**Graduate Projects**

University of Colorado at Boulder

Aerospace Engineering Sciences

ASEN 5018/6028 –Fall 2015

|  |
| --- |
| **FlyNet**  **Planning Subsystem Summary/Continuity Document** |

|  |  |  |
| --- | --- | --- |
| **Approvers List** | | |
|  | Title | Name |
| Prepared By | Planning Team | Bryce Hill |
|  |  | Edward Meletyan |
| Approved By | Planning Lead | Bryce Hill |
| Approved By | Systems Engineer |  |
| Approved By | Project Manager |  |

**1: Introduction & Summary**

To complete our objective of searching a space for targets a path must be generated which: avoids obstacles, arrives at desired final destination and is possible to compute in real time environments.

Two algorithms have been evaluated: the first is a hybrid Voronoi/A\* method which uses the Voronoi algorithm to generate points between obstacles and the A\* algorithm to plan the most efficient path from start to goal. The second is a probabilistic road map approach.

# **2: Semester Report**

## 2.1: Objectives and Tasks List

**Completed**:

1. Matlab implemented Voronoi/A\* 2D path planning algorithm
2. Matlab implemented probabilistic road map 2D path planning algorithm
3. Designed a perpetual motion machine

**Incomplete**:

1. Test each planning algorithm in Matlab simulation

## 2.2: Issues

What problems prevented you from being able to complete the tasks above?

1. This tasks remains incomplete due to time constraints.

## 2.3: Lessons Learned

So you just identified some things that prevented you from completing some tasks – is there anything you can learn from that? Having more time to do things isn’t really a lesson learned, so anything like Issue 3 in my example above won’t really have a lesson learned. Remember: lessons learned are anything that you **wish you knew before**, something **learned the hard way**, or a **piece of advice**. Note that you don’t have to address each numbered issue above directly. Also note that you are encouraged to include lessons learned in the process of completing the tasks that you *did* finish.

1. Don’t check your email past 9pm. The PM will email you expecting something at 8am the next day.
2. Decide who is going to keep the team yacht *before* it is finished being built, otherwise you may end up with a lot of bickering and fighting.
3. Don’t tell anyone about the perpetual motion machine – they either won’t believe you or will try to steal it.
4. Always define an acceptable tolerance when defining a specific value for a requirement. Finding out the meaning of life was 42 didn’t really help because we didn’t have a requirement set on the acceptable tolerance around the nominal value of 37.

## 2.4: Procedures

This section will likely be the bulk of your report. What did you actually do? Be as detailed as you can. **List any software used, including the version of the software**. You started from nothing, how did you get to the completed tasks? What progress have you made on incomplete tasks and what have you been doing to make that progress? Where can more detailed documentation be found?

Table 2.: Software list

|  |  |  |
| --- | --- | --- |
| Program Name | Version | Purpose |
| Matlab | R2015a | Algorithmic development and testing |
|  |  |  |

# **3: Next Semester/Future Expectations**

## 3.1: Prioritized List of Tasks and Objectives

Include your incomplete tasks ***and*** next steps for your subsystem. It is important for you to be thinking ahead. If your project is not continuing next semester, summarize what you think could be done if your project was reinstated in the future.

1. Finish preliminary design of toaster
2. Finalize the decision on the peanut butter – jelly interface
3. Start analysis of fish stick crumbliness

## 3.2: Starting Points

For each one of the tasks and objectives in 3.1, describe to the best of your knowledge where the person reading this can start. The person reading this is either going to be you after over a month off, or someone totally unfamiliar with how to go about things. **Point to the locations in the server where you have pertinent files saved.** Now the person has the file open in whatever software, what should they work on doing to start with?

1. Structural design of the toaster can be found on the project server under Working Directories > By Subsystem > Structures > CAD > Toaster.cad. We suggest first adding a best estimate for the heating coils to place volume constraints on the real coils that can be used. Other work that needs to be done is ensuring the polish will be shiny enough to see your face in the reflection.
2. The lingering issue in finalizing the peanut butter – jelly interface is the type of jelly to be used. We have collected some data on various jellies, but a trade study should be performed ASAP to determine the best jelly option.
3. Fish sticks have already been chosen but their crumbliness has not yet been analyzed. We suggest importing the CAD model of the fish stick and performing FEA stress analysis on the fish sticks to determine how much mass will be lost to rogue crumbs.

## 3.3: Improvement, Updates, Verification

For tasks you have completed, what could/should be done to improve or update them in the future? Here is a good place to blatantly state all the assumptions you have made, and prioritize them in order of the impact the assumption has on your result. As assumptions later get filled with more concrete data, your analysis will need to be updated and/or verified to ensure no issues have been raised.

Note: Be careful with improvements -- remember the goal is always to meet the requirement and not go any further.

1. We have assumed that there will be a large body of water near Boulder, CO for the team yacht. When that assumption is verified, the yacht should be moved to the body of water. Should this assumption prove to be invalid, drastic redesign of the boat may be necessary.
2. We have assumed that the question “what is the meaning of life” is one that makes sense. We have an answer to this question, but the question may need to be updated in the future as it may be out of the scope of this project.
3. In the design of the perpetual motion machine, we assumed the existence of tachyon particles to provide superluminal ghost-forcing on our gyroscopic, electromagnetic tether. Specifically, the tachyons are expected to produce a force of 30N. This number was determined using superluminal, reverse-time Feynman diagrams but should be updated when a better understanding of the interaction between the Higgs boson, muons, and tau-neutrinos is obtained.