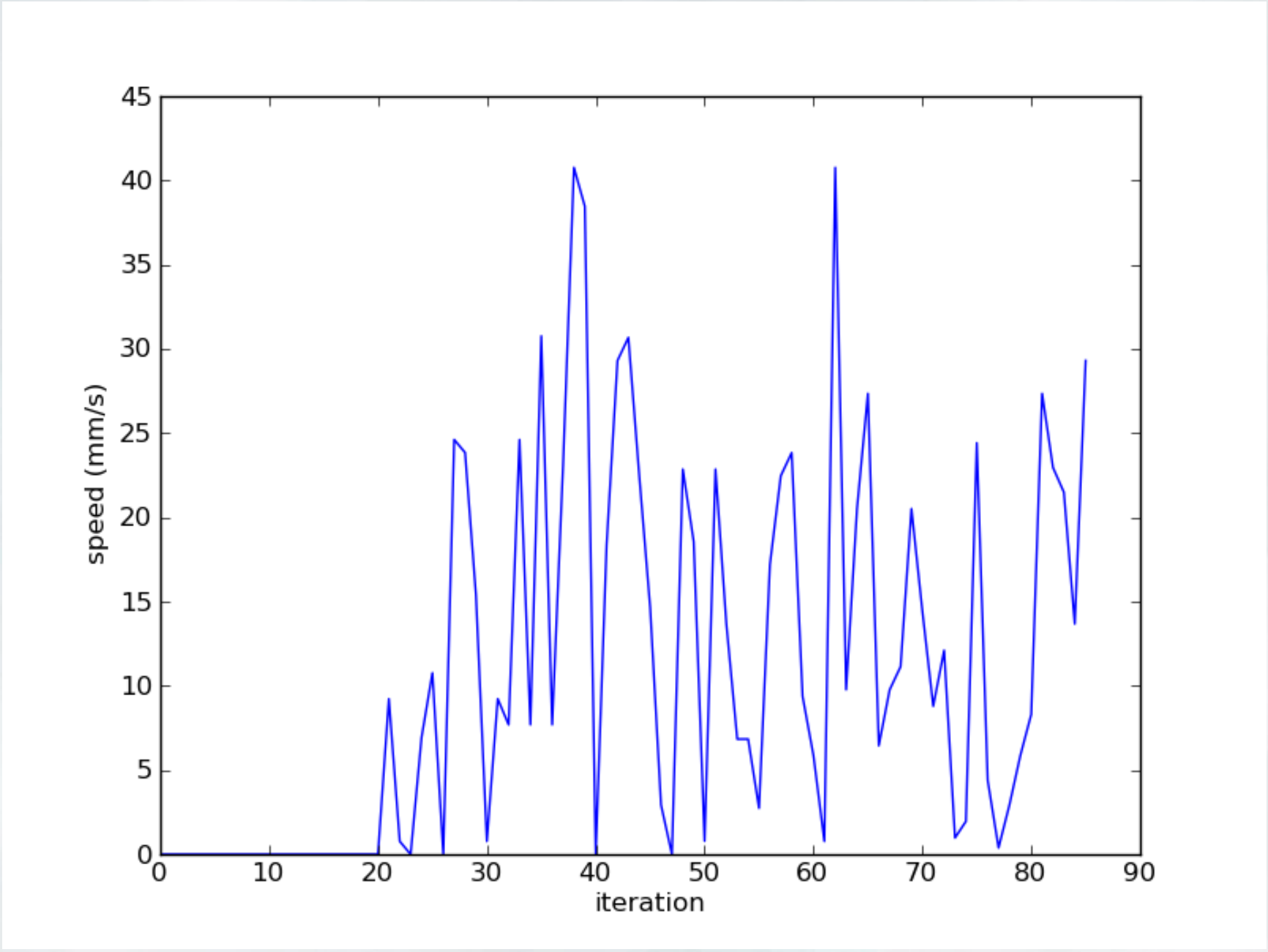


$m_{in/out}$

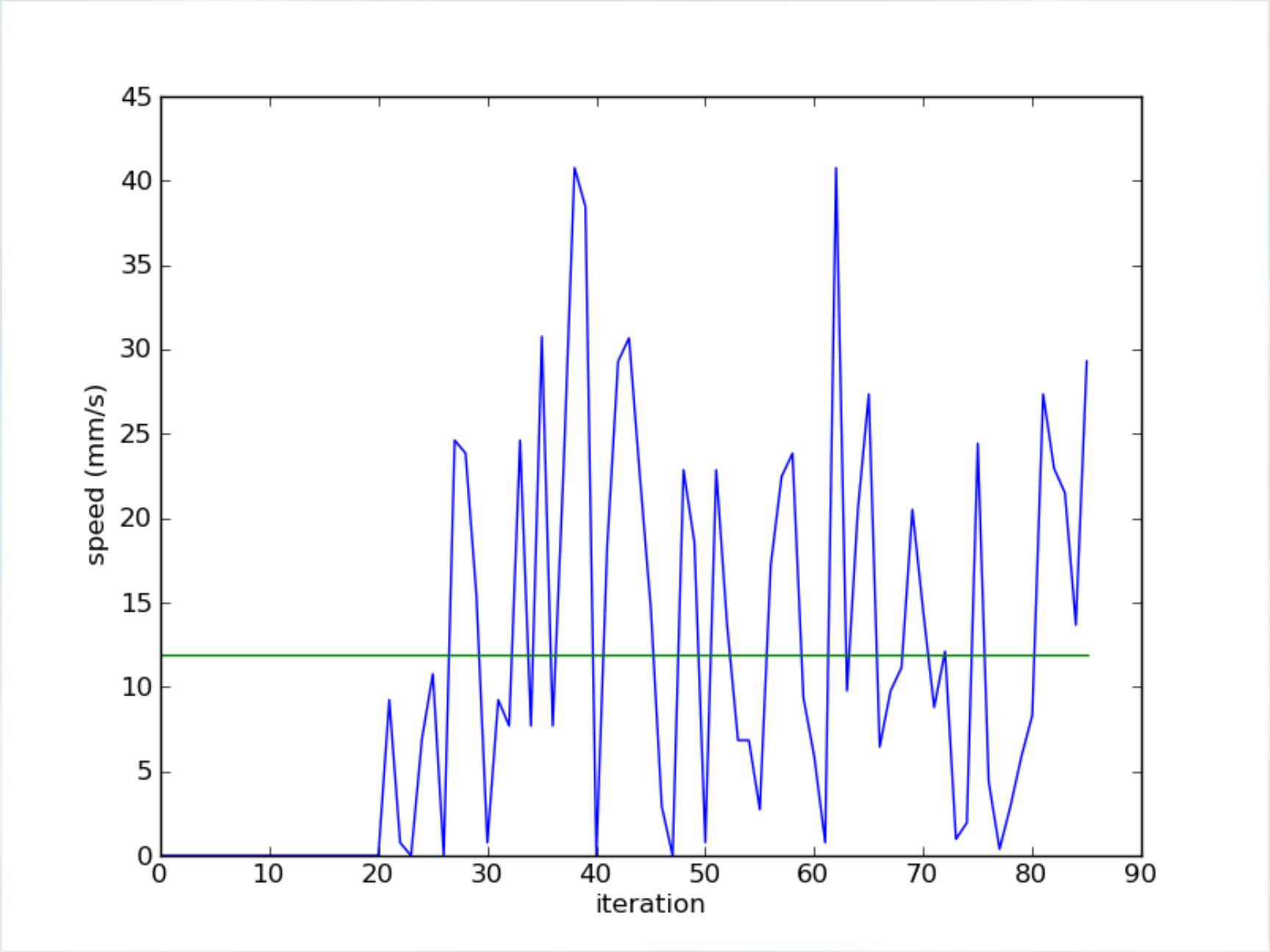
$m_{F/B}$

$m_{L/R}$

Results


45
40
35
30
25
20
15
10
5
0
speed (mm/s)
0
10
20
30
40
50
60
70
80
90
iteration

Results


45
40
35
30
25
20
15
10
5
0
speed (mm/s)
0
10
20
30
40
50
60
70
80
90
iteration

Done

General Approach

Where's the AI?

- Optimization (gradient descent)
- Evolutionary algorithm / Genetic algorithm
- Reinforcement learning

Domain specific hints

- Reduced dimensionality of parameter space (periodic, symmetric, ...)
- Parametrized gait
- Geometric constraints

Background Reading

- An Evolutionary Approach to Gait Learning for Four-Legged Robot
  by Sonia Chernova, Manuela Veloso
- Policy Gradient Reinforcement Learning for Fast Quadrupedal Locomotion
  by Nate Kohl, Peter Stone
- Evolving Dynamic Gaits on a Physical Robo
  by Viktor Zykov, Josh Bongard, Hod Lipson

Evaluation Plans

- AI will be evaluated on the following, compared to its initial hard-coded walking, and compared to other quadraped robots in the lab
  - Speed taken to walk a certain distance
  - Number of failures (falling down, getting stuck) over a certain number of attempts
- The evaluation for our particular project is an ongoing process, but we anticipate a final evaluation taking place early-to-mid November

I/O Specification

Input:

- Distance walked (input by user)
- Eventually hope to have camera feedback system

Output:

- Motor position commands over time

System Architecture

- Robot with on-board computer running Linux
- Lower level drivers are in C and we are implementing the system in Python
- Feedback about distance travelled is currently user-inputted, but will eventually be provided by Optitrack positioning system.