SigSent

Group #11

Critical Design Review—Friday, February 9th, 2018

Making Introductions

Who we are.

Joshua Lee Franco, John Millner, Jeff Strange Jr., & Richie Wales

Problem Spotted!

- Campus Security
  - Requires multiple guards 24/7
  - 3 Ds
  - Traditional Robots don’t cut it

What does success look like?

- Can traverse long distances regardless of terrain
- User-friendly interaction for operators
- Minimal operator intervention necessary
- Automated surveillance patrols

Requirements

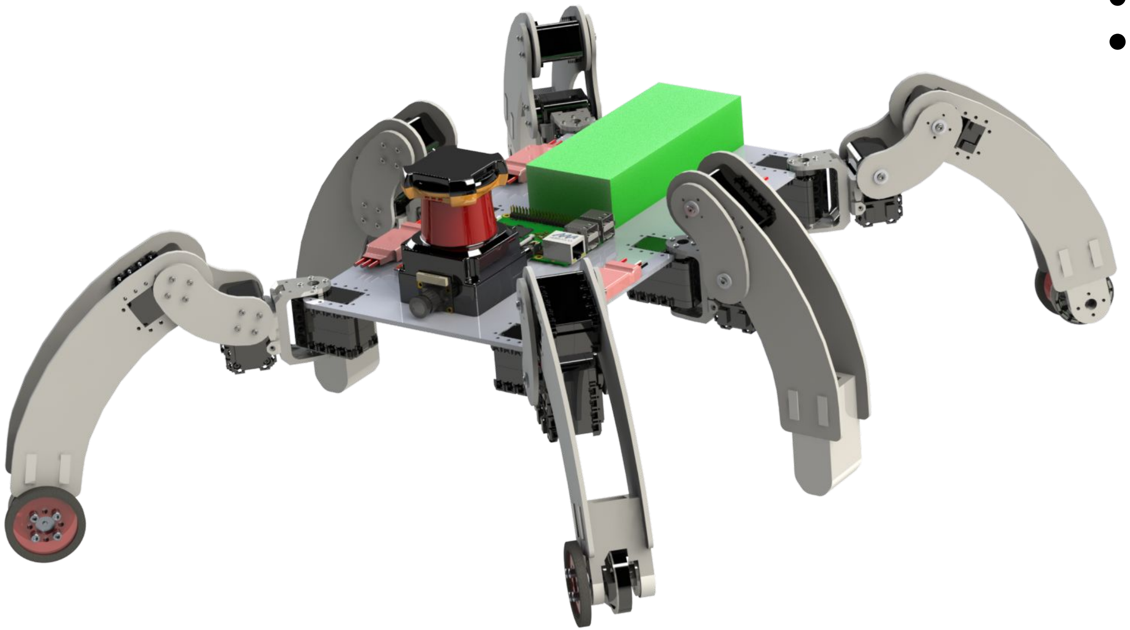
- Man-portable
- Keep pace with a running person
- Traverse rough terrain at walking speed
- Useful battery life
- Outdoor Localization
- Wireless Networking

Specifics

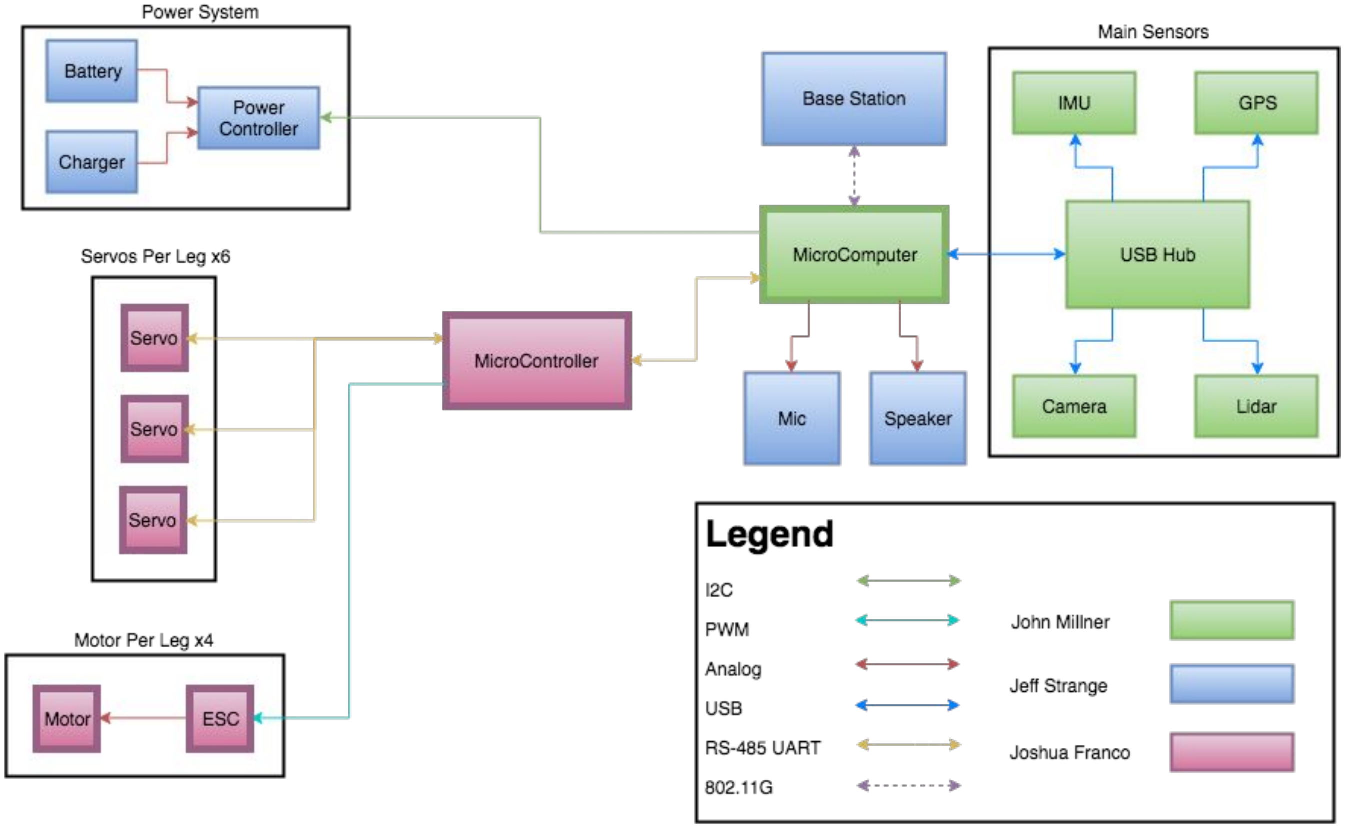
Requirement	Specification		Requirement	Specification
Weight	< 25 kg		GPS accuracy	5 m
Durability	Survives a 0.5 m fall		Comm. distance	32 m
Longevity	1 year life-cycle		Bandwidth	5 Mb/s
Availability	> 50%			
Top Speed	Smooth Terrain: 12 mph Rough Terrain: 1 mph			
Battery	1 hour in Typical Use			

The Design

What's a SigSent?



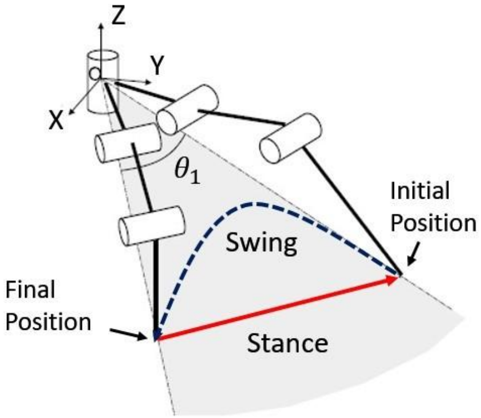
- Hexapod
- Multimode Locomotion
- Uses AI to determine appropriate mode



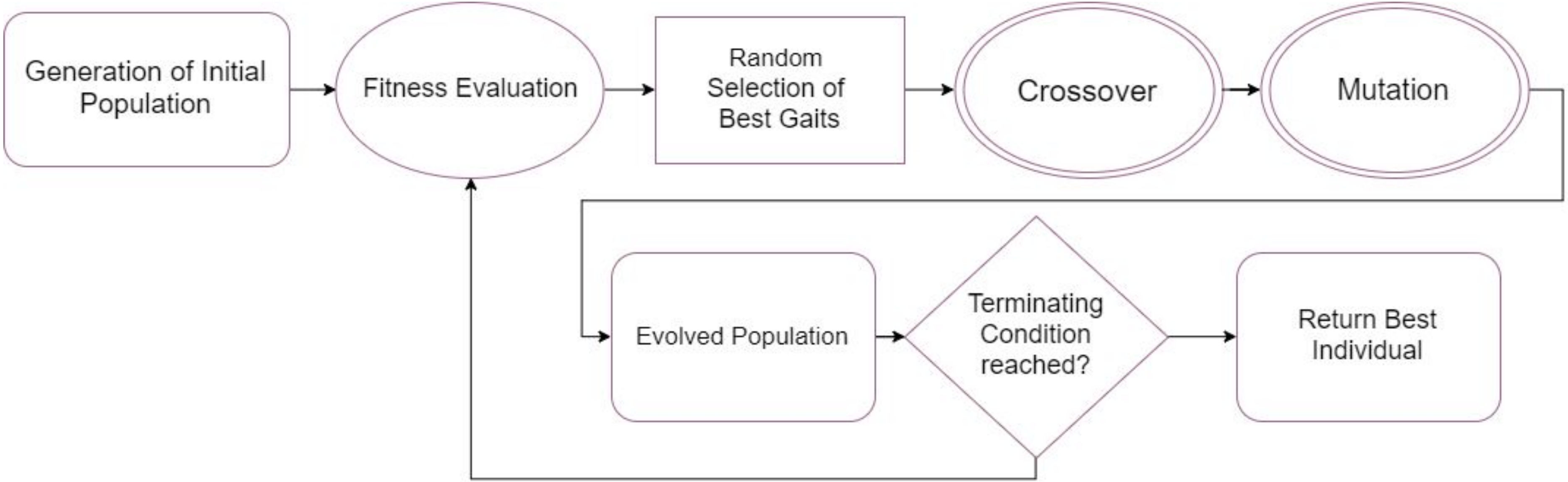
High-Level Block Diagram Overview

The Mechanics - Locomotion

- Each leg will have 3 servos providing 3 DOF.
- SigSent's gait will be pre-generated.
  - Derived from Inverse and Forward Kinematics using Denavit–Hartenberg parameters



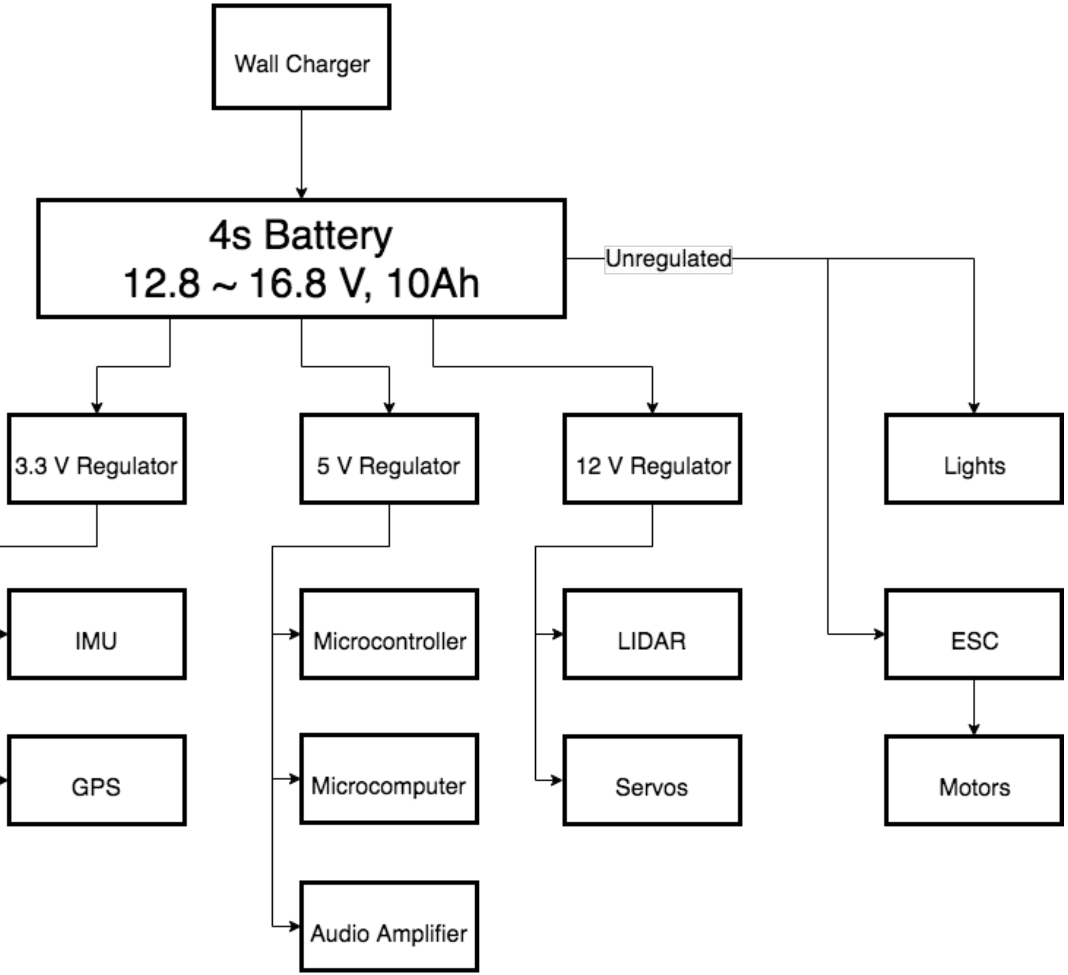
The Mechanics - Gait Generation



The Mechanics - Controls

- ATmega1280 microcontroller unit (MCU) will control all 18 servos through RS-485 serial communication
  - Each servo has an ID/address.
- MCU will receive general direction goals from the Raspberry PI and execute gaits to its own discretion to achieve said goals.
- Feedback Signals
  - From Servos to help maintain proper gait
  - IMU to help maintain direction given by the Raspberry PI

The Flow



Wall Charger
4s Battery
12.8 ~ 16.8 V, 10Ah
Unregulated
3.3 V Regulator
5 V Regulator
12 V Regulator
Lights
IMU
GPS
Microcontroller
Microcomputer
Audio Amplifier
LIDAR
Servos
ESC
Motors

The Juice

MultiStar 4S, 10 Ah LiPo

- 10 C Constant Discharge
- 804 g
- 160 x 65 x 60 mm
- 148 Wh

The Senses - Overview

Sensors of SigSent

- Camera
- Lidar
- Inertial Measurement Unit
- GPS

The Senses - Camera

Requirements

- Night vision
- Day/Night IR Filter
- Fast FPS
- Good Documentation

Selection

WaveShare RPi IR-CUT Camera

Specifications

- Uses OV5647 sensor
- Up to 120FPS at 640p
- Directly communicates to Raspberry Pi

The Senses - Lidar

Requirements

- Day/Night
- >180° horizontal FOV
- Long & accurate range
- Fast update rate
- Good Documentation

Selection

Hokuyo UTM-30LX Lidar

Sponsored by the Robotics Club at UCF

Specifications

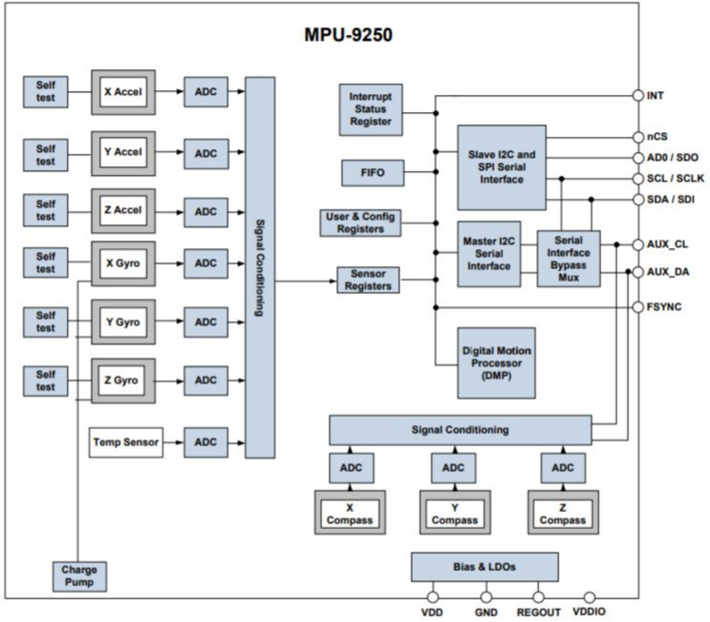
- 905nm laser for Day/Night
- 270° view
- 30m range with <30mm inaccuracy
- 2400rpm
- Easy Communication through USB

The Senses - IMU

Requirements

- High Refresh rate
- Low Power Consumption
- Low Accelerometer and Gyroscope Noise (Stability Scale Factor)

Selection



Invensense MPU-9250

Specifications

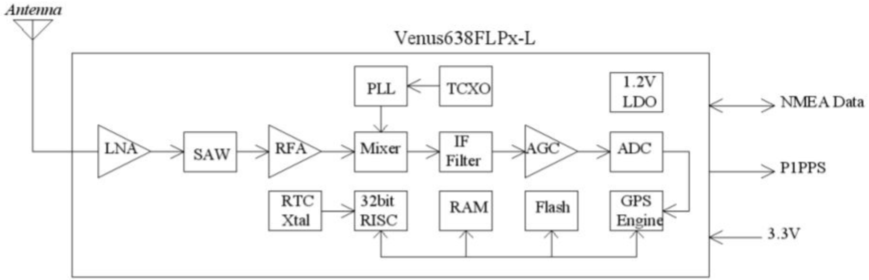
- Sample rate of 200hz
- .061 deg/s Accel. SSF
- 16.4 deg/s Gyro SSF
- Communicates via I2C
- 9 DOF

The Senses - GPS

Requirements

- High Refresh rate
- Low Power Consumption
- High Sensitivity

Selection



SkyTrac Venus638FLPx-L

Specifications

- 20hz refresh rate!
- 60mW power consumption
- -165db sensitivity
- Communicates through TTL

The Brains - Microcomputer

Requirements

- Low Price
- Multicore
- 1GB RAM
- USB Inputs
- Able to smoothly run Linux & ROS

Selection

Raspberry Pi Model 3

Specifications

- $35
- 4 Core processor
- 1GB RAM
- 4 USB Ports
- Binaries and official support already compiled for both Linux & ROS

The Brains - Robot Operating System

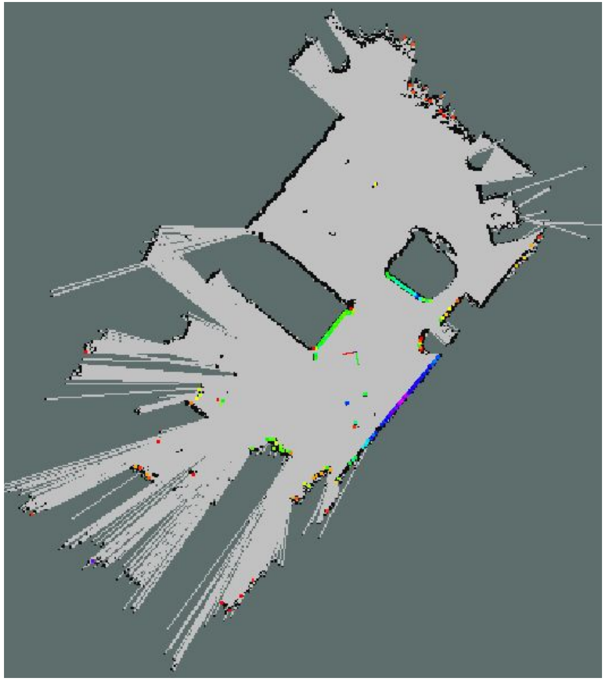
ROS is a system on top of Ubuntu Linux that:

- Handles communications between programs
- Has a large repository of robotic tools
- Provides a framework for robots to be built.

This allows us to focus on making a robot, and not homebrew already designed sub-systems

The Brains - ROS - gmapping

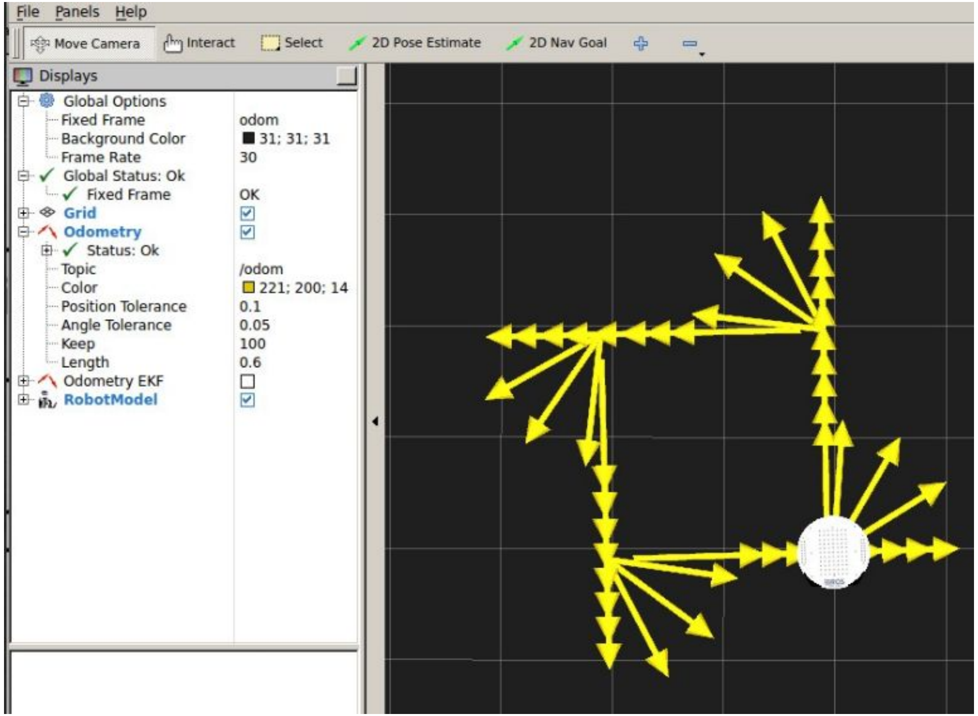
- ROS Node that uses the Simultaneous Localization and Mapping (SLAM) algorithm to do just as the name says
- This will help SigSent move smoothly and efficiently around obstacles by knowing and documenting its surroundings



The Brains - ROS - move_base

Move_base is a goal planner that allows the robot to take in a goal position and navigate towards it

Move_base allows SigSent to safely path around objects and people towards a given goal, be that via tele-op, a determined path, or by investigating/following a suspect

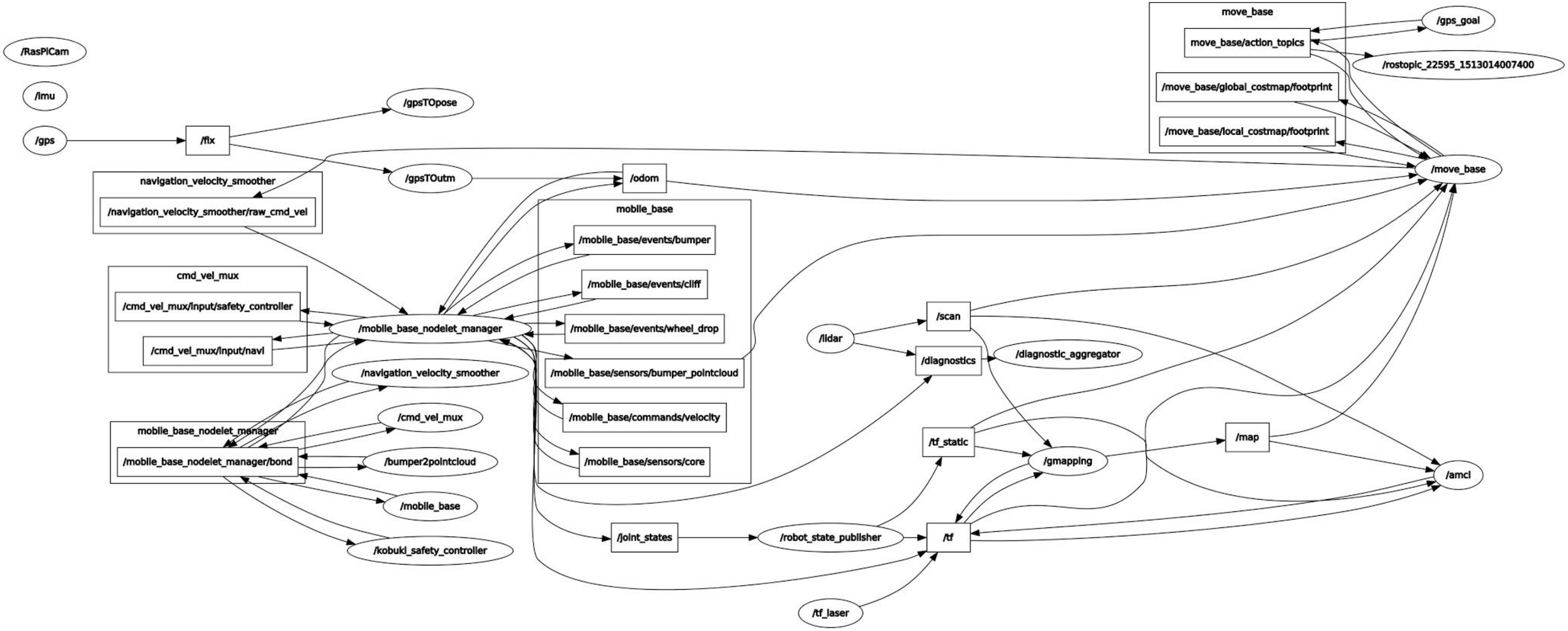


The Brains - ROS - gps_goal

- Simple node that takes in a GPS coordinate and orientation
- Transforms the GPS coordinate and orientation to a local goal
- Sends to move_base

Gps_goal allows SigSent to follow a predetermined path that can be set up by the user to patrol around.

The Brains - What it actually looks like!



The Smarts

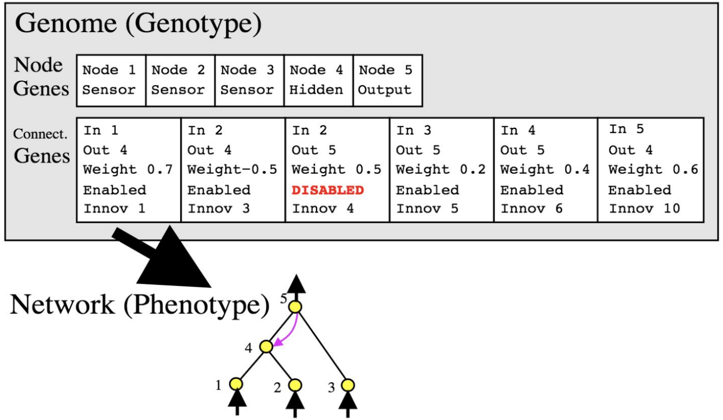
- Biologically inspired learning process
  - **N**euro**E**volution of **A**ugmenting **T**opologies
  - Dr. Kenneth Stanley
- Computer Vision analysis on surrounding terrain obstacles
- LIDAR
- Analyze IMU data for uniformity
- Servo positions describe robot’s current pose

The Smarts - NEAT

- Reinforcement learning
- Evolve a population of Artificial Neural Networks (ANN) over some generations
- Assign fitness values to each individual based on performance
- Reproduce between individuals to make a new population
    - More likely for high performers
    - Successful traits propagate
- Redo process with new population of individuals

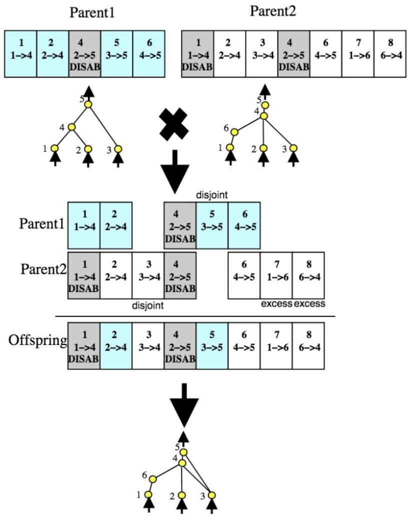
The Smarts - NEAT

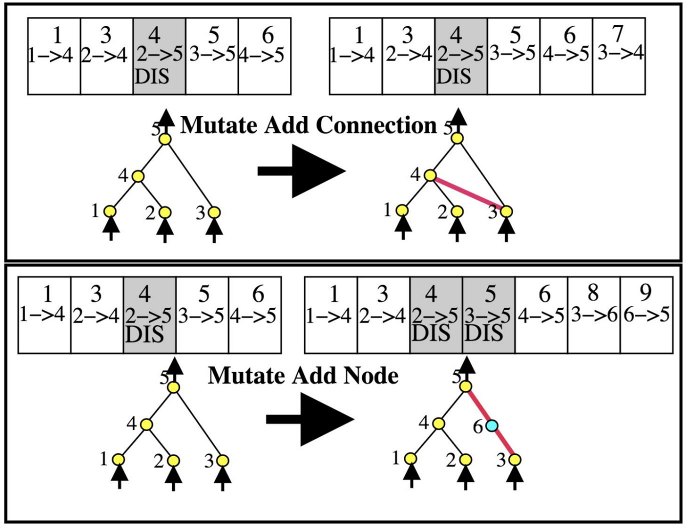
- Genome is the low level representation of the ANN
- Genes represent nodes and their connections between another
- Connections can be enabled or disabled



The Smarts - NEAT

- Selected individuals have their genes modified
- Innovation number
  - Compare two topologically dissimilar networks easily





The Smarts - Inputs

- Terrain Classification
  - OpenCV simple obstacle detection
  - LIDAR detects distances between obstacles
  - Robot pose taken from servo positions
  - Assign a value between 0-10 based on these values
- IMU measurements detect rough/smooth terrains

The Smarts - Outputs

- Inputs propagate forward in the network
- Boolean decision to engage or disengage crutches

The Smarts - Fitness

- Distance traveled
- Smoothness of traversal
- Reward will be altered throughout testing process

The Smarts - ROS Integration

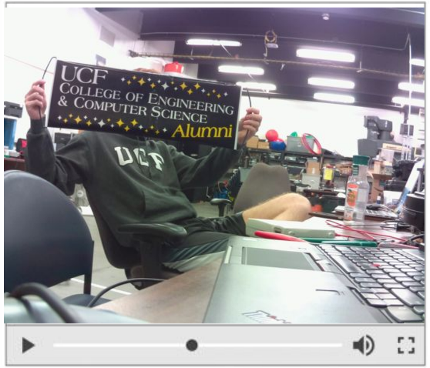
- NEAT package is a ROS node
- Most fit ANN used
  - Subscribes to sensor topics for ANN inputs
  - Publishes to a “mobility_type” topic
- During training
  - Sensor topics feed into tested ANN individual
  - Also subscribes to sensor topics for fitness evaluation

The Audio

The Interaction

The View

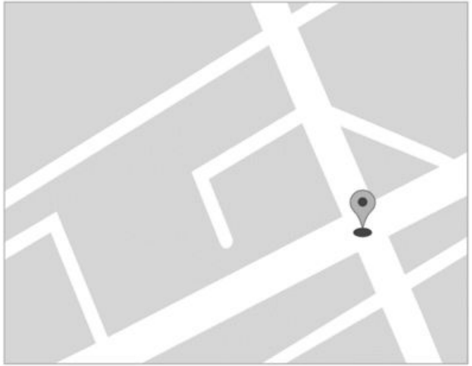
Video Feed



Sentry Mode

TeleOp Mode

GPS Location



Remaining Battery

NEAT Classification

▾ Terrain Classification	▾ IMU Orientation	▾ NEAT Output
8 (Rough)	<2.75 m/s^2, 0.70 m/s^2, 0.12 m/s^2>	Wheeled-Mode

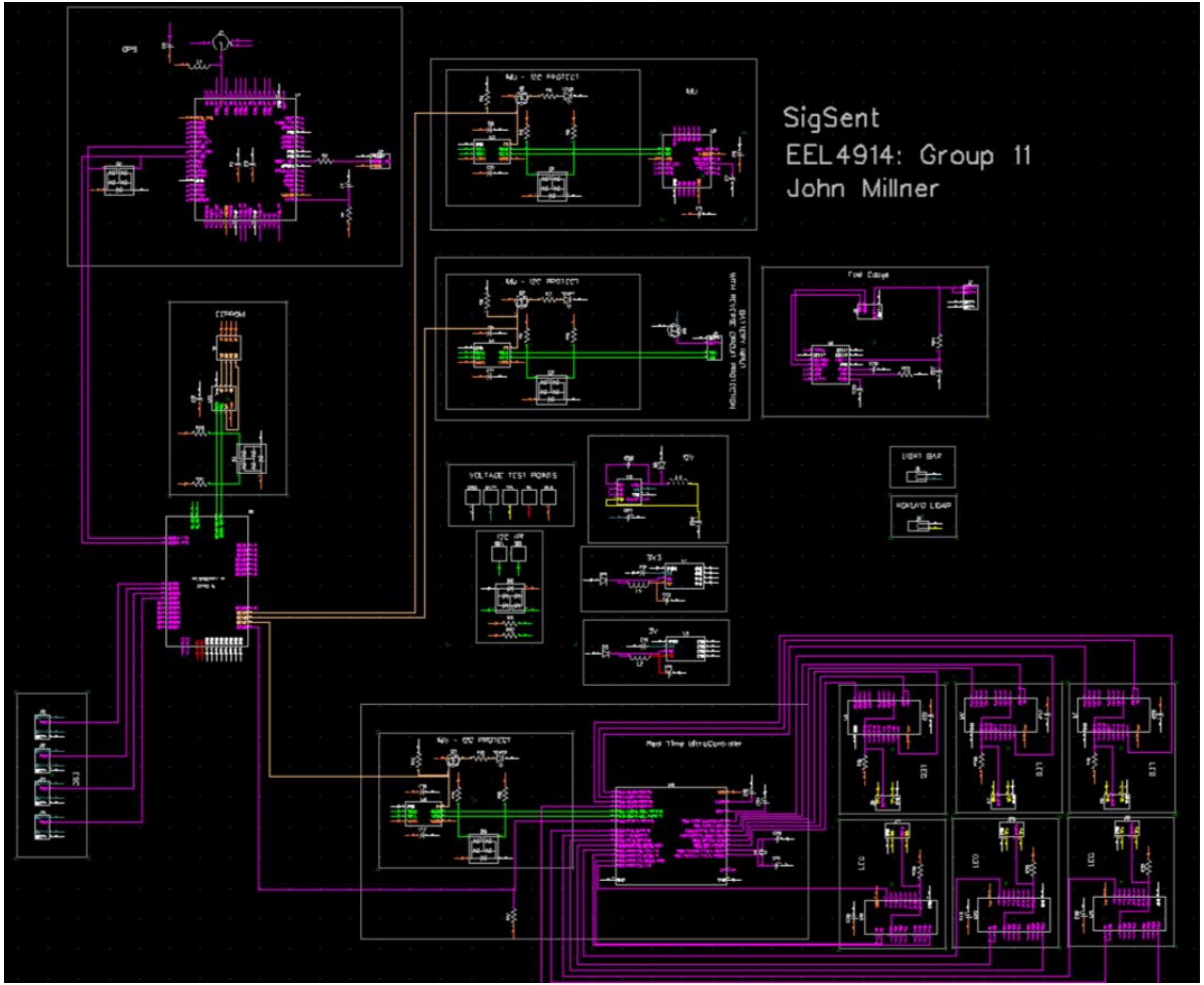
Where we’re at.

The Prototype

ClearPath Robotics: Turtlebot 2 as testbed

- Allows team to implement and test high level actions such as:
  - Path planning
  - interacting with base station
  - patrolling a GPS-coordinate path
  - Reporting suspects
- Contains the same exact sensors that will be used on SigSent
- Known platform with a significant amount of documentation and demos

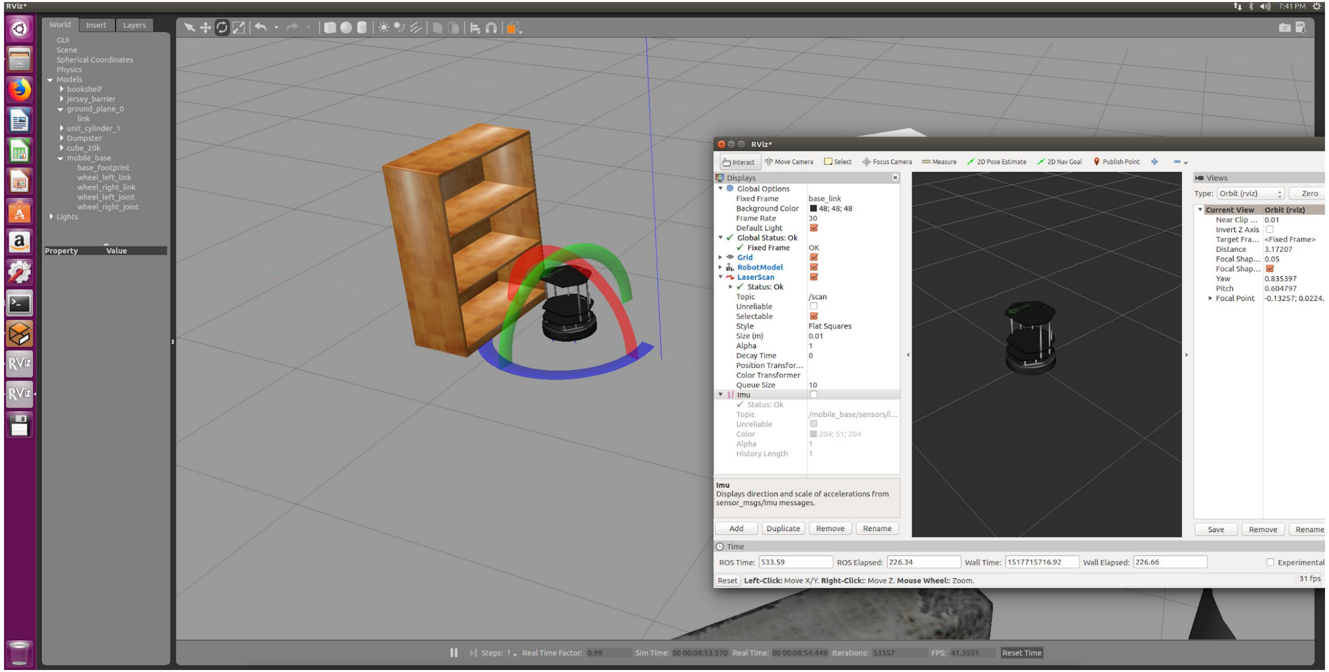
The Schematic



The Simulation

- Using Gazebo for simulation
- Integrated into ROS
- Abundant documentation and code online
- Simulation decouples code from hardware completion contingency
- RViz used to show robot and sensor topics in simulation
- Run best ANN from simulation on physical bot
  - Train again to minimize errors between sim and physical
  - Hopefully reduces search time for best results

The Simulation



The Budget

Part	Unit Price ($)	Total Price ($)	Project Price ($)		Part	Unit Price ($)	Total Price ($)	Project Price ($)
Camera	25.99	25.99	25.99		Motor	24.93	99.72	99.72
IMU	44.46	44.46	10.63		ESC	13.51	80.03	80.03
GPS	49.95	49.95	39.95		Servo (6pcs Bulk)	224.5	673.5	673.5
Raspberry Pi	34.49	34.49	34.49		Servo mounts			
SD	24.6	24.6	24.6		Joystick			34.99
Battery	40.87	40.87	40.87		Headset			10
LIDAR			4800		Laptop			200

Total Development Price: 1073.61
Total Production Price: 6074.77*

Wins & Losses

- Eliminated pressure sensors from design
- No more solar panels
- Extended Kalman filter difficult to implement
- Simulation getting off the ground
- Base station networking and teleoperation successful
- Basic ROS nodes setup

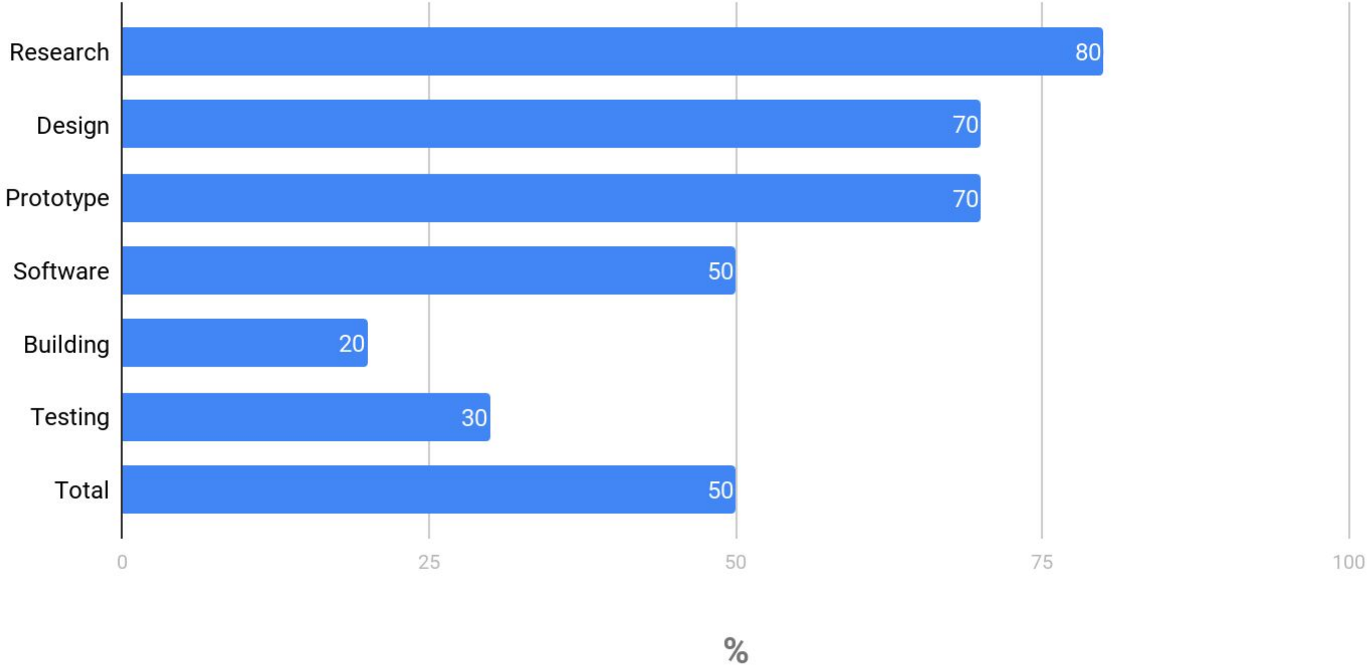
What's Next?

- PCB Fabrication
- Building Mechanical Prototype
- Hexapod Assembly
- Gait Optimization
- Complete ROS stack and NEAT AI nodes
- Simulate operation over various terrains
- Test, test, & test

The Schedule

TASK TITLE	OWNER	2/12 - 2/18	2/19 - 2/25	2/26 - 3/4	3/5 - 3/11	3/12 - 3/18	3/19 - 3/25	3/26 - 4/1	4/2 - 4/8	4/9 - 4/15
User Interface	Jeff									
Simulation Setup	Richie									
Sensor Fusion	John									
Navigation	John									
FSM	John									
Microcontroller Integration	Josh									
Electrical Design	Jeff									
Electrical Fabrication	OSH Park									
Electronics Integration	Team									
Leg Fabrication	Team									
Prototype Assembly	Team									
Gait Optimization	Josh									
NEAT Training	Richie									
Testing	Team									

Progress Estimation



Questions?