ECE 498SMA: Principles of safe autonomy	Spring 2020
Your name coursewebsite	*Homework 3: Safety and invariants*
	4/24/2020

Homework is individual work. **This is even more important now that our course is going online. Do not share your answers. Violation of academic integrity code will be penalized.** *Create a .pdf file with your solution, ideally using the provided LaTeXtemplate, and upload the file on compass. In order to prepare for exams, we recommend that you do these problems after studying and understanding the lecture materials and the reference text. Looking for answers online will impair your learning.*

Problem 1. Conditional probabilities (10 points) In deriving the Bayes filter update rule we used the following

$$P(x \mid y, z) = \frac{P(y \mid x, z) P(x \mid z)}{P(y \mid z)}.$$

Derive this relationship using the definition of conditional probabilities.

Problem 2. Particle filters (10 points)

(a) What is importance sampling? Write the pseudocode. What role does it play in the particle filtering algorithm?

(b) List all the information you need to implement a particle filter for a completely new system.

Problem 3. Search (10 points) Prove that if the heuristic function h is *consistent*, that is, $h(u) \leq w(u, v) + h(v)$ for any edge (u, v) in the graph, then the total cost function $f(v)$ is non-decreasing along all paths. Recall, the total cost $f(v) = g(v) + h(v)$, where $g()$ is the *cost to come*.

Problem 4. Billiards automaton (30 points) Consider an idealized billiard table of length a and width b. The table has no pockets; its surface has no friction; and its boundary bounces the balls perfectly. Write an automaton model $\mathcal{A}$ for the position of two balls of equal mass on this table. Use discretized state space for your model. That is, the state for each ball is given by 4 integers (x_i, y_i, vx_i, vy_i) where $0 \leq x_i \leq a$, $0 \leq y_i \leq b$ and $vx_i, vy_i \in \{-1, 0, 1\}$. At each transition, each ball can **move** to only one of its 9 neighboring cells based on vx_i, vy_i.

Each ball has some initial velocity (one of 9 possibilities). A ball **bounces** off a wall when its position is at the boundary. Balls **collide** whenever $|x_1 - x_2| = 1$ and $|y_1 - y_2| = 1$

and their velocity vectors are pointing towards each other. Whenever a bounce occurs, the appropriate velocity changes sign. Whenever a collision occurs, the balls exchange their velocity vectors.

(a) Write the definition of the transition relation for this automaton. Specifically, write the **move**, **bounce**, **collide** transitions using mathematical formulas or as programs.

(b) State conservation of speed along each axis as an invariant property of the automaton.

(c) Prove the above invariant using the theorem from the last verification lecture.

(d) Write a program to compute the reachable states of a set of balls. That is, given a set of initial positions for each ball Θ, and a fixed initial velocity for each ball, and a time bound $T > 0$, your program should output and plot, for each time $t = 0, 1, \ldots, T$ the set of possible positions for each ball.

(e) For a single ball starting at $\{(6,0),(7,0)\}$ with velocity moving it in the NE (north-east) direction, compute and plot the reach sets for $T = 10, 20, 30, 40$ time units.

(f) *Can* this ball reach a pocket located at $(5,7)$? Is it *guaranteed* to reach the pocket?

Problem 5. Checking invariants (30 points)

(a) Give an algorithm that takes as input a finite state automaton $\mathcal{A} = \langle Q, Q_0, A, D\rangle$ with $|Q| < \infty$ (as defined in the lectures) and a candidate set $I \subseteq Q$, and checks whether I is an invariant of $\mathcal{A}$. That is, your algorithm should return "yes" if indeed I is an invariant of $\mathcal{A}$ and otherwise it should return "no". Write the pseudocode of the algorithm clearly (every statement in the pseudocode should be a well defined function).

(b) Prove (argue precisely) why your algorithm gives the right answer.

(c) What are the time and memory requirements of your algorithm?

Problem 6. Hybrid A* (10 points) This problem asks you to work through a few steps of the Hybrid A* planner similar to MP3. Let us suppose that the discrete coordinates of the vehicle are integers. For each node n, the discrete coordinates are obtained by rounding the computed continuous coordinates to nearest integers. We will work with the heuristic function:

$$h(n) = \sqrt{(x_n - x_{goal})^2 + (y_n - y_{goal})^2}.$$

The cost function $g(n) = g(previous(n)) + 1$. The steering angle $\delta \in \{0, -45°, 45°\}$. The speed $v = 1.0$. For heading angle θ (x-axis points to $0°$). Vehicle length $l = 2$. The goal position is at $(0,0)$. Suppose we use the following vehicle model:

$$\begin{aligned}\theta_{t+1} &= \theta_t + \delta \\ x_{t+1} &= x_t + v\cos(\theta_{t+1}) \\ y_{t+1} &= y_t + v\sin(\theta_{t+1})\end{aligned} \tag{1}$$

Suppose the hybrid A*'s open set pops out a node n_0. Its continuous coordinates are $(x, y, \theta) = (3.0, 5.0, 90°)$ and $g(n_0) = 13$. After n_0 is popped out, the open set has one node n_1 left. Its continuous coordinates $(x, y, \theta) = (2.7, 6.1, 90°)$ and $g(N_1) = 11$.

(a) Calculate the coordinates in continuous space for all the new nodes generated from n_0.

(b) Write down the continuous coordinates (x, y, θ) of all the nodes in open set after this iteration is finished.

(c) Which node will be popped out from the open set in the next iteration? Write down its coordinates in continuous space.

MPi survey (10 points) When you are done with each programming assignment (MPi) with your group, you have to give an assessment of how much time each member of your group devoted to this MP each week in the following table format. This could be submitted as a separate document.

	Name 1	Hours 1	Name 2	Hours 2	Name 3	Hours 3
Week 1						
Week 2						