**EE106A/ 206A/ Bioe 125 Project Proposal**

**Contact Information**

|  |  |
| --- | --- |
| Name | Email |
| Robert Spark | rspark@berkeley.edu |
| Anders Lewis | anderslewis@berkeley.edu |
| Andrew Blatner | ablatner@gmail.com |
| Henry Smith | Henry.smith@berkeley.edu |

**Project Proposal Abstract**

We plan on having Baxter learn how to play dominoes. We will likely be modelling our project upon last year’s Baxter chess project, as we will have similar goals; learn the state of the game board, understand when it's time to make an action, get the next optimal move from an API, and move the piece correctly and efficiently to its correct location.

**Project Description**

We plan to play a simplified version of dominoes in order to relax some constraints of the game and provide possibly simpler motion planning tasks for domino placement by Baxter’s gripper.

The rules for the basic game of dominoes are as follows:

* The goal of the game is to be the first player to reach a score of 150 (can be reduced for speed)
* There will be two players in this game, the challenger (human) against baxter.
* We will only be playing this game without double sided dominoes (tiles with matching number of dots on both sides), so there will be 21 unique dominoes in play.
* The game will start by first placing all of the dominoes face-down on the table and mix them (the human will do this part)
* Each player will then select 6 dominoes from the face-down pile and place them in their own hand without the other seeing (this rule may become relaxed later)
* Since the human is the guest, they will place the first domino on the table to start the game.
* To play a valid tile, you must place a matching sided tile on any of the four exposed sides of the first tile.
* The first tile is the only tile on which others can be placed on all four of it’s sides. The other dominoes must be placed end to end afterwards.
* If the player’s tile will go out of bounds on the board, they may connect a tile at a right angle to a matching tile.
* If a player is not able to play any of their tiles correctly, they must pick a new tile from the remaining face down tiles (the bone yard) until they pick a tile that can be played.
* If they are unable to pick any playable dominoes and the boneyard is empty, they must pass their turn to the opponent.
* Points for each move are awarded based on the total number created by the end dots on each chain. They are only awarded if this total number is a multiple of 5 and that many points will be awarded (ex. 5, 15, 25…)
* The first player to play all of their dominoes wins the hand (current round). They then add up all of the opponent’s dots on all tiles together and round to nearest multiple of 5. They are awarded this many points for winning the hand. The first player to reach 150 points wins.

We've broken down our main goal into 4 rough subgoals:

* Recognize dominoes and their position on the board
* Ascertain the state of the game from domino positions
* Get the board state into a format that our domino solver will recognize and translate it back onto the board.
* Actually carry out the move with a valid move by baxter that doesn’t collide with objects in the environment or interfere with the board state. This will be done by forcing checkpoints at various time intervals to be met during Baxter’s motion.

This project is interesting because it incorporates many different coordinated pieces together to produce a worthy real life game adversary that could compete against humans. It also poses a challenging motion planning task: how do you efficiently place dominoes on the board without disrupting the current game state in addition to playing competitively against a real opponent.

Our project involves sensing by using a camera with computer vision and potentially AR tags to track positions of tiles on the board, game board boundaries, and other necessary constraints. The planning and actuation includes incorporating a Domino AI game engine to help calculate strategic moves and then using this information to produce planned motion of Baxter’s hands and suction cup gripper to move the tiles to their corresponding locations in a reasonable manner.

Our project is similar to previous year’s projects involving Baxter playing board games such as Chess or Athelo. We will need to keep track of the board state, other game environment constraints like the table surface, tiles in play, and board boundaries.

**Tasks**

**Critical:**

1. Develop a model on RViz for testing.

**Parallelizable Tasks:**

Movement:

1. First, move an AR-Tag brick using MoveIt on ROS.
2. Have baxter follow series of checkpoints when moving brick.
3. Provide Baxter with necessary constraints for collision avoidance (lab 7).
4. Then, move large dominoes.
5. Be able to place dominoes in correct orientation on board.
6. Then, be able to flip dominoes
7. Then, be able to pick out of the boneyard to start the game.

CV:

1. First, be able to detect board state for the solver engine
   1. This will likely end up with us doing ML

AI:

1. Random move at first for testing other components
2. Random legal move
3. Greedy one move lookahead
4. Solvers that track game statistics/n-player expectimax
   * There is surprisingly little attention given to optimal strategies for Dominoes, but we did email one blogger who was able to get a marginal improvement over the greedy strategy.
   * He provided us with a link to his GitHub Repository, which can be found at <https://github.com/jackdied/muggins> to use the engine

Stretch:

If we have extra time, it would be fun to have Baxter set up a line of dominoes and knock them down.

**Milestones**

Note: Tiers may vary in order depending upon progress in certain areas etc.

* **Tier 1: Manipulating Domino Tiles with Baxter**
  + Be able to set the constraints on which Baxter is able to move around such as the table, items on the table, and other constraints needed. (week 1-2)
  + Move block with gripper to correct location (hard coded) (week 1-2)
  + Be able to move AR-tagged objects on a table surface (week 3-4)
  + Be able to move dominoes on a table surface (week 3-4)
  + Stretch: Have Baxter able to flip over dominoes efficiently (If all previous milestones completed)
* **Tier 2: Computer Vision**
  + Develop working image homography of table surface used for Baxter
  + Having computer vision model be able to identify dominoes on table
  + Have Baxter keep track of the board boundaries
  + Be table to detect positions for valid move
* **Tier 3: AI Domino Game Engine**
  + Have separate working version of game engine, not using computer vision
* **Tier 4: Connect Computer Vision and Baxter**
* **Tier 5: Connect Tiers 2 and 4**
  + Connect Computer Vision Model and Baxter to move objects in environment and sense/ record changes
  + Connect Game engine and Computer Vision Model and test basic moves, placement
* **Tier 6: Test Basic Moves**
  + Connect Game Engine, CV Model and Baxter to make basic moves

**Assessment**

Given how many past groups have seemed to struggle with grasping objects and setting them in a regular grid, we expect this to be a pretty ambitious project. Realistically, we think we should be able to position enlarged dominoes with AR tags in valid move states.

Baxter should be able to make a correct decision in whether it can place a domino in the correct position and orientation on the board. Baxter should not disrupt any of the currently pieces in play when placing a new tile on the board.

Baxter should also know when to take it’s turn. On past projects, if error propagated with Baxter, it would continuously make more incorrect moves. There needs to be some sort of fail-safe in case the board is disrupted.

Finally, if Baxter plays correctly, it should be able to win some basic games against a standard opponent. A stretch goal would be for Baxter to work flawlessly, having made perfect moves each time without any board disruption. We predict there will be some error produced by gripping and moving the dominoes.

**Team Member Roles**

Anders

* Role(s): Implementation of the Domino AI game engine, assist in helping Baxter move objects
* Background: CS 188, CS 170, EE 126 a lot of personal projects in unix and scripting; all lower div CS, no EE.

Robert

* Role(s): Motion planning of domino placement with suction cup, help configure domino game engine, prepare model in Rviz
* Background: CS 188, CS 70, CS61A, CS61B, CS61C, Robotics Research in Fearing Lab Spring 2016, EE20, EE40, MATLAB

Andrew

* Role(s): Interface subsystems, motion planning and Baxter control
* Background: EE 123, 126, 128, C220B, CS 162, 188, 170; Python ninja

Henry

* Role(s): Developing computer vision portion, and interface with game engine.
* Background: CS 188, 61 series, EE16B, working on another CV project

**Bill of Materials**

Resources / Items that will plan to use and are currently available:

|  |  |
| --- | --- |
| Item | Quantity |
| Baxter | 1 |
| Suction Cup | 1-2 |
| Lab Computer w/ ROS installed | 1 |
| Robot Hand Gripper | 1 |
| Table | 1 |
| Camera | 1-2 |
| Black Table Cloth | 1 |
| White tape roll | 1 |

For **new** items we want **purchased**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item | Quantity | Price | Website | Justification |
| Large dominoes set | 1 | $14.99+tax | <https://www.amazon.com/Learning-Resources-Jumbo-Foam-Dominoes/dp/B000NVNMW4/ref=sr_1_1?ie=UTF8&qid=1477628501&sr=8-1&keywords=large+dominoes> | Require large dominoes to allow relaxed motion planning constraints |

Total Price: $14.99 + tax

**Other**

**GSI response:**

Good idea. I like that you broke the problem into manageable steps. However, you will need to be more concrete with how you plan to solve each step. A brief explanation of the rules of the game you plan to play would be helpful too. We have suction cups for Baxter, you may want to talk to one of the GSI's so that you can try them out. Sensing:- You'll need to figure out how to recognize each domino and how to represent the state of the game. I am guessing you will be making use of a camera here. Planning:- Do you have a game engine in mind already? Actuation:- consider Baxter's erratic movement and think of how you can constrain it. You may want to talk to Shivani ( one of the lab assistants) to see how her team was able to overcome this last year. Your concerns about the effect of inaccuracies are also valid. You may want to test this out to see if it will be a big problem. Finally, we will need you to carefully document what each team member does. We need to make sure everyone has something reasonable to do.