# Technical report on Agent-Based Model of drunks planning using Python

## Executive summary

Drunks leaving the bar alone and returning home at midnight is normal, but they may encounter a multitude of hazards on their way home, and in the worst-case scenario, they may be wounded or even die. An Agent-Based Model (ABM) is created in order to find and predict the potential locations where the drunks will go. In predetermined areas where a home and bars have already been built. The model would simulate intoxicated people moving around at random before they find their way home and plot their movements on a map. The ABM model is made up of five parts: 1. Set the canvas for map plotting by reading the location results. 2. On the canvas, draw a pub and a house. 3. Construct arbitrary points to represent drunks in bars and pass the drunks about at random, recording each movement at the same time. 4. Render a graph of the drunks' motions. 5. Create a text file that contains all of the steps. In short, the model performs as predicted and produces good performance.

## Introduction

### Agent-Based Model

Since its inception in the 1950s, the mathematical approach has been improved and used in a variety of disciplines. In geography, an agent-based model is used to solve problems involving place and time. As a result, complex variations can be quantified to aid analysis (Heppenstall et al., 2011). According to Gilbert (2020, p.1), agent-based modelling is “a computational method that enables a researcher to create, analyse, and experiment with models composed of agents that interact within the environment”. The agents in this analysis are drunks, and the environment is referred to as a map of various locations.

### Data

A 300\*300 text file is given in this analysis. Each value represents each pixel on the map and separated by commas, and the file is laid out at one line per image line from top left to bottom right. In the text file, pubs are assigned to 1 and each 10 in 10-250 stands for houses for every drunk. There are 25 drunks in total from 1 to 25 and each drunk has the corresponding house (10-250). Other features, such as soil, lane, houses, and so on, are represented by the number zero.

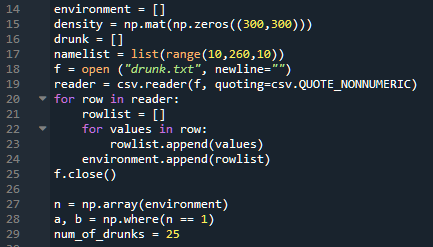
## Methodology and Results

### Algorithm

The basic logic includes 4 parts. Firstly, complete data preparation such as input and read data, initialize drunk class, set up empty lists. Second, generate a random position among pubs and allocate a value between 1 and 25 as the identify, then move the drunk randomly until they find their home. Thirdly, cycle the process until 25 drunks all find their home. Last, plot the map and output position text.

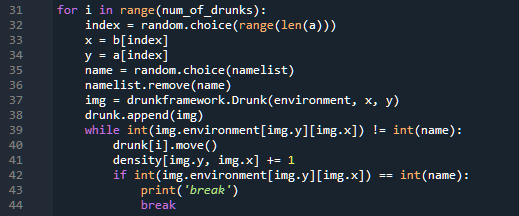
### Agent frame and basic variables

To realize the first part of algorithm, a Drunk class is built with three parameters: environment, x, y. x and y represent for the coordinate of the drunk and environment stands for the geographic information. Next, initialize each parameter as screenshot below:



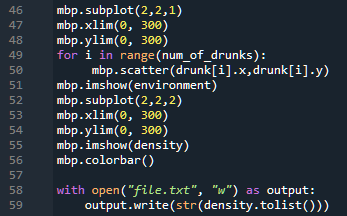
A matrix is set up with 300 rows and columns filling with zero that used to plot the density map (line 15). In line 18 to 25, the position text file is read and append into environment list which will be used later. In order to create initial points, the pub’s coordinate is split from environment list in line 27-29. It is developed through transforming environment list to array and allocating each x and y coordinate of pubs to list a/b. List and array are looks similar but have different uses in python. At first, there always an error message when I want to access positions of value 1. The problem got fixed through printing arrays in console, I found that the output from “np.where(n == 1)” has two list respectively are x, y coordinates.

### Functions and loops



Here is the main part of the model. In the first few lines, the function random choose an index value of list ‘a’ and assign the corresponding value as x, y coordinate. Similarly, random choose a value from ‘namelist’ and remove itself in the list. The drunk’s movement is realized by a ‘while’ loop, in each loop, drunk will move one step and count one to density. The loop will end until the environment consistent with the name value. A ‘print’ function is used to exam whether each loop end.

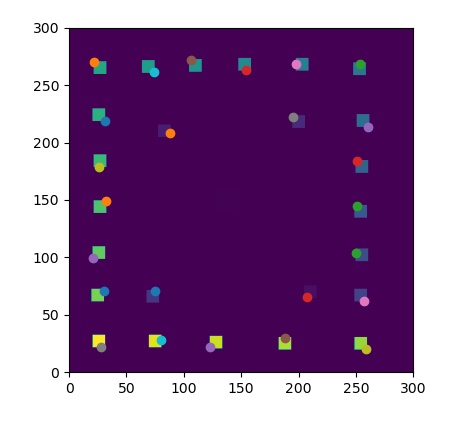
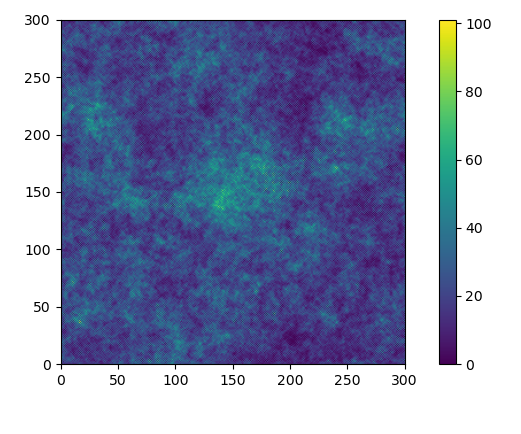
### Plot and output



There are two graphs been created in the model. One is used to display final position of each drunk and another plots the density map. In line 58-59, the density matrix is transformed to list and write to “file.txt”

## Discussion

The model runs as expected in final test, one of the results is shown below. Problems are all fixed during the developing process. For example, the initial plan is to use one map showing places and density at the same time, the question is that the position’s value may change when counting density. The problem finally fixed through plotting two maps. Nevertheless, some points still need notice. In the places map plotted, the pubs position is inconspicuous because the map using heat map in matplot library and the map render according to values. One solution is to change the pub value at first, while it may have more suitable way to plot the data which is not realized in this project. Another thing needs to consider is the iterations. In the project a while loop is used for iteration, situation exists that the loop running too many times. A limit should set up to avoid keeping iterating if the data is given in a larger area.

There also few points to improve the model to make it real. For example, let the drunk’s movement avoid pubs and turning round or let drunks back if they walk in a opposite direction too far.

## References

Heppenstall, A., Batty, M., Crooks, A., and See, L. 2011. Agent-Based Models of Geographical Systems. [Online]. Dordrecht: Springer Netherlands. [Accessed 16 May 2021]. Available from: <https://doi.org/10.1007/978-90-481-8927-4>

Gilbert, N. (2020). Agent-based models. [Online]. Los Angeles, CA: SAGE Publications, Inc. [Accessed 16 May 2021]. Available from: <https://methods.sagepub.com/book/agent-based-models-2e>