An Agent-based Approach to ARGESIM Comparison C16 'Restaurant Business Dynamics' with SeSAm

Patrick Herrler, University of Würzburg
pherrler@ki.informatik.uni-wuerzburg.de

Simulator: SeSAm (www.simsesam.de) is a Java™-based Multi-Agent Simulation Environment providing a generic environment for modelling and experimenting with agent-based simulation. It is focused on easy construction of complex models with the help of visual editors. Nevertheless SeSAm has the power of a programming language.

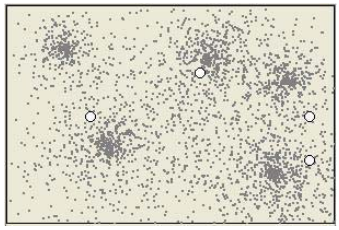

Figure 1: Runtime visualization of the model.

Model: The Model is implemented using an agent-based way with a timestepped schedule. Both restaurants and persons are agents. All of the agents' behaviour is defined by UML-like diagrams. Cells are defined as objects with a value *PeopleDensity* and a coordinate *CellUpperLeft* and are stored in a hashmap for fast access. All configurable parameters are part of a world, where agents "live" in. The world is some sort of main agent which has also a behaviour model. For each tick the world's actions are executed first, after that the order of agents is chosen randomly.

- *Space:* Everything takes place in a two-dimensional continuous map. At startup persons are randomly distributed according to the specifications. The people density of each cell is once calculated. The restaurant density is not stored but calculated every time a new restaurant is opened.
- *Time:* Each tick represents one day. After seven ticks all restaurants pay taxes to the government and open new restaurants or close down, according to financial situation and given probabilities.
- *Persons:* Every person keeps a list of restaurants in range. The list is updated every time the person wants to visit a restaurant. Persons try to go out for dinner even when their list is empty, because when it's time for dinner the list of restaurants in range is updated. Persons have a variable attribute *TimeBetweenDinner* which counts up to *NextTimeForDinner*, This variable is updated with a random number every time these two variables are equal.
- *Restaurants:* Each restaurant has a variable *Profit* accumulating the current week's profit. Another variable *TotalProfit* stores all accumulated profit.
- *Animation:* SeSAm provides a spatial map, where all agents can be observed visually. Since persons don't move, the only animation is the opening and closing of restaurants.
- *Experiments:* Experiment definitions allow simulating multiple scenarios varying start parameters.

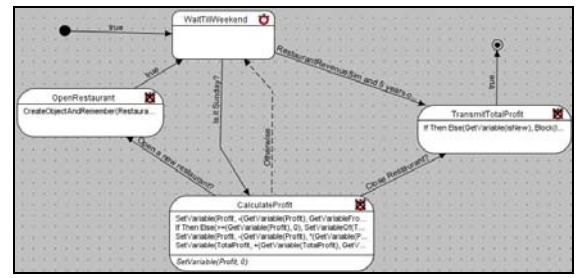

Figure 2 Visual modelling: restaurant's activity graph.

Task a: Time domain analysis. Evaluating the number of restaurants over 10 years, there is a mean of 5 restaurants, which is reached after about 100 days. From then there's no significant change in the average number of restaurants.

Task b: Tax income maximisation. Figure 3 shows the results of task b. The tax rate has been varied between 5% and 60%. Several simulation runs have been accomplished. The highest tax income is reached by a tax rate of 26%, indeed there's no real peak value, since the difference between the highest tax incomes is minimal. Very low and very high tax rates result in low tax income.

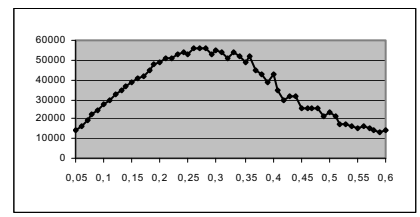

Figure 3: Tax income maximisation

Task c: Restaurants' revenue analysis. To find the best value for parameter *k*, the profit of new restaurants opened during the simulation has been summed up and divided by their number. Analysing the values it seems that a value of **0.5** is optimal. The worst value is 0, from 0.5 up to 6 the revenue slowly decreases.

A detailed description and model files can be found at www.simsesam.de/ArgesimC16

C16 Classification: Agent-based Approach

Simulator: SeSam v 1.9, 2004