traffgo.ht

PedGo Tutorial v1.2

(Referencing PedGo v2.6.0.2, PedEd v2.6.0.0)

This tutorial introduces the application of the PedGo Editor and the Simulation. It is designed to give you a step by step instruction into all functions, so after having completed it, you should be able to use PedGo to its full extend. If some points remain unclear or if you have suggestions on how to improve this tutorial, please give us feedback, so we have a chance to improve it (info@traffgo-ht.com).

Content

1 Basic Geometry Modelling

1. Start the Editor (PedEd) and familiarize yourself with its layout.
2. At the top of the main window, you will find buttons for general functions as well view filters.
3. The *View* window controls your view of the floor plan as well as the zoom.
4. The *Tools* window controls all edit functions. It has three tabs (*Geometry*, *Routes* and *Agents*), so only the elements of the active tab can be edited. To a new user, it sometimes is confusing, but if you cannot edit an element, make sure, you are in the right mode.
5. At the bottom of the main window, the coordinates of your cursor are displayed (continuously and cell based).
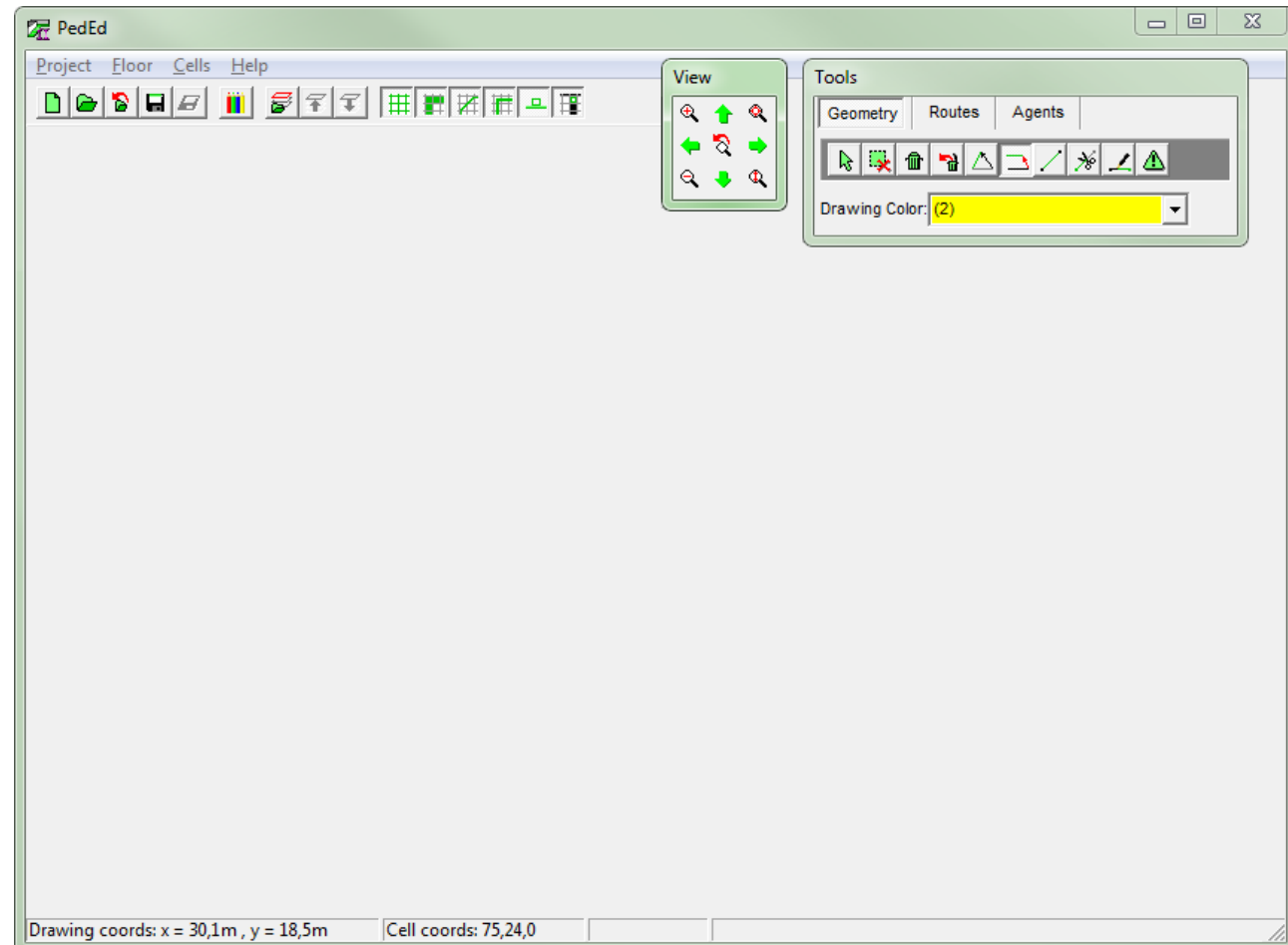
1.1 Drawing/Deleting Geometry Elements

1. Start the Editor.
2. Click on *Floor→New* in the main windows menu.
3. You can edit the given values, but this will be fine for now. Press *Ok*.
4. Choose the *Geometry* Tab of the *Tools* window and draw the geometry shown on the right. Use the following buttons:

   (0) ▾ Select drawing color,

   ◺ Draw line (click & hold, move, release).
5. Only use the left half of the grid.
6. Your window should now look like shown on the right.
7. You can delete elements by pressing the button 🗑 and then by clicking on the elements.
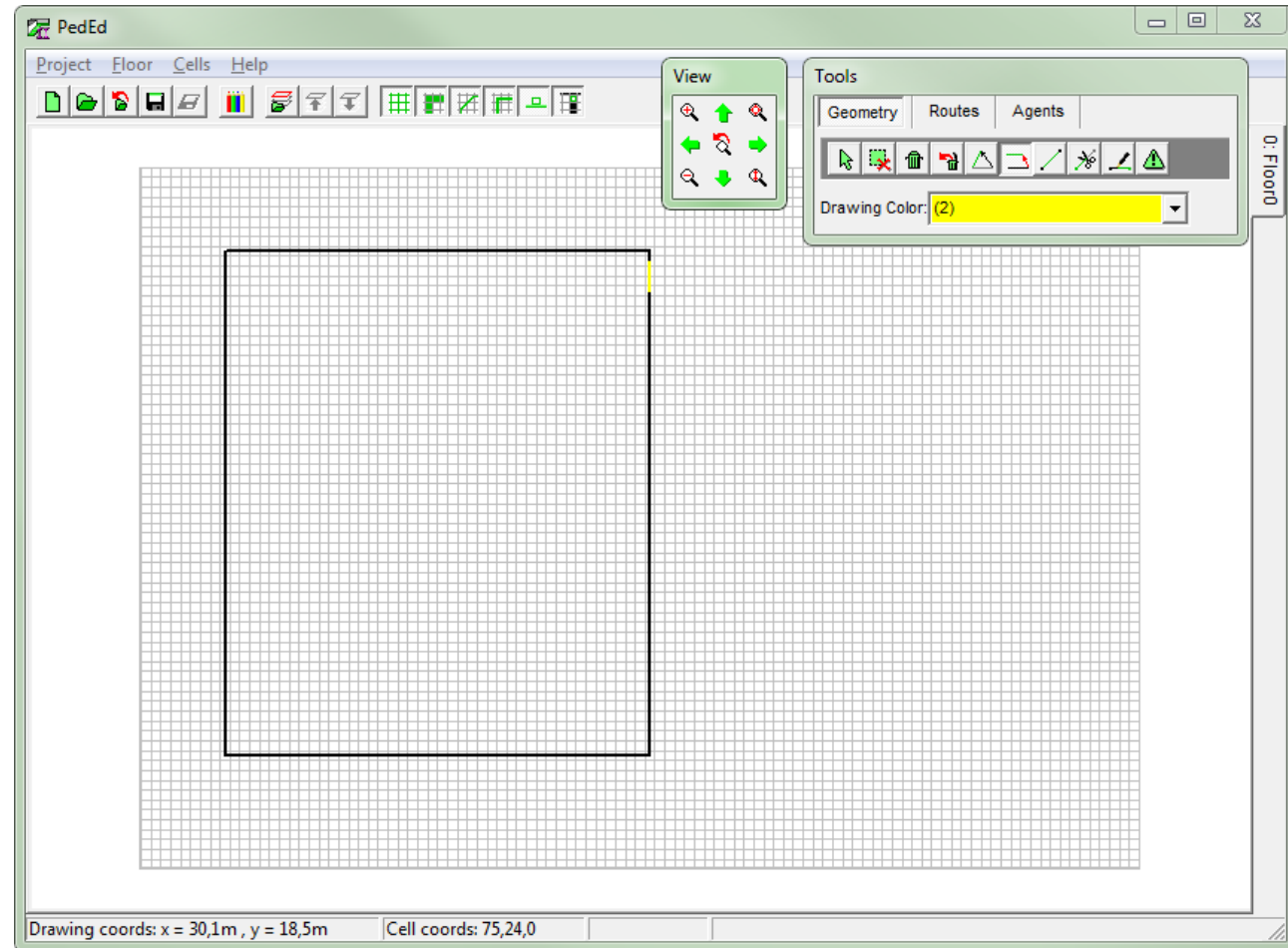
1.2 Defining Color Coding

1. Since no properties are assigned to the drawing colors, no attributes are assigned to the cells. Click 🔳 or *Project→Color coding* to assign properties to element colors.
2. First click on a color in the list and then on a button on the right to assign properties.
3. By clicking on a pressed button, you can reset the property of a color.
4. If your company uses the same color scheme for all drawings, you can define a default distribution by pressing *Make default*.
5. Your color coding window should now look like this.
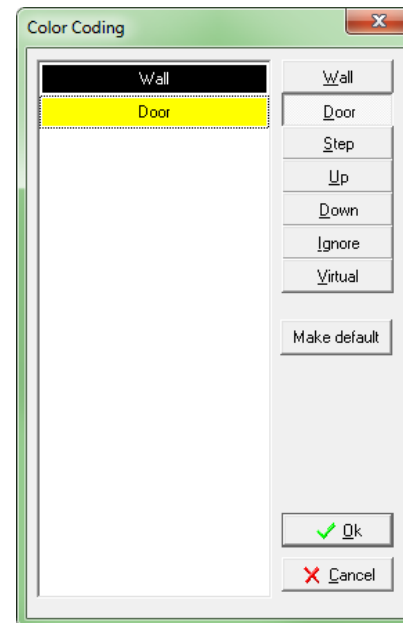
1.3 Saving, Filters, Pan and Zoom

1. As you can see, the cells are now painted in the colors of the elements responsible for their property (see image to the right).
2. Save the project ("Step 1") by clicking *Project→Save as* or the save button 🖫. The path is now shown in the header of your window.
3. Play around with the filter buttons at the top of your main window:
   - ⊞ Toggle cell grid (combined with cell data),
   - ▦ Toggle cell information,
   - ▨ Toggle dxf drawing (no effect yet, since no dxf imported),
   - ▦ Toggle line elements rounded onto grid,
   - ▫ Toggle tags of route elements (no effect yet, since no route elements used),
   - ▦ Mark bottlenecks which are only one cell wide.
4. You can change the zoom and pan the view by pressing the buttons of the *View* window.
5. You can also zoom with your mouse wheel or by selecting a zoom section with right-clicking the mouse.
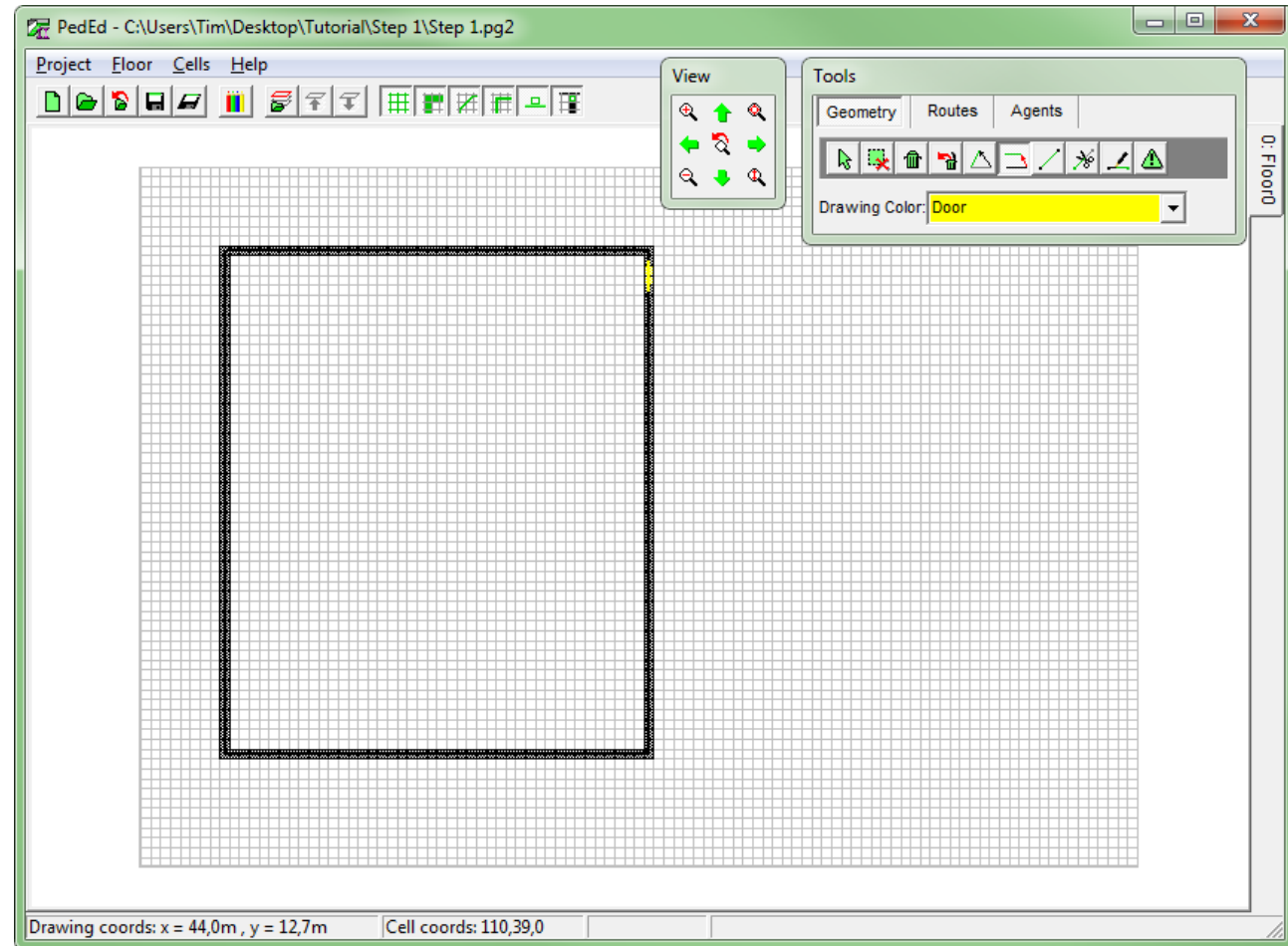
1.4 Selecting/Deselecting and Editing Elements

1. We now edit the geometry, so it will look like shown at the right. There are several ways to do so.

2. By pressing △, you can move nodes by dragging them. If several elements end at one node and none are selected, all elements will be moved.

3. Alternatively, you can also move elements by pressing ◲ and dragging them. All nodes of other elements ending at the nodes of the moved element will be moved as well.

4. If only one element is supposed to be moved, you can select it. Whenever an element is selected, all following actions only work for the selected element. There are three ways of selecting elements:

   ⬚ Select/deselect elements by clicking on them.

   Shift + left mouse button: Select/deselect elements by clicking on them.

   Ctrl + left mouse button: Select elements by dragging a rectangle around them.

5. In this case, you could select all vertical elements at the right and move them using the ◲.

6. Selected elements can be deleted by pressing the *del* key or unselected by pressing ▣
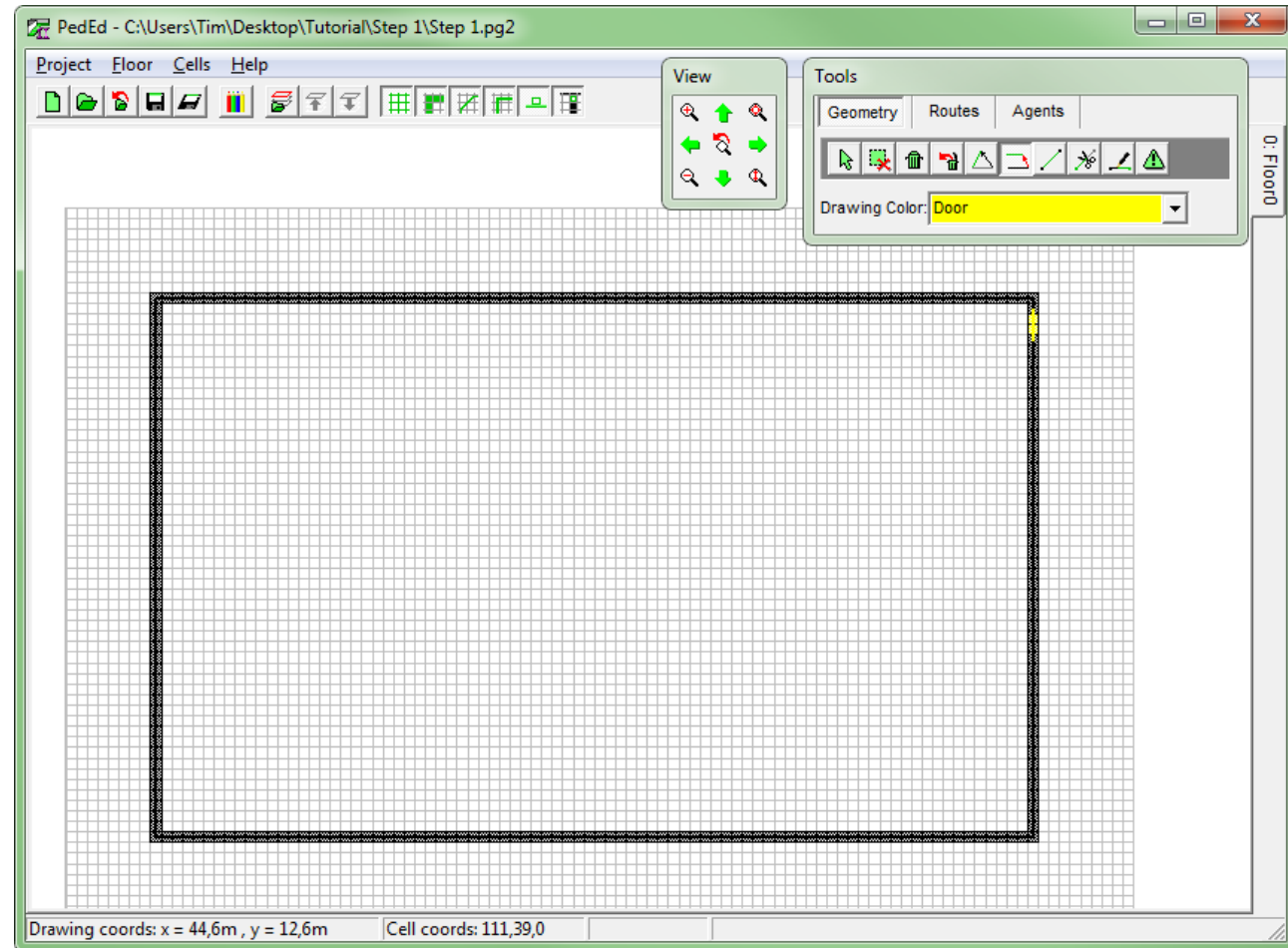
1.5 Adding a Route and Agents

1. Switch to the tab *Routes* in the *Tools* window. The route *Default* is selected in the pull down menu. At the moment, one route is sufficient.

2. Note, that only elements of the active route can be altered, so if you are handling several routes, you can only edit the elements of the active route. The active route elements are displayed green, others grey.

3. Press [ ] to add exit cells as shown (click & hold, move, release). Agents are taken out of the simulation when they reach these cells. Alternatively they can change their goal, if a follow up route is defined.
Exit cells should be positioned a few cells (3-4) away from exit doors and should be wider in order to prevent influences on the exit flow.

4. Switch to the tab *Agents*. Next to the *Default* route, a *Default* population is defined, which we will use for now.

5. Press [ ] to add agents. Click into the created room. You can enter the amount of agents (here: 500), select the route to which the added agents are appointed to and define the group they are supposed to belong to. If you have a demo version, you can add 20 agents max.

6. Your project should now look like shown on the right.
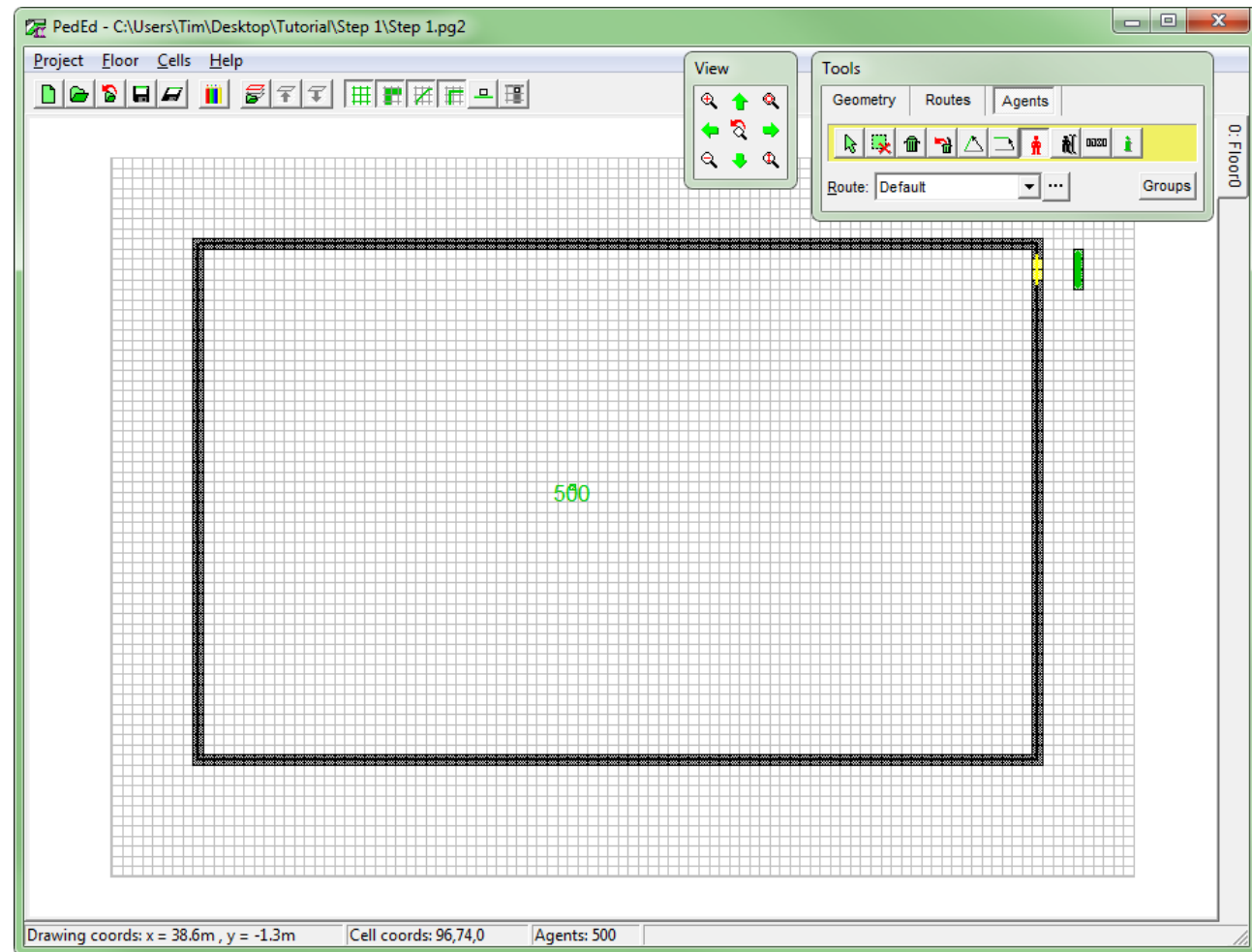
2 First Simulations

1. If you have a dongle, plug it in. If not, your simulation is restricted to 20 Agents.
2. Start the Simulation (PedGo).
3. Press *Project→Load project* or click 📂 to load your project.
4. PedGo offers the following simulation functions:
   - Demo run, no data generated, just for testing,
   - Mean run for statistical analyses, delivering statistical data,
   - Single run to generate detailed result data,
   - Stop simulation.
5. Run the *Demo run* and see, how all agents move to the exit cells of their assigned goal, their color thereby representing their walking speed. At the goal cells, they are saved/taken out of the simulation.
6. Below the menu buttons, you can see, how many agents are distributed in the project, how many agents have been saved, what time has passed and which seed is currently used for the random number generator (this comes later).
7. If you are using the demo license, it will run very fast, due to the small amount of agents.
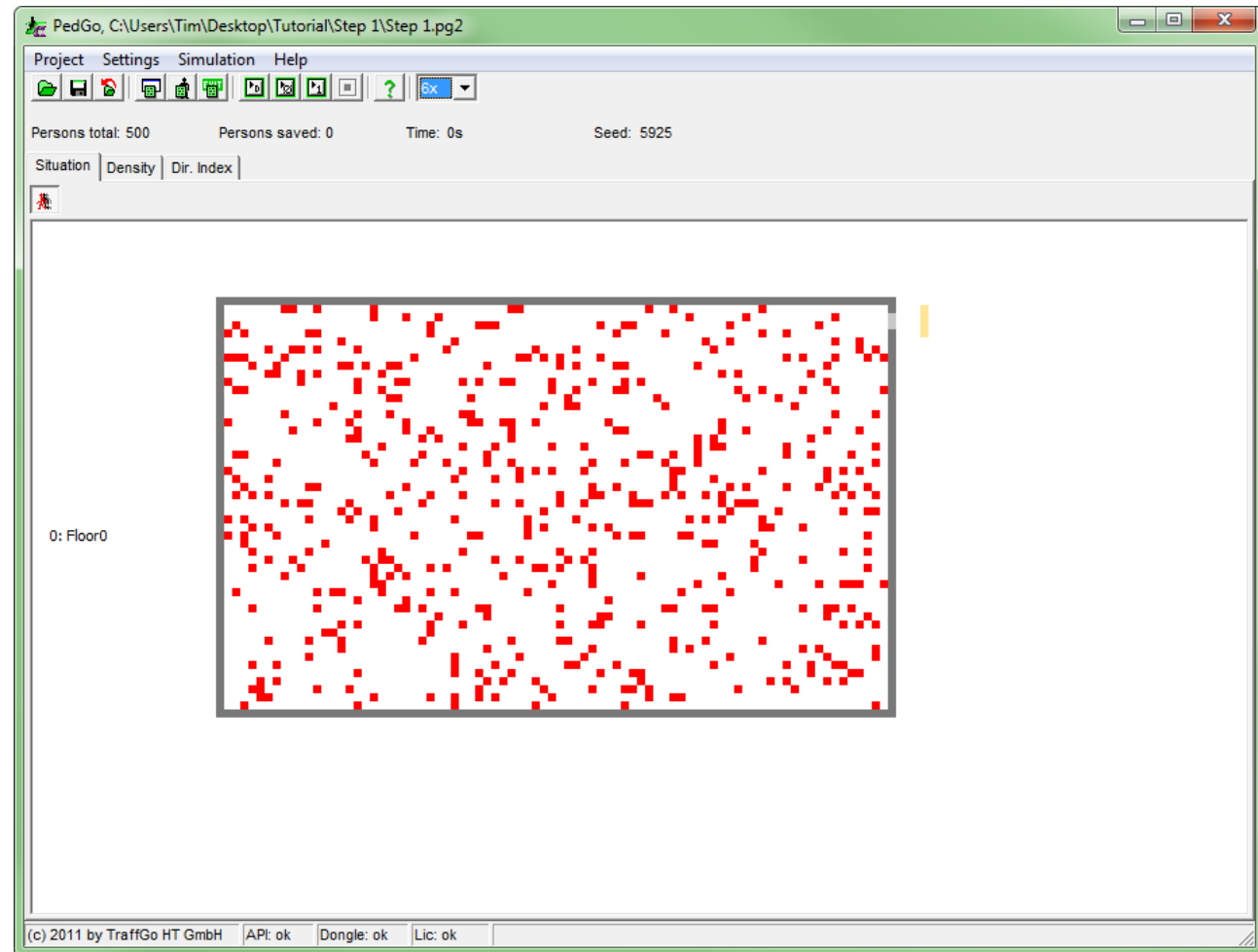
2.1 Density Plot

1. Press the *Density* tab.
2. The density plot visualizes how long agents have been standing in densities of over 4 persons per square meter. By default, the value is set to 10%.
   In the example shown, the overall evacuation duration was 5:44 min, so 10% is 34 s.
   The areas marked red thus are areas in which agents have been standing in densities of over 4 P/m² for 34 s or longer.
3. You can set the time threshold to other values by moving the track bar or by pushing one of the buttons.
4. If you are using the demo license, you will not see red areas, since no congestions occur.
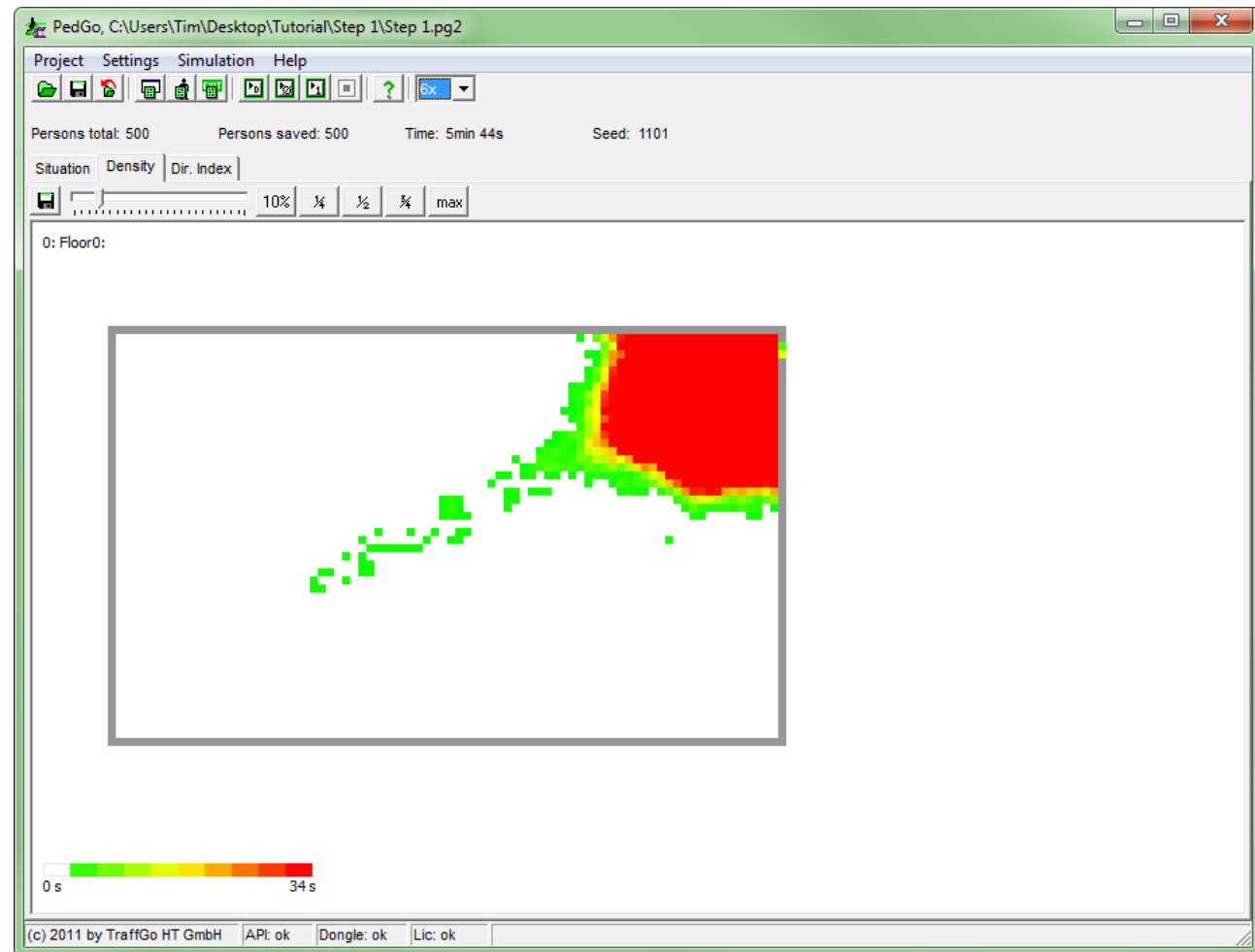
2.2   Directional Index

1. Press the *Directional Index* tab.
2. The directional index plot visualizes the directional index values of every cell in a repeating green to blue color gradient.
3. When you click on a cell, its directional index value is shown in the bar at the bottom of the window.
4. The color gradients show, how PedGo has spread the potential, starting at the exit cells through all cells of the outer room and then through the door into the inner room.
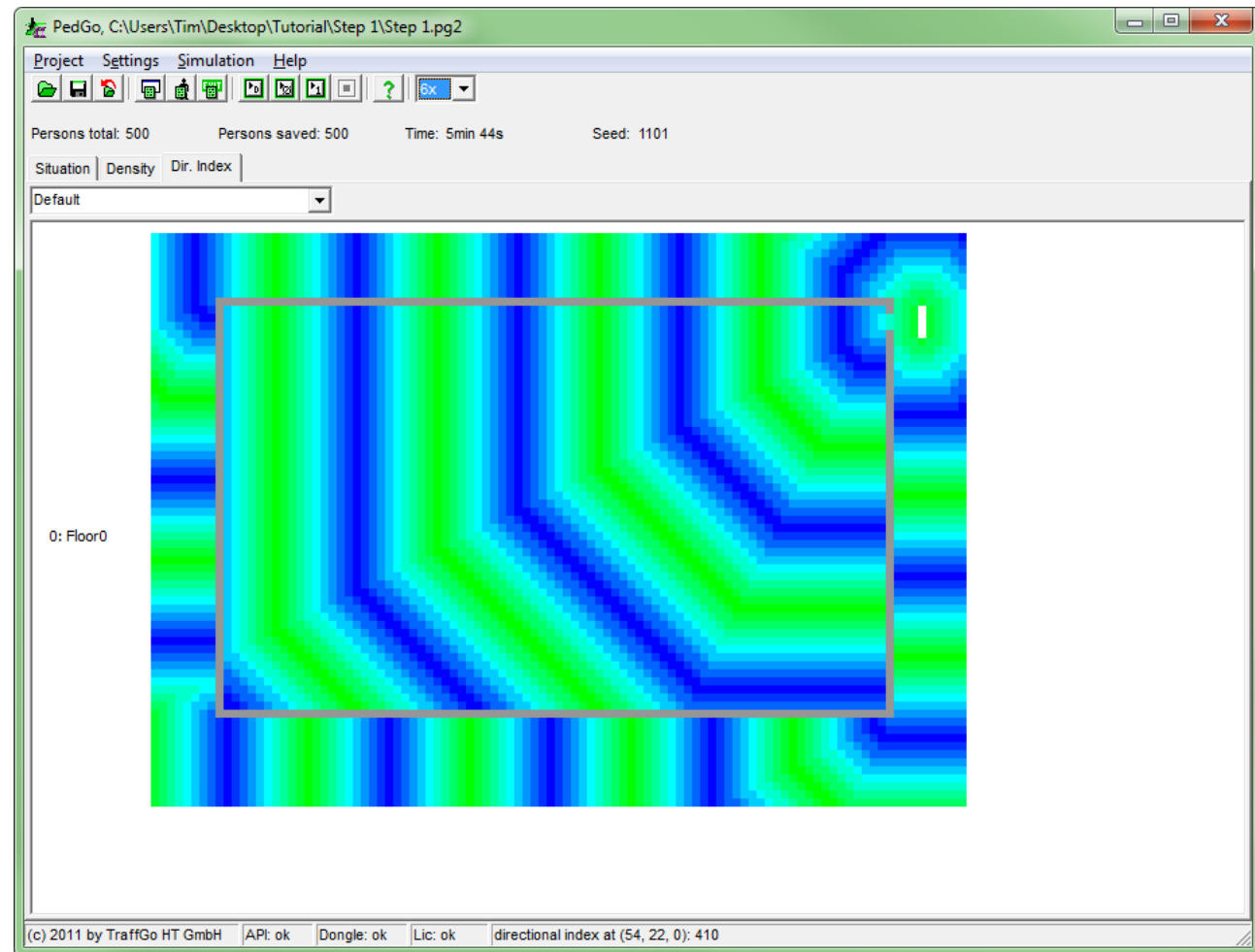
2.3   Demographic Parameters

1. Press [icon] or *Settings→Demographics* to open the demographics dialog.

2. You can use this dialog to assign individual parameters to each defined agent group. You can choose the group through the pull down menu at the top and assign parameters of the active group to others by pressing [icon].

3. PedGo has a few standard distributions which can be assigned by pressing the following buttons:

   Default Default parameters representing an average population,

   RiMEA Parameters according to the German RiMEA guideline,

   [☀ icon] Day time parameters according to the IMO guideline,

   [☾ icon] Night parameters according to the IMO guideline,

   [icon] Passenger parameters according to the IMO guideline,

   [icon] Crew parameters according to the IMO guideline.

4. For detailed information about the parameters, have a look at the manual, for now, play with the parameter *Reaction* and see, how it influences the agents reaction durations.
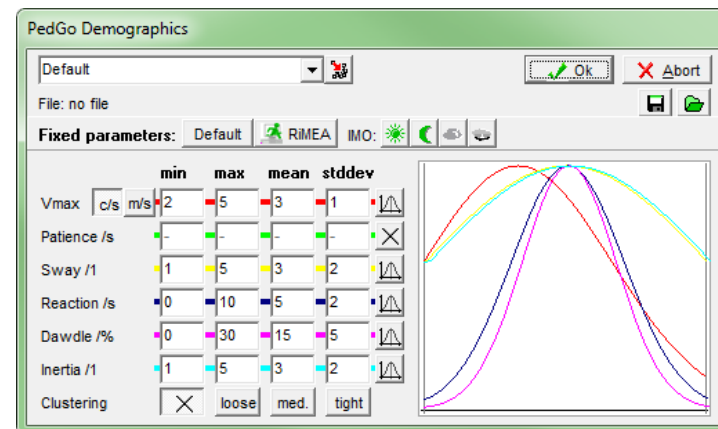
3 Routes, Index Spread

1. Switch to the Editor.
2. Revise your floor plan as shown at the right.
3. Save it as a new project ("Step 2") using the command *Project→Save as*.
   Since the Editor generates additional files to the pg2 project file, every Scenario should be saved in its own directory. Otherwise, project files (e.g. the *.tdf files) of one scenario will be overwritten by the files of another project.
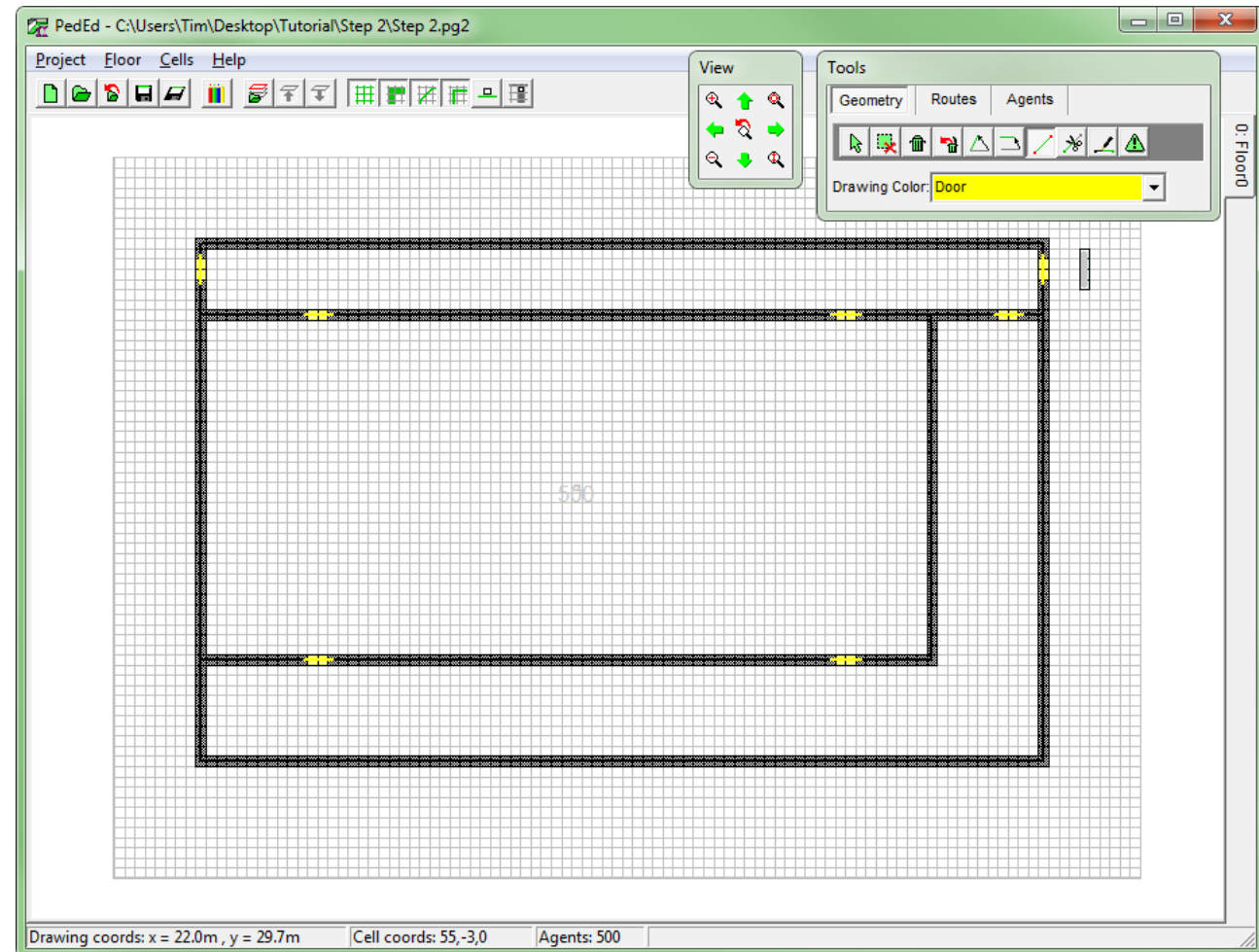
3.1 Directional Index - Background

1. Switch to the Simulation and load the project.
2. Starting a *Demo Run* you will notice, that the agents will not all use the exit to the right, but will also leave the "building" via the left exit. The reason for this is the way, the directional index is calculated. Understanding the way it is spread and how you can control it is a key factor in working with PedGo.
3. Open the Tab *Dir. Index*, it visualizes the directional index, how it is finally spread.
4. PedGo spreads the directional index by room. A room is surrounded by walls, doors, virtual and up/down cells. In this case, the index is spread in three steps:
   Step 1: From the exit cells along the outer room.
   Step 2: Through both doors into the top hallway.
   Step 3: Through the next doors into the remaining rooms.
5. As a result, agents will choose the two upper doors of the middle room and from there move to the left end right door of the top corridor.

Step 1:
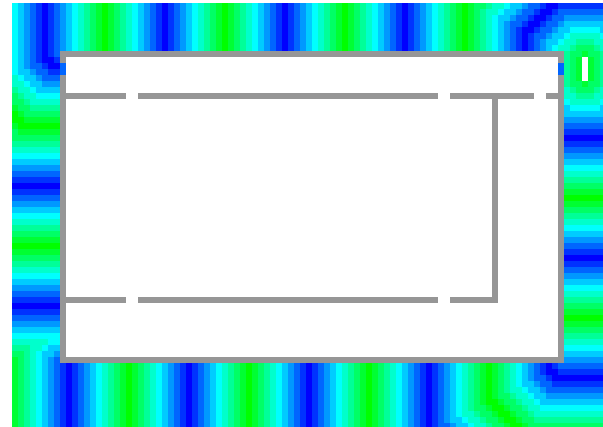
Step 2:
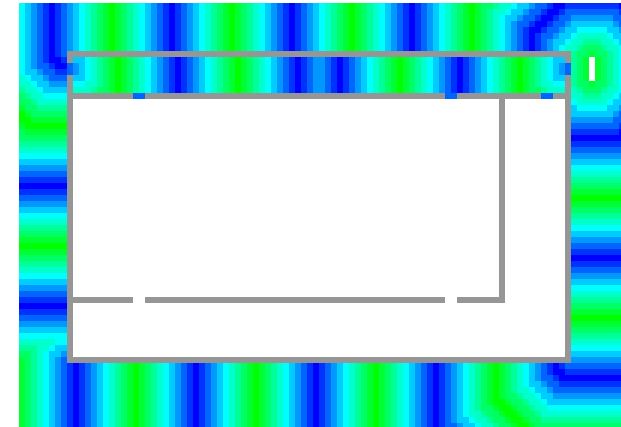
Step 3:
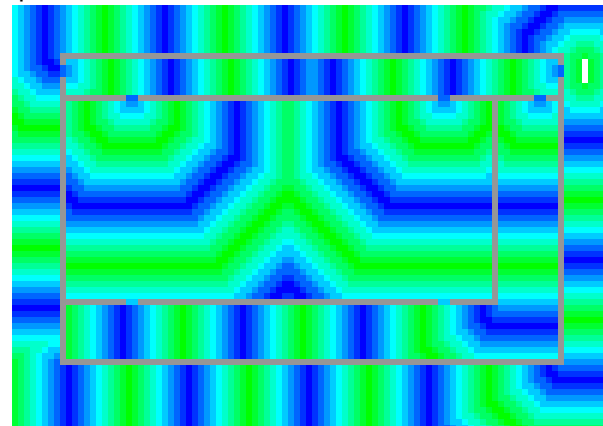
3.2 Route-Guiding, Example 1

1. Switch to the Editor.
2. You can influence the spread of the directional index by using route lines. They mark doors and up/down cells, through which the index is primarily spread.
3. Press ⬛ in the *Routes* Tab of the *Tools* window to draw route lines. Basically, it only matters to mark the appropriate door cells, but for a good overview, it is better to draw long and connected lines.
4. Add route lines as shown on the right.
5. Save the project, switch to the Simulation and press ⬛ to reload.
6. Have a look at the directional index. It now has spread through the outer room, but then into the inner room through the door at the top right, which was marked by the route line.
7. All agents will now use the marked door.
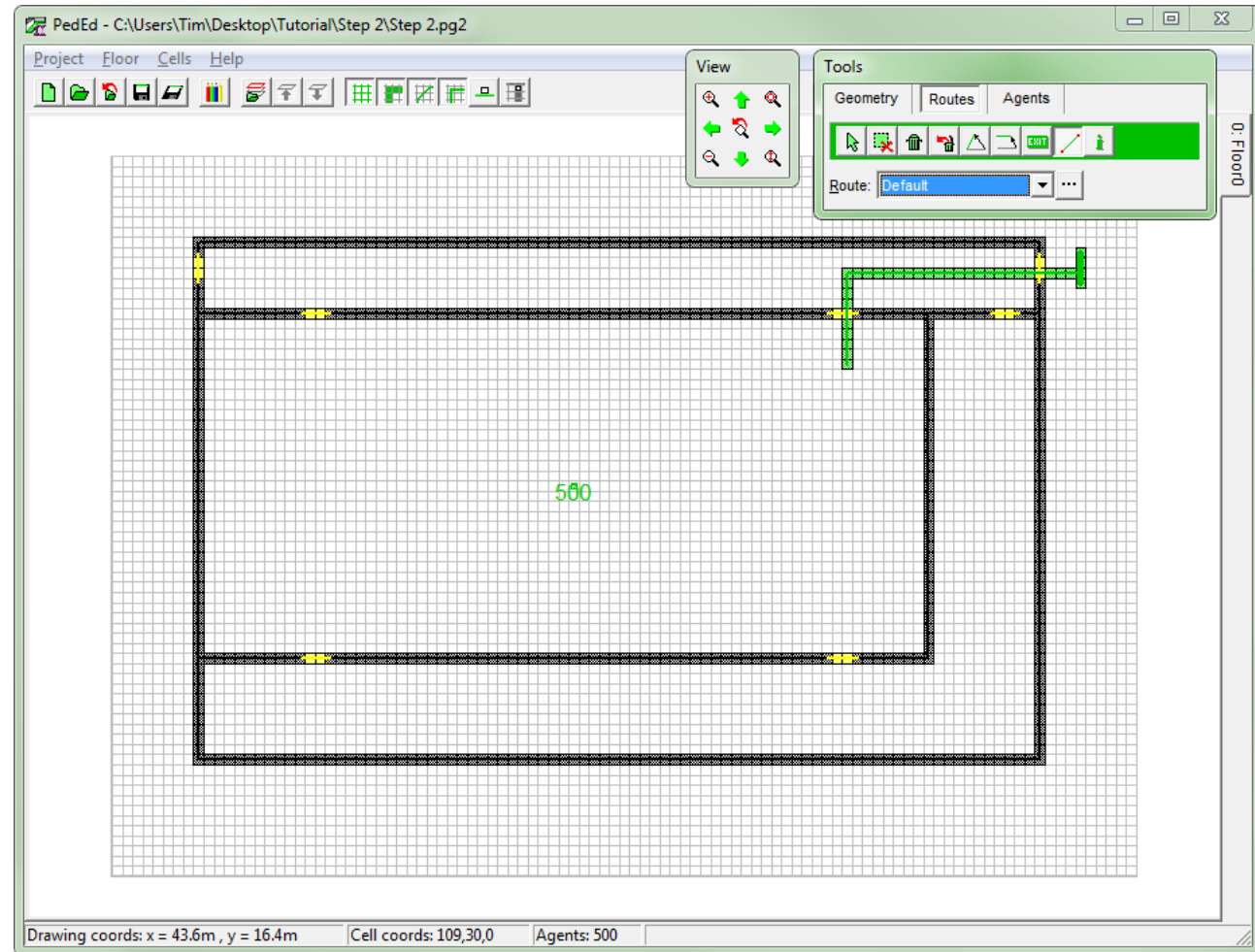
3.3   Route-Guiding, Example 2

1. Switch to the Editor.
2. Press 🗑 and delete the route element marking the last door.
3. Save the project, switch to the Simulation and reload.
4. Have a look at the directional index and see how it has changed. Since the index is now primarily only spread into the upper corridor, it afterwards spreads room wise through the remaining doors.
5. Resulting, the agents will use both upper doors.
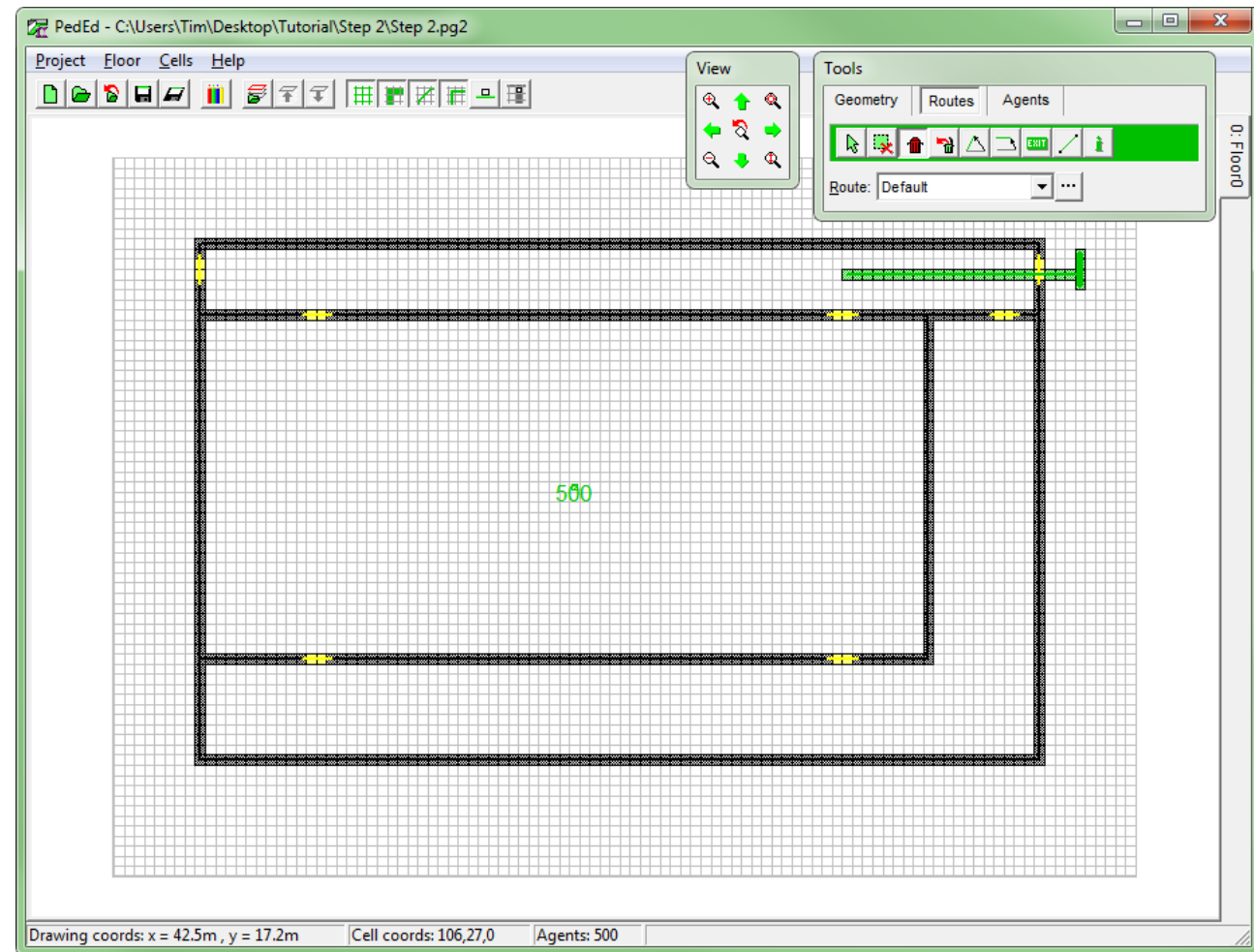
3.4 Route-Guiding, Example 3

1. Switch to the Editor.
2. Extend the route lines as shown.
3. The directional index will now primarily spread through the two corridors adjacent and then through all doors into the inner room.
4. Agents will now use all exits available.
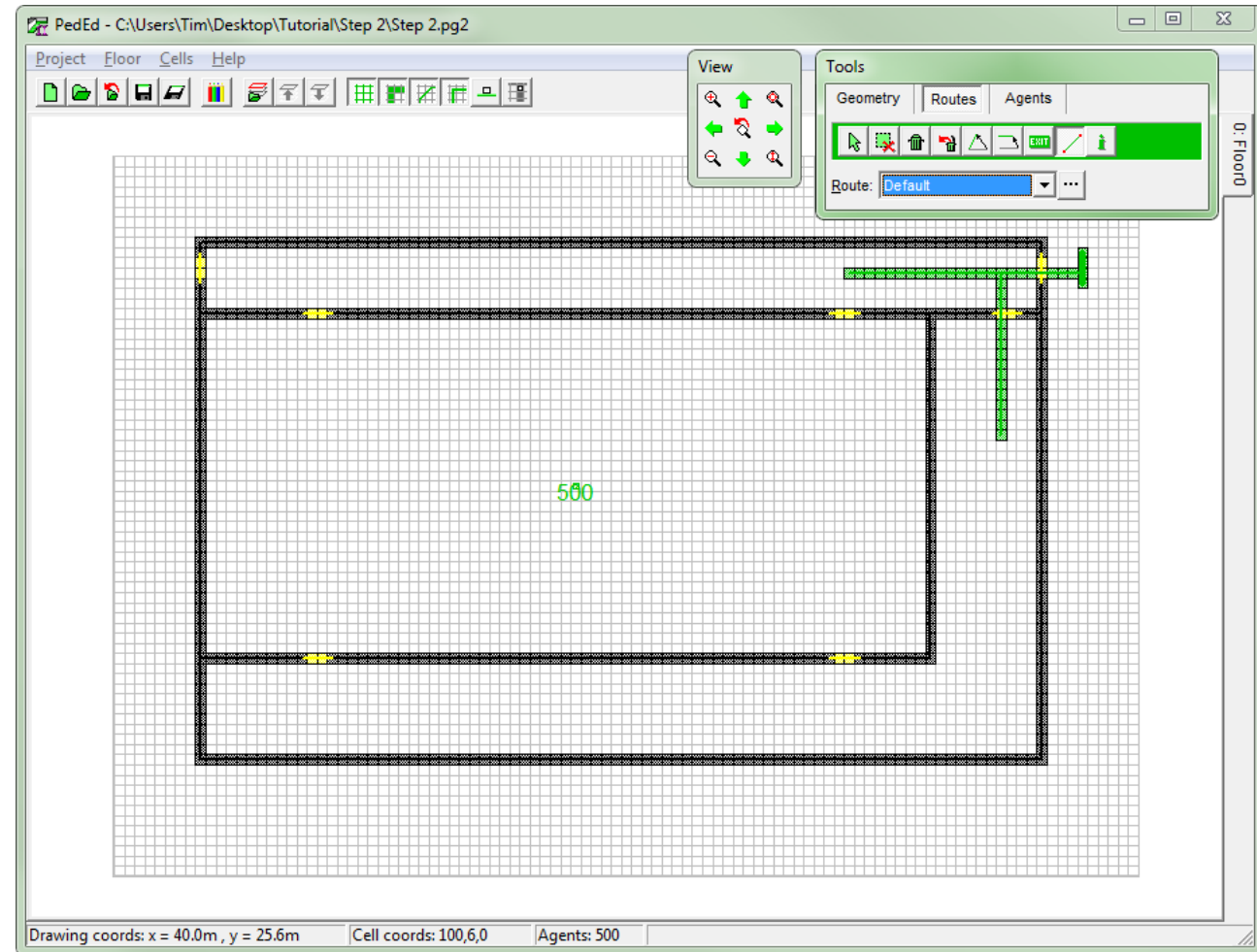
3.5 Multiple Routes, Defining

1. Switch to the Editor.
2. We will now add a second route, press the button ⬚ to open the route dialogue, shown on the right.
3. Currently, only one route ("Default") is displayed in the route list on the left. Click on the route name, rename it "Right" and press ENTER.
4. Press ⬚ to add a new route and name it "Left".
5. An existing route, its elements (exits or route lines) or its appointed agents can be deleted by pressing the appropriate button 🗑.
6. As you can see from the table, all route elements and agents are assigned to route "Right", route "Left" contains no elements.
7. Press *Ok*.

Routes

Overview | Details

Routes:

Caption	Stay	Save	Agents	Elements	A	F
Left	100	100	0	0	0	0
Right	100	100	500	3	0	0

Alternatives:

Probability	Route

Followups:

Probability	Route

✔ Ok
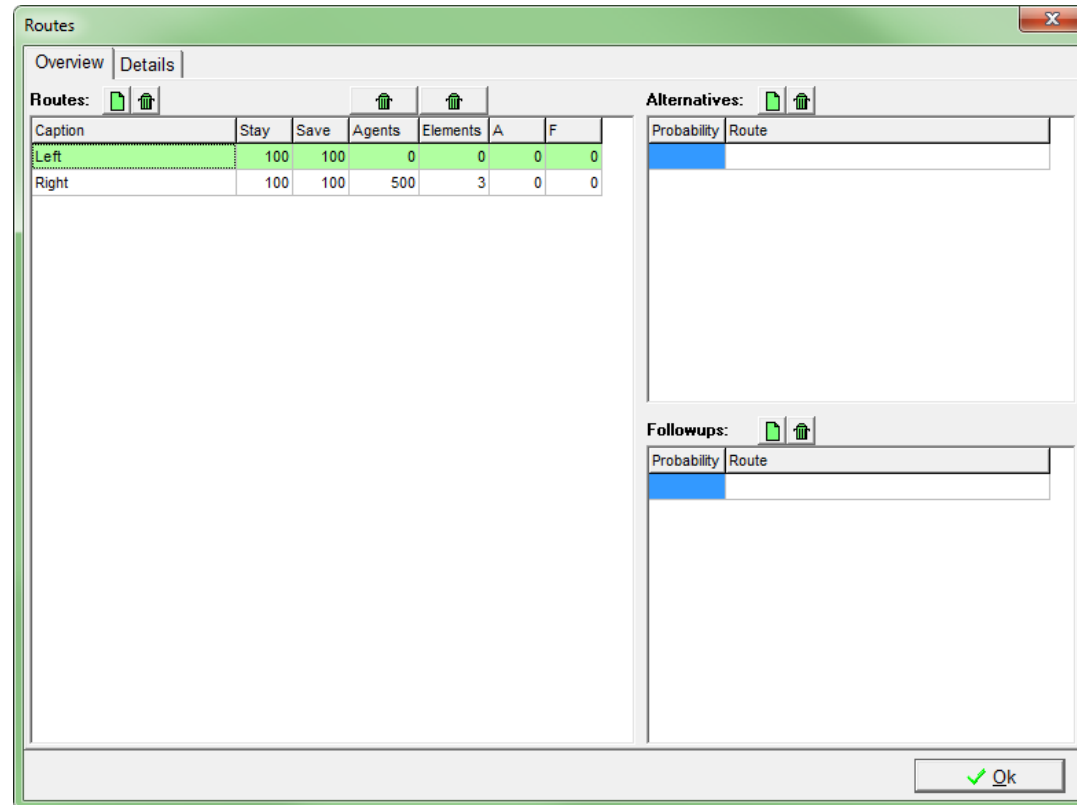
3.6   Multiple Routes, Modeling

1. Switch to the Editor.
2. Edit the project like described before, so you finally have the route "Right" leading to the exit on the right and the route "Left" leading to the left side equivalently. 500 agents are assigned to each route.
3. Note, that you can only edit the agents and elements of the active route, which is displayed green. The inactive route elements are displayed grey.
4. Until now, the agents were distributed randomly over the cells of the inner room. We now have stretched the agent elements using the button △ of the *Agents* tab in the *Tools* window. Whenever the agent element is larger than one cell, agents will be distributed on the resulting rectangle.
5. Save the project in a new directory ("Step 3") and run it in the Simulation.
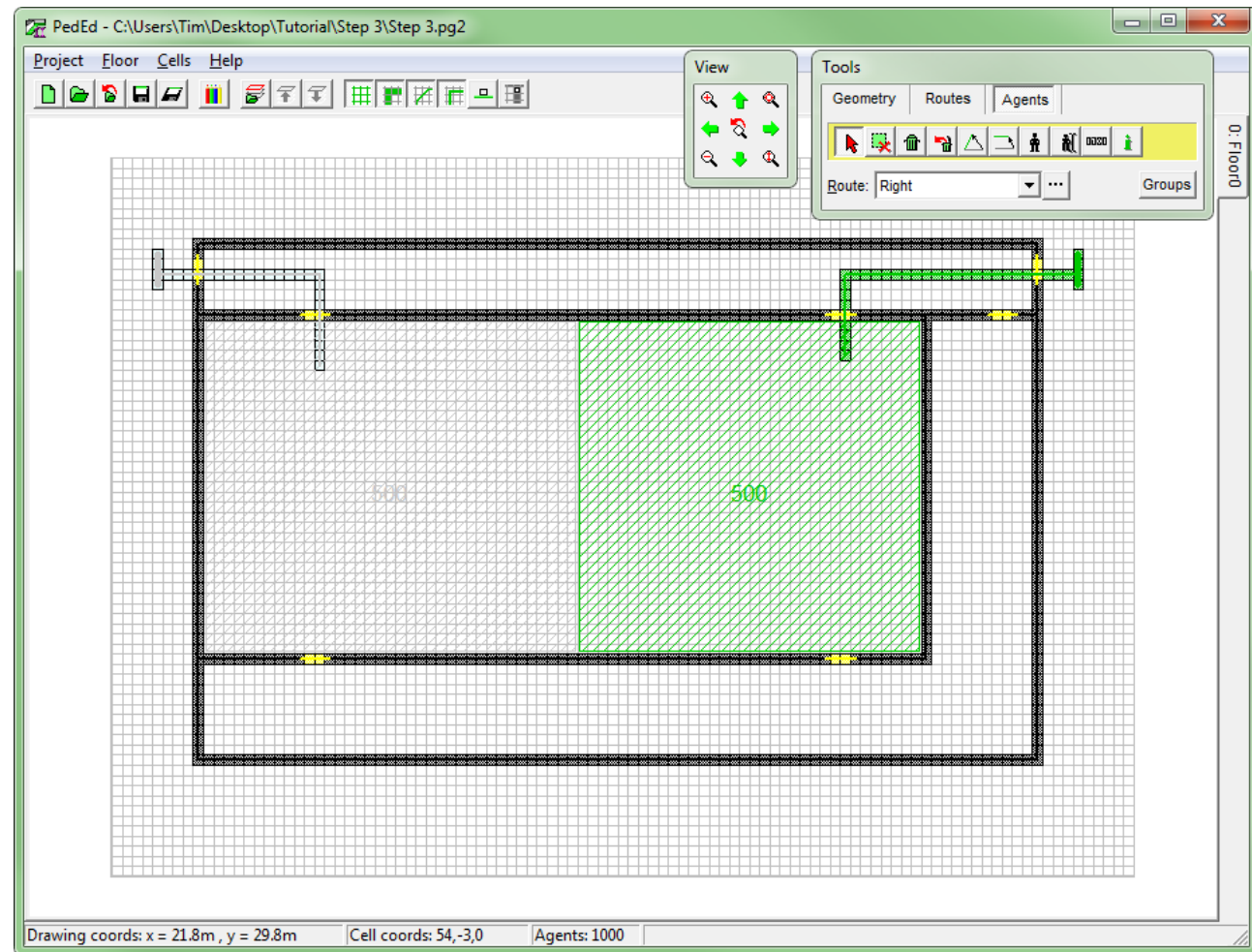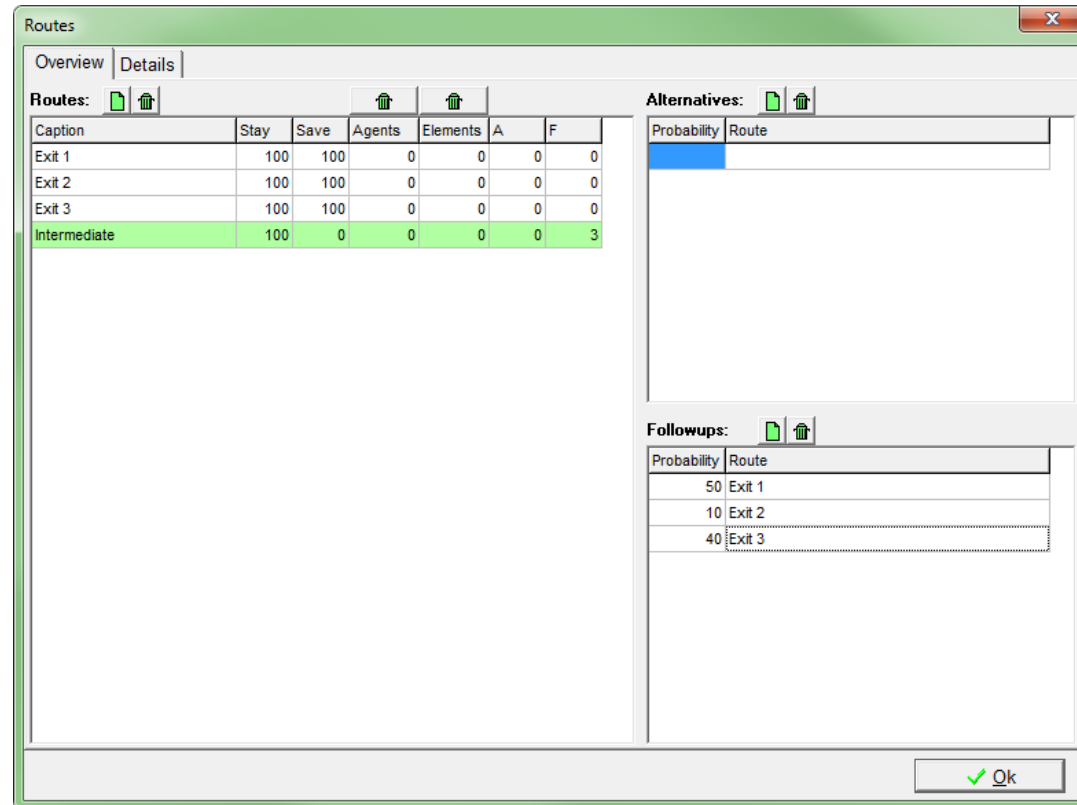
4   More Complex Routes

1. Start a new project in the Editor by pressing ▯ and model a new floor as described before.
2. The geometry should look like the one on the right.
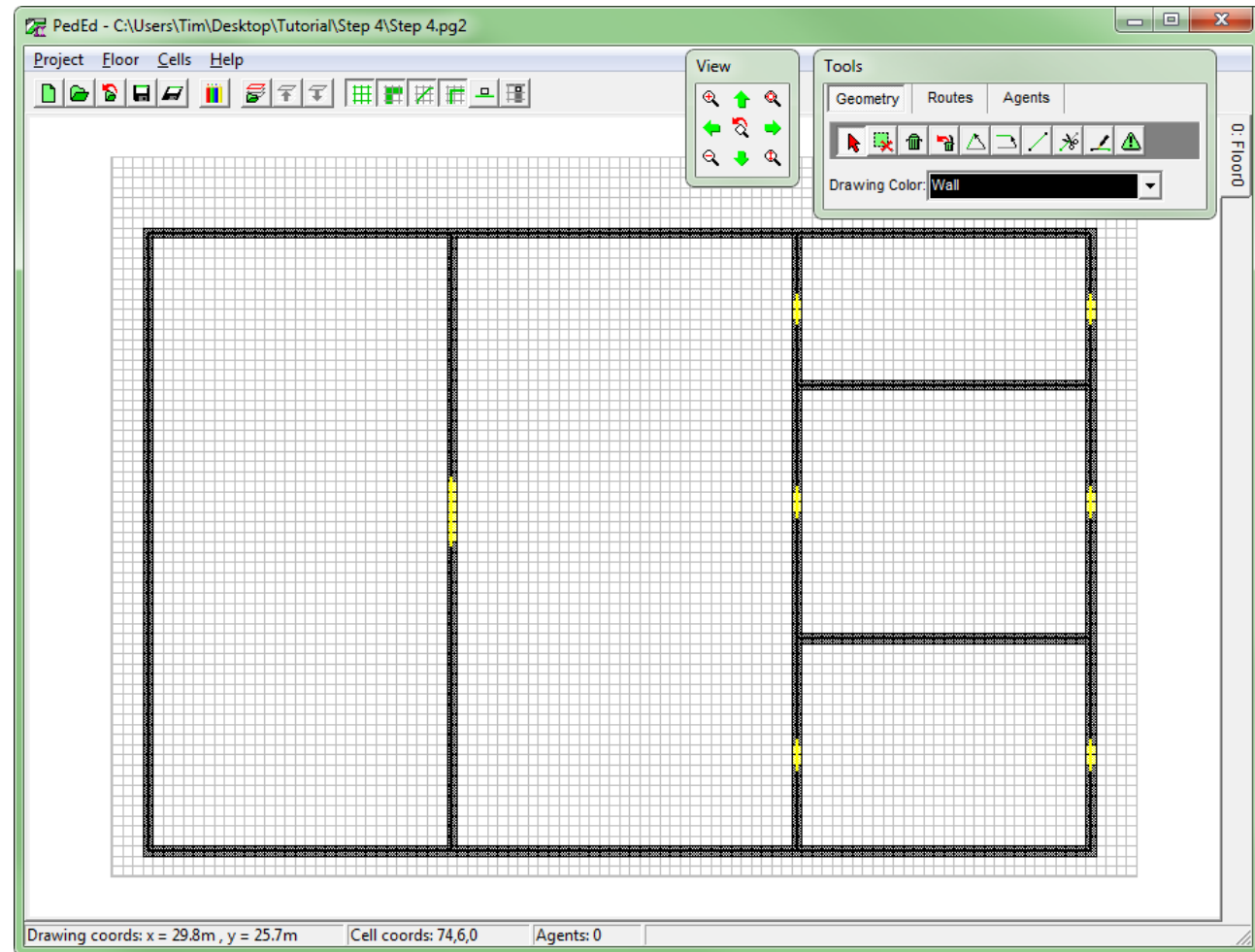
4.1 Follow Ups - Defining Routes and Properties

1. Create 4 routes as shown on the right.
2. The following scenario is planned: Agents will walk to the "Intermediate" goal, from where they are supposed to walk on to "Exit 1", "Exit 2" and "Exit 3" according to a stochastic distribution.
3. Select route "Intermediate" in the *Routes* table and add the "Exit" routes in the *Followups* table by pressing button ⬜.
4. Edit the *Probability* values as shown (assign by pressing ENTER).
5. Note, that the *Save* value in the *Routes* table of route "Intermediate" is automatically set to 0, since all agents are supposed to choose a follow up route.
6. Press *Ok*.

Routes

Overview | Details
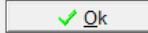
Routes:

Caption	Stay	Save	Agents	Elements	A	F
Exit 1	100	100	0	0	0	0
Exit 2	100	100	0	0	0	0
Exit 3	100	100	0	0	0	0
Intermediate	100	0	0	0	0	3

Alternatives:

Probability	Route

Followups:

Probability	Route
50	Exit 1
10	Exit 2
40	Exit 3

✓ Ok
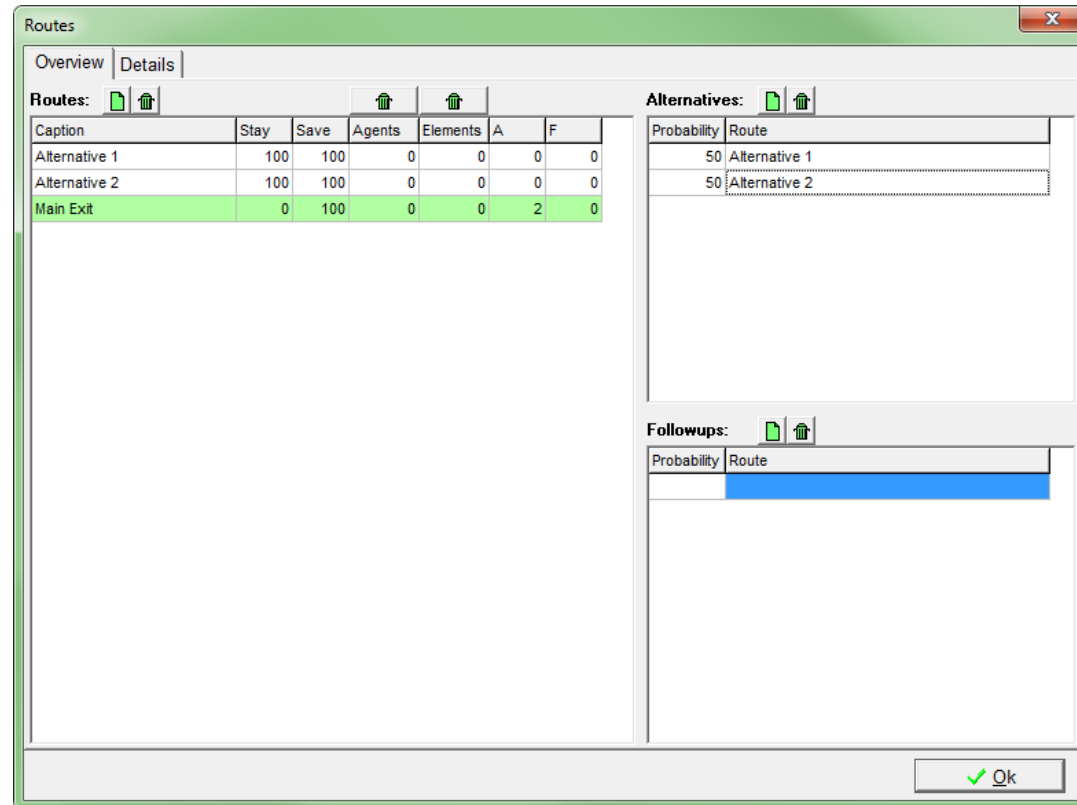
4.2 Follow Ups - Adding Route Elements

1. Add route elements (exit cells and route lines) according to the screen-shot.
2. Note that we have turned on the *Tags filter* by pressing ▣, so all elements are labeled.
3. Add agents (in this case 500) of the population group "Default" and assign them to the route "Intermediate".
4. Save the project ("Step 4") and run it in the simulation.
5. Reaching the exit cells of route "Intermediate", the agents will choose a route of the *Followup* list according to the assigned probabilities and head to the appropriate exit cells.
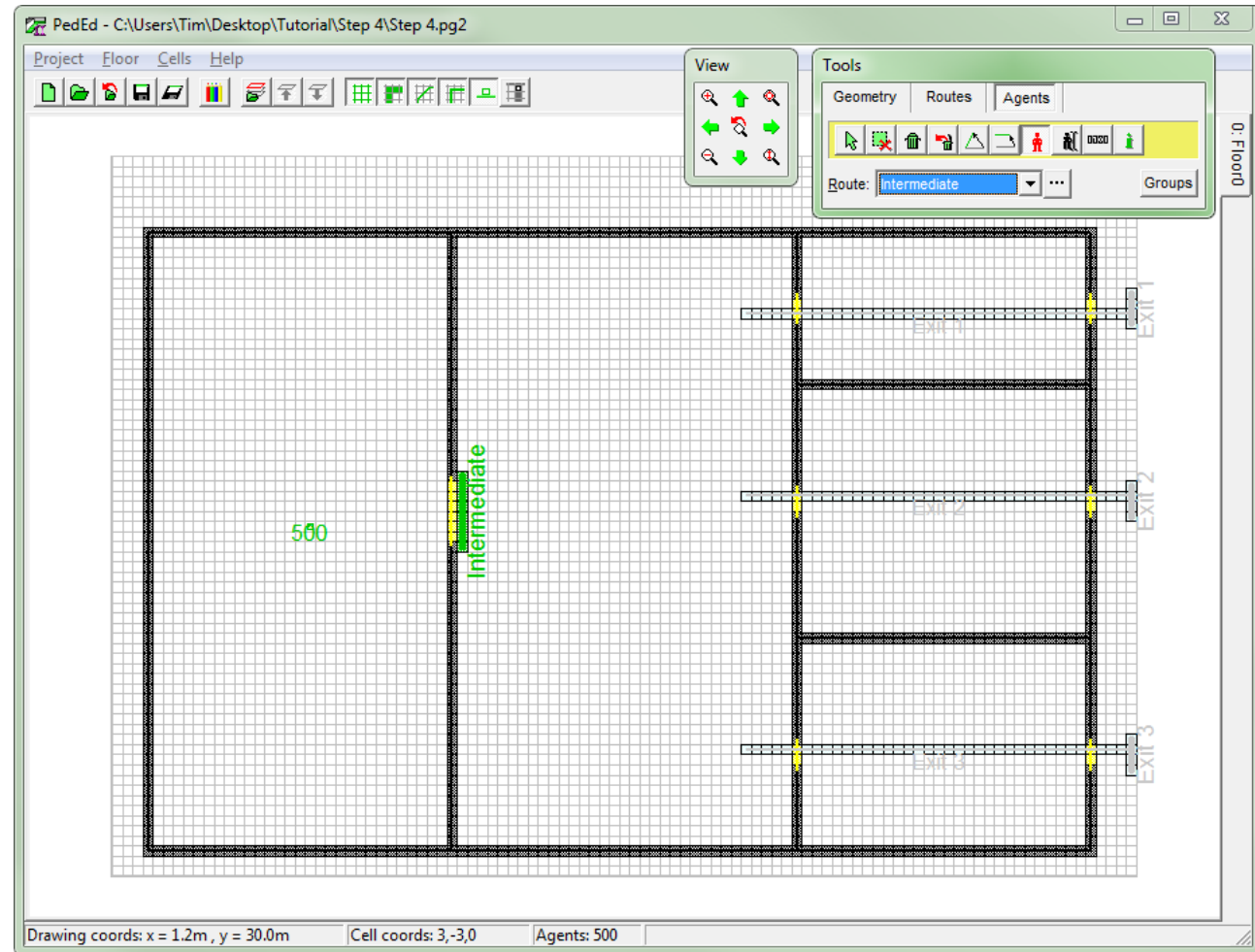
4.3   Alternatives - Defining Routes and Properties

1. Open the routes dialog and delete all routes by pressing 🗑.
2. Create 3 routes as shown on the right.
3. The following scenario is planned: Agents will walk to the "Main Exit" goal until this route is blocked by a hazard. From this moment on they will use the alternative routes "Alternative 1" and "Alternative 2" according to a stochastic distribution.
4. Select route "Main Exit" in the *Routes* table and add the "Alternative" routes in the *Alternatives* table by pressing button 📄.
5. Edit the *Probability* values as shown.
6. Note, that the *Stay* value in the *Routes* table of route "Main Exit" is automatically set to 0, since all agents are supposed to choose an alternative route when the hazard blocks the room.
7. Press *Ok*.

Routes

Caption	Stay	Save	Agents	Elements	A	F
Alternative 1	100	100	0	0	0	0
Alternative 2	100	100	0	0	0	0
Main Exit	0	100	0	0	2	0

Alternatives

Probability	Route
50	Alternative 1
50	Alternative 2

Followups

Probability	Route

✓ Ok

4.4 Alternatives - Adding Route Elements

1. Add route elements (exit cells and route lines) according to the screen-shot.
2. Add agents (in this case 500) of the population group "Default" and as-sign them to the route "Main Exit".
3. Add a hazard by clicking on button ⚠ of the *Geometry* tab of the *Tools* window. Then click into the middle room at the right side. You can de-fine a stochastic distribution for the hazard to happen, set it to 60 s (min and max value). It stochastically oc-curs at the defined time and will last until the end of the simulation.
4. Save the project ("Step 5") and run it in the simulation.
5. When the hazard occurs, the appro-priate room turns grey and the agents will choose a route of the *Al-ternatives* list according to the as-signed probabilities and head to the appropriate exit cells.
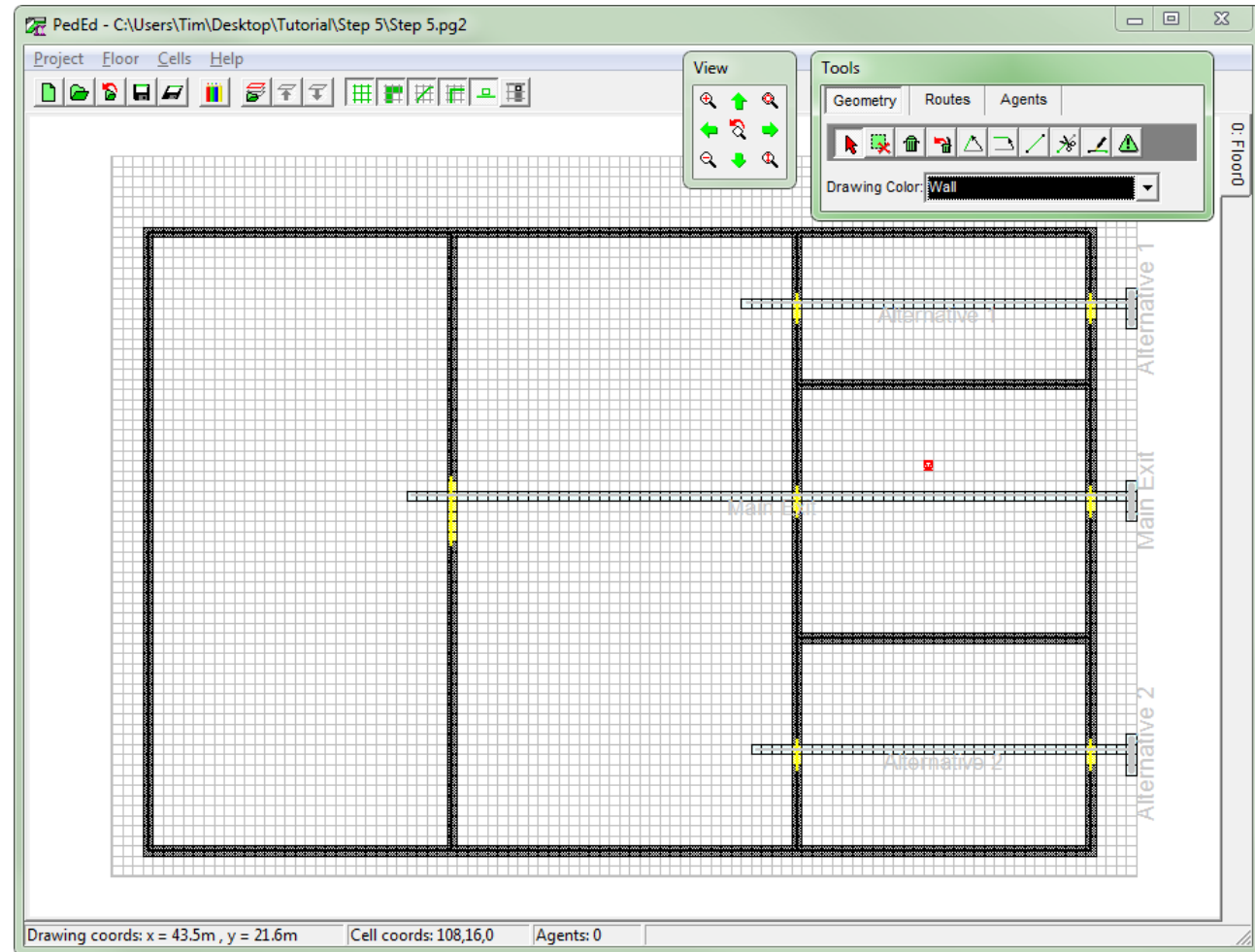
5 Levels and Stairs

1. Load project "Step2" and save it as "Step6".
2. Draw a stair as shown on the right, using new colors for steps, up and down cells and define their properties via the *Color definition* dialogue. The cell types have the following properties:
   Up defines the first row of cells, so basically the lower end of a stair.
   Down defines the last row of cells, so basically the top end of a stair.
   Steps are used to model the cells within the up and down cells or just single steps which result in no change of level.
3. Only the information of the cells is relevant for your simulation model, so the orientation of the line elements assigning this information is irrelevant. That is why it is more efficient to use longitudinal lines to assign the *Step*-cells in this example.
4. For details regarding stair modeling, see the manual.
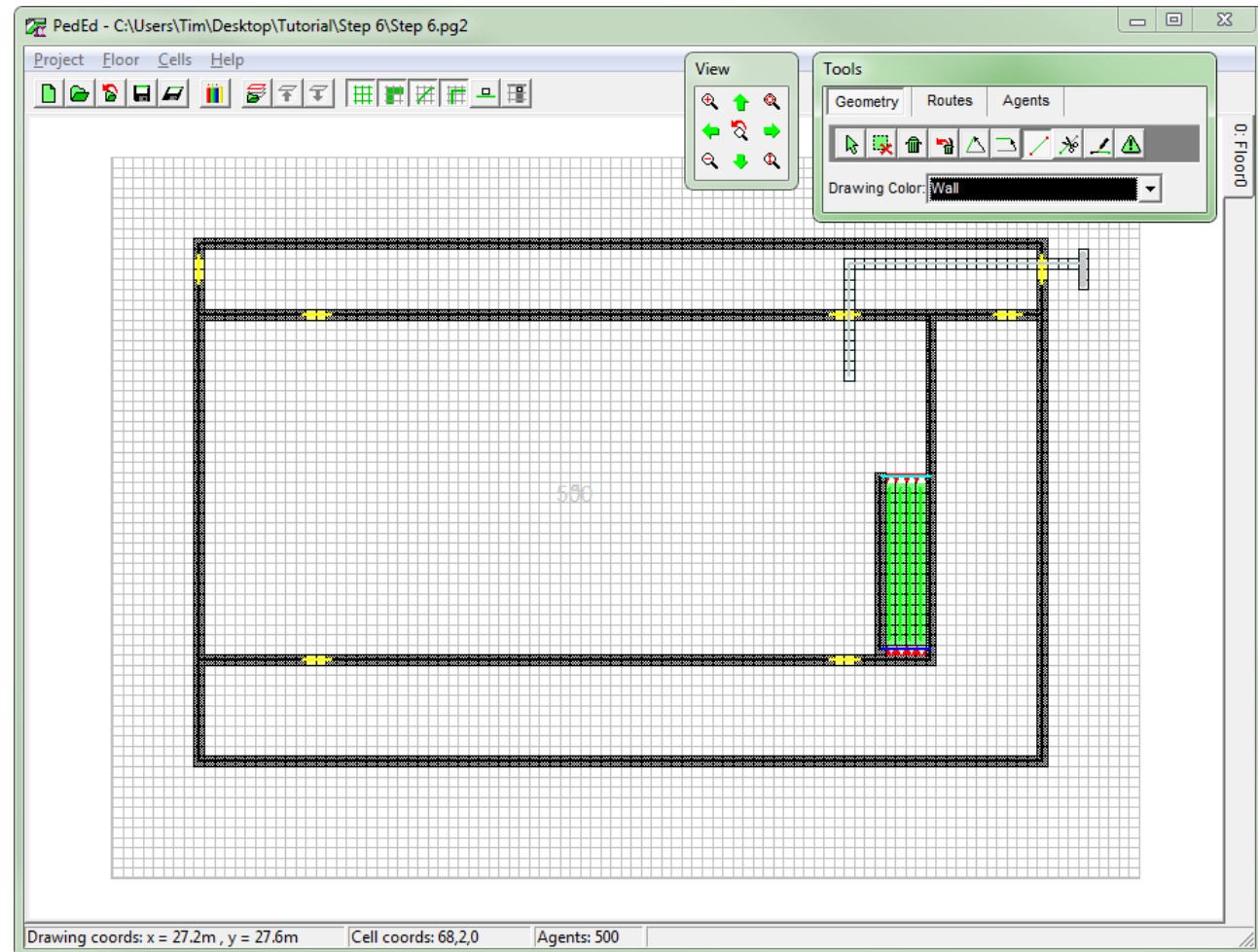
5.1 Duplicating a Floor

1. Copy the floor via *Floor*→Duplicate. The Editor proposes a name for the new floor ("Floor1") and a level above your current floor. You can choose, what to copy, but for now, you can disable everything, so only the geometry will be duplicated. Press *Ok*.

2. Your project now has two floors which can be selected via the tabs at the right side.
   Switch to the upper floor and delete 🗑 and recolor ✎ the elements, so it looks like shown on the right. As an alternative, you can select elements (e.g. Ctrl + left mouse button for dragging a rectangle) and delete them by pressing the Del key.

3. Add agents to the level.

4. Save and run the project in the simulation.

5. As you can see, the agents will change the floor when passing between down and step cells. Generally, stairs are always projected onto the bottom floor. Although a complete stair run is modeled in this example, one row of down and step cells is sufficient.
   In order for the stairs to work, floors have to be aligned on top of each other.
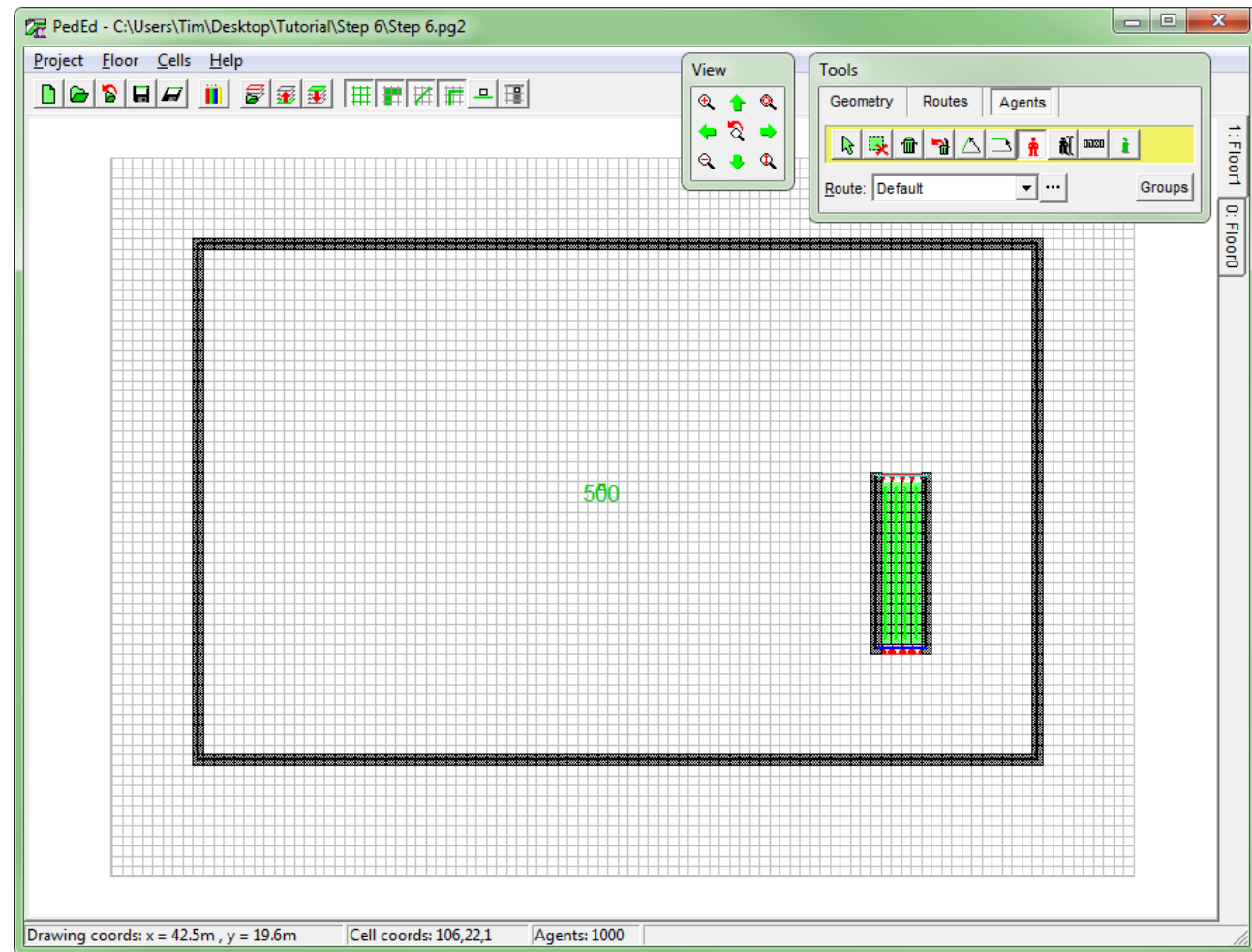
5.2   Another Stair Type

1. Add a stair with a platform and a turn as shown.
2. As you can see, the stair is fully modeled only on the bottom floor with the physical stair being bordered by up and down cells.
3. For the upper part (shown in the red encircled inlet) only the necessary elements have been used, being wall cells for the edges, one row of down cells and the adjacent step cells. For this being the uppermost floor, this modeling is sufficient.
4. Agents of the upper floor will now use both stairs, depending on their initial position. If you want them to use only one stair, mark its up and down cells by route lines as shown earlier for the door cells in order to control the spread of the directional index.
5. Save and run the project in the simulation.
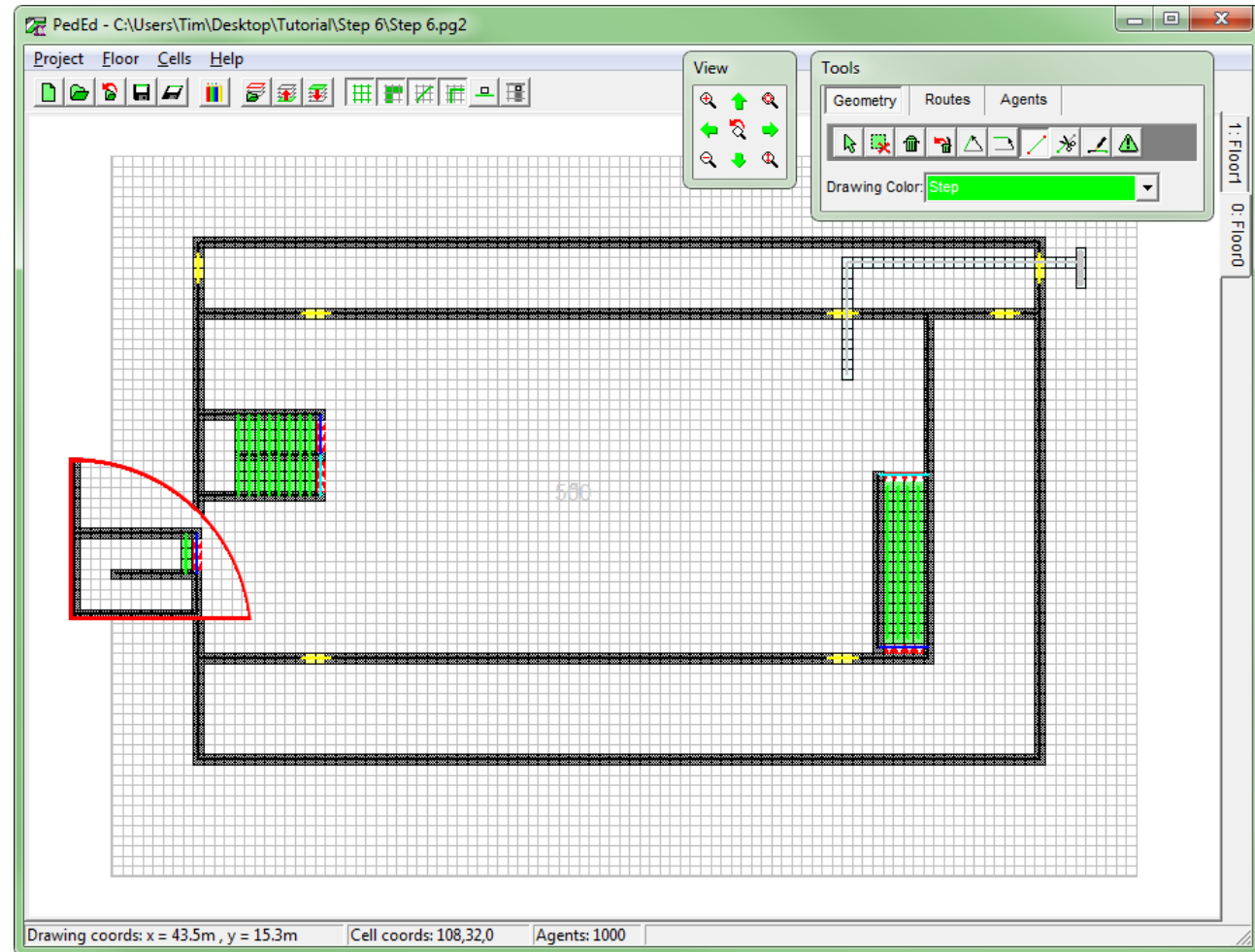
6 Groups and Parameters

1. Load project "Step 2" and save it as "Step 7".
2. Bisect the amount of agents. Press button 🔧 in the *Agents* tab of the *Tools* window and click on the agent element. Edit the amount in the dialog window.
3. Open the groups dialog by pressing Groups.
4. Edit the name of the existing group into "Fast" by clicking on it and add a second group by pressing the *Add* button and name it "Slow".
5. Add agents to the middle room (in this case 250), assign them to route "Default" but also assign them to the new group "Slow".
6. Save your project which should look like shown on the right.
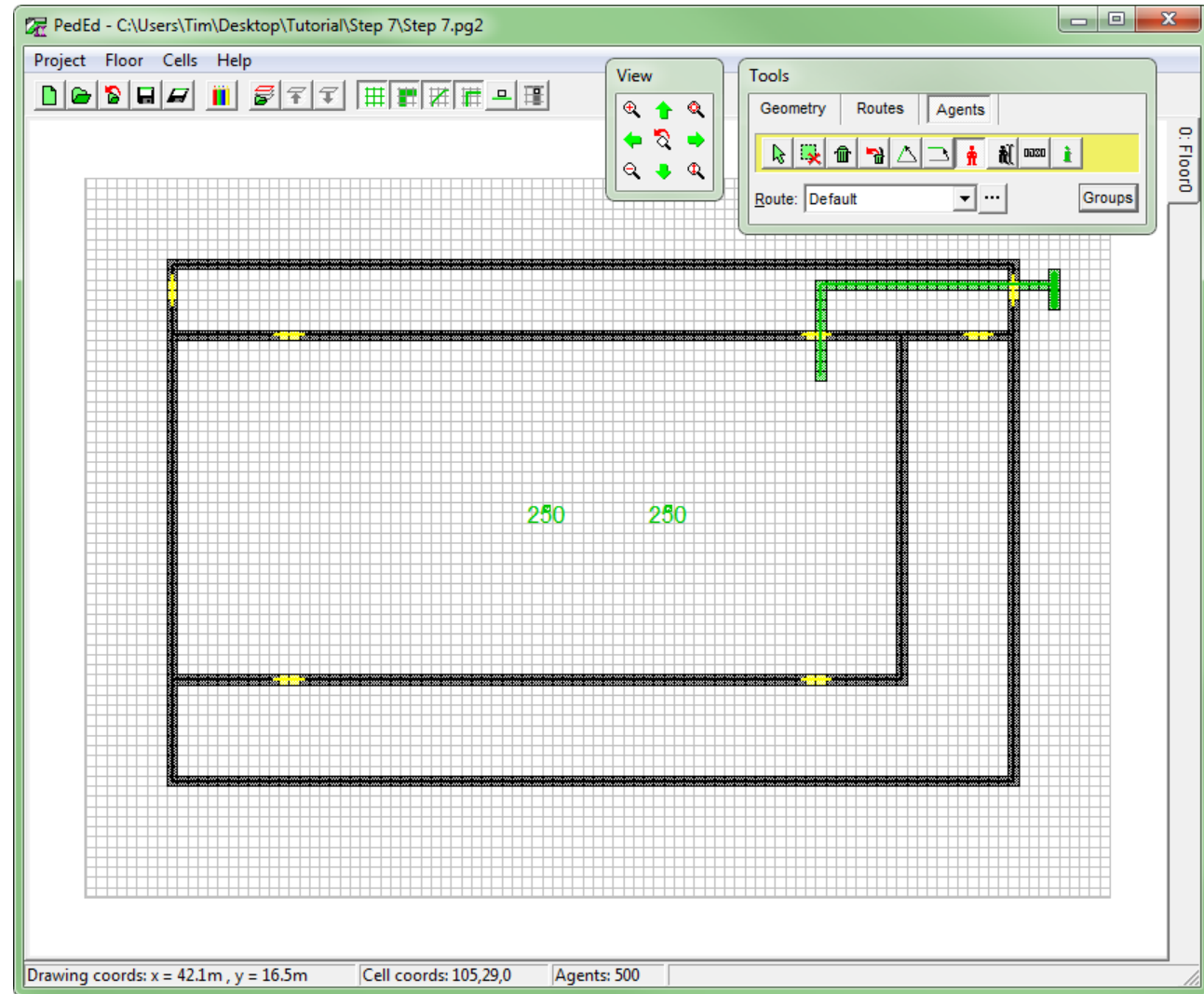
6.1 Demographics: Reaction

1. Switch to the Simulation.
2. Open the demographics dialog by pressing 🔧.
3. Choose group "Slow".
4. Edit the reaction parameter distribution from normal 📈 to equal ⬜.
5. Set the reaction duration of the "Slow" group to 6 min (=360 s), so the min and max value are equal.
6. Press *Ok* and run the simulation. As you can see, the "Fast" group starts walking immediately, while the "Slow" group reacts at a later point.
7. Open the demographics dialog again and set the min value of the reaction parameter of group "Slow" to 0.
8. Run the simulation and see, how the agents of group "Slow" now start moving in a time frame of 0 to 6 min.
9. Save the project and reload it in the Editor by pressing 🔧.
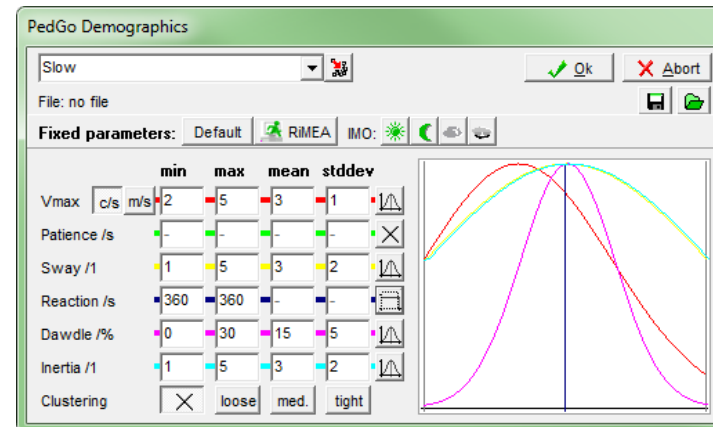
6.2   Demographics: Clustering

1. Delete all groups in the *Groups* dialog by pressing the *Delete* button.
2. We now want to create several small groups which react at different durations and stay together (=cluster) while walking.
3. Create six groups "Cluster 1" to "Cluster 6" in the *Groups* dialog.
4. Add six groups as shown on the right (in this case 50 agents each).
5. Save the project and reload it in the Simulation.
6. Set the parameters of each group as follows:
   Cluster 1: React: 0 s, Cluster: tight
   Cluster 2: R: 300 s, Cluster: tight
   Cluster 3: R: 600 s, Cluster: tight
   Cluster 4: R: 900 s, Cluster: tight
   Cluster 5: R: 1200 s, Cluster: tight
   Cluster 6: R: 1500 s, Cluster: tight
7. Save the project and run the simulation. Note, how each cluster starts walking at a different time and how the agents cluster. If you model a more complex geometry with bigger distances, the cluster effect will become more visible.
   Clusters will try not to mix with other clusters and can be used to simulate people evacuating in groups.
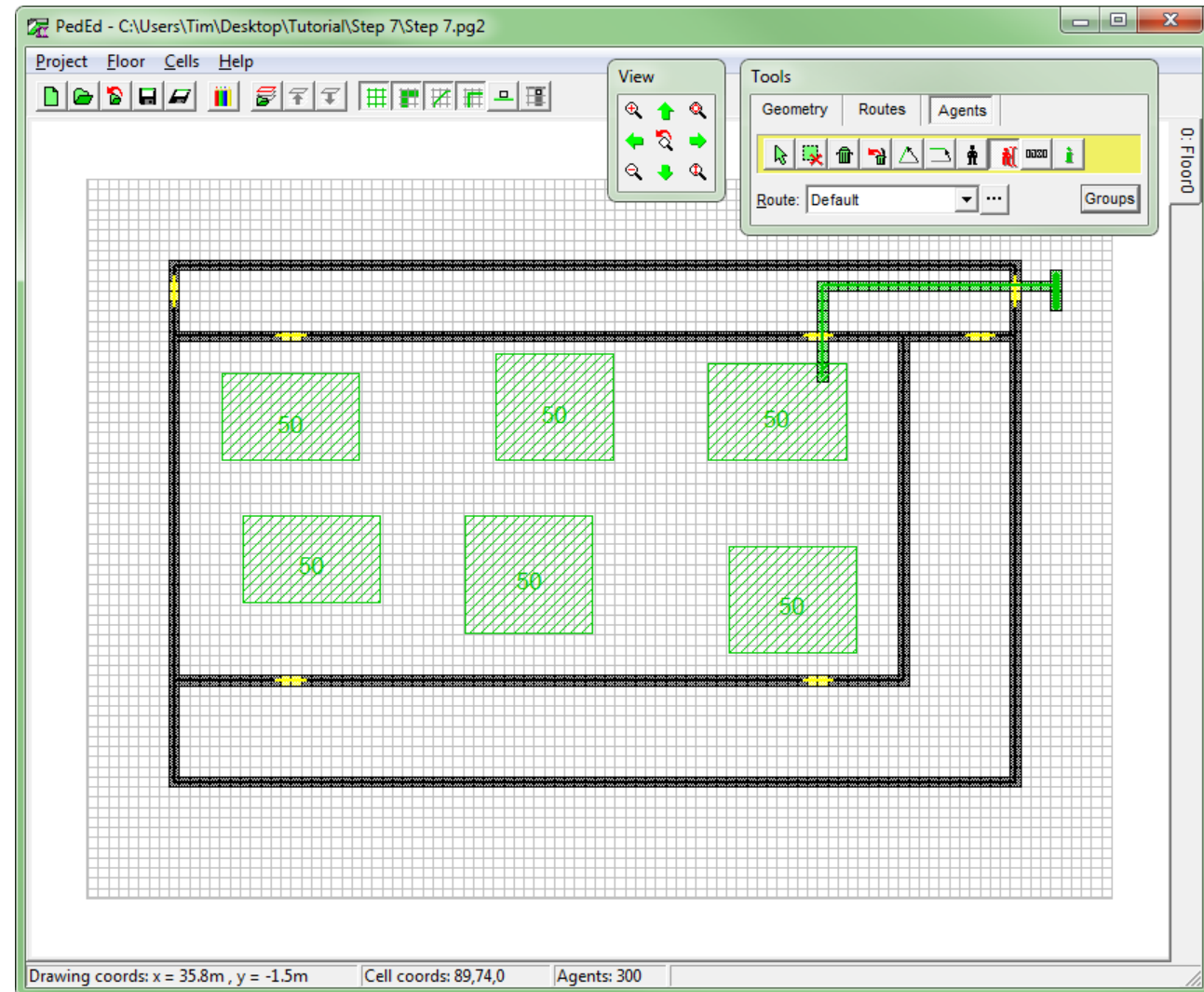
7   Evaluation

1. Open project "Step 7" in the Editor.
2. It sometimes is necessary to log the amount of persons over time in a certain area. For this, log points can be used.
3. Add log points to the middle room and the top corridor. Click on 🔲 in the *Agents* Tab of the *Tools* window. Then click into the rooms to add the log points. Give them individual names as shown.
4. Save the project and open it in the Simulation.
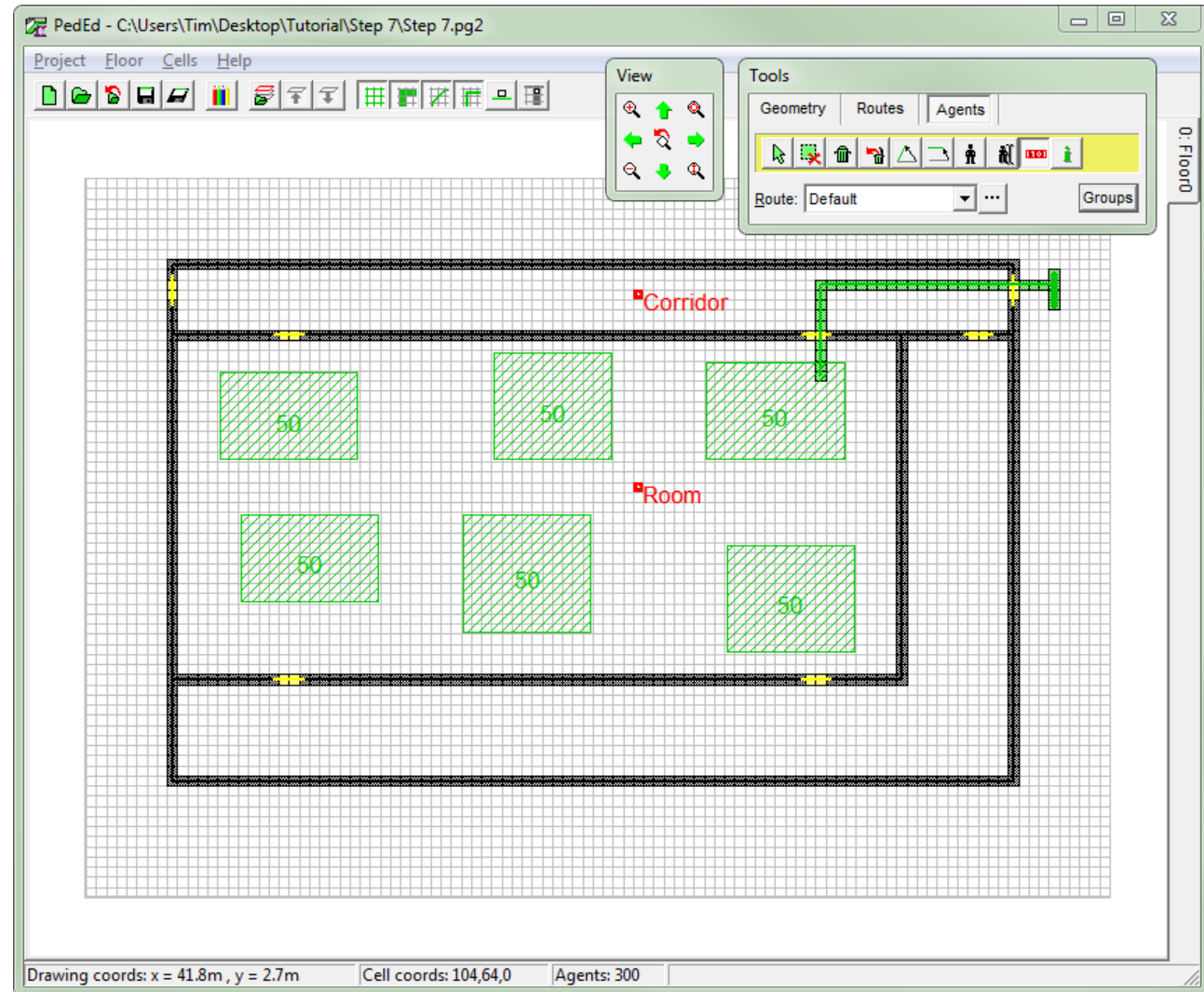
7.1   Mean Run

1. Since many Parameters are defined by statistic distributions, PedGo is optimized towards statistical analyses. This means, that a large amount of simulation runs is required to gain a reliable result. This is why we start with a mean run simulation by pressing ⊠.

2. For explanations about the input values see the manual.

3. Please note, that many log points and/or large rooms with log points significantly reduce the calculation speed, so if log point data is not required for the statistical analysis, this data should not be generated during the mean run.

4. Start the mean run simulation by pressing *Ok*.

5. PedGo now performs run after run. The newly opened duration distribution window thereby shows, which duration has been calculated how many times.

6. You can speed up the calculation by disabling the update of the situation plot by pressing 🏃.

7. PedGo signals the end of the mean run by a beep.

Mean run settings

Starting seed: 4711

Number of simulations: 500

Save results in:

C:\Users\Tim\Desktop\Tutorial\Step 7\results\

☑ Duration distribution diagram

☑ Log point files

☐ Hibernate after simulation is finished

✓ Ok   ✗ Abort
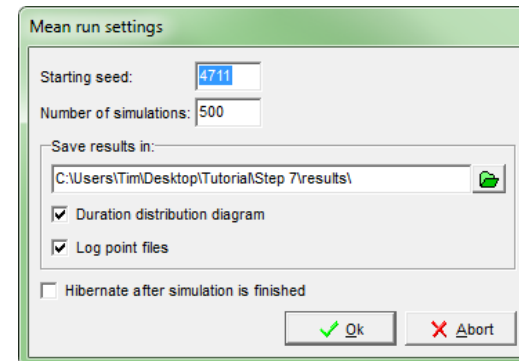
7.2 Mean Run Results

1. In the results directory, you will find a file called
   "seed=4711-5210.xls"
   By default, PedGo will create the results directory in the project directory. However, you can also define another path.
   This file is an ASCII formatted text file with tab separated values, so you can open it in any spreadsheet software.
2. For detailed information about the result files, please see the manual.
3. For the detailed analysis, the part of the mean run result file shown on the right is important. It shows, which run (defined by its seed) represents which result.
4. It is common practice, to use the significant run for a detailed analysis. This is the run delivering a duration which is larger than 95% of all runs. Its duration is marked by a green column in the duration distribution diagram, which is also generated by PedGo. In this example, the corresponding seed is 4905, it can be a different value for your project.

Duration Distribution File:
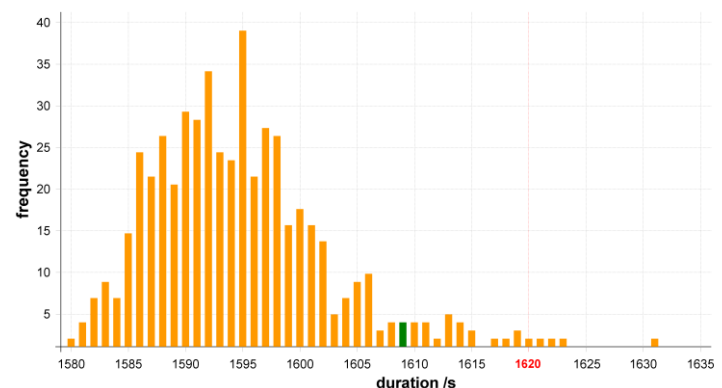
Content of Mean Run Result File:

	/s	/(h:min:s)	seed
15			
16	/s	/(h:min:s)	seed
17 mean	1595	00:26:35	4711
18 stddev	8	00:00:08	---
19 significant	1609	00:26:49	4905
20 min	1580	00:26:20	4737
21 max	1631	00:27:11	5087
22			

7.3 Single Run

1. Start a single run by pressing ⊞.
2. Enter the seed of the significant run as determined in 7.2.
3. You can select, which individual results should be saved or all by clicking on *toggle all*.
4. For detailed information about all results, please see the manual.
5. After pressing *Ok*, the single run will start. Notice, how the simulation runs slower, since more data is gathered, which affords computational time.
6. Once again, the end of the simulation is marked by a beep.
7. The generated 3dl file can be opened by the Viewer which you can download from our homepage.

Single run settings

Seed: 4905 ✓ Ok ✗ Abort

Save results:

☑ toggle all

☑ Agent data: C:\...\Tutorial\Step 7\results\seed=4905.xls

☑ Animation: C:\...\Tutorial\Step 7\results\seed=4905.avi

☑ Density plot: C:\...\Step 7\results\Density,seed=4905.bmp

☑ 3D log file: C:\...\Tutorial\Step 7\results\seed=4905.3dl   0   --> END

☑ Log points: C:\...\Step 7\results\LogPoints,seed=4905.xls

☑ Screen shots C:\Users\Tim\Desktop\Tutorial\Step 7\results\   Interval: 60
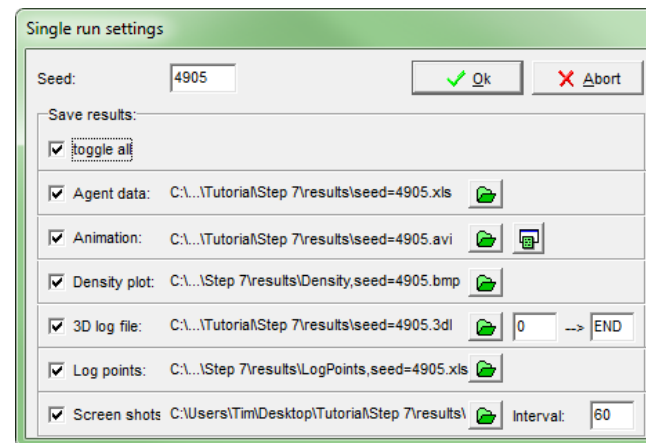
8 Dxf Import

1. The Editor offers the ability to import dxf files in order to speed up the process of modeling.

2. Press 📑 to choose a dxf file which will be imported as a new floor.

3. After the file is chosen, the dialog shown on the right opens. Choose the unit of the imported file. The numeric dimensions are shown, to help you choose the right unit.

4. Dxf elements under a certain size can be neglected by the import. After *Ok* is pressed, the Editor will discretize the dxf data onto the Grid, thereby converting the dxf elements to discrete line elements with the nodes rounded onto the grid. You can also define a minimum size for the imported, discrete elements.

5. The discretization sometimes leads to rounding errors, so the imported geometry should be checked and altered by manually.

6. For big projects with many elements, it might be useful to use the *Floor→Cleanup* command. It selects superfluous elements, which can then be deleted by pressing the Del key.

DXF Import

Select unit used in the drawing file:
(numeric dimensions 59.3225×42.85)

Units
- ○ mm   ○ inch   ○ ft   ○ km
- ○ cm   ○ dm   ◉ m

☐ Discard DXF elements smaller than [0.2] m

☐ Discard discrete elements smaller than [0.2] m

✓ Ok   ✗ Cancel