

IE 588 Project Report

January 8, 2018

Mine Melodi Caliskan 2015705009

1 Party

1.1 Problem Description

Students living in Hisarustu, Istanbul have limited free time and they want to spend their Friday night at the one and only pub, La Liberta Plus, which is a very small. Therefore, it's very difficult to find a place to hang out with their friends. If the pub is too crowded when a student enters, she feels happy however if it is more than that and also if she decides to stay home she feels upset. Our task is to capture the effects of students types to overall happiness and themselves. It's important to remark that if all students use the same decision rule to decide whether to go or not, it's unavoidable to fail since in this sense the bar would be either full of people or full-empty. Therefore, there should be a group of decision rules and students must learn from their history in order to choose the following decision rule. Every students' action results from the other students' actions and also themselves. Our task is to answer the question of whether or not the equal happiness with higher than a certain degree of total happiness could be provided. Therefore it is interesting to analyze.

Modeling Questions

- How would overall happiness change depending on the type of the students?
- How would groups effect each others happiness?
- Which agent type results in better in the long-run?

Method Justification

We can't predict easily when to go to party since there is highly complex interaction effect, however components of the system are simple, therefore it's better to use ABM for this problem.

- **the puzzle component:** If everybody prefers the same strategy to decide whether go bar or not, all of them would make themselves unhappy since the bar will be full or empty and our happiness criteria would not be satisfied. So that, there should be multiple strategies and these strategies outputs will be affected with each other and these interactions are complex to define directly.
- **dynamic nature:** This model is dynamic since the decision strategies depend on historical data which changes over time. People will choose a strategy based on their experience in advance and their experience change over time and they depend on themselves (autocorrelation) and the other people.(cross-correlation)

- **importance of agent heterogeneity:** Every person has different behaviors and characteristic features.

1.2 Agent Types

Copies

- Shape: Star
- At each run randomly picks an agent type and copies behavior of one-of them

Analyzers: * Shape: Human * Keeps History for 3 days * Has 3 strategies * Scores of strategies are kept for 3 days also.

Always: Always goes to party * Shape: Face Happy

Nevers: Never goes to party * Shape: X

Randomms: Goes party with probability %50 * Shape: Butterfly

1.2.1 Global Variables

- **number-of-people-at-bar:** the current number of people at the bar
- **outside-patches:** outside of the bar
- **party-patches :** inside of the bar
- **days-passed :** days since the beginning passed
- **total-total-happiness :** total happiness of all agents
- **total-agents:** total number of all type of agents
- **number-of-randomms:** number of agent type “randomms” which is determined by : total-agents- sum(number of other type of agents)
- **crowd-threshold:** threshold for the crowiness of the party of agents which is determined by: total-agents * crowd-percentage

Turtles (All) Variables

- happiness
- go?

1.2.2 Agent Variables

Agents are assumed to have local information, i.e when they go to the party they can count the total number of agents perfectly. However, before entering the bar they don't know.

Copies

- **prediction-copy**

They are assumed to have contact of all agents and learn their decision to decide to go to party or not.

Analyzers

- **history-analyzer**
- ***prediction-analyzer****
- **score-1 :** score of strategy-1

- **score-2** : score of strategy-2
- **score-3** : score of strategy-3
- **chosen** : corresponds to chosen strategies symbol as s1,s2,s3

Always * prediction-alway

Nevers

- **prediction-never**

Randomms

- **prediction-randomm**

1.3 Procedures

Setup

- total-agents are set to 100
- bar patches drew ask disk centered at the patch (0, 0) with color pink
- outside patches are in color blue
- turtles are created as follows: * number of user specified * for randomm type agents only, they are create with the number of left size of total agents 100.
- variables of turtles are initialized as follows: * for analyzers

```

;initialize history of 3 days' attendance randomly between 0 and number of total-agents
set history-analyzer (list (random total-agents) (random total-agents) (random total-agents))
show history-analyzer
set prediction-analyzer (random total-agents)
show prediction-analyzer
;initialize scores randomly (0 or 1) (failure or success) for 3 days
set score-1 (list (random 1) (random 1) (random 1))
set score-2 (list (random 1) (random 1) (random 1) )
set score-3 (list (random 1) (random 1) (random 1))]
```

- happiness of all turtles initialized randomly between 0,100

**** Go****

to go

```

; scale agensts color with respect to their happiness levels
ask turtles [ set color scale-color magenta happiness 0 100 ]
go-analyzers
go-always
go-nevers
go-randomms
go-copies
```