Distributed Swarming Robotic System

Armando Hernandez-Christian, Capitol College, Team Lead
Lars Osborne, Montana State University
Katherine Blackburn, Louisiana State University
Robert Rucker, Vanderbilt University
Principle Investigator: Robert E. Ray, NASA Marshall Space Flight Center (Jacobs Engineering)

Background:

The project was created in response to the potential consequences of Kessler Syndrome, a concept described by Donald Kessler in 1978. Kessler syndrome, also known as collision or ablation cascade, describes the increasing cascading effect of multiple collisions of space debris with satellites and space artifacts found in low earth orbit. In such a situation, each of the thousands of pieces of debris created in an orbital collision could each destroy a spacecraft, creating more debris. As this process continues it will accelerate, possibly making future spaceflight too dangerous.

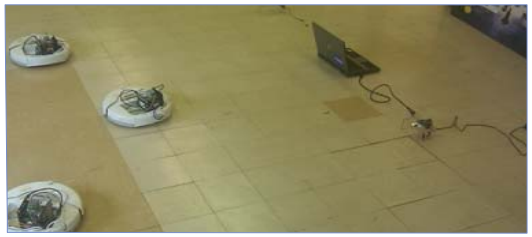
Distributed swarming robotics is a new approach to the coordination of multi-robot systems based on the concept of swarm intelligence. The multi-robot system consists of simple, homogenous robots which follow a set of simple rules to coordinate their movement. This approach emerged from observations of the behavior of social insects, such as bees or ants.

Left: view of the Freescale Tower

Right: the IR towers used in location

Abstract:

The robots built by this team will provide a framework for combating Kessler syndrome. A psuedo-swarm of small satellites with this framework will intercept and deorbit space debris in order to clean up Low Earth Orbit. The robotic system uses several modified iRobot Creates capable of moving in coordinated patterns. The Creates have been modified with a microcontroller, voltage regulator, and a wireless adapter to communicate with a central computer via a wireless TCP connection. The location of the robots will be determined using information gathered from two rotating infrared emitters at fixed locations. Using C++ with Qt, a graphical user interface (GUI) was developed to control the robots and the infrared emitter. In addition to removing and extracting space debris, this project will provide a framework for future exploration and research of practical applications including, but not limited to, military or geographical reconnaissance, transportation of supplies, formation flying or movement, or multiple satellite coordination.



Methodology:

Software used to run the IR tower's servo motors, wireless routers, robots, and TWR microchips:

- Code Warrior for communication with TWR microchip
- Qt Creator with C++ for the control program

•Hardware used:

iRobot Create Robot

- TWR Microcontroller
- 5V voltage regulator
- Serial connectors
- Netgear wireless router
- ASUS wireless adapter
- Lexan plate and standoffs

Lighthouse (IR Tower)

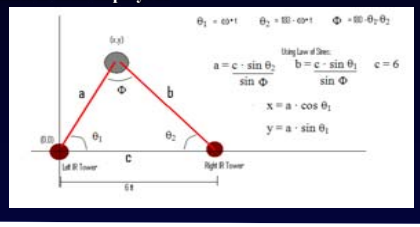
- Hitec Servo Motor
- Pololu Servo Control
- Batteries and electrical wires
- Lexan plating
- iRobot Virtual Wall assembly
- Mini USB to USB connector

iRobot Create Dimensions:

· Height: 5.5"
· Diameter: 14"
· Weight: 9 lbs

• Positioning and Algorithms:

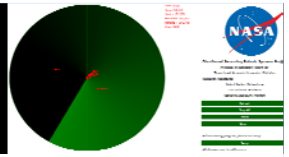
Coordinate origin is based from location of left IR tower. A timer is started and both IR beams sweep from right to left. When the beams hit a robot, a signal is sent to the main computer and a time stamp is given to the signal. The time stamp is then changed into an angle, and then, using trigonometry, the angle between both IR beam is changed into relative distances to each tower. The robot's X-Y coordinate is then displayed.

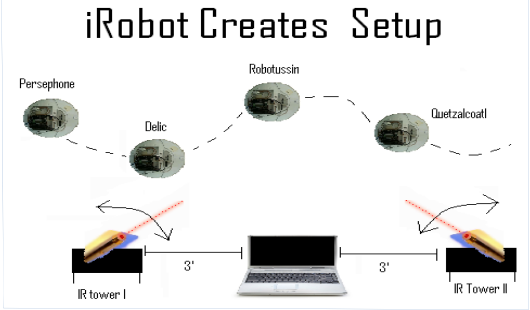


Left: view of the project setup

Right: GUI used

Below: flowchart of the project setup





Conclusions:

Our psuedo-swarming robotic system moved together in coordinated patterns. The computer and IR towers were able to locate the position of each of the robots. The robots moved in several coordinated preprogrammed patterns including a line, in a circle, and a wave. Our methods and algorithms are open to future development by Dr. Ray, his colleagues, and future Robotics Academy students.

Future Applications:

· Coordinated space debris clean up
· CubeSat, Satellite, or telescope coordinated movement
· IED diffusion
· Underwater exploration or oil spill control
· UAV formation flying or reconnaissance

References:

•Blanchette, Jasmin, Mark Summerfield. C++ GUI Programming with Qt4, 2/e (New Edition). 2nd Ed. Westford, MMA: Pearson Education India, 2008. Print.
• Donald J. Kessler and Burton G. Cour-Palais (1978). "Collision Frequency of Artificial Satellites: The Creation of a Debris Belt". Journal of Geophysical Research 83: 63.
•Lippman, Stanley, and Josée Lajoie. C/C++ users journal. 3rd ed. Reading, MA: Addison-Wesley Professionalield, Sreekrishnan Venkateswaran, and Blanchette, 2001. Print.
•Background image: http://www.eclipsewallpapers.com/wallpaper/Space-Eclipse-