Project Modelling of Complex systems: Idp

First I needed to model the vocabulary following the LTC formalism, to do so, I defined the inertia axioms for GetPos and I defined these predicates:

-C\_GetPos(Time,Entity,Pos): True if and only if the position of entity at this time has been caused at this moment

-Cn\_GetPos(Time,Entity,Pos): True if and only if the position of entity at this time has been uncaused at this moment

To define those, I made my first helper function :

-NewPos(Pos,Move) : Pos returns the position of the tile where you would land if you moved from the given Position with the given Move.

This does not take other entities in count, it just maps to another tile. I used this function a lot while defining other predicates, such as the predicates described above. This is a partial function as you can’t always move in any direction because of the length of the board.

Another helper predicate is

-Occupies(Time,Entity,Pos) is true if and only if the given entity occupies the tile at a given time. This means that that position is not free. A tile being free is denoted by the following extra predicate:

-isFree(Time,Pos) is true if and only if the given tile is not occupied by any car or obstacle at the given time.

These 2 predicates entirely describe the movement freedom on the board, and is used intensively in the definition of CanMove.

The theories in the Verifications have been correctly implemented, however, I had a lot of problems trying to use other procedures such as isinvariant, so I used isSat everytime. I know however that showing that a theory and structure are satisfiable is not enough to prove that the verifications always hold.

Hours worked on Project: 15.