Memo

To: Dr. Berry

From: Christopher Collinsworth, Jordan Patterson

Date: 2/4/2016

Re: Lab 7 – Path Planning

The purpose of this lab was to use topological and metric navigation to move the mobile robot from a starting position to a goal position in a given world. The lab was separated into three parts: 1) topological navigation, 2) metric path planning and execution, and 3) topological path planning and execution.

Implementation of topological navigation in part one was achieved by using the robot's sensors to identify various gateways and distinctive places in a given world so that the robot could navigate accordingly. In addition, the robot was initially given a series of commands through the use of push buttons to assist in the successful execution of its path.

Implementation of metric path planning and execution in part two was achieved by using a wavefront expansion algorithm to plan a path from the robot's starting position to a goal position. A given world map was provided, consisting of a 4x4 array of 0's and 99's representing free spaces and obstacles. Starting and goal position coordinates were given to the robot via pushbuttons and an optimal path was generated using a wavefront expansion algorithm. The robot was then able to use this generated path, along with our wall following algorithm from lab 4, to navigate the world without hitting any obstacles until arrival at the specified goal position.

Implementation of topological path planning and execution in part three was achieved similarly to part two, however the world map was represented by the topology of the space including walls, hallways, corners, and junctions as salient features represented by an integer between 0 and 15, dependent upon the location of the wall(s) around the space. A wavefront expansion algorithm was used again to generate an optimal path from a starting position to a goal position. The robot was then able to navigate this path using various behaviors and rules.

In conclusion, the robot performed all tasks well during the lab demonstration. It successfully was able to use topological navigation to move from a starting position to a goal position in part one. It was also able to successfully navigate from a starting position to a goal position using wavefront expansion with an occupancy grid in part two and a topological map in part three.

1.) **What was the strategy for implementing the wavefront algorithm?**
Our strategy consisted of first locating the goal, designated by the number 98 on the map. We then worked our way backwards through the map in waves until reaching the robot's starting position. On the first wave, each free space within proximity to the space containing the goal was given a 97. On the second wave, each free space within proximity to a space containing a 97 was given a 96. This process was repeated until arriving at the robot's starting position (1).
2.) **Were there any points during the navigation when the robot got stuck? If so, how did you extract the robot from that situation?**
Yes, sometimes the robot would run into a wall and get stuck due to wall following error. To correct for this, we simply fine-tuned the proportional and derivative gains.
3.) **How long did it take for the robot to move from the start position to the goal?**
On average, it usually took no longer than 20 seconds.
4.) **What type of algorithm did you use to select the most optimal or efficient path?**

Memo

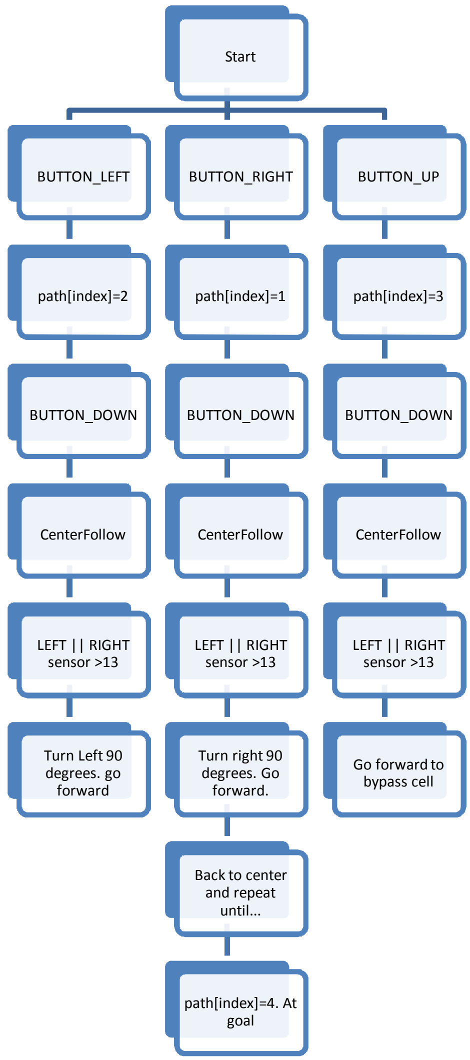
Our algorithm for navigating the map along the most optimal path begins by finding the robot's starting position on the map. It then determines the direction of the next highest number in proximity to the current space, checks that it is not an obstacle (99), and changes the orientation of the robot so that it's facing direction matches the direction of the aforementioned next highest number. The robot then moves to that space. This process is repeated until the robot's current position matches that of the goal position.

5.) **How did you represent the robot's start and goal position at run time?**
The robot's starting position was represented by the number 1 and the goal position was represented by the number 98. This was shown on the map.

6.) **Do you have any recommendations for improving the robot's navigation or wavefront algorithm?**
The wavefront algorithm seems to be about as efficient and effective as it could possibly be. As for navigation, there are probably a multitude of different ways for implementing a navigation algorithm. Our navigation algorithm is very efficient and effective, probably as good as any other. There are probably a few ways to improve upon its efficiency and effectiveness though, but none that come to mind at the moment.

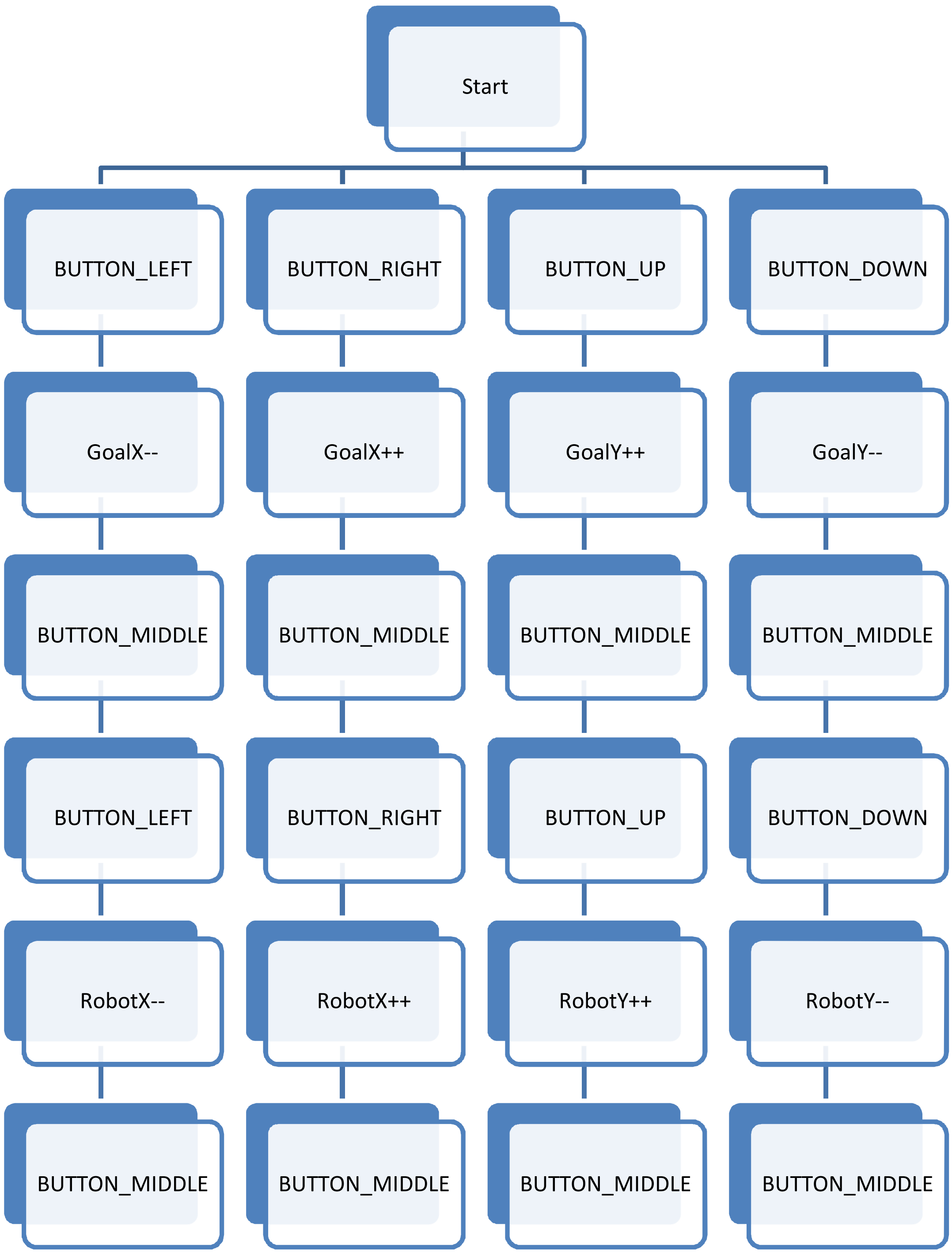
Memo

Appendix:
Part 1:



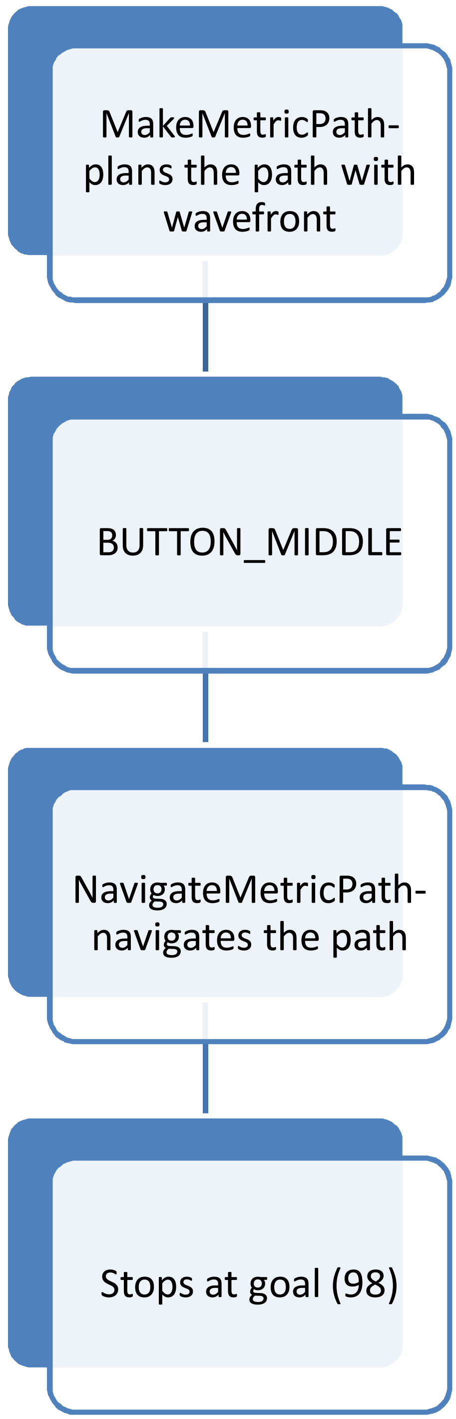
Memo

Part 2:



Continued on next page…

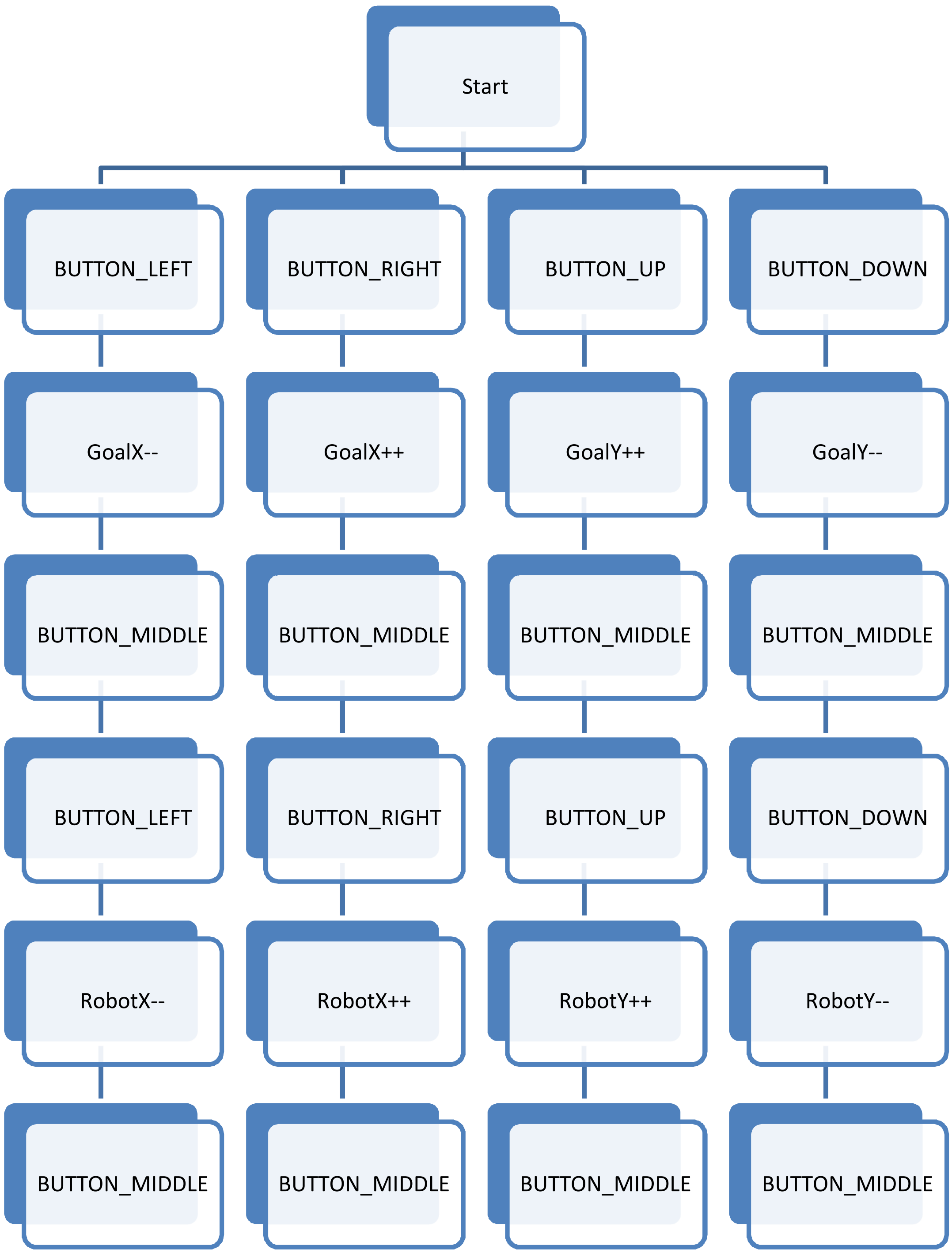
Memo



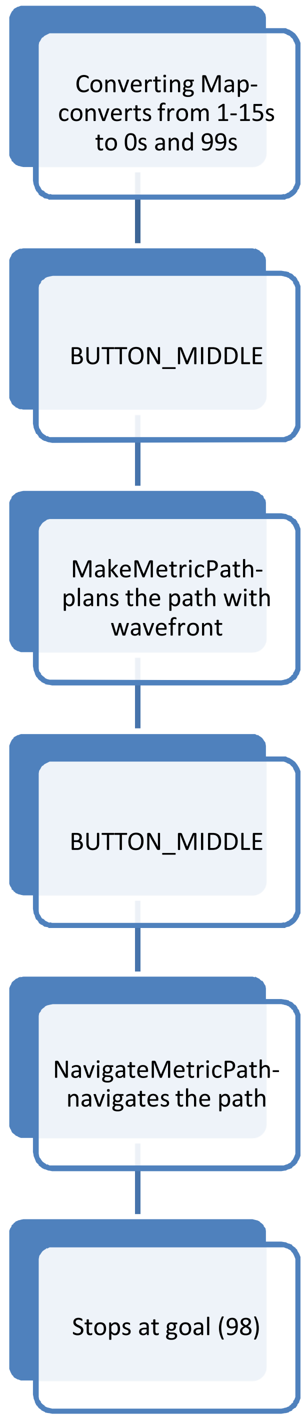
MakeMetricPath- plans the path with wavefront
BUTTON_MIDDLE
NavigateMetricPath- navigates the path
Stops at goal (98)

Memo

Part 3:


Start
BUTTON_LEFT
BUTTON_RIGHT
BUTTON_UP
BUTTON_DOWN
GoalX--
GoalX++
GoalY++
GoalY--
BUTTON_MIDDLE
BUTTON_MIDDLE
BUTTON_MIDDLE
BUTTON_MIDDLE
BUTTON_LEFT
BUTTON_RIGHT
BUTTON_UP
BUTTON_DOWN
RobotX--
RobotX++
RobotY++
RobotY--
BUTTON_MIDDLE
BUTTON_MIDDLE
BUTTON_MIDDLE
BUTTON_MIDDLE

Memo


Converting Map-converts from 1-15s to 0s and 99s
BUTTON_MIDDLE
MakeMetricPath-plans the path with wavefront
BUTTON_MIDDLE
NavigateMetricPath-navigates the path
Stops at goal (98)

Memo

Table 1:

	Left Sensor	Front Sensor	Right Sensor
Left Corner	5.6	5.43	19
Right Corner	20	4.5	5.5
Front Corner	23	6.0	18
Dead End	5.0	5.0	4.75
Left Hallway	20	20	5.5
Right Hallway	5.5	20	20
Hallway on both sides	20	20	20
T-Junction	20	5.5	20