Benchmarks for handling STL∗

ABSTRACT

We modeled 10 hybrid automata benchmarks for applying the contents of the paper. We will explain each model briefly. After that, we will explain how to execute the benchmarks.

© 2018 Copyright held by the owner/author(s). 978-1-4503-0000-0/18/06...$15.00

https://doi.org/10.1145/1234567890

1 Networked thermostat controllers

There are two rooms, they are interconnected by an open door. The temperature of each room is separately controlled by each thermostat, depending on both the heater’s mode and the temperature of the adjacent room. Initially, the temperature of each room is higher than 19 degrees and less than 21 degrees, and the heater is off. Every thermostat controller synchronously performs its discrete transitions with common action . The number of actions of each thermostat controller is two. One of them is “The heater is turned off”. It is performed when the temperature of the room is higher than 20. The other is “The heater is turned on”. It is performed when the temperature of the room is smaller than or equal to 20. The temperature of each room( is changed according to the following functions:

1. Linear ODE

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

1. Nonlinear ODE

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

where are constants depending on the size of door, the size and the heater’s power of , is set of adjacent rooms of . The safety property is that the temperature of each room is in the range (10, 30). We designed following STL formulas for the model:

The temperature of the is greater than 21 degrees within 40 steps.

For 20 steps, if the heater’s mode of each room is both off, then either one of them eventually is turned on within 5 steps.

The temperature of the should be less than or equal to 20 degrees, until the heater’s mode of each room is both on.

For 10 steps, if the heater’s mode of each room is both on, then the temperature of the is less than or equal to 20 degrees during two steps

2 Networked water tank controllers

There are two water tanks, they are connected by a pipe. The water level of each tank is separately controlled by each pump, depending on both the pump’s mode and the water level of the adjacent water tank. Initially, the water level of each water tank is higher than 4.9 and less than 5.1, and the pump is off. Every water tank controller synchronously performs its discrete transitions with common action . The number of actions of each water tank controller is two. One of them is “The pump is turned off”. It is performed when the water level of the tank is greater than or equal to 5. The other is “The pump is turned on”. It is performed when the water level of the tank is smaller than 5. The water level of each tank() is changed according to the ODEs:

1. Linear ODE

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

1. Nonlinear ODE

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

where are constants determined by the size of the tank, the power of the pump, the width of the I/O pipe, and g is the standard gravity constant. The safety property is that the temperature of each room is in the range . We designed following STL formulas for the model:

The water level of the is greater than 5.1 within 40 steps.

For 20 steps, if the pump’s mode of each tank is both on, then either one of them is turned off within 5 steps.

For 10 steps, if the water level of the is less than 5, then pump of should be always turned on with in 5 steps.

For 10 steps, if the water level of the is less than 4.9, then the level is greater than or equal to 5.1 within 10 steps eventually.

3 Driving Simple Cars

3.1 Linear

Two cars are running in a straight road. The precedes the . The velocity of each car depends on the distance between and . There are three modes for each car. Initially, the position of is in range [0, 1], is [3, 10] and the mode of each car is ‘acc’. If the distance is less than 1, the mode of ( is changed to ‘dec’ and is changed to ‘acc’. If the distance is bigger than or equal to 1 and smaller than 2, the mode of ( is changed to ‘keep’ and is changed to ‘acc’. If the distance is bigger than or equal to 2 and smaller than 3, the mode of ( is changed to ‘acc’ and is changed to ‘acc’. If the distance is bigger than or equal to 3 and smaller than 4, the mode of ( is changed to ‘acc’ and is changed to ‘keep’. If the distance is bigger than or equal to 4 and smaller than 5, the mode of ( is changed to ‘keep’ and is changed to ‘dec’. If the distance is bigger than or equal to 5, the mode of ( is changed to ‘acc’ and is changed to ‘dec’. The value of are changed according to the ODEs:

The safety property is that the always precede .

We designed following STL formulas for the model:

The distance between the and the is less than or equal to 3 within 40 steps.

At some time, if the distance between the and the is less than 4, then the keeps its velocity, and the raise its velocity within 5 steps.

The distance between and is always less than 4 with in 20 steps.

For 10 steps, if the keeps its velocity, and the raise its velocity, then one of them changes its velocity with in 5 steps.

3.2 Nonlinear

Two cars are running in sequence, while each car follows the behavior of the car in front (the first car moves according to its own scenario). There are three modes for each car. The position and the direction of each depends on its seed , and steering angle . Initially, the x position of is in (0,3) and y position is in (3, 10). The x position of is in (5,10) and y position is in (3, 10). The velocity of and are in [3, 4]. The direction of is in (0, 1) and is in (-1, 0). The steering angle of is in (0, 1) and is (-1, 0). We used taylor series for linearization of trigonometric functions. For each step, if the distance between two cars are bigger than 6 smaller than 9, then the mode of car( is ‘keep’. If the distance between two cars are smaller than 9, then the mode of car( is ‘dec’. If the distance between two cars are bigger than 9, then the mode of car( is ‘acc’. The values of variable are changed according to the ODEs:

The goal is to find mode change schedule for driving cars safely. We designed following STL formulas for the model:

The distance between the and the is less than 6 within 40 steps.

At some time, if the distance between the and the is greater than 10, then the distance is greater than or equal to 10 for 10 steps.

For 20 steps, if the distance between the and the is smaller than 6, then the car2 turns to the right

For 20 steps, the difference of each position of and should be less than or equal to 0.5.

4 Railroad

This is a system of modeling a crossing barrier controller and a train on a track. There is a circular railroad track and there is a crossing barrier on the track. The size of the track is 100. A train is going around and around the track. There are four modes for the train. The barrier opens when the train approaches to the barrier and closes when the train passes the barrier. The variable is angular between ground and the barrier. If is 0, then the barrier closes completely. If is 90, then the barrier opens completely. The angular velocity of barrier is constant and the value of velocity is different according to train’s mode. Initial relative distance of the train to the barrier is greater than or equal to 60 and less than or equal to 70, and the mode of train( is ‘Far’. If the relative distance is greater than or equal to 40 and smaller than 50, then the mode is changed to ‘Approach’. If the relative distance is greater than or equal to 20 and smaller than 30, then the mode is changed to ‘Near’. If the relative distance is greater than or equal to -5 and smaller than or equal to 0, then the mode is changed to ‘Past’. If the relative distance is greater than or equal to -10 and smaller than or equal to -5, then the mode is changed to ‘Approach’ and the relative distance is updated to ‘. The values of variables are changed according to the ODEs:

1. Linear
2. Nonlinear

where is the current degree between the crossing bar and ground. We designed following STL formulas for the model:

The position of train is less than or equal to zero within 40 steps.

For 40 steps, if the bar is almost open, then train will cross the bar.

If the angle between bar and ground is bigger than 80 degrees, the degree will be smaller than 40 degrees in the future.

For 40 steps, if the train pass the crossing bar, then it stays its mode during 7 steps.

5 Battery

There are two of fully charged batteries, and a control system switches load between these batteries to achieve longer lifetime out of the batteries. There are three modes for each battery. Initially, the total energy of is 8.5 and is 7.5. The mode of each battery is ‘on’. If the total energy of battery is smaller than specific value, the battery’s mode becomes dead. Otherwise, the mode can be either ‘on’ or ‘off’. The values of variable are changed according to the ODEs:

1. Linear
2. Nonlinear

with variable is its kinetic energy, variable is its total charge, and constant is its threshold. We designed following STL formulas for the model:

The total charge of the is less than or equal to 1 within 40 steps.

If two batteries are alive and used, then the system doesn’t make mode change during 17.5 steps.

For 40 steps, the total energy of both batteries are greater or equal to zero until they both dead.

For 20 steps, if the is dead, then there are only two possible ways when the mode is changed. One of them is the is alive and used. The other is both batteries are dead.

How to run benchmarks

We use python3.5 and z3 for running the benchmarks. For running all the benchmarks, just run “run.py” in python3.5. Then, the result data of each benchmark is generated. The filename of the result data is “Linear/Poly + Model name + Report + f1/f2/f3/f4(STL formula)”.