Group 10

Amit Maole
Brent Eskridge
Klo Utley
Tony Lopez

PROJECT 2: SOFTWARE SYNOPSIS

For this project, we chose to implement an architecture (Fig. 1) similar to the "Sensor Fusion Effects" (SFX) architecture, as described in our textbook "Introduction to AI Robotics" by Murphy. The behaviors were broken down into two categories: strategic (Fig. 2) and tactical (Fig. 3). The strategic behaviors determine the long term goals for the robot, such as traveling to a light or searching for a light. The tactical behaviors take the resulting output of the strategic behaviors as a guide and determine the current direction the robot should move in. This direction is determined by the tactical behaviors processing the inputs provided by the range finders and bump sensors and then generating commands that allow the robot to reach the strategic goal while avoiding obstacles in its path.

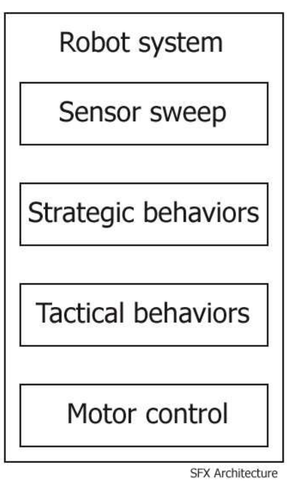
In our initial design, the robot was to sense its environment through the use of a rotating sensor turret. The turret would rotate through 180 degrees and 7 pre-defined regions (each covering roughly a 30 degree arc) (Fig. 4). In each region it would take measurements using each of its three sensors: a light sensor and two range finders. The light sensor was used to determine if a lit light bulb was in the region. The first range finder was used to detect any low obstacles present in the region and the second was used to detect any high obstacles present in the region. All three readings were put into a global structure containing all the sensor data last sensed for use by the various behaviors. These behaviors basically used potential fields and would perform the various calculations and output a force vector. The tropotaxis behavior was to simply determine which region contains the strongest light above a certain threshold and output a force vector for that region. The obstacle avoidance behaviors, both low and high, were to create force vectors pointing away from each region with an obstacle, sum the vectors, and determine the resulting tactical output. When the range finders proved unable to provide accurate data, both the robot's hardware and software design needed to be refactored.

In the final design, only the light sensor was mounted to the sensor turret. The range finders were statically mounted on the edges of the robot to detect high obstacles and bump sensors were added to detect collisions with low obstacles. It was also determined that while the overall software architecture was adequate for the task, but the implementation was too complex. The use of potential fields, as such, was abandoned and replaced with a few simple directions, one for each sensor region plus one for reverse. The tropotaxis behavior performed the same calculations, but only had the desired region of travel as the output. The high obstacle avoidance behavior was redesigned to

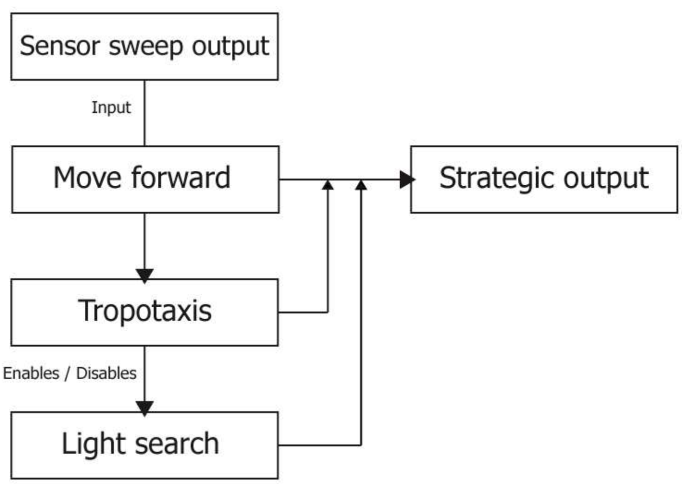
Group 10 Amit Maole
Brent Eskridge
Klo Utley
Tony Lopez

such that it turned away from obstacles. It remembered the last direction it turned to avoid an obstacle in order to prevent cyclic patterns. In testing however, this problem still appeared. The low obstacle avoidance behavior was entirely rewritten to use the inputs of the bump sensors. If the robot collides with an obstacle, it backs up and turns to avoid it. To help avoid cyclic patterns, the amount of time the robot spends turning is variable and depends on the position of the servo in the sensor turret (a pseudo-random value).

The Handy Board has limitations on the number of concurrent process. As a result, choices had to be made on the organization of processes. The sensor sweep was given its own process so the gathering of light data would be continuous. All of the strategic behaviors were put into one process and were called in succession inside a while loop. The move forward behavior was called first as a default if the other behaviors did not produce an output. The tropotaxis behavior was called second so that it had an opportunity to find a light source. If it did not find one, it triggered the search light behavior. The tactical behaviors were also put into a single process. The low obstacle avoidance behavior executed first and then the high obstacle avoidance behavior. This order was chosen because avoiding tall object has higher precedence. Though it is not a behavior, motor control was placed into the tactical behaviors process for two reasons: first to cut down on the total number of processes and second to ensure that an up to date tactical output is available.



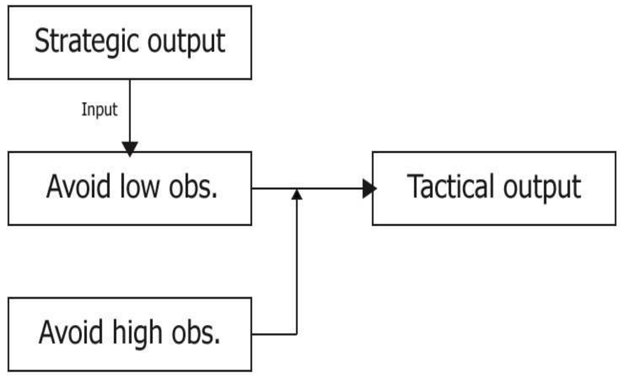
(Figure 1)



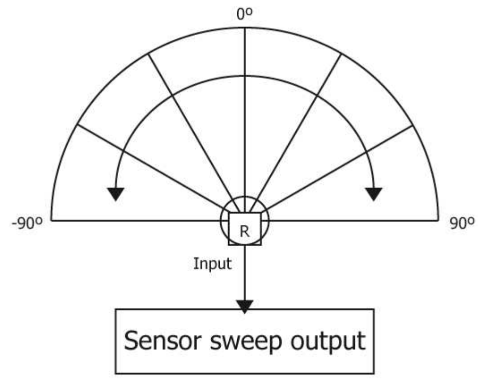
(Figure 2)

Group 10

Amit Maole
Brent Eskridge
Klo Utley
Tony Lopez



(Figure 3)



(Figure 4)