

Spike Outcome Report

Number: 3

Spike Title: Goal Oriented Behaviour

Personal: Michael Williams (7668481)

Goals:

Create a simple goal insistence (SGI) model simulation of goal-oriented behaviour (GOB) that demonstrates the both the effectiveness and the limitations of the technique.

Technologies, Tools, and Resources used:

- Python IDE
- Sample Lab Code
- Lecture Material

Tasks undertaken:

1. The first task in getting this spike started was read over the appropriate guide material, in this case the lecture notes, and get an understanding of what needs to be accomplished.
2. Then, a basic class diagram needs to be drawn up describing the interactions and responsibilities between the different parts of the program.

This was done, by morphing the code that was shown in the lecture material, into to an object-oriented manner.

3. After the diagram was constructed, the design was implemented in the code using Python.
4. When the implementation is finished, it is time to test it out. Some test goals and actions were created to do this.
5. Finally, the program needs to be evaluated.

What we found out:

We found out that using simple goal-oriented behaviour is a fantastic method for simulating simple choices based on priority. This is due primarily to its overall simplicity and effectiveness. Below is an example output for this implementation:

```

"C:\Program Files\Anaconda3\python.exe" "C:/Users/willi/Google Drive/Swinburne/COS30002/Spik
Goal with highest insistance is:Have fun!
Chosen action is Party!!
Goal with highest insistance is:Sleep
Chosen action is Go to bed
Goal with highest insistance is:Have fun!
Chosen action is Party!!
Goal with highest insistance is:Sleep
Chosen action is Go to bed
Goal with highest insistance is:Get a high distinction
Chosen action is Study...
Goal with highest insistance is:Sleep
Chosen action is Go to bed
Goal with highest insistance is:Have fun!
Chosen action is Party!!
Goal with highest insistance is:Sleep
Chosen action is Go to bed
Name:Get a high distinction,Insistance:0
Name:Have fun!,Insistance:0
Name:Sleep,Insistance:0

Process finished with exit code 0

```

We chose to implement this in an object-oriented manner and found that the overall solution really is quite nice and easy to follow. The main code boils down to simply:

```

while not goals_fulfilled:

    #find goal with highest insistance
    current_goal = max(goals, key=lambda item: item.GetGoalInsistance())
    print("Goal with highest insistance is:{}".format(current_goal.GetGoalName()))

    #find all the actions that will satisfy this goal
    actions_for_goal = list(filter(lambda item:item.AlleviatesGoal(current_goal),actions))

    #find action that will satisfy the best - by having the lowest resulting discontent
    if len(actions_for_goal) > 0:
        current_action = min(actions_for_goal, key=lambda item: item.GetDiscontent())
        print("Chosen action is {}".format(current_action.GetActionDescription()))

        #apply the action
        current_action.PerformAction()

    #determine if there are more goals to fulfil
    goals_fulfilled = True

    for goal in goals:
        if goal.GetGoalInsistance()> 0:
            goals_fulfilled = False

```

We have learned that this sort of a system does have its limitations. Through our experimenting we have found out that once the system is unable to satisfy itself, it can be quite difficult to manage the goal insistance to ensure appropriate action is always taken in the AI's best interest. A simple example of this is a soldier on a battlefield, who has lost half of their health. At this point I would expect "health" to have the highest insistance and then "take cover" followed by "attack/shoot". In this situation we may not have access to any actions that will improve the health situation, so what do we do?

In this sort of a system, with external influences and factors, it becomes much more difficult to maintain a functioning AI that will make appropriate decisions.

Find below an example where this does not work well:

```
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
Goal with highest insistance is:Sleep
```

```
Process finished with exit code 1
```

This has simulated the situation where the AI's most insistent goal is "Sleep" but there is not bed, and thus we are stuck in a loop constantly trying to sleep. Obviously, this can be easily rectified in several ways (i.e. If no action for current goal, check for the next goal) but that is beyond the scope of this task.