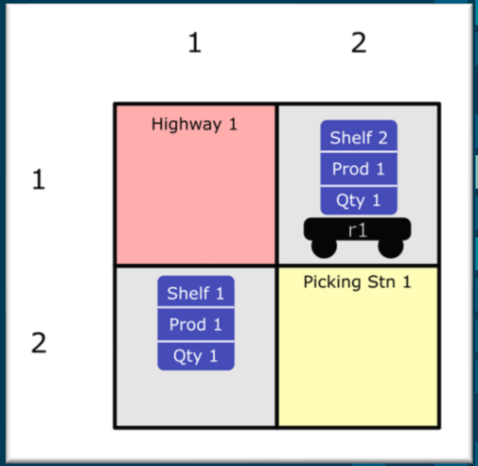
CSE 579 Group Project: Automated Warehouse Scenario

Riley Tallman, Kirk Manegold,
Matthew Jibben
12/8/2019

Introduction

- Automated warehouse - dynamic and more interesting
- The challenge: given a warehouse full of robots, shelves, products, highways, and picking stations, use ASP to fulfil customer orders in minimal time
- Output: robot moves and actions
- Ex. `occurs(object(robot,1),move(0,1),6)`
- Much more behind-the-scenes work with shelves, orders, highways, and deliveries



Approach

- warehouse.lp
  - Shelf, order, and product logic
  - End goal
  - Minimizing moves and time
- robotPhysics.lp
  - Robot movement
  - Moving shelves
  - Fulfilling orders

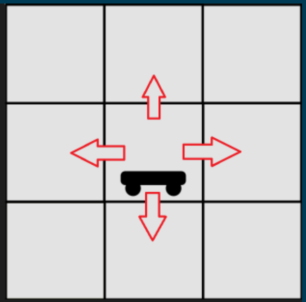
Robot Movement

Robots can move to adjacent spaces, or stand still

```
% we can "move" a robot to the same location,  (commonsense law of inertia)
{object(robot,N,X,Y,T+1)} :- object(robot,N,X,Y,T), T = 0..t-1.

% effect of moving a robot. The first move is at time step 1, not 0.
object(robot,N,X + DX,Y + DY, T) :- occurs(object(robot,N), move(DX,DY),T), object(robot,N,X,Y,T-1).

% these rules allow moves to be chosen. Can only move onto a valid node
actionSpace(-1;1).
{occurs(object(robot,N), move(DX, 0), T+1)}1 :- actionSpace(DX), object(robot,N,X,Y,T), object(node,NodeN,X+DX,Y), T = 0..t-1.
{occurs(object(robot,N), move(0, DY), T+1)}1 :- actionSpace(DY), object(robot,N,X,Y,T), object(node,NodeN,X,Y+DY), T = 0..t-1.
```



Robots can pick up shelves

```
% robots can pick up shelves if they are at the same location
{occurs(object(robot,RN),pickup,T)} :- object(shelf,SN,X,Y,onRobot(0),T-1), object(robot,RN,X,Y,T-1).
% effect of picking up a shelf
object(shelf,SN,X,Y,onRobot(1),T) :- occurs(object(robot,RN),pickup,T), object(robot,RN,X,Y,T-1), object(shelf,SN,X,Y,onRobot(0),T-1).
```

Robot Actions

Robots can deliver product from shelves

```
{occurs(object(robot,RobotN),deliver(OrderN,ProductN,UnitsToDeliver),T)} :- object(robot,RobotN,X,Y,T-1),
object(pickingStation,PickingStationN,X,Y), object(shelf,ShelfN,X,Y,onRobot(1),T-1),    % the robot must be at the pickingstation location carrying a shelf
object(order,OrderN,PickingStationN,fulfilled(0,T-1)),   % the order must be unfulfilled and we need the right pickingstation
object(order,OrderN,ProductN,UnitsNeeded,T-1),    % the needed number of units for this product on this order
object(product,ProductN,ShelfN,UnitsOnShelf,T-1),      % the product number must match,
UnitsNeeded > 0, UnitsOnShelf > 0,
(UnitsOnShelf - UnitsNeeded) >= 0,
UnitsToDeliver = UnitsNeeded.
```

Robots can put down shelves

```
% robots have the ability to put down shelves
{occurs(object(robot,RN),putdown,T)} :- object(shelf,SN,X,Y,onRobot(1),T-1), object(robot,RN,X,Y,T-1).
% effect of putting down a shelf
object(shelf,SN,X,Y,onRobot(0),T) :- occurs(object(robot,RN),putdown,T), object(robot,RN,X,Y,T-1),
object(shelf,SN,X,Y,onRobot(1),T-1).
```

End goal

At the final timestep, all orders must be fulfilled

```
% end goal: have all orders filled
:- not object(order,OrderN,PickingStationN,fulfilled(1,t)), init(object(order,OrderN),value(pickingStation,PickingStationN)).
```

Minimize timeCount to get the optimal plan

```
timeCount(N) :- #count{T: occurs(object(robot,RN), Action, T)} = N.
#minimize{T: timeCount(T)}.
```

Results

Satisfying all 5 instances!

1. Find minimal time

```
#const t=13.
#minimize{T: timeCount(T)}.
```

```
(clingo) clingo warehouse.lp robotPhysics.lp inst5.lp
clingo version 5.4.0
Reading from warehouse.lp ...
Solving...
Answer: 1

Optimization: 12
Answer: 2

Optimization: 11
Answer: 3

Optimization: 10
Answer: 4

Optimization: 9
Answer: 5

Optimization: 8
Answer: 6

Optimization: 7
Answer: 7

Optimization: 6
OPTIMUM FOUND

Models       : 7
  Optimum    : yes
Optimization : 6
Calls        : 1
Time         : 16.549s (Solving: 14.62s 1st Model: 0.44s Unsat: 13.77s)
CPU Time     : 15.453s
```

Results Cont.

Satisfying all 5 instances! 🎉

2. Find minimal actions for a specific time

```
#const t=6.
#minimize{X: actionCount(X)}.
```

```
(clingo) clingo warehouse.lp robotPhysics.lp inst5.lp
clingo version 5.4.0
Reading from warehouse.lp ...
Solving...
Answer: 1
occurs(object(robot,2),move(-1,0),1) occurs(object(robot,1),move(-1,0),1)
 occurs(object(robot,1),move(-1,0),2) occurs(object(robot,2),move(0,1),3)
 occurs(object(robot,2),move(0,-1),5) occurs(object(robot,1),move(-1,0),5
) occurs(object(robot,2),pickup,2) occurs(object(robot,1),pickup,4) occur
s(object(robot,2),deliver(1,3,4),4) occurs(object(robot,1),deliver(1,1,1)
,6)
Optimization: 10
OPTIMUM FOUND

Models       : 1
  Optimum    : yes
Optimization : 10
Calls        : 1
Time         : 0.112s (Solving: 0.02s 1st Model: 0.01s Unsat: 0.00s)
CPU Time     : 0.094s
```

Conclusion

- The team verified all results with Inkscape
- We are happy that all instances are satisfied
- Start earlier
- Become more confident with git
- `git diff robotPhysics.lp`