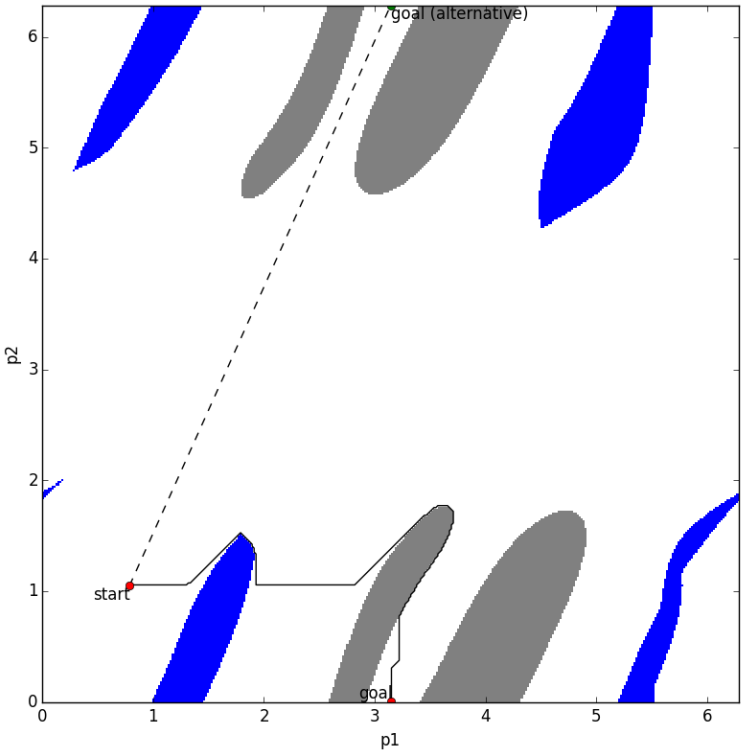
Robotics Assignment #06

Paul Bienkowski, Konstantin Kobs

29. Juni 2015

Task 6. Our implementation makes heavy use of the two libraries *shapely* (for obstacle intersection calculations) and *matplotlib* (for plotting the results). We also attach a pathfinding module (`astar.py`)[1].



In this figure, we have plotted the results of all the tasks from this assignment:

- Gray C-obstacles correspond to the obstacles o_1 and o_2.
- Blue C-obstacles correspond to the obstacles o_3 and o_4.
- We have plotted the point-sized manipulator configurations "start" and "goal", as well as the path between them (using our A* pathfinding algorithm).
- We have plotted an alternative goal under the assumption that our joints can rotate over the 2π boundary, hence wrapping around the plotted space. This leads us to a simpler path (dashed) between "start" and "goal (alternative)" without obstructions.

[1] modified from http://code.activestate.com/recipes/578919-python-a-pathfinding-with-binary-heap/