Max O'Cull
0028687211
CS 471
Homework 4
02/24/19

1. PDDL vs. Situation Calculus
   a. Convert the cake problem and the cargo problem from PDDL to situation calculus.
      1. Cake Problem

$$\exists c\, Have(c, s_0)$$
$$\exists c_1 \exists c_2\, Have(c_1, s_g) \wedge Eaten(c_2, s_g)$$
$$\forall c\, \forall s\, Have(c, s) \Rightarrow Poss(Eat(c), s)$$
$$\forall c\, \forall s\, \neg Have(c, s) \Rightarrow Poss(Bake(c), s)$$
$$\forall a\, \forall c\, \forall s\, Poss(a, s) \Rightarrow Eaten(c, Result(a, s)) \Leftrightarrow (a = Eat(c)) \vee (Eaten(c, s) \wedge \neg(a = Eat(c)))$$
$$\forall a\, \forall c\, \forall s\, Poss(a, s) \Rightarrow Have(c, Result(a, s)) \Leftrightarrow (a = Bake(c)) \vee (Have(c, s) \wedge \neg(a = Bake(c)))$$

      2. Cargo Problem

$$\exists c_1 \exists c_2 \exists p_1 \exists p_2\, At(c_1, SFO, s_0) \wedge At(c_2, JFK, s_0) \wedge At(p_1, SFO, s_0) \wedge At(p_2, JFK, s_0)$$
$$\exists c_1 \exists c_2\, At(c_1, JFK, s_g) \wedge At(c_2, SFO, s_g)$$
$$\forall c\, \forall p\, \forall a\, \forall s\, At(c, a, s) \wedge At(p, a, s) \Rightarrow Poss(Load(c, p, a), s)$$
$$\forall c\, \forall p\, \forall a\, \forall s\, \mathrm{In}(c, p, s) \wedge At(p, a, s) \Rightarrow Poss(Unload(c, p, a), s)$$
$$\forall p\, \forall f\, \forall t\, \forall s\, At(p, f, s) \Rightarrow Poss(Fly(p, f, t), s)$$
$$\forall a\, \forall c\, \forall p\, \forall i\, \forall s\, Poss(a, s) \Rightarrow \mathrm{In}(c, p, Result(a, s)) \Leftrightarrow (a = Load(c, p, i)) \vee (\mathrm{In}(c, p, s) \wedge \neg(a = Load(c, p, i)))$$
$$\forall a\, \forall c\, \forall p\, \forall i\, \forall s\, Poss(a, s) \Rightarrow At(c, i, Result(a, s)) \Leftrightarrow (a = Unload(c, p, i)) \vee (At(c, i, s) \wedge \neg(a = Unload(c, p, i)))$$
$$\forall a\, \forall p\, \forall t\, \forall f\, \forall s\, Poss(a, s) \Rightarrow At(p, t, Result(a, s)) \Leftrightarrow (a = Fly(p, f, t)) \vee (At(p, t, s) \wedge \neg(a = Fly(p, f, t)))$$

   b. Discuss one advantage of each planning language for each problem.

      It is much easier for humans to read the original PDDL problem than it is to read situational calculus.

      It is likely easier to program problems into situational calculus given its rigidity and simple rules.

2. Inference in Propositional Logic
   a. Explain how to modify SATPLAN (Fig. 7.22) so that it only calls the SAT solver once.

      Modify SATPLAN so that instead of using a goal g_T for $T \in \{0, \ldots, T_{max}\}$, a compound goal written in disjunctive normal form $g_0 \vee g_1 \vee \ldots \vee g_{T_{max}}$ is used instead which is then passed to the SAT solver.

   b. Are there any new spurious solutions?

      It is possible that SATPLAN will reach the goal, leave it, then return back to the goal (possibly multiple times), but SATPLAN will not produce incorrect solutions.