Ian Cornelius

CSE-40793

September 24, 2018

Homework 5

Design Rationale

To handle correctly opening and closing gates based on train proximity, I changed the conditions under which the gate state is changed. I added a buffer, or "overlapped zone", where the trains would be seen by both gates at once. If one train is leaving a particular gate, the gate checks if the other train is either approaching or within the station before changing its state. Effectively, the gates will not open when both trains are between them – only when both trains are registered as leaving. This seemed like the simplest way for me to correctly open/close gates based on the positions of both trains.

To correctly switch gate/car observing, I turned the collection of gates into an array to easier access each particular gate. Once it was determined that a car would switch, I would remove the gate observer for the original gate and add the observer for the other gate. An array of gates was easier to manage than a collection, and a cast to a CrossingGate object is all that was needed.

In the event that the South Bend mayor asks if this simulation can be upscaled, I am not sure of the answer. I don't think it would effectively upscale because some aspects of this program are handled explicitly knowing that there are two gates, two roads, and two trains. I think upscaling would require a refactoring of the train and road classes to something akin to CarFactory.