Emma Posega Rappleye and Ju Yun Kim
1/17/2018
Assignment 1
CS321 - Artificial Intelligence

Introduction

The goal is to create an agent that rotates through a floor of 12 offices, one office at a time, and adjusts the temperature and the humidity intelligently so that the employees can be both comfortable and efficient. To maintain comfort, we want the robot to only be able to raise or lower the settings by 1 in each visit.

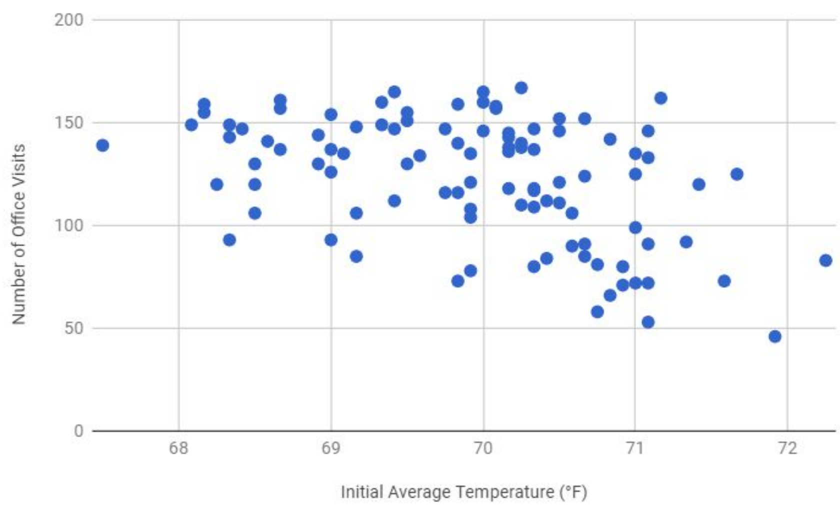
In this write-up, we will assume that HeatMiser is an agent that exists in physical space, and describe the environment accordingly. It could also be a non-physical agent that exists as an online program with access to office controls, in which case some of our PEAS items--especially environmental and actuators--and our simulation improvement suggestions could be less relevant.

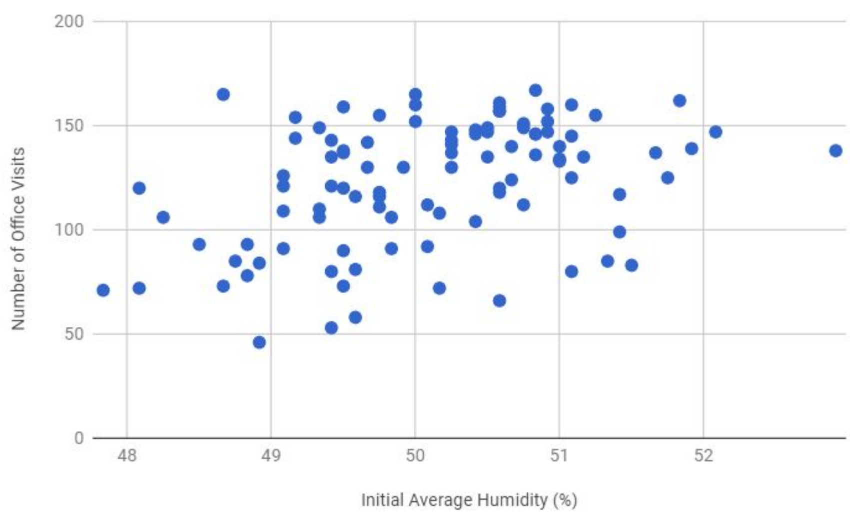
PEAS

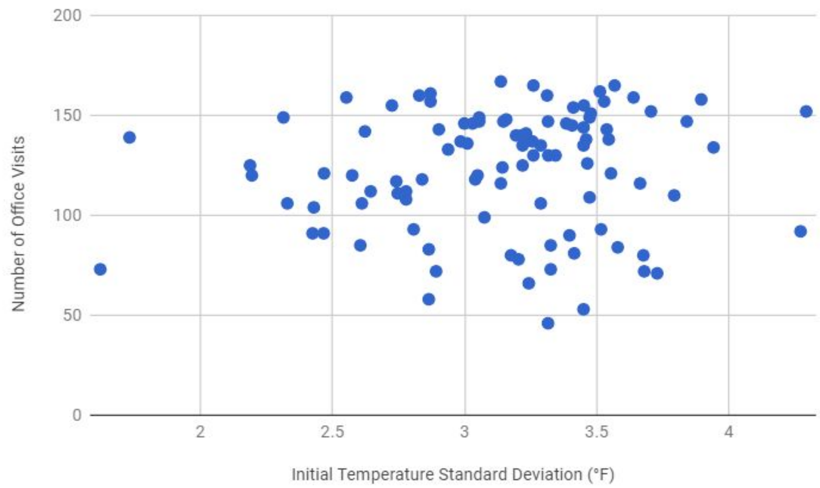
Performance measure	• The number of rounds across the floor until the average floor temperature is 72 degrees Fahrenheit, the average floor humidity is 47%, and the standard deviation from these among offices is 1.5 and 1.75, respectively
Environment	• A series of 12 offices each of which can have an initial temperature between 65 and 75 degrees and an initial humidity between 45-55 percent • The hallway
Actuators	• The tool (either internal via WiFi or external via physical interaction with the control panel) that allows HeatMiser to adjust the temperature or humidity by 1 degree • The locomotion tools that allow HeatMiser to move from room to room
Sensors	HeatMiser has access to the current office's temperature and humidity as well as the floor's averages and standard deviations for temperature and humidity This can be accomplished either through: • a thermometer and humidity sensor attached to HeatMiser, • a camera attached to HeatMiser, and thermostats available in every room and in the hallway that HeatMiser could read, or • thermometer and humidity sensors in the room that upload their data to HeatMiser

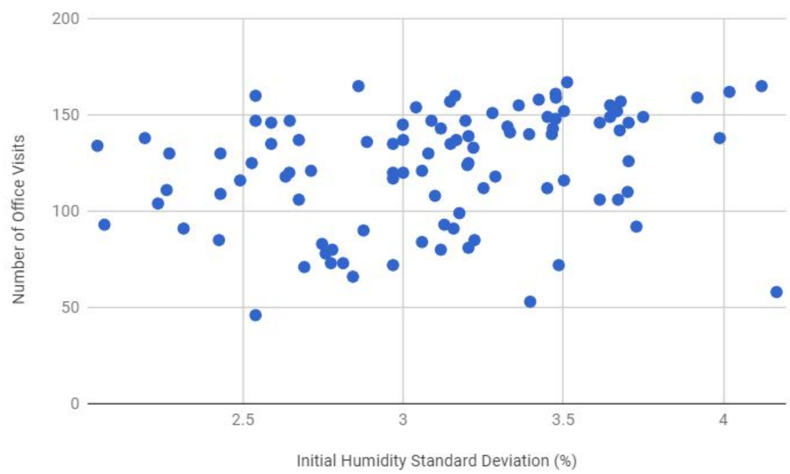
	Additionally, to navigate, find the temp/humidity controls (if not internal and controlled via WiFi), HeatMiser could use: ● a camera ● a GPS or location sensor

Analysis









For context, our algorithm is this: when HeatMiser visits an office, if the office's temperature is more different from the desired 72 degrees than the humidity is from the desired 47%, it adjusts the temperature to be 1 degree closer to the goal. Otherwise, it adjusts the humidity to be 1% closer to the goal. When all the metrics are in the desired range, it stops running.

There was an observed relationship between the number of office visits and the initial average temperature and humidity of the floor. Generally, lower average initial temperatures led to more visits, which is expected since the goal of 72F is towards the upper bound of allowed temperatures. Similarly, we found that higher initial average humidity led to more visits, since the goal of 47% humidity is very close to the lower bound of allowed humidities. From these findings, we can tell that the initial, randomly-assigned conditions in offices played a large part in how successful our algorithm was.

Additionally, there was a positive relationship between office visits and initial standard deviations for both measurements. Again, this is expected for our algorithm since it tries to homogenize the temperatures and humidities across all of the offices. However, the relationship is not too strong, which may be attributed to the fact that moderately low and moderately high values may be close together enough that the standard deviation specs are met quickly.

Over 100 runs of the simulation, our HeatMiser had an average of 122.92 office visits with a standard deviation of 29.84. This and the graphs show that the agent's performance is fairly dependent on the initial state, but the aggregated results seem to be consistent.

HeatMiser's performance should scale linearly for a simulation environment of 100 floors, assuming that each floor's climate is independent of the others. The agent would simply average around 123 office visits per floor 100 times. If the simulation environment was instead a floor with 100 offices, the relative performance may increase. On a floor with many offices, a change in the value of one office will have a smaller impact on the floor averages, which will reduce the occurrences of overshooting the target metrics. This ability to fine tune the global state will allow fewer visits per office than in smaller floors.

Future Considerations

Performance/Behavior improvement

The current approach naively alters the temperature or humidity if they are deviant from the goal. This leaves a lot of room for improvement, since moving away from the goal or moving towards the average may be an effective strategy. For instance, a 72 degree room would not be adjusted by our current algorithm if the hallway average is 75 degrees, but an improved technique might be to make HeatMiser adjust the room to 71 degrees, cooling down the hallway average to be closer to the goal temperature.

We hypothesize that if HeatMiser were to calculate and base its decision on the hypothetical effects of the 5 possible actions (raise temp, lower temp, raise humidity, lower humidity, do nothing) on the averages, it would be able to reduce the number of office visits needed to reach the goal. Without refining this approach, however, HeatMiser may focus too much on achieving an average that is overall perfect, but has unacceptable standard deviations.

While the performance measures are fine, they are still relatively simplistic and don't take into account how HeatMiser affects employees--just how quickly it affects the average conditions of the hallway. Implementing performance measures like measuring how large of a change HeatMiser ended up causing in each office (did HeatMiser eventually turn the temperature of a cold office all the way up to 72 when it could have left the temperature at 70 and still arrived at a solution?), maximizing its amount of "do nothings," or otherwise taking its effect on employees into account could help us arrive at a more refined, realistic solution.

Simulation improvement

While the current simulation provides an easy and straightforward way to model the temperature and humidity changes, if HeatMiser were put into a real-world situation it would undoubtedly encounter more obstacles from external forces than we have modeled here. For instance, we assume that HeatMiser's changes are permanent, but chilly or grumpy employees may readjust their temperature or humidity after HeatMiser has passed through--especially in the time it takes HeatMiser to visit all 12 offices once or twice. Making some of the temperatures or humidities randomly adjust after HeatMiser begins his visits would allow us to see how HeatMiser's algorithm holds up to human conditions.

Additionally, to get a really involved simulation, we must take into account that HeatMiser's actions may differ in impact depending on the current "state" of the hallway, including factors like:

- who has an open window
- who has an open door, or a door that is frequently used
- how many people are standing in the hallway and raising the overall temperature

For instance, if one office has closed windows and a closed door that is rarely opened, its impact on the hallway's averages may be less significant than other offices. In the interest of preserving a sense of individual autonomy, HeatMiser's algorithm may be changed to only adjust these low-impact rooms only when absolutely necessary.

Finally, this simulation allows us to abstract HeatMiser to a point where we don't account for its physical presence in the rooms and hallways (assuming HeatMiser is, in fact, physical). While this is fine to let us come up with a basic algorithm for changing the temperature and humidity, we don't take into consideration issues like HeatMiser interrupting meetings by entering an office every five minutes, dealing with locked doors or getting bumped in a crowded hallway. For a simulation that takes some of these physical concerns into account, we could implement a grid-based system, or even just a "locked door/important meeting" event.

New PEAS based on our suggestions

Performance measure	• The number of rounds across the floor until the average floor temperature is 72 degrees Fahrenheit, the average floor humidity is 47%, and the standard deviation from these among offices is 1.5 and 1.75, respectively • Maximizing the amount of "do nothing" actions • Minimizing disruption for the workers
Environment	• A series of 12 offices each of which can have an initial temperature between 65 and 75 degrees and an initial humidity between 45-55 percent • The doors and windows of each office • The hallway • Employees

<table>
<tr><td>Actuators</td><td>The tool (either internal via WiFi or external via physical interaction with the control panel) that allows HeatMiser to adjust the temperature or humidity by 1 degree Fahrenheit
The locomotion tools that allow HeatMiser to move from room to room</td></tr>
<tr><td>Sensors</td><td>HeatMiser has access to the current office’s temperature and humidity as well as the floor’s averages and standard deviations for temperature and humidity This can be accomplished either through:
a thermometer and humidity sensor attached to HeatMiser,
a camera attached to HeatMiser, and thermostats available in every room and in the hallway that HeatMiser could read, or
thermometer and humidity sensors in the room that upload their data to HeatMiser
Additionally, to navigate, find the temp/humidity controls (if not internal and controlled via WiFi), HeatMiser could use:
a camera
a GPS or location sensor</td></tr>
</table>