We propose here a computational model of school segregation that is based on an ABM model of Schelling-type for residential segregation.

Model description

Our world consists of agents, resources, and locations. The locations are defined by a two dimensional grid with 100x100 locations, embedded on a torus. Each location can hold a household (an agent) or a school (a resource), but some locations are left empty to allow for movement across neighborhoods or schools.

The geographical space is divided into Voronoi cells. To generate the Voronoi diagram we use randomly assigned locations for the schools as so-called "generator" points. Each cell contains exactly one school, located at its generator point. Each school has a maximum number of students, equal to the number of locations in its corresponding cell. Because each cell actually defines the geographic region that is nearer to the focal school than to any other school, we call it school neighborhood or school 'catchment area'.

The model allows for two types of dynamical evolutions, which we call residential model and school model, respectively. In the residential model, each location defines a local residential neighborhood, modeled as a diamond-shaped neighborhood, approximating a circular region around the residential location. We consider that a location belongs to the local residential region if the distance between its location and the center of the local residential region is smaller or equal to the radius. When using a radius r = 1.5 location sizes, one finds the classical Moore neighborhood with exactly eight locations.

Each household belongs to one ethnic group and has exactly one child of school age. A household has an ethnic preference for the ethnic composition of its school or residential neighborhood. We use for the ethnic preference a single-peaked linear function that maps the local proportion of members of the same group on the attractiveness of the school or residential location for an agent. The ethnic preference is considered to be the same for all agents belonging to a specific ethnic group.

Agents can evaluate a school not only in terms of its ethnic composition, but also in terms of the distance between their residential location and the school's location. The distance preference (D) represents the "nearness" of the school, implemented as the mirror image of the geographical distance between residential location of the household and the school. The distance preference is modeled as a linear falling function with its maximum value 1 at the agent's residential location and minimum value 0 at a maximum distance, after which the function value remains null.

An agent's utility U for the school chosen is obtained from a Cobb-Douglas utility function. Technically,

$$U = P^{\alpha} * D^{(1-\alpha)}$$

where α is a parameter which controls how much weight is put on the ethnicity preference (P) relative to the distance preference (D).

In the residential model, agents can change where they live and can choose between a set of available residential locations with known location and local ethnic composition. Each evolution cycle, a given number of agents are tested whether their satisfaction with respect to their local neighborhood is smaller than a given threshold (T). If their level of satisfaction is smaller than the given threshold, the agents will move to the first available location where their level of satisfaction with respect to the location's local neighborhood is above a threshold. If no suitable location has been found within a maximum of tries, the agent will move to the first location where his satisfaction level becomes larger than his current satisfaction level.

In the school-choice model, the residential location is fixed and agents face the choice of sending their child to one out of a set of available schools with a known location and ethnic composition. The evolution of the system takes place in a similar way as for the residential model. If the agent's level of satisfaction is smaller than the given threshold, the agent will move its child to the first schools with an available spot, where their level of satisfaction with respect to the distance and ethnic composition is above a threshold. If no suitable school has been found within the available schools, the agent will move its child to the first school where his level of satisfaction increases.

Source code

The model is provided in the form a Visual Studio Project created using Microsoft Visual C++ Express 2010. The entry point for the console application is *schelling_model2.cpp*.

Input file

The input parameters and other information are specified in an ASCII input file. The name of the file (now: *agenti.txt*) can be changed in the same cpp file *schelling_model2.cpp*. For the case that the executable is started from the integrated environment of Visual Studio, the input file should be located in the project directory (…\schelling_model2\schelling_model2). Otherwise, the file should be located in the same folder as the executable or the path on disk should be given together with the filename. The input file is structured as a collection of parameters. Each parameter definition takes up two lines as follows: the first line contains the parameter's description and the second line contains the parameter's value.

The following input parameters can be edited:

- Parameters specifying the initial world composition
    - *generate setup 1 = Generate, 0 = READ:*
        - If the value on the next row is equal to 1, the program will generate the initial distribution.

        - If the value on the next row is equal to 0, the program will read the initial distribution from a user defined file containing an initial distribution of schools, agents and free locations.
    - *no of agents, no of white agents OR fileName (if 0):* The following two lines contain the information required to create/read the initial distribution. For the case that the input method is "Generate =1", the number of agents and the number of white agents must be provided. For the case that the initial distribution will be read, the name of the file as described at the previous point must be provided.
    - *Export setup (1 = Yes 0 = NO), To OutputFileName:* This parameter set provides the facility to export a generated distribution to an output file, which can in turn be used at a later time as input for the world distribution.

- Parameters specifying the dynamic evolution of the system
    - *no of repetitions for Schelling:* this parameter sets the number of cycles for both the school model and the residential model;
    - *no of agents per cycle:* number of agents that are selected for evaluation each cycle;
    - *no of tries per Agent:* number of tries for each agent in order to establish his happiness threshold, in the residential model;
    - *number of runs:* sets the number of repetitions for the simulation, in order to evaluate statistical deviations;
    - *time step for writing into file:* time intervals at which the results are written into the output files;
    - *step for producing bmp:* time intervals at which a bmp snapshot is produced in the residential model;
- Preference parameters
    - *T:* satisfaction threshold, as described above;
    - *radius_moore_extended:* sets the radius (in units of location size) for variable neighborhood index;
    - *radius Evaluation Utility Schelling:* sets the radius (in units of location size) of the local neighborhood that is used for the evaluation of the ethnic preference function;
    - *utility parameters white agents MA and fA:* sets the utility parameters for white agents for the ethnic preference function, Ma and fa ;
    - *utility parameters black agents MB and fA:* sets the utility parameters for black agents for the ethnic preference, Mb and fb ;
    - *utility parameters for distance: alpha and dist_max:* sets the value of the weight α between ethnicity preference and the distance preference and the value of the maximum distance, at which the distance preference becomes null.

Output files

The source code provides several output files.

1. The information obtained for every evolution of the residential model is written to an output file. The name of the file is given as parameter to the cartier::schelling(char *)

function. The file contains on each line the information corresponding to the specified cycle as follows:

- the number of the cycle for which the information has been evaluated (is always a multiple value of the *time step for writing into file*),
- the school-neighborhood segregation index,
- variable-neighborhood index for a radius of 1.5,
- variable-neighborhood index for the specified parameter *radius_moore_extended,*
- the number of white agents, and respectively black agents, whose level of satisfaction with respect to the local residential neighborhood is above threshold.

The name of the file in the current version is *segregare_in_time.txt*.

2. The program automatically generates an output file containing statistical information of the school-neighborhood segregation index, across the total number of repetitions. The file contains on each line the information corresponding to the specified cycle as follows:
   - the number of the cycle for which the information has been evaluated (is always a multiple value of the *time step for writing into file*),
   - the average school-neighborhood index corresponding to the given cycle, across the total number of repetitions.
   - the standard error of the school-neighborhood index corresponding to the specified cycle.

   The name of the file in the current version is *statistica_schelling.txt*

3. The information obtained for every evolution of the residential model is written to an output file. The name of the file is given as parameter to the cartier::segregare_scoli(char*) function. The file contains on each line the information corresponding to the specified cycle as follows:
   - the number of the cycle for which the information has been evaluated (is always a multiple value of the *time step for writing into file*),
   - the school-neighborhood segregation index,
   - the number of white agents, and respectively black agents, whose level of satisfaction with respect to their assigned school is above threshold.

   The name of the file in the current version is *segregare_scoli_in_time.txt.*

4. At the end of a run using school-choice model, a final output file (*indice_segregare_scoli.txt*) is written containing detailed information of the schools status. The file contains on each row,
   - the school's number (logistical index associated to a school in order to identify it across the simulations),
   - the school-segregation index, at the end of the run,
   - the school's X, Y coordinates with respect to the rectangular grid,
   - the maximum number of students that can attend the school
   - the total number and the number of white students attending,
   - for convenience, the percent of white students at the given school

5. The program automatically generates an output file *(*called *statistica_scoli.txt)* containing statistical information of the school-segregation index, across the total number of repetitions. The file contains on each line the information corresponding to the specified cycle as follows:
    - the number of the cycle for which the information has been evaluated (is always a multiple value of the *time step for writing into file*),
    - the average school-segregation index corresponding to the given cycle, across the total number of repetitions,
    - the standard error of the school-segregation index corresponding to the specified cycle.