

**Question 1.****part a.** Situation Calculus World Specification:

Action Possibilities (note, we do not need “Is Location” or “Is Room” predicates as “In” encodes those already):

$$\begin{aligned} & [\text{At}(\text{Shakey}, x, s) \wedge \exists r (\text{In}(x, r) \wedge \text{In}(y, r))] \Rightarrow \text{Poss}(\text{Go}(y), s) \\ & [\text{At}(\text{Shakey}, x, s) \wedge \text{At}(b, x, s) \wedge \text{IsBox}(b) \wedge \exists r (\text{In}(x, r) \wedge \text{In}(y, r))] \Rightarrow \text{Poss}(\text{Push}(b, y), s) \end{aligned}$$

Fluent Conditions:

$$\begin{aligned} \text{Poss}(a, s) \Rightarrow & \left( \text{At}(x, y, \text{Result}(a, s)) \Leftrightarrow \right. \\ & \left[ (a = \text{Push}(x, y)) \vee (x = \text{Shakey} \wedge \exists b a = \text{Push}(b, y)) \vee (x = \text{Shakey} \wedge a = \text{Go}(y)) \right. \\ & \left. \left. \vee (\text{At}(x, y, s) \wedge x \neq \text{Shakey} \wedge \neg \exists z a = \text{Push}(x, z)) \right] \right) \end{aligned}$$

Starting Knowledge:

$$\begin{aligned} & \text{At}(\text{Shakey}, x, S_0) \Leftrightarrow (x = P_0) \\ & \text{At}(B_1, x, S_0) \Leftrightarrow (x = P_1) \\ & \text{At}(B_2, x, S_0) \Leftrightarrow (x = P_2) \\ & (x \neq \text{Shakey} \wedge x \neq B_1 \wedge x \neq B_2) \Rightarrow \neg \text{At}(x, y, S_0) \\ & \text{In}(x, r) \Leftrightarrow \left[ (x = P_0 \wedge r = R_4) \vee \right. \\ & \quad (x = P_1 \wedge r = R_3) \vee \\ & \quad (x = P_2 \wedge r = R_2) \vee \\ & \quad (x = P_3 \wedge r = R_1) \vee \\ & \quad (x = D_1 \wedge (r = R_1 \vee r = R_4)) \vee \\ & \quad (x = D_2 \wedge (r = R_2 \vee r = R_4)) \vee \\ & \quad \left. (x = D_3 \wedge (r = R_3 \vee r = R_4)) \right] \\ & \text{IsBox}(b) \Leftrightarrow (b = B_1 \vee b = B_2) \end{aligned}$$

**part b.** Situation Calculus Goal:

$$\text{At}(B_1, P_3, g)$$

**part c.** STRIPS Specification:

Actions:

Action(Go(x,y,r),

PRECOND: At(SHAKEY,x)  $\wedge$  In(x,r)  $\wedge$  In(y,r)  $\wedge$  NotSameLoc(x,y)EFFECT:  $\neg$ At(SHAKEY,x)  $\wedge$  At(SHAKEY,y)

Action(Push(b,x,y,r),

PRECOND: IsBox(b)  $\wedge$  At(SHAKEY,x)  $\wedge$  At(b,x)  $\wedge$  In(x,r)  $\wedge$  In(y,r)  $\wedge$  NotSameLoc(x,y)EFFECT:  $\neg$ At(SHAKEY,x)  $\wedge$   $\neg$ At(b,x)  $\wedge$  At(SHAKEY,y)  $\wedge$  At(b,y)

Initial Conditions:

At(SHAKEY,P0)  $\wedge$  At(B1,P1)  $\wedge$  At(B2,P2)  $\wedge$ In(P0,ROOM4)  $\wedge$  In(P1,ROOM3)  $\wedge$  In(P2,ROOM2)  $\wedge$  In(P3,ROOM1)  $\wedge$ In(D1,ROOM1)  $\wedge$  In(D1,ROOM4)  $\wedge$ In(D2,ROOM2)  $\wedge$  In(D2,ROOM4)  $\wedge$ In(D3,ROOM3)  $\wedge$  In(D3,ROOM4)  $\wedge$ IsBox(B1)  $\wedge$  IsBox(B2)  $\wedge$ NotSameLoc(P0,P1)  $\wedge$  NotSameLoc(P0,P2)  $\wedge$  NotSameLoc(P0,P3)  $\wedge$ NotSameLoc(P0,D1)  $\wedge$  NotSameLoc(P0,D2)  $\wedge$  NotSameLoc(P0,D3)  $\wedge$ NotSameLoc(P1,P0)  $\wedge$  NotSameLoc(P1,P2)  $\wedge$  NotSameLoc(P1,P3)  $\wedge$ NotSameLoc(P1,D1)  $\wedge$  NotSameLoc(P1,D2)  $\wedge$  NotSameLoc(P1,D3)  $\wedge$ NotSameLoc(P2,P0)  $\wedge$  NotSameLoc(P2,P1)  $\wedge$  NotSameLoc(P2,P3)  $\wedge$ NotSameLoc(P2,D1)  $\wedge$  NotSameLoc(P2,D2)  $\wedge$  NotSameLoc(P2,D3)  $\wedge$ NotSameLoc(P3,P0)  $\wedge$  NotSameLoc(P3,P1)  $\wedge$  NotSameLoc(P3,P2)  $\wedge$ NotSameLoc(P3,D1)  $\wedge$  NotSameLoc(P3,D2)  $\wedge$  NotSameLoc(P3,D3)  $\wedge$ NotSameLoc(D1,P0)  $\wedge$  NotSameLoc(D1,P1)  $\wedge$  NotSameLoc(D1,P2)  $\wedge$  NotSameLoc(D1,P3)  $\wedge$ NotSameLoc(D1,D2)  $\wedge$  NotSameLoc(D1,D2)  $\wedge$ NotSameLoc(D2,P0)  $\wedge$  NotSameLoc(D2,P1)  $\wedge$  NotSameLoc(D2,P2)  $\wedge$  NotSameLoc(D2,P3)  $\wedge$ NotSameLoc(D2,D2)  $\wedge$  NotSameLoc(D2,D2)  $\wedge$ NotSameLoc(D3,P0)  $\wedge$  NotSameLoc(D3,P1)  $\wedge$  NotSameLoc(D3,P2)  $\wedge$  NotSameLoc(D3,P3)  $\wedge$ NotSameLoc(D3,D2)  $\wedge$  NotSameLoc(D3,D2)**part d.** STRIPS Goal:

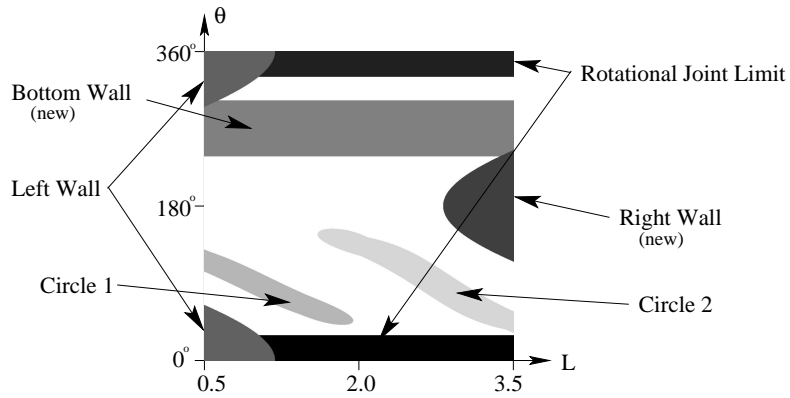
At(B2,R1)

**part e.** The goal of “Box 1 being in any room other than Room 3” would be simple in situation calculus. In STRIPS it could not be done with the above formulation. Yet, if we kept a “NotAt” predicate and updated it as well, it could be formulated.

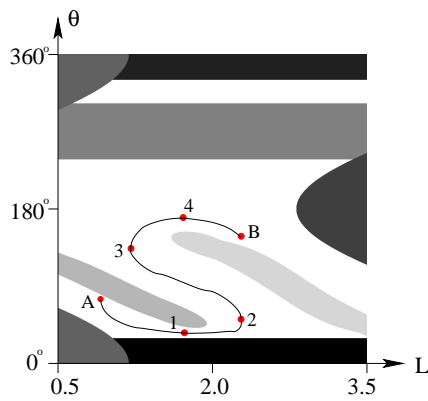
The goal of “Box 1 being in either Room 1 or Room 2” is also simple in situation calculus and not easily possible in STRIPS. But, yet again, it could be done in Strips if we reformulated the problem to keep track of “AtOneOf.” However, at that point things become very difficult and we are pretty much solving the problem for the STRIPS planner.

**Question 2.**

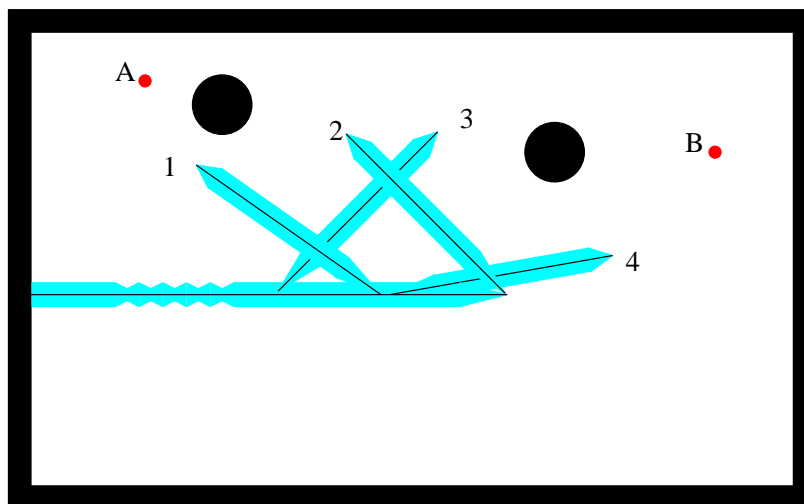
**parts a and b:**



**parts c and d:**



**part e:**



**Question 3.**

Code available on the course website. Results of test runs:

Museum File	Average correct	Std. of number correct	Ave. distance error	Std. of ave. distance error
museum1.plan	8.3	2.45	0.435	0.728
museum2.plan	7.45	2.78	0.465	0.624
museum3.plan	1.1	1.95	3.40	0.958
museum4.plan	29.7	12.8	3.67	8.78

For comparison, we compare it to the random guard provided:

Museum File	Average correct	Std. of number correct	Ave. distance error	Std. of ave. distance error
museum1.plan	0.8	0.75	2.58	0.406
museum2.plan	0.65	0.853	2.53	0.527
museum3.plan	0.7	0.9	2.72	0.429
museum4.plan	0.05	0.218	29.3	6.19

All of the numbers are better for our “intelligent” agent. Even the numbers for museum3 are better! In museum3, the sensors provided no information! Yet, the smart agent could reason where the robber would be more likely to be found (due to starting position and how he tends to move) and so could do better even with no sensor information.