Math 4020 Project Manual

"Do Dogs Know Calculus?"

(no)

Overview

The inspiration for this research project comes from Timothy Pennings and his dog Elvis. In particular, Pennings had a seemingly ridiculous question: Do dogs know calculus? Make no mistake, *it is* a ridiculous question, however this thought experiment proved interesting.

Essentially, our project's genesis comes down to an overeager dog on the beach. When Pennings would throw a tennis ball into the water, he noticed that Elvis did not travel in a straight line from the beach to the ball, rather Elvis entered the water at an angle with respect to his path on the beach. Understanding the canine's propensity for enthusiasm, Pennings conjectured that this was an instinctive response on Elvis's part to minimize the time to reach the ball, as opposed to minimizing the distance. It was fairly obvious that Elvis could run in the sand much faster than he could swim in water, thus the optimal route with respect to time would not be in a straight line. Pennings ultimately ran some trials to conclude Elvis was indeed doing this. Despite this statistical evidence, Pennings determined Elvis did not know calculus admitting that, "In fact, he has trouble differentiating even simple polynomials." Even still, how could Elvis learn calculus if he, as a dog, is under the constant temptation to eat his homework?

Timothy Pennings's question has manifested itself in the form of software that this class has developed. Our goal was fundamentally simple, yet complex in practice. Given two regions with two different speeds, how could we minimize the time to travel from point A to point B? Our software takes the more general approach of giving the user a reachable set, that is the set of all points reachable given a certain amount of time to travel. We considered three cases: a trajectory completely within region 1, a trajectory from region 1 into region 2, and a trajectory from region 1 then along the boundary and finally back into region 1 (so called "Swim Run Swim" or SRS). The motivation for the SRS trajectory is to show the reachable points that would not be reachable with a trajectory completely contained in region 1.

Using the GUI

The interface has been designed with ease of use and readability in mind. Upon opening the program, the user will see the following:

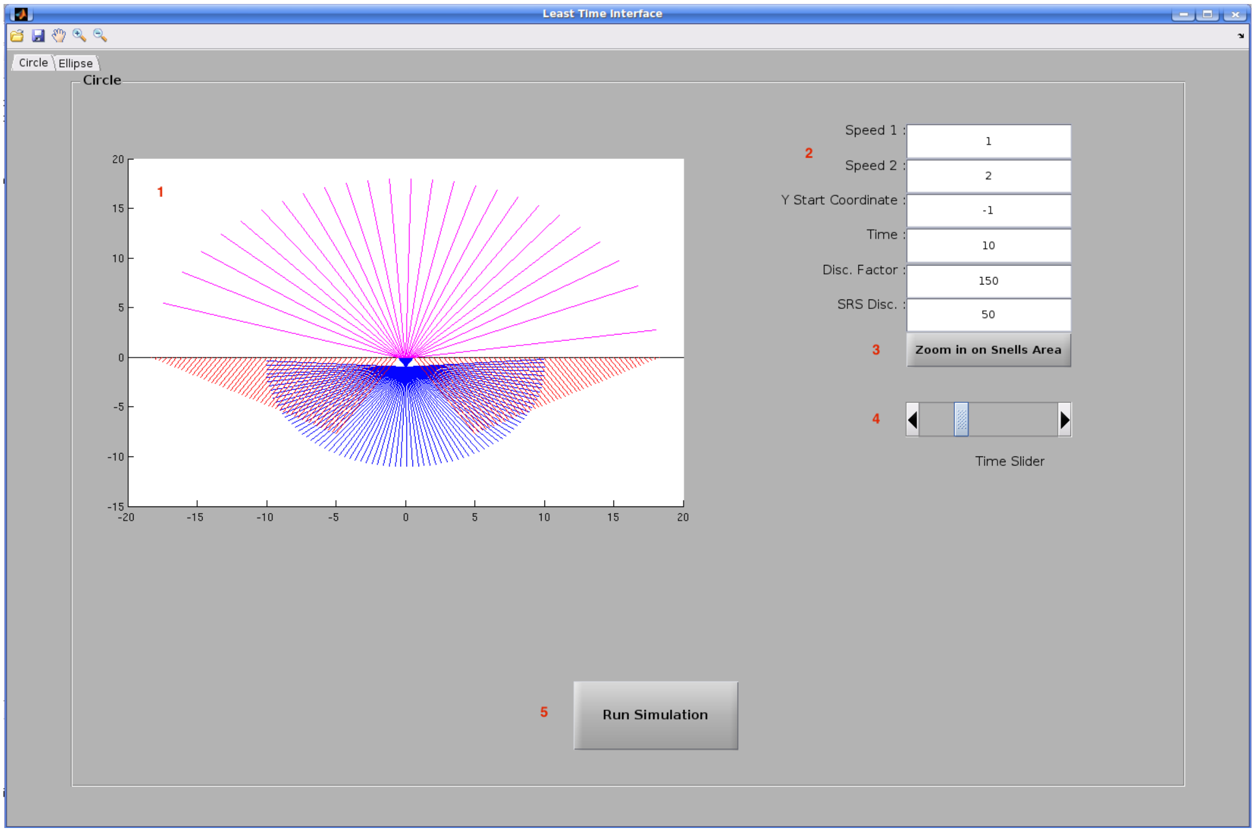
The basic layout consists of two tabs, one for analysis on a circle and one for analysis on an ellipse. For the circle, the user specifies the region speeds, which graphically are separated by the x-axis in the Cartesian plane, the starting position on the y-axis, and the time. When the user runs the simulation, a graphic appears showing the reachable set in three cases:

1. Traveling from the first region into the second region.
2. Traveling only in the first region.
3. The Swim Run Swim (SRS) trajectory.

The user can also customize how many trajectories are drawn graphically by discretizing the reachable set at the point of origin. Additionally, A zoom button allows the user to instantly zoom in on the boundary to more easily show the trajectories as they travel into the second region. Since the angles formed by the paths in region 1 and the paths in region 2 obey Snell's Law, this has been named "Snell's Area".

The same analysis is done in the second tab for elliptical regions, except that the speed of each region depends on the size of the ellipses and varies depending on the angle of the path out of the origin point. Additionally, in the ellipse analysis the SRS reachable set is not computed. We shall go over each tab now.

Circle Tab



1. **Graphical window:** This window displays the results of the analysis. Each of the three colors denote a particular path. Blue denotes the reachable set into the first region as well as the initial path from region 1 into region 2. Pink denotes the reachable set into the second region. Red denotes the SRS reachable set as we travel from the first region, along the boundary, and back into the first region.

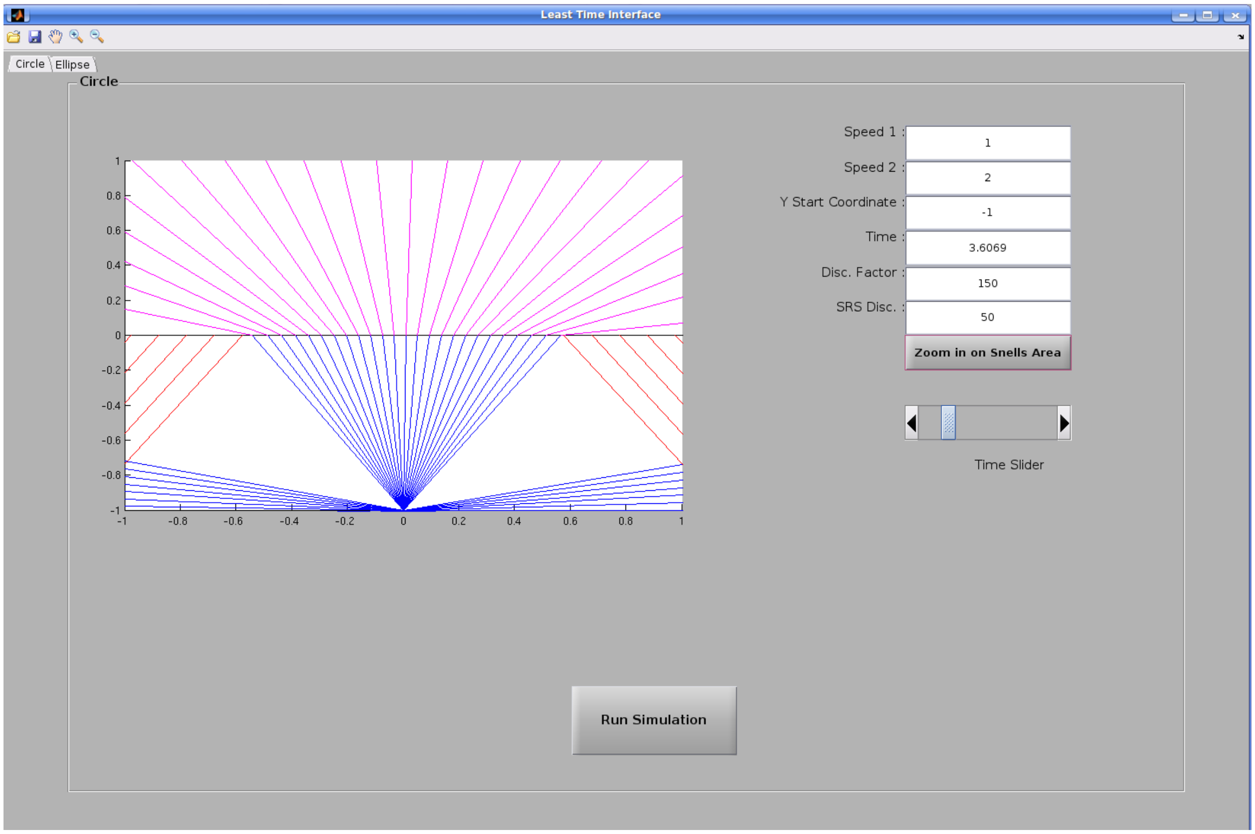
2. **Text fields:** Here the user may alter the speed of the each region, the position of the starting point, and the travel time. Additionally, the discretization of the paths can be changed as well as the discretization of the Swim Run Swim scenario of returning to the first region (denoted in red).

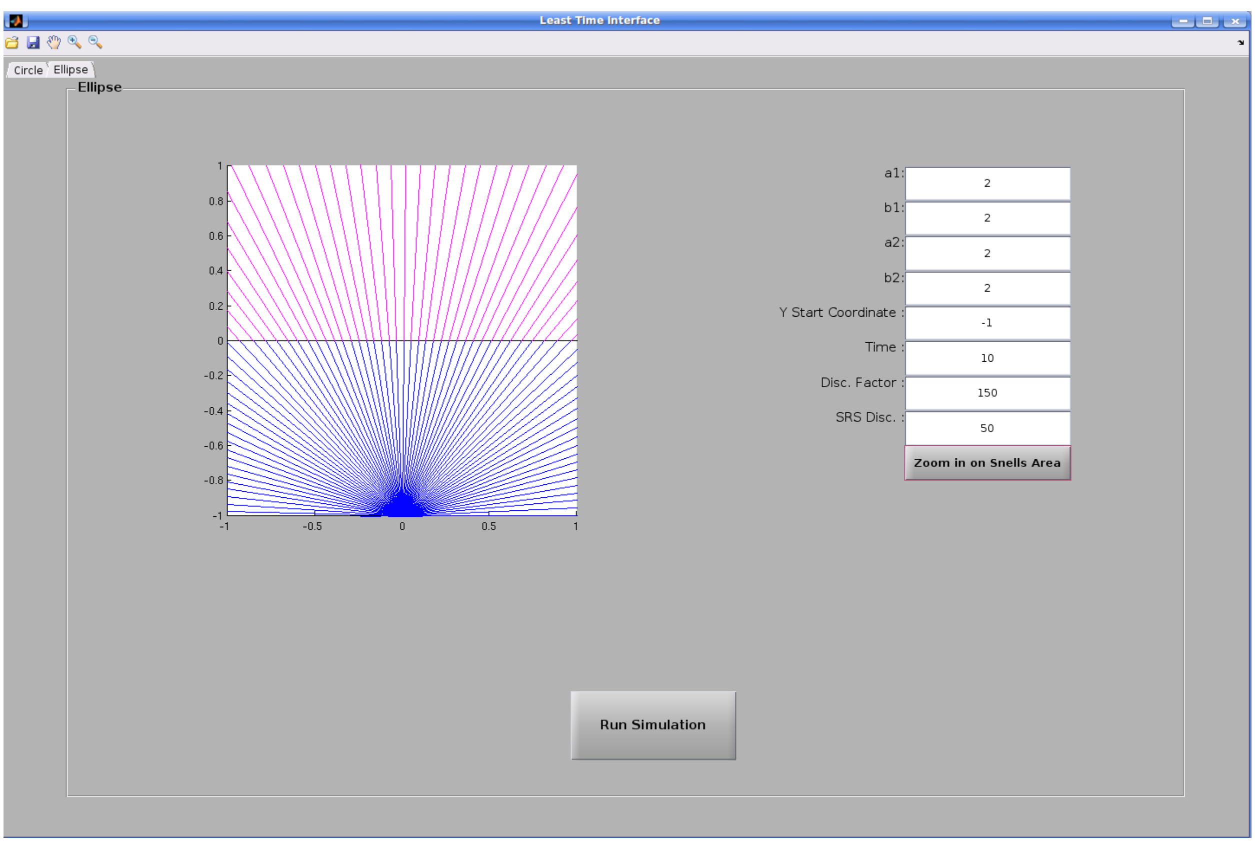
3. **Zoom:** This button simply zooms into the graphic at the point of entry into the second region to better show Snell's Law in action.

4. **Time Slider:** This allows the user to adjust the time quickly and run the program.

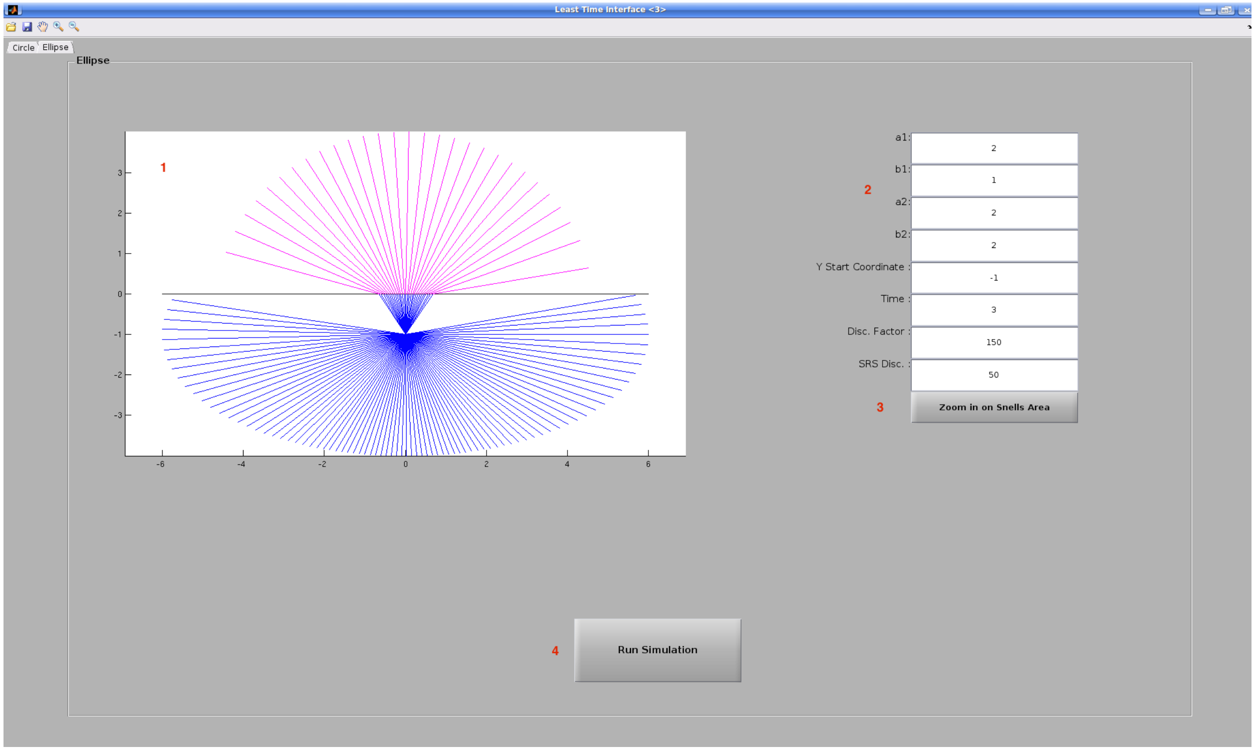
5. **Run Button:** Once clicked, the text will change to "Running…" so that the user knows it is working.

Here are some screenshots of the Snell's zoom in action:





Ellipse Tab



6. **Graphical window:** This window displays the results of the analysis. Each of the two colors denote a particular path. Blue denotes the reachable set into the first region as well as the initial path from region 1 into region 2. Pink denotes the reachable set into the second region.

7. **Text fields:** Here the user may alter the size of the ellipses in regions 1 and 2 as well as the position of the starting point and the travel time. Additionally, the discretization of the paths can be changed.

8. **Zoom:** This button simply zooms into the graphic at the point of entry into the second region to better show Snell's Law in action.

9. **Run Button:** Once clicked, the text will change to "Running…" so that the user knows it is working.

Here are some more screenshots of the Ellipse analysis in action: