SPARK-PL: Installation and User Interface

Alexey Solovyev

Abstract

Table of Contents

1. Installation of SPARK-PL

SPARK is a Java library for developing agent based models in Java. SPARK-PL is a special programming language which greatly facilitate the model developing process. Each SPARK-PL distribution includes a copy of the latest SPARK library along with all necessary tools to implement and run your own SPARK models.

First of all you need to obtain a copy of SPARK-PL. It can be downloaded from the official SPARK site at http://www.pitt.edu/~cirm/spark [http://www.pitt.edu/~cirm/spark] in the Download section. There you need to select the latest version of a SPARK-PL distribution for your operation system. Also you can download a universal distribution which works on any platform. This universal distribution is smaller than other distributions but it lacks fast graphics.

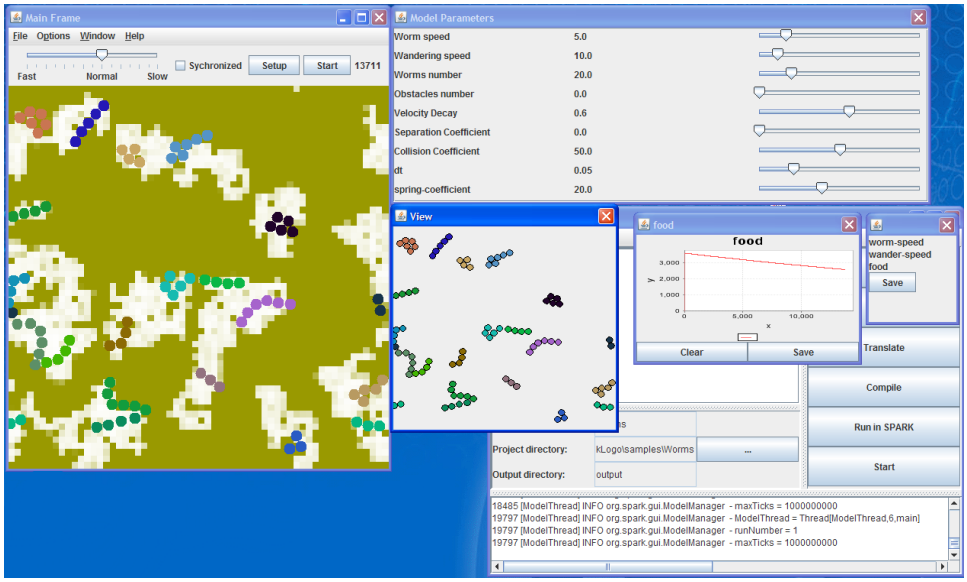
You don't need to get anything else except Java runtime of version 1.5 or higher. All modern operation systems have this version of Java already installed. You can test whether you have Java or not by visiting the Web page http://java.com/en/download/help/testvm.xml [http://java.com/en/download/help/testvm.xml]

The installation process of SPARK-PL itself is very simple. Extract the distribution archive into any folder on your computer and you are done. After that you can run SPARK-PL by opening 'SPARK Manager.jar' in Explorer, Finder, or another file manager.

2. SPARK GUI

Before creating your own model, it is a good thing to look at some sample models. After running the 'SPARK Manager.jar' click 'Open project...' in the 'File' menu. Go to the 'samples' folder, select any folder there, and open an xml file inside the selected folder. Now click the 'Start' button and the SPARK user interface will appear with the selected model (here a screenshot of Worms model is shown).

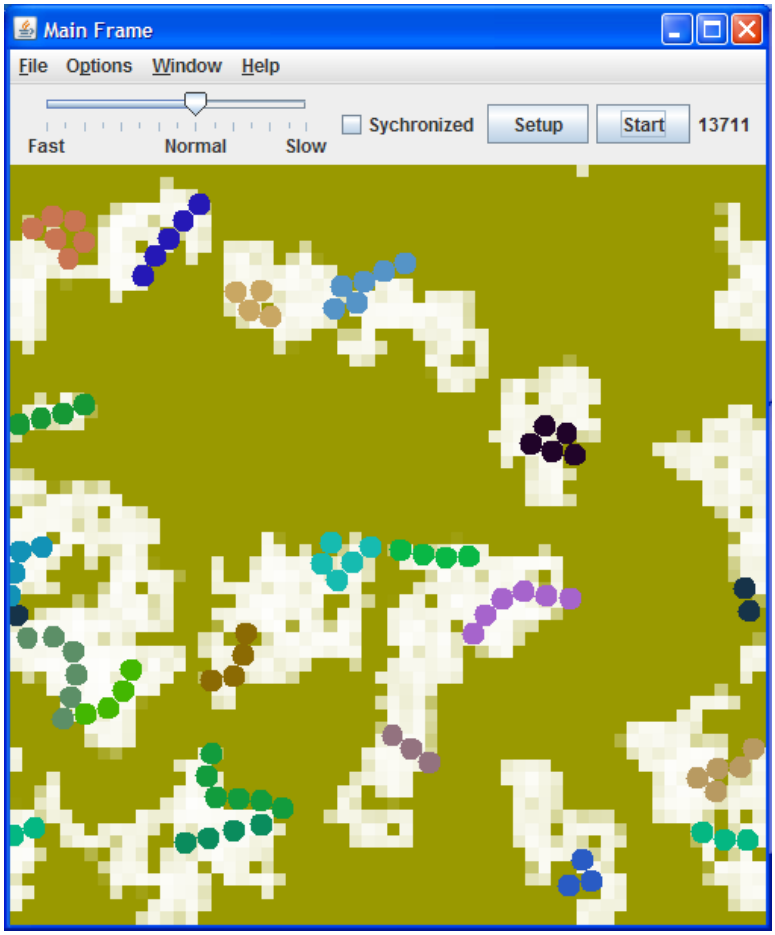
Figure 1. SPARK GUI



2.1. Main Window

The SPARK user interface consists of several windows. First of all look at the main window with simulation control elements inside it.

Figure 2. Main Window

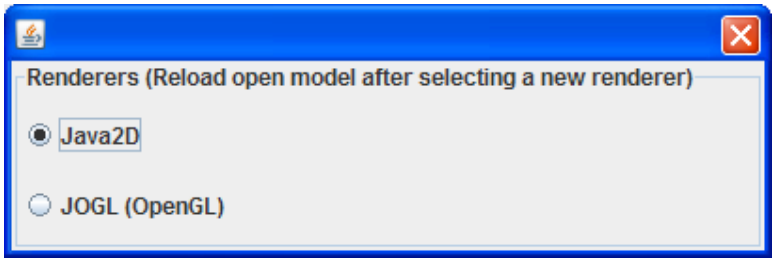


The button 'Start' starts the simulation. When this button is clicked, then it will become the 'Pause' button which can be used for pausing the active simulation process. The button 'Setup' stops the simulation and initializes the model again. Also it resets all information gathered during the previous simulation process. The slider on the left controls the simulation speed. The 'Synchronized' check box changes the way in which the simulation process and visualization interact. If it is checked, then the simulation process will always wait for visualization process to redraw everything completely before starting a new simulation step. If it is unchecked, then the simulation and visualization processes will work independently.

The first menu of the main window is the 'File' menu. It has only one item: Open. It is used for opening SPARK models directly from the SPARK interface. Since you are using SPARK-PL, you don't need to open SPARK models directly.

The next menu is 'Options' menu. The first item there is 'Graphics'. If you select this item, then you get the following window

Figure 3. Graphics options



Here you can select a library which will be used for performing all visualization operations. OpenGL is a fast library, but Java2D is more robust. If you have an universal SPARK-PL distribution, then you cannot select OpenGL option here (in fact, you can do this, but you get an error). If you have another version of SPARK-PL and have some problems with graphics or get some errors referring to 'jogl', then select Java2d here. In any case, after changing a graphic library, you need to close the SPARK user interface and run it again.

The next item in the 'Options' menu is 'Batch Run Parameters'. It is used for running the same model several times automatically (see below about saving simulation data for more details). The last item is 'Data Layer Parameters'.

Figure 4. Data layers parameters

Here you can specify the visualization properties of data layers (whatever they are). You can change two values and colors associated with that values. The first value corresponds to the minimum value of a data layer and the second value corresponds to the maximum value of a data layer. These values can be arbitrary, they do not affect data layers but only used for their visualization. If you want to get exact minimum and maximum values for a data layer, then click 'Normalize' button.

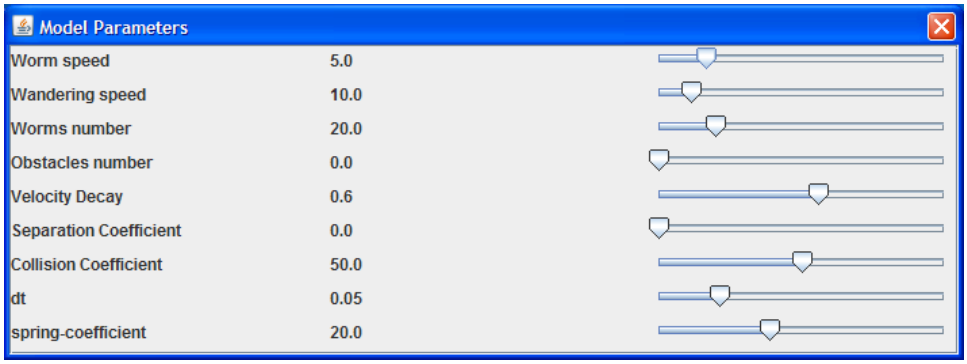
The next menu is 'Window' menu. The first item 'New view' creates a new 'View' window where a simulation process is visualized. In SPARK you can have several visualization windows with different visualization parameters. Next items in the 'Window' menu contain the names of all open windows. A check box on the left for each window indicates whether the window is visible or not. If you click on a window name, then you either make it visible or hide depending on its current state.

The last menu 'Help' contains information about SPARK.

2.2. Parameters window

The next window to look at is the parameters window.

Figure 5. Parameters window



It contains all parameters of the current model. Parameters are model variables that can be changed during a simulation process. You can change them before starting a simulation or when a simulation runs in real time. Some parameters represented by sliders (numerical parameters). Other parameters represented by check boxes (boolean parameters).

2.3. Chart windows

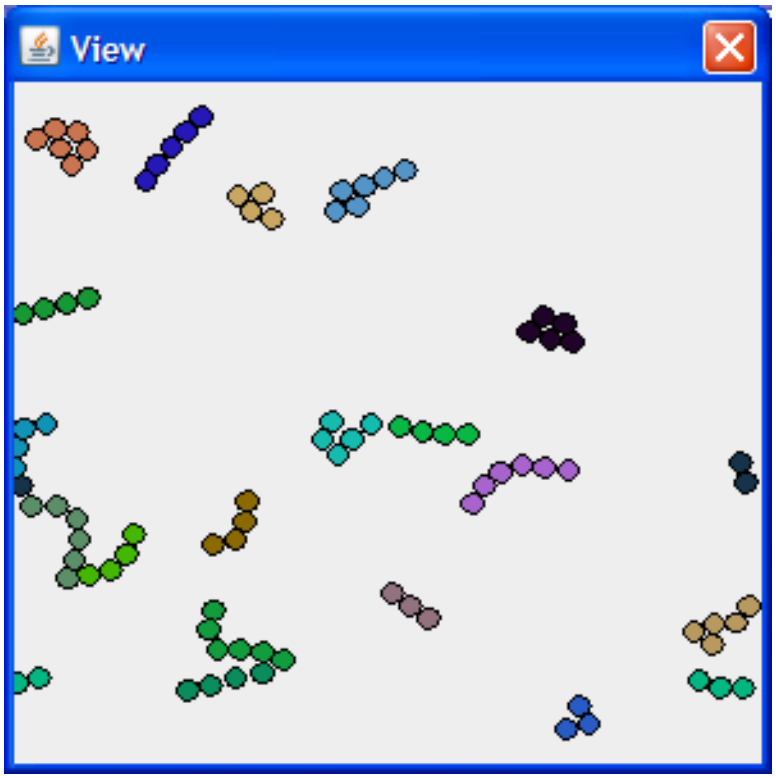
There could be several chart windows in a model.

Figure 6. A chart

Each chart window shows a plot of some model variable. If you click the 'Clear' button, then the plot area will be cleared. If you click the 'Save' button, then you can select a file in which to save the numerical data of a plot.

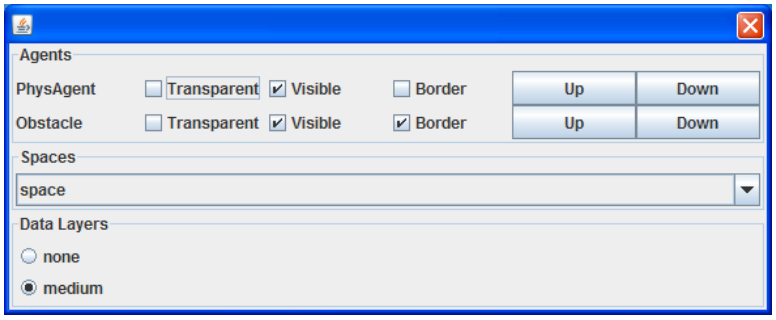
2.4. View windows

Figure 7. A view window



These windows give the additional visualizations of the running simulation. You can customize each view window and the main window view by right clicking on them and selecting 'Properties' from the pop-up menu. Also you can remove any additional view window by selecting 'Remove view' from the same pop-up menu. 'Properties' opens a visualization properties dialog.

Figure 8. Visualization Properties dialog

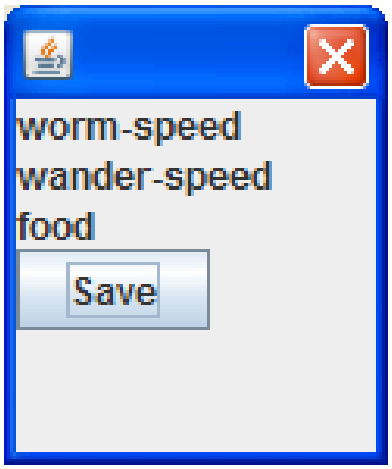


Here you can set visualization properties of agents: they can be transparent, visible/invisible, with or without borders. Also you specify the order in which agents will be visualized. Agents on the top of the list are drawn atop of other agents. In the 'Spaces' section you can select which space is visualized in the given window. In the 'Data Layers' section you can select a data layer to be visualized.

2.5. Dataset window

It is possible to save the data collected during the simulation. Just click the Save button in the Dataset window.

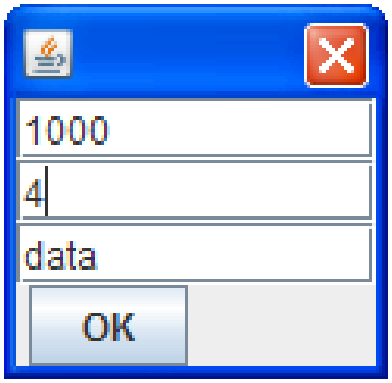
Figure 9. Dataset window



The standard dialog will appear where you can specify the name of a data file. All data will be saved as a plain text CSV file which can be easily opened in Microsoft Excel. Note that the saved data will also contain the current values of all parameters.

Sometime it is required to run a model several times and save the results after each run. To simplify this procedure, you can use batch run feature of SPARK. Go to Options menu and click Batch Run Parameters.

Figure 10. Batchrun Dialog



The first line in the batch run dialog is the number of steps (ticks) in each run. The second line is the number of runs. And the third line is the [name of file] where results will be saved. Batch run works as follows: first you specify all parameters and click OK. Then you click Setup button and Start button (you may also turn off synchronization and close some windows for increasing simulation speed). After that the simulation will run exactly specified number of steps. When the maximal number of steps is reached, the results are automatically saved in the file named [name of file][#run].csv, where [name of file] is the name specified in the batch run parameters and [#run] is the number of the current run. After the results are saved, model is reseted and a new simulation starts.