

SwarmVis - A Tool For Visualizing Swarm Systems

User Guide

Don Miner and Niels Kasch

December 5, 2008

Contents

1	Introduction	3
2	The User Interface	3
2.1	The Tab Controls	3
2.1.1	The Agents Tab	3
2.1.2	The Tracks Tab	3
2.1.3	The Types Tab	3
2.1.4	The Velocity Tab	5
2.1.5	The Misc Tab	5
2.2	The Playback Control	5
2.3	The Render/OpenGL Control	5
2.4	The Menu Bar	5
3	Compiling and Running	5
4	Data File Format	6

1 Introduction

This document describes the user interface of SwarmVis, a tool that allows swarm researchers to visualize swarm systems in two and three dimensional space. The tool enables researchers to track swarms and agents of a swarm in a continuous time simulation. It is the goal of the toolkit to provide insight in the development and evolution of swarms systems by allowing researchers to apply agent and swarm level visualization while browsing back and forth through the discrete time steps in the lifetime of a swarm.

2 The User Interface

SwarmVis consists of graphical user interface (GUI) in the form of a window embedded with control and rendering widgets. The main window is divided into three major components:

1. a tab control - allowing access to various visualizations and settings (Figure 1(a))
2. a playback control - controlling the progression of time frames(Figure 1(c))
3. a rendering window - responsible for rendering the visualizations (Figure 1(b))
4. a menu bar - allowing the loading of datasets (Figure 1(d))

2.1 The Tab Controls

2.1.1 The Agents Tab

The *Agents* tab facilitates selection of features for individual agents. Using the *Agents* tab, a user is able to manipulate the visual representation of an agent, such as color and size. To change the color of an agent, select the agents followed by pressing the *Select Color* button. A color dialog is displayed facilitating the selection of an RGB color. The size of an agent can be adjusted via a *spinbox* and the *Set Size* button. Multiple agents can be selected by (1) dragging the mouse over multiple agents, (2) pressing and holding the **CTRL** key on the keyboard while clicking on individual agents in the list (individual selection) and (3) pressing and holding the **Shift** key (continuous selection).

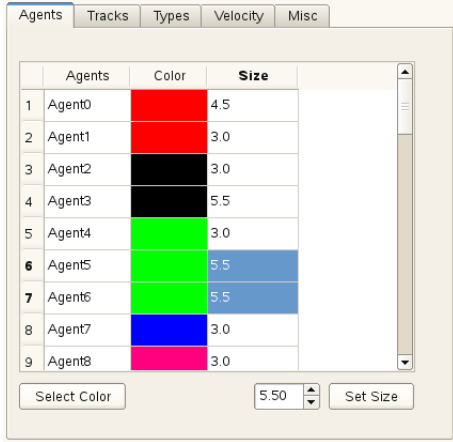
2.1.2 The Tracks Tab

The *Tracks* tab allows a user to track the path of selected agents over the entire simulation. Tracks can be visualized using the specific color of an agent or a selected general color for all agents. Visualizing tracks with agent specific colors allows users to provide an instant overview of an agents past and future positions. This visualization technique is particularly useful during “freeze frame” analysis. Select an agent to visualize its track. Multiple selections are allows similarly to selecting agents on the *Agents* tab.

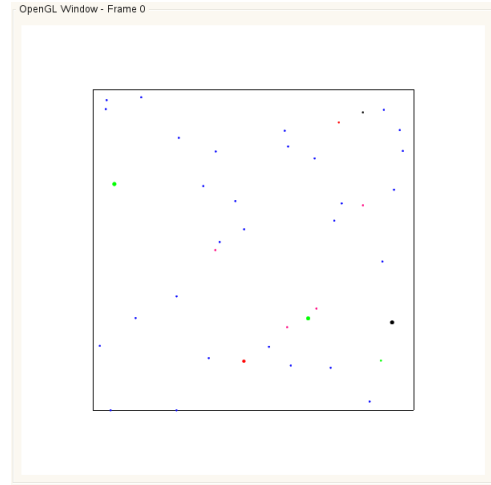
Tails can be enabled via the *tails spinbox* provided on the tab. The tail view shows the position of all agents in previous time steps. The length of the trail can be adjusted by increasing the value of the spinbox. Tails progressively fade according to the tails temporal distance in relation to the current time step.

2.1.3 The Types Tab

The *Types* tab shows an overview of the different types of agents and provides a facility to distinguish the types according to selected colors. Particular datasets can annotate the data with several types of agents. Each agent type may operate according to different rules and hence behaves differently. The *Types* tab facilitates visual distinction of types which makes the task of tracking types easier for a user.



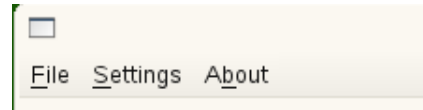
(a) The Tab Control gives access to the various visualization capabilities in SwarmVis. The image illustrates the *Agents Tab* where the color and size of individual agents can be manipulated.



(b) The Render (OpenGL) window where all visualizations are rendered. A typical visualization includes a bounding box for reference.



(c) The Playback Control enables the *browsing* through the visual representation of a swarm dataset. The *Play* button starts a continuous time simulation of a dataset whereas the *slider* allows for the rewinding and fast-forwarding of the simulation.



(d) The File Menu allows for the loading of datasets. The Settings Menu can be used to load/save user specific settings for a particular visualization.

Figure 1: The SwarmVis User Interface.

2.1.4 The Velocity Tab

The *Velocity* tab implements functionality to visualize the velocity of the agents. A rainbow scale shown on the tab, maps color according to speed. When enabled the agents and/or agent tracks/trails change color according to the speed an agent travels. It is noteworthy that track segments are colored according to the speed at which the agent was traveling for a particular segment.

2.1.5 The Misc Tab

The *Misc* tab provides additional functionalities to control the visualization. Via a *checkbox*, a user can enable or disable the bounding box surrounding the visualization. The bounding box can be useful in providing context when the view window is rotated or its size is changed. OpenGL depth checking can be enabled via the respective checkbox. Depth checking is a feature of OpenGL to access the depth of a pixel from the viewing plane. This allows a pixel at a certain depth to assess whether or not other pixels overlay the current viewport.

2.2 The Playback Control

The playback control functionality is comparable to a standard media player. It allows for the playing, pausing, rewinding and fast-forwarding of the swarm animation in the render window. A slider bar is provided to give an overview and control over the currently displayed frame. Advancing and moving back the slider skips to future and past time frames, respectively.

2.3 The Render/OpenGL Control

The render/OpenGL window displays the frames of the swarm animation. Since SwarmVis supports 3D visualizations of swarm systems, a bounding box for providing orientation in 3D space can be enabled via the *Misc Tab*. The render window supports zooming and rotating the view. The zoom feature can be activated by hovering a mouse over the render window and utilizing the scroll wheel on a mouse (scroll up = zoom in; scroll down = zoom out). Rotation of the view can be accomplished by clicking the left or right mouse button while simultaneously moving the mouse. The left mouse button activates rotation over the x and y axes, whereas the right mouse button activates rotation over the x and z axes.

2.4 The Menu Bar

The menu bar consists of three components, namely the *File*, *Settings* and *About* menus. The *File* menu allows a user to (1) exit the application and (2) load swarm datasets to be visualized. The swarm dataset are loaded through an information file (the structure of the information file is described in Section 4) which can be selected via an open file dialog.

The *Settings* menu enables a user to load and save the settings (i.e. color and size of agents, time frame displayed in the render window and state of all check boxes and selected agents) for a visualization. *Save Settings* stores all settings in a file specified by the user. *Load Settings* restores the settings from a selected file. This feature permits a user to save and restore “interesting” situations or occurrences in a visualization.

The *About* menu simply displays information about the authors of SwarmVis.

3 Compiling and Running

SwarmVis is a cross-platform tool and is intended to run on Unix-like systems, such as Mac OS X and GNU/Linux, as well as Microsoft Windows. The source code is available at:

<http://code.google.com/p/swarm-vis/>

The following programs are required to compile SwarmVis from source:

- Qt 4.2¹
- gcc/g++ 4.2.2
- make

Note that some libraries from Qt4.2 may be needed to run the SwarmVis binary. We have tested SwarmVis on Mac OSX 10.5 and openSUSE Linux 11.

Run `qmake`, and then `make`, both in SwarmVis' root directory (where the file `swarmvis.pro` is located), to compile SwarmVis. A binary will be created in the `bin/` directory. Execute this binary to run SwarmVis.

4 Data File Format

SwarmVis requires a specific format for data sets that are to be loaded. The agents' position data is segmented into separate files that each represent a single time step. These files contain space-delimited data, with each row representing an agent. For example, a swarm system with 100 agents depicted over 500 time steps will have 500 files in a folder, each with 100 lines of text.

Each row entry in a file must follow a specific format. The first two or three columns (depending on dimensionality) are the position data (X, Y) or (X, Y, Z) , respectively. The last column is reserved for the group label, which may be used to pass group membership data to SwarmVis. For example, a well-formed entry that conveys a three-dimensional position with group information could be:

```
10.15 5.24 84.85 Corner
```

The order of the listing of agents in each file must be consistent. That is, the k^{th} entry in one file and the k^{th} entry in another file must represent the same agent.

A plain-text information file containing important meta-data must accompany the frame files in the same directory. The following variables must be defined (i.e., `VARNAME = VALUE`) in this file in order for the data to be loaded appropriately:

- `DIMENSIONS` (2 or 3)
- `AGENTS` (the number of agents)
- `FRAMES` (the number frames/time steps)
- `RANGEX` (the maximum X value)
- `RANGEY` (the maximum Y value)
- `RANGEZ` (the maximum Z value)
- `AGENTTYPES` (1 to track agent types, 0 if not)

At the bottom of this file, the keyword `FILES` must appear, followed by a list of frame files, in ascending temporal order. The number of files listed here must equal the number specified by the `FRAMES` variable. Also, the number of lines in every file must match the number specified by the `AGENTS` variable. To load a data set in the SwarmVis application, navigate to "Load Data" and select the info file.

A sample info file for a 150 agent three-dimensional swarm over the course of 446 frames:

```
DIMENSIONS = 3
AGENTS = 150
FRAMES = 446
RANGEX = 600
RANGEY = 600
```

¹Qt is an open source cross-platform application framework. It is available at <http://trolltech.com/downloads>

```
RANGEZ = 600
AGENTTYPES = 1
FILES
frame000001.txt
frame000002.txt
frame000003.txt
frame000004.txt
frame000005.txt
frame000006.txt
...
frame000442.txt
frame000443.txt
frame000444.txt
frame000445.txt
frame000446.txt
```