ET NAVSWARM

Extra Terrestrial Navigation with Particle Swarm Optimization

Team Members: Joshua Proulx, Nathan Beaulieu, Haley Breedlove, Colin Pellerin, Samuel Wilkinson, Gregory Hatfield, Brody McClure, Brian Teune, Nolan Dinero, Aaron Rist, Garrett Malagodi, Iskandar Fendi, Ben Yandell, Zachary Plumer

Advisors: Professor May-Win Thein, Professor Se Young Yoon, Michael Johnson

www.unh.edu/navswarm

Abstract

ET-NavSwarm's mission is to design, build, and test a swarm of autonomous robots. These will serve as a testing platform for a graduate research algorithm: Particle Swarm Optimization (PSO). PSO is derived from the concept of swarm behavior found in nature (birds, bees, etc.). By communicating and using mathematical principles, the swarm can find the best/most of a given objective within a specified area. The swarm operates without a centralized leader to search for natural resources, signs of life, or items of interest.

The robots are designed to traverse rugged terrain, climb 45° inclines, avoid obstacles, and collect data on position and elevation. To accomplish this they were made to be water-resistant and have the longest possible run time (battery life). The robots' sensors and electronics were selected to fulfill the requirements of each algorithm. As a proof of concept for NASA, ET-NavSwarm will perform a large scale field test with these robots, searching for the highest and lowest elevations.

Design Constraints

- Drive up a 45 degree incline
- Traverse rugged terrain
- Water resistant
- 1 hour of run time
- Cost
- Sensors to provide feedback

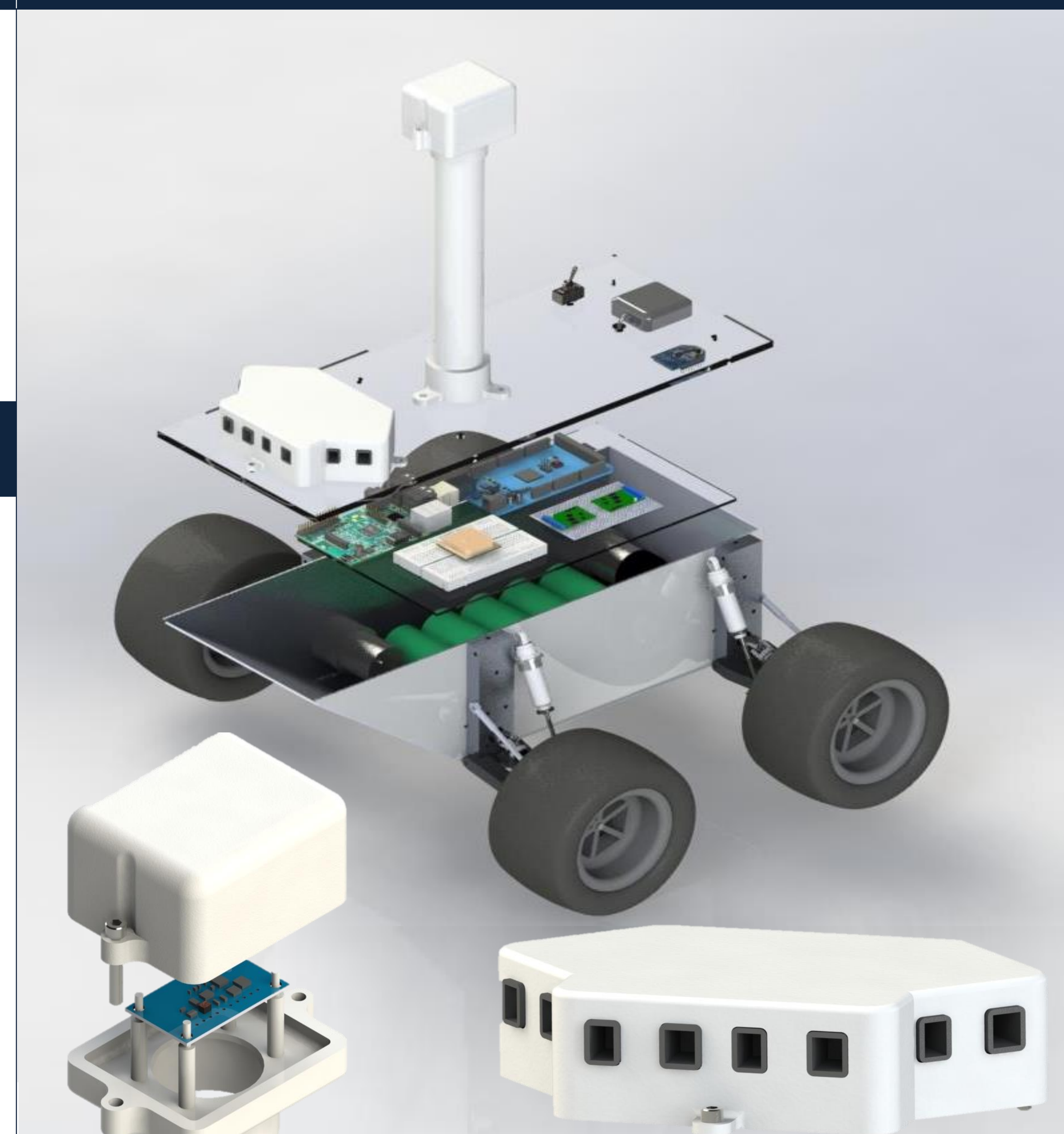
Sensors

Inertial Measurement Unit (IMU)
- 10 Degrees of Freedom
- Determines roll, pitch, and yaw
- Determines altitude via barometric pressure

Global Positioning System (GPS)
- Determines location

Infrared (IR) Sensors
- Detects obstacles up to 5 feet

Design



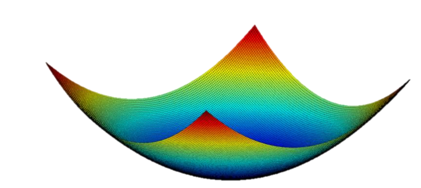
IMU Holder

IR Sensor Housing

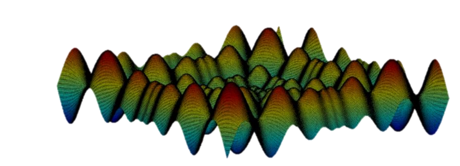
Particle Swarm Optimization

Description:
- Created by Graduate Student Michael Johnson
- Proof of concept for NASA to aid in Space Exploration
- Algorithm created from studying swarms of animals
- Decentralized with no leader
- Each member acts independently towards a goal

How it will be Tested:
- Will be tested on Earth using the two functions below:



Sphere Function

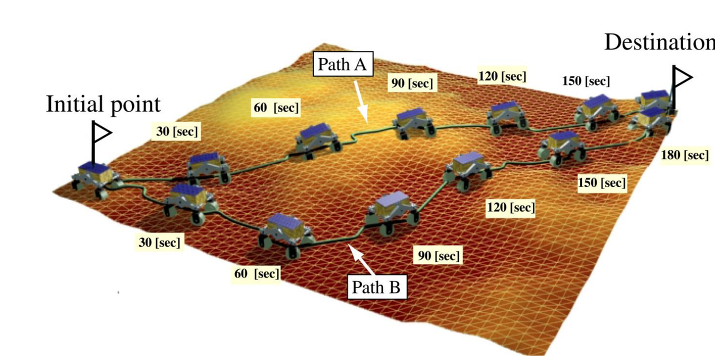


Schwefel Function

- The Sphere Function will serve as the first test platform due to its simplicity
- The Schwefel Function will be later used to test the complexity

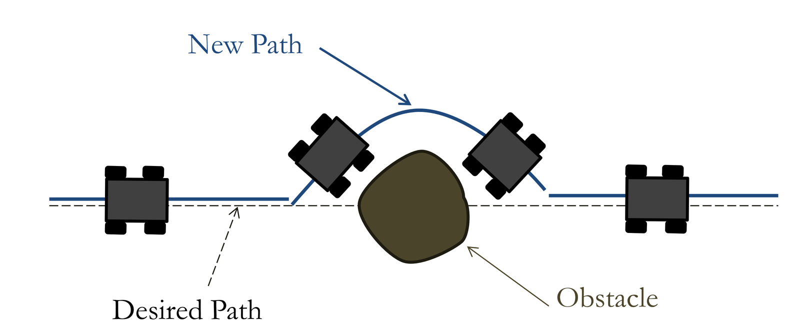
Path Planning

Description:
- Determines the best way around a stationary obstacle
- Obstacles are known prior to the start of the mission

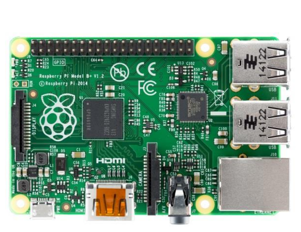


Obstacle Avoidance

Description:
- Avoids stationary obstacles that are in the desired path
- Obstacles are not known prior to the start of the mission



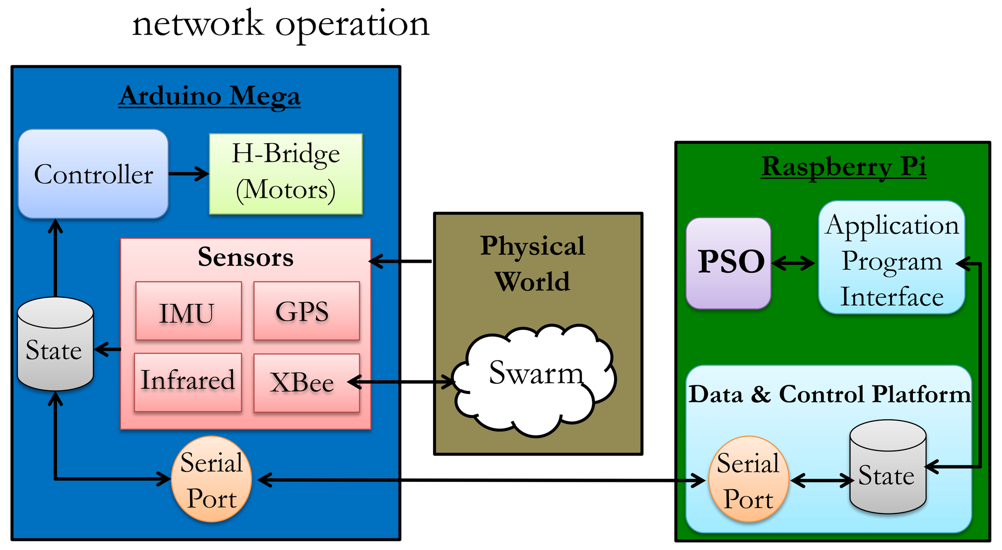
Software and Autonomy



Raspberry Pi
- Runs the PSO (programmed in Java)
- Uses sensor data and the PSO algorithm to select search points and plans a path to the destination

Arduino Mega
- Runs an LQR control algorithm (programmed in C/C++)
- Analog and digital input/output pins read sensor data
- Sends sensor data to the Raspberry Pi

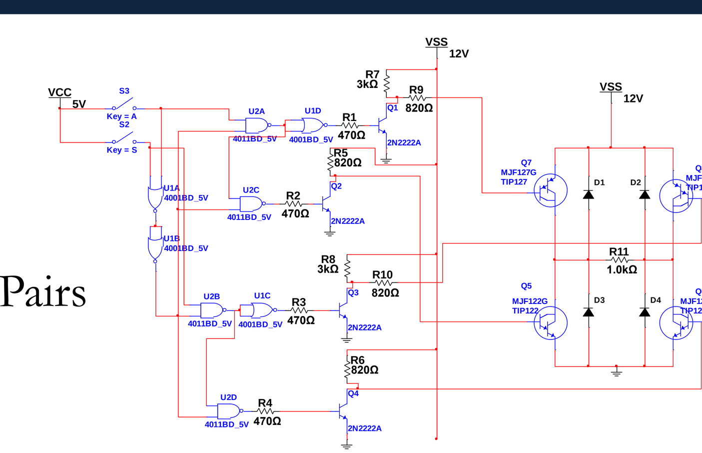
Xbee
- Wirelessly transmits and receives data allowing communication between bots
- Sends packets of data through a mesh network
- Adds network stability through self healing, self discovery, and dense network operation



Circuit Analysis

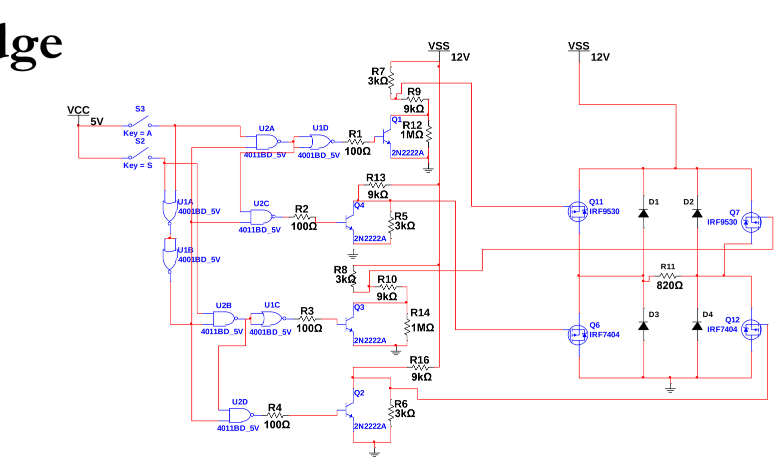
Original Design: BJT H-Bridge

Original BJT H-Bridge Design Problems
- Low Current Capacity (2A)
- Voltage Drop Across Darlington Transistor Pairs
- Slower Switching Times



Updated Design: MOSFET H-Bridge

MOSFET H-Bridge Design Problems
- Soldered on Perf-board
- Time consuming to produce
- Prone to wiring/soldering errors
- Difficult to fix errors

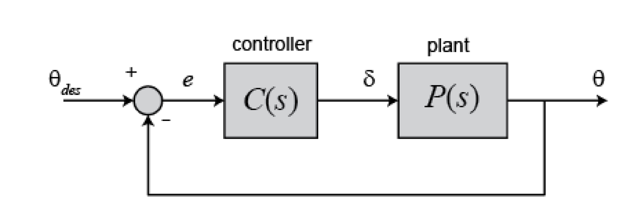


Final Design: Pololu 18v17

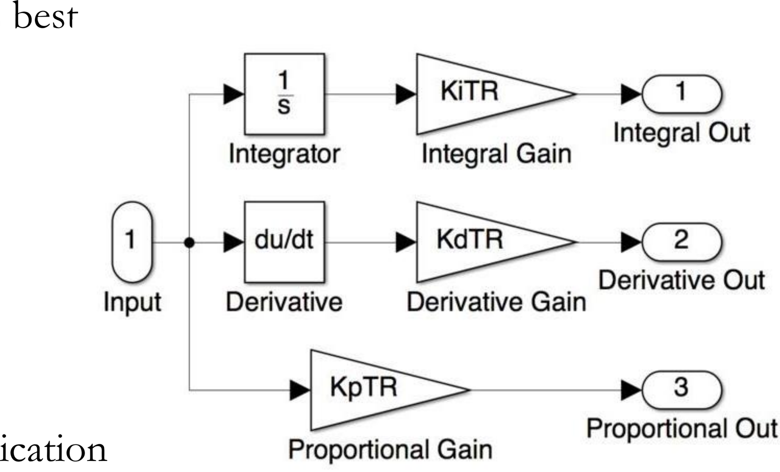
Pololu 18v17 High Power Motor Driver
- Handles 17 Amps continuous at 12 volts
- Very Small (1"x1")
- Minimal connections
- Consistent operation

Controls Analysis

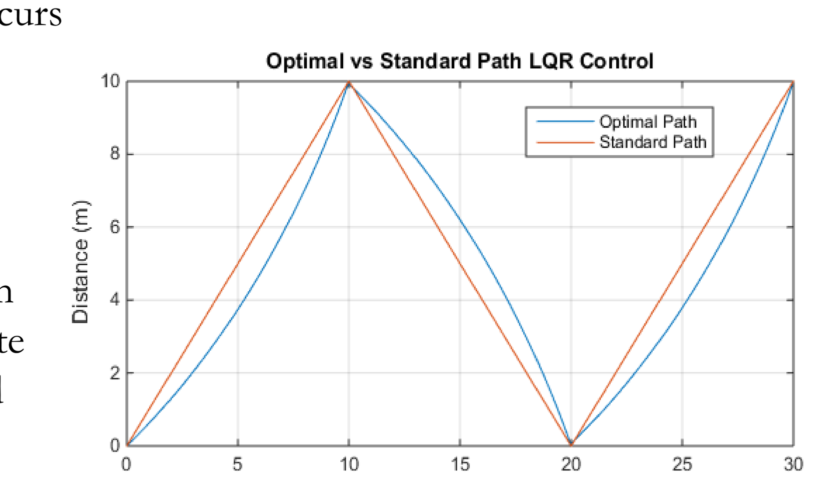
Description:
- Method to determine inputs to the motors
- Inputs are based on relative error
- Error is based on position relative to goal set by particle swarm optimization
- Diverse controller options
- PID and LQR modeled, LQR selected as best
- Based on closed loop control system



Controller Characteristics:
- PID (Proportional Integral Derivative)
  - Simple
  - Highly susceptible to noise
  - Loses effectivity at longer distances
- LQR (Linear Quadratic Regulator)
  - Complicated preparation, simpler application
  - Inputs calculated before operation occurs
  - Effective for all ranges
  - Less controller effort is required



Resulting System:
- Travels towards goal based on the calculated inputs from the control system
- Robots not already facing their goal rotate until it is more efficient to move forward
- End result is a curved path to location
- New path is faster with higher efficiency



2015-2016