

A REVIEW OF SOFTWARE FOR CROWD SIMULATION

Author: Thomas Richards, Data Science Intern, Leeds Institute for Data Science ([LIDA](#)), [University of Leeds](#).

AIM

To review a number of different software libraries and platforms that can be used to create agent-based pedestrian simulations, in particular to find a library that will allow us to use data assimilation to update the state of the model at runtime.

METHOD

Search Google, GitHub, SourceForge, Wikipedia etc, and personal suggestions/recommendations.

Priorities are software which: 1. able to run agent-based pedestrian simulations; 2. is open source; 3. is modular/accessible; 4. Allows run-time intervention (e.g. to add, change, remove agents whilst simulation is running); 4. uses a language popular with the research team, namely Python and/or Java.

Software that have not been updated or appear to be no longer supported, have no documentation, or are unsuitable have been included in the list so that there is a record they have been checked but may not include much detail.

Sections for software which was not short listed may be incomplete.

RESULTS

Models are given a rating out of three ^{*}'s. Note that these ratings indicate the *suitability of the software for the research project*. They are not a general assessment of the quality of the software; we do not suggest that a library with a high rating is inherently better than one with a low rating, just that it may be more suitable for use in the DUST project. If the authors of the software would like us to amend any inaccuracies or errors with respect to their software we will be happy to. Please contact the Principal Investigator, [Nick Malleson](#).

[SL]	Shortlisted
[NE]	Not explored.
	= No stars. Could not run or excluded for other reasons
*	= May not start, but potentially promising
**	= Looks useful but not ideal
***	= Could be ideal, recommended

CONTENTS

<i>Aim</i>	1
<i>Method</i>	1
<i>Results</i>	1
<i>Contents</i>	2
<i>Open source</i>	4
Vadere ***[SL]	4
Cromosim **[SL]	6
Menge *[SL]	8
Crowd Dynamics *[SL]	10
PEDSIM *[SL]	12
JuPedSim [SL]	14
jCrowdSimulator [NE]	16
Mesa [NE]	18
RVO2 Library [NE]	19
Fire Dynamics Simulator [NE]	21
Agents.jl: agent-based modeling framework in Julia [NE]	22
<i>Licensed or 3d</i>	24
AnyLogic Pedestrian simulation ***	24
INCONTROL Simulation Software **	26
Oasys MassMotion *	27
Quadstone Paramics *	29
UCrowds	30
CrowdSim3D [NE]	32
CrowdMaster [NE]	33
Crowd.lab [NE]	34
Urho3D [NE]	35
Golaem Crowd [NE]	35
MASSIVE [NE]	35
PTV Vissim [NE]	35
CrowdVision [NE]	36
Crowd Dynamics [NE]	36

Modelling Environments/Engines	37
Blender [NE]	37
Unity [NE]	37
NetLogo [NE].....	37
Repast [NE]	37
Mason [NE].....	37
SteerSuite [NE].....	38
Crowd Nav [NE].....	38
A crowd simulation using OpenCL™ and based on the continuum approach [NE]..	38
TEMPLATE ***[SL][NE]	39

OPEN SOURCE

VADERE ***[SL]

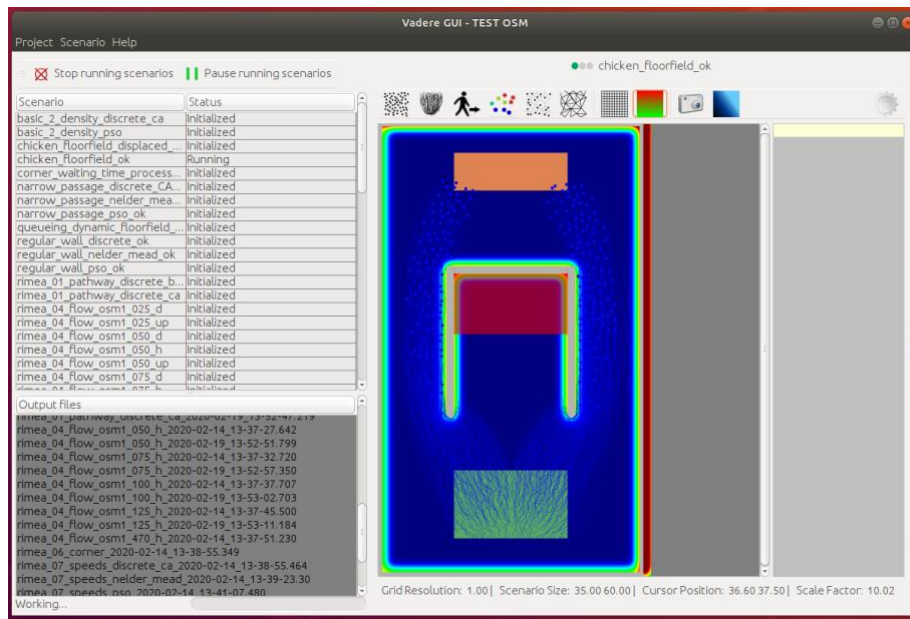
Vadere is built and maintained by the research group of Gerta Köster <gerta.koester@hm.edu> at the Department of Computer Science and Mathematics at the Munich University of Applied Sciences. The platform is open source and available for free. This is the best open source platform I came across, and Gerta seems very responsive and keen to collaborate. Gerta seems very positive that it would be easy to implement dynamic data assimilation and has students working in making it accessible from a software architecture perspective. The software itself is very nice and includes a GUI through which the code seems to be accessible (for simulation, model, topography, events etc). It looks easy to create an environment through the GUI. As the simulation runs, the visuals are clear and you can toggle features such as trajectory, density etc, and generates output files at the end through which you could presumably save data for analysis. Several models are already implemented in the program: optimal steps model, the gradient navigation model, and the social force model. Many examples are can be run straight out of the box.

URL

<http://www.vadere.org/>

Rating (/3^{*})

Documentation:	***	http://www.vadere.org/publications/
Complexity:	***	
Accessibility:	***	
Usefulness:	***	
Modularity:	***	
Maintenance:	***	



SUCCESSFUL INSTALLATION/TRIAL RUN?

Yes – Linux installation successful

COMPATIBLE OPERATING SYSTEMS

Windows, OS X and Linux

LANGUAGE?

Java

2D/3D?

2D

TUTORIALS?

Installation: <https://gitlab.lrz.de/vadere/vadere>

Tutorials: <http://www.vadere.org/video-tutorials/>

INTENDED USE

Simulation of microscopic pedestrian dynamics

MODELS IMPLEMENTED

Optimal steps model, the gradient navigation model, and the social force model

COMMAND LINE/GUI?

GUI, code accessible

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

Yes

FEATURES

Integrates EikMesh (<http://www.vadere.org/the-eikmesh-library/>) to generate meshes for spatial discretization of pedestrian dynamics.

OPEN TO COLLABORATION?

Yes

SOURCE

SourceFourge

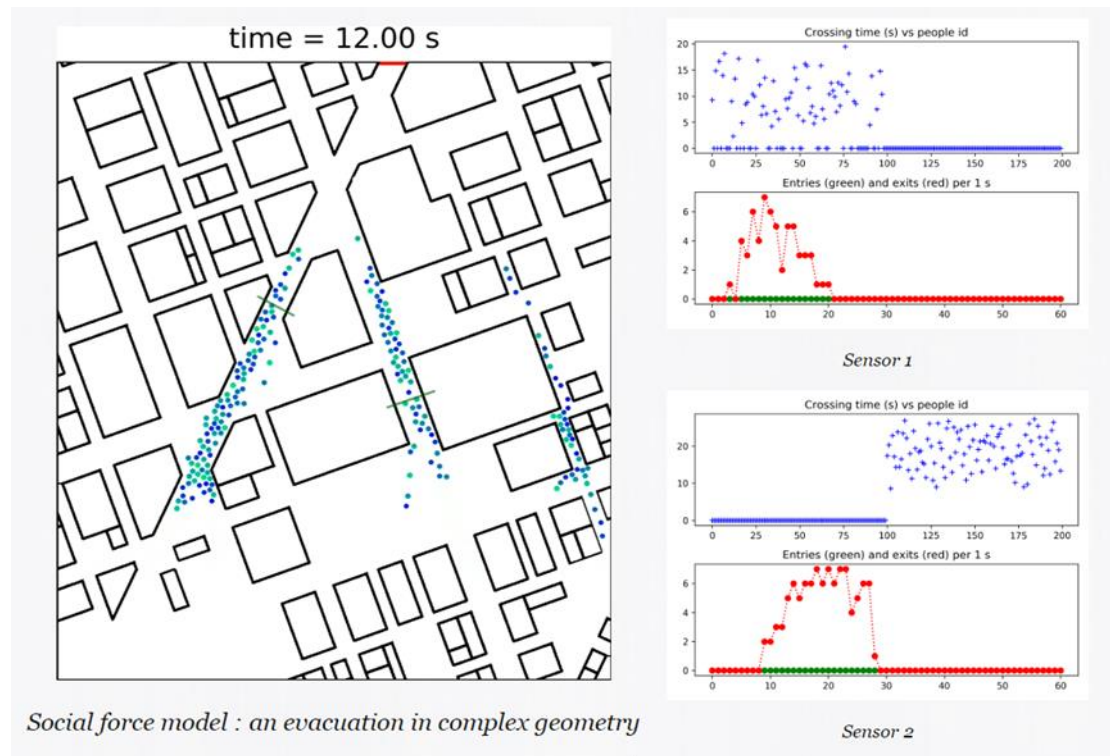
CROMOSIM **[SL]

Cromosim is a Python library for crowd motion simulation maintained by Sylvain Faure <sylvain.faure@math.u-psud.fr>. On the surface it seems to be very useful, mainly because of its simplicity of use. There are several very simple example models which run well and seem like they would be easy to integrate with each other and adapt for your purposes. On closer examination the code seems messy and not very well commented. However the examples run well and look like they could be adapted for use in an agent-based model. When you run the example models in Python, some plots are generated of data taken from the simulation (such as sensor crossings – see social force model example: http://www.cromosim.fr/example_micro.html#social-force-model).

URL

<https://github.com/sylvain-faure/cromosim>

	Rating (/3 [*])	
Documentation:	***	http://www.cromosim.fr/example_micro.html#social-force-model
Complexity:	**	
Accessibility:	***	
Usefulness:	**	
Modularity:	***	
Maintenance:	Not contacted	



SUCCESSFUL INSTALLATION/TRIAL RUN?

Yes on Windows

LANGUAGE?

Python

2D/3D?

2D

TUTORIALS?

Cellular automata: http://www.cromosim.fr/example_dom.html

Microscopic simulation: http://www.cromosim.fr/example_micro.html

INTENDED USE

Crowd motion simulation

MODELS IMPLEMENTED

1D: follow-the-leader

2D: cellular automata, social forces, granular, compartment model

COMMAND LINE/GUI?

Python code

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

Seems likely

OPEN TO COLLABORATION?

Not contacted but all available on GitHub so plausibly

SOURCE

GitHub

MENGE ^{*}[SL]

Menge looks like it would be ideal for building and adapting models with a very modular and accessible design. For example, it can be used to different compare models in the same situation. It allows the comparison of different models. It is only maintained by one person, Sean Curtis, <seanc@cs.unc.edu> in his spare time, and I couldn't get it installed as there appears to be some quite specific requirements in terms of versions for installation (e.g. Visual studio 2005, 2012). It also runs on C++ which could complicate things unless the user is C-savvy.

URL

<http://gamma.cs.unc.edu/Menge/>

Rating (/3^{*})

Documentation: *** <http://gamma.cs.unc.edu/Menge/files/MengeTechReport.pdf>

Complexity: ***

Accessibility: ***

Usefulness: *

Modularity: ***

Maintenance: **

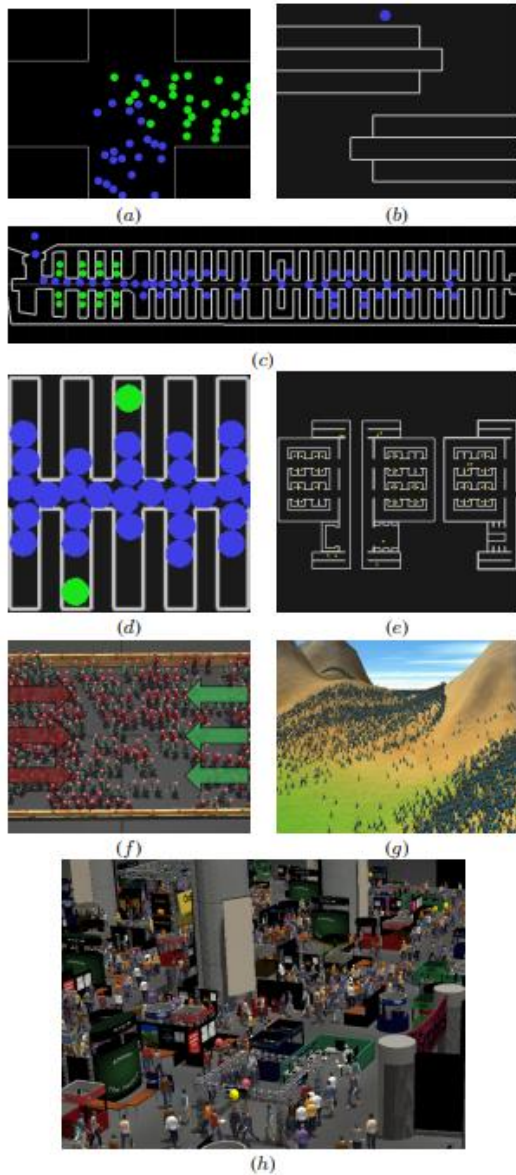


Figure 4: Images from the various prototype scenarios in Menge. (a) A cross flow highlighting pedestrian model comparisons. (b) A benchmark translated from SteerBench XML. (c) Airplane loading using random goal selection. (d) Agents (green) waiting for the aisle to clear using a custom transition in the airplane. (e) Agents work at desks and perform other activities in a three-story office building. (f) General Adaptation Syndrome algorithm simulation. (g) A battle scene showing 32,000 agents moving across complex terrain at interactive simulation rates. (h) The trade show scene

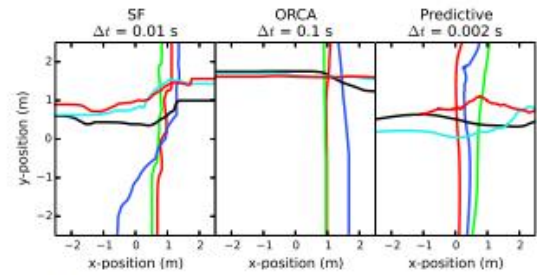


Figure 5: Trajectories plotted for three different pedestrian models in the Cross Flow scenario: a velocity-obstacle model (ORCA), a simple social force based model (SF), and a predictive forces model (Predictive). With all other simulation elements the same, these trajectories illustrate differences in the model behaviors.

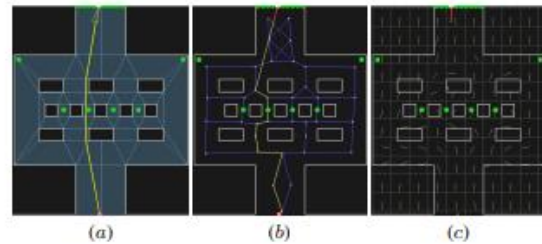


Figure 6: Visualization of three different global navigation methods applied to the Obstacle Course scenario. The green discs are agents; the yellow line represents the path computed for each algorithm. In the case of the guidance field, each cell's direction vector is shown in yellow. (a) the navigation mesh, (b) the roadmap, and (c) the guidance field. These navigation structures can be swapped by changing a single line of XML, the Velocity Component

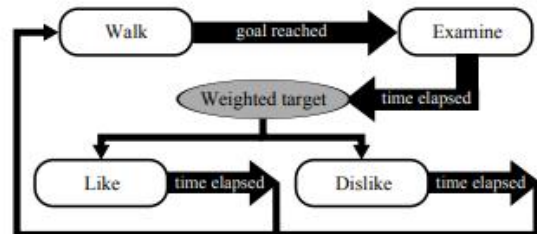


Figure 7: An illustration of the BFSM used in the trade show scenario. The white boxes represent FSM-states, the black arrows represent transition Conditions, the grey circle is a transition Target. Agents walk to an exhibit. When they reach the exhibit they enter the "Examine" state and stay there for a random amount of time after which they randomly enter the "Like" or "Dislike" state based on weighted probabilities. Finally, after a random amount of time in those states, they select and move to a new ex-

SUCCESSFUL INSTALLATION/TRIAL RUN?

No. Failed to build on Linux

COMPATIBLE OPERATING SYSTEMS

Windows with Visual Studio (2005 and 2012)

Linux (Ubuntu 14) using g++

LANGUAGE?

C++, Uses tinyxml to parse XML

2D/3D?

Both

TUTORIALS?

<http://gamma.cs.unc.edu/Menge/learn/gettingStarted.html>

INTENDED USE

Cross-platform, modular framework for crowd simulation

MODELS IMPLEMENTED

Multiple. Includes Behaviour Finite State Machine (BFSM) which may be native model, and particle model.

COMMAND LINE/GUI?

Not specified but states is "framework for simulating crowds and developing novel pedestrian models and techniques"

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

Should be possible writing plugins

FEATURES

Users can submit own add-ons

OPEN TO COLLABORATION?

Yes

SOURCE

GitHub

CROWD DYNAMICS *[SL]

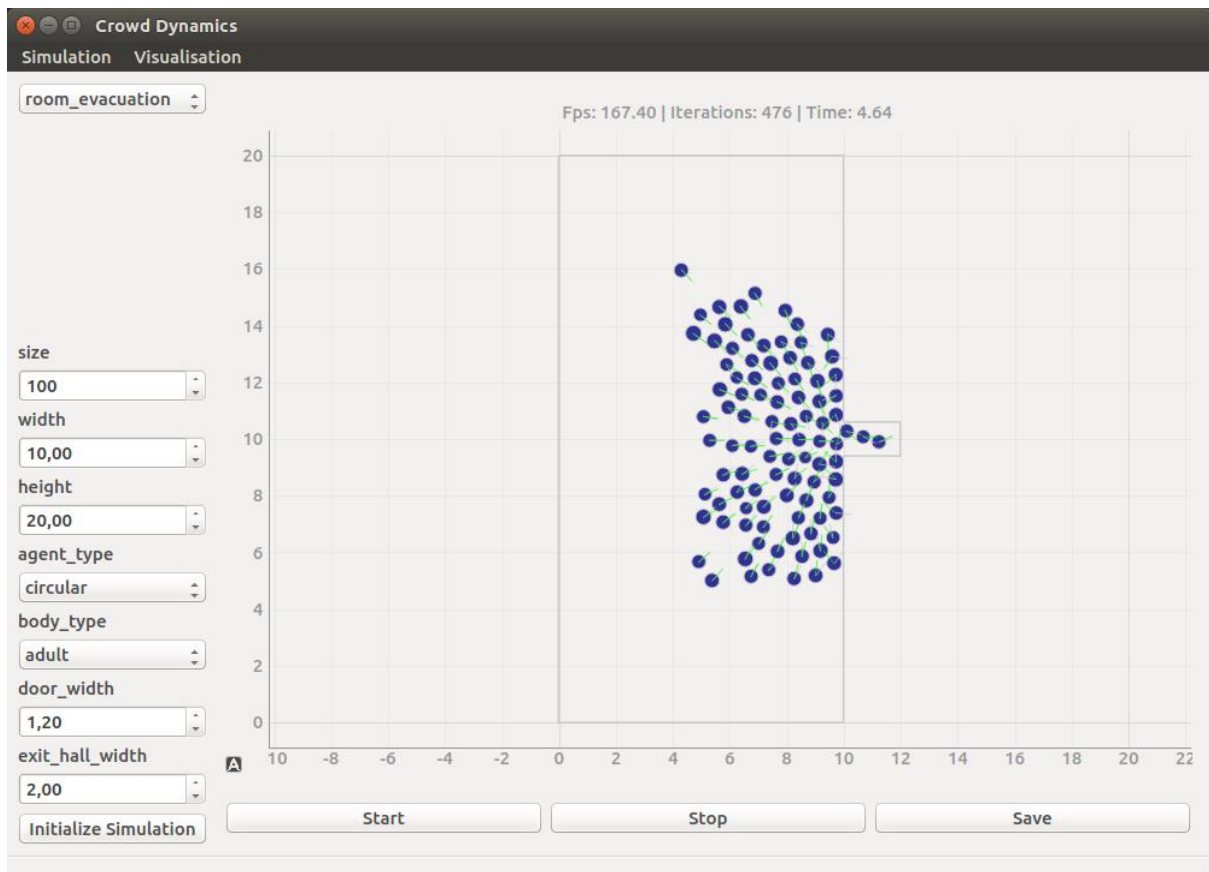
Initially appeared as though it may be useful, but is in early stages of development and failed on installation. The creator (Jaan Tollander, <de.tollander@aalto.fi>) has not updated it and it does not

appear to be a priority. Although the software could be installed successfully on the command line, the GUI did not install properly. It is a Python package created in summer 2016 for Systems Analysis Laboratory at Aalto University in Finland. There appears to be plans to integrate it into Anaconda, though this does not appear to have been implemented as yet.

URL

<https://github.com/jaantollander/crowddynamics>

	Rating (/3 [*])	
Documentation:	***	https://jaantollander.com/crowddynamics/multiagent.html
Complexity:	***	
Accessibility:	**	
Usefulness:	*	
Modularity:	**	
Maintenance:	*	



SUCCESSFUL INSTALLATION/TRIAL RUN?

No. Though the Crowd Dynamics software itself appeared to build successfully, the GUI did not and the creator could not get it to run due to apparent compatibility issues and needs to update the code from pyqt4 to pyqt5 but has not done this yet.

COMPATIBLE OPERATING SYSTEMS

Tested (by creator) on Ubuntu 16.04.

LANGUAGE?

Python

2D/3D?

2D

TUTORIALS?

Incomplete: <https://jaantollander.com/crowddynamics/usage.html>

INTENDED USE

A simulation environment written as a Python package for simulation movement of crowds

MODELS IMPLEMENTED

Agent-based models

COMMAND LINE/GUI?

GUI

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

?

OPEN TO COLLABORATION?

?

SOURCE

GitHub

PEDSIM ***[SL]**

Looks like nice software but could not install. Runs on C++ without additional packages. At current there is no PEDSIM application that allows you to design, run and analyse your own scenario. At this point, PEDSIM is a library, plus some helper applications. This means you basically have to write your

own application. You need a C++ compiler and some computer science knowledge. Updated very recently, but no response by email from the creator (Christian Gloor <chgloor@silmaril.org>).

URL

<http://pedsim.silmaril.org/>

Rating (/3^{*})

Documentation: *** <http://pedsim.silmaril.org/documentation/libpedsim/libpedsim.pdf>

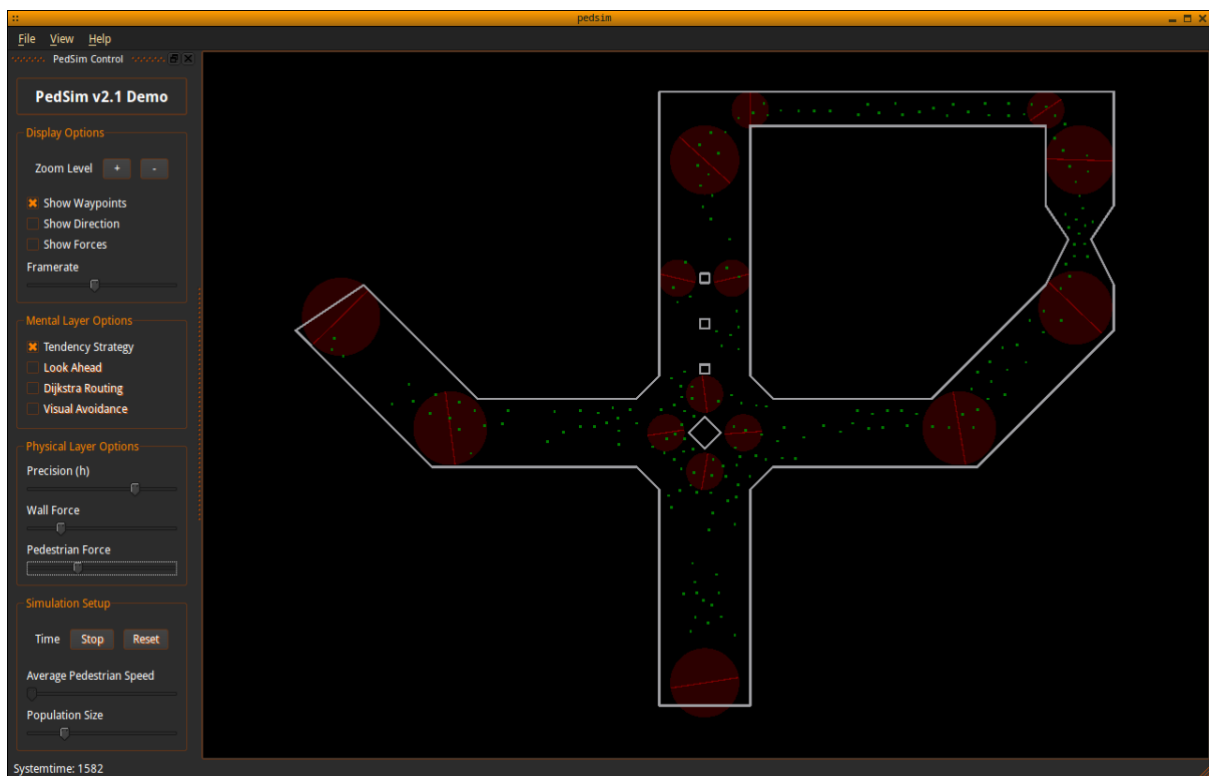
Complexity: ***

Accessibility: **

Usefulness: *

Modularity: ***

Maintenance: *



SUCCESSFUL INSTALLATION/TRIAL RUN?

No, not built on Linux

COMPATIBLE OPERATING SYSTEMS

Linux: tried and tested (by creator)

Windows: supported on visual studio

LANGUAGE?

C++

2D/3D?

2D with 3D capabilities

TUTORIALS?

Some examples with videos: <http://pedsim.silmaril.org/examples/>

INTENDED USE

PEDSIM is a microscopic pedestrian crowd simulation library.

MODELS IMPLEMENTED

Social force, cellular automata details available from documentation:

<http://pedsim.silmaril.org/documentation/libpedsim/libpedsim.pdf>

COMMAND LINE/GUI?

GUI, code accessible

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

Unknown

FEATURES

?

OPEN TO COLLABORATION?

No response

SOURCE

SourceFourge

JUPEDSIM [SL]

Looked promising, but failed on installation. Appears to be a work in progress or in the process of being heavily updated. Written in C++ and Python. From website:

“The primary goal of JuPedSim is to provide students and researchers with a framework to investigate pedestrian dynamics and focus on research, i.e. development and validation of new models or model features, analysis of experiments and proper visualization of results.”

Built from 4 core modules.

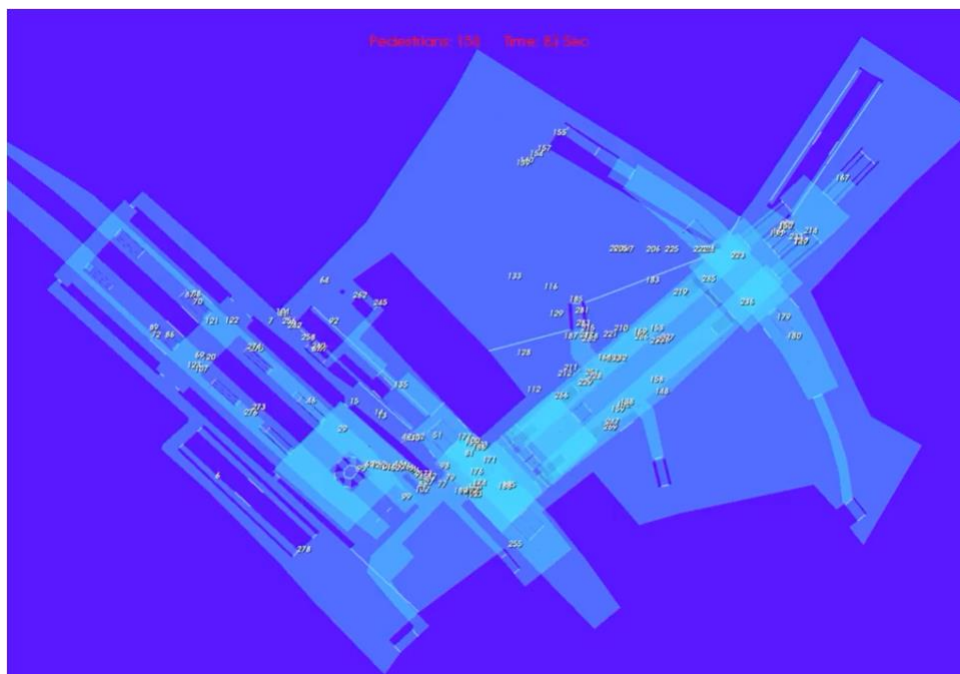
<https://www.youtube.com/channel/UCKS8w8CUCIHEeN4K1SUSMBA>

URL

<https://github.com/jupedsim/jpscore>

<http://www.jupedsim.org/>

	Rating (/3 [*])	
Documentation:	***	http://www.jupedsim.org/
Complexity:	***	
Accessibility:	***	
Usefulness:	**	
Modularity:	***	
Maintenance:	**	



SUCCESSFUL INSTALLATION/TRIAL RUN?

Failed to install

COMPATIBLE OPERATING SYSTEMS

Linux, Windows

LANGUAGE?

C++, Python

2D/3D?

2D, appears to have 3D implementation for building design etc.

TUTORIALS?

Installation: http://www.jupedsim.org/jupedsim_install_on_linux.html

Use: http://www.jupedsim.org/jpsreport_howto.html

INTENDED USE

Pedestrian Dynamics for researchers. Validation of new models or model features, analysis of experiments and proper visualization of results

MODELS IMPLEMENTED

Many http://www.jupedsim.org/jupedsim_publications.html

COMMAND LINE/GUI?

GUI

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

OPEN TO COLLABORATION?

Yes

SOURCE

Crowd Dynamics documentation website (resources section
<https://jaantollander.com/crowddynamics/resources.html>)

JCROWDSIMULATOR [NE]

Not explored as it was a known quantity; we have used this simulation with success in the past and have had good discussions with the developer about how it might be adapted to allow for dynamic

manipulation of the model at run time. Written in Java. Implements the Social Force Model of Helbing et al. (2005) / Helbing et al. 1995.

URL

<https://github.com/FraunhoferIV/jCrowdSimulator>

Rating (/3^{*})

Documentation: * [Click here](#) for German

Complexity: ?

Accessibility: ?

Usefulness: ?

Modularity: ?

Maintenance: ?

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

Java Virtual Machine

Windows

LANGUAGE?

Java

2D/3D?

2D

TUTORIALS?

?

INTENDED USE

Crowd simulation

MODELS IMPLEMENTED

Social Force Model of Helbing et al. (2005) / Helbing et al. 1995

COMMAND LINE/GUI?

?

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

Can pause

FEATURES

Multiple crowds, each having an individual route can be simulated simultaneously

Point/Line/Polygon data can be used to define obstacles

The outline of crowds can be computed

Resulting forces within the simulation can be visualised

The simulation library may be used independently from the user interface

OPEN TO COLLABORATION?

?

SOURCE

Suggestion

MESA [NE]

Not explored as it was a known quantity. Mesa is a python library for agent-based modelling rather than a pedestrian simulation in its own right.

URL

<https://mesa.readthedocs.io/en/master/#>

	Rating (/3 [*])	
Documentation:	***	https://mesa.readthedocs.io/en/master/#
Complexity:	?	
Accessibility:	?	
Usefulness:	?	
Modularity:	?	
Maintenance:	?	

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

Python

2D/3D?

2D

TUTORIALS?

<https://mesa.readthedocs.io/en/master/#>

INTENDED USE

Agent-based models

MODELS IMPLEMENTED

?

COMMAND LINE/GUI?

Python based, presumably command line

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

Built-in tools for analysis, browser-based visualization.

OPEN TO COLLABORATION?

yes

SOURCE

Suggestion

RVO2 LIBRARY [NE]

Reciprocal Collision Avoidance for Real-Time Multi-Agent Simulation. An algorithm for interactive navigation and planning of large numbers of agents in two-dimensional (crowded) environments. Does not appear to have been updated since 2016 - not explored in detail for this reason. Was applied in various video games (Warhammer 40K). Might be useful as a library that can be integrated into a pedestrian simulation that we build ourselves.

URL

<http://gamma.cs.unc.edu/RVO2/>

Rating (/3^{*})

Documentation: *** <http://gamma.cs.unc.edu/RVO2/documentation/2.0/>

Complexity: ?

Accessibility: **

Usefulness: *

Modularity: ***

Maintenance: *

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

C++

2D/3D?

2D, includes 3D implementation

TUTORIALS?

?

INTENDED USE

Crowd simulation, navigating a team of mobile robots, video games, studying natural flocking behavior, traffic engineering, architecture and design, emergency training simulations, etc.

MODELS IMPLEMENTED

Optimal Reciprocal Collision Avoidance (ORCA) formulation for multi-agent simulation:

<http://gamma.cs.unc.edu/ORCA/>

COMMAND LINE/GUI?

?

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

?

OPEN TO COLLABORATION?

?

SOURCE

Crowd Dynamics documentation website (resources section

<https://jaantollander.com/crowddynamics/resources.html>)

FIRE DYNAMICS SIMULATOR [NE]

Perhaps too specific and not broad enough applications so not explored in detail

URL

<https://github.com/firemodels/fds>

	Rating (/3 [*])	
Documentation:	***	https://github.com/firemodels/fds/tree/master/Manuals https://drive.google.com/drive/folders/0B_wB1pJL2bFQUiJwMmNfaHlqMEQ
Complexity:	*	
Accessibility:	*	
Usefulness:	*	
Modularity:	?	
Maintenance:	?	

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

Windows

Mac OS

Linux

LANGUAGE?

?

2D/3D?

2D

TUTORIALS?

?

INTENDED USE

Fire Dynamics Simulator (FDS) is a large-eddy simulation (LES) code for low-speed flows, with an emphasis on smoke and heat transport from fires.

MODELS IMPLEMENTED

?

COMMAND LINE/GUI?

?

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

?

OPEN TO COLLABORATION?

Yes

SOURCE

Crowd Dynamics documentation website (resources section
<https://jaantollander.com/crowddynamics/resources.html>)

AGENTS.JL: AGENT-BASED MODELING FRAMEWORK IN JULIA [NE]

Agent-based modelling framework. Presents itself as a competitor to Mesa. Not explored in detail.

URL

<https://github.com/JuliaDynamics/Agents.jl>

Rating (/3^{*})

Documentation: *** <https://juliadynamics.github.io/Agents.jl/dev/mesa/>

Complexity: ?

Accessibility: ?

Usefulness: ?

Modularity: ?

Maintenance: ?

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

Julia

2D/3D?

2D

TUTORIALS?

<https://juliadynamics.github.io/Agents.jl/dev/mesa/>

INTENDED USE

It provides a structure and components for quickly implementing agent-based models, run them in batch, collect data, and visualize them.

MODELS IMPLEMENTED

?

COMMAND LINE/GUI?

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

Simple, concise and extendable API for running ABMs.

Default grids to run the simulations, including simple or toroidal 1D grids, simple or toroidal regular rectangular and triangular 2D grids, and simple or toroidal regular cubic 3D grids with von Neumann or Moore neighbourhoods.

Allows arbitrary graphs as spaces for the ABMs.

Several different default schedulers.

Running the simulations in parallel on multiple cores.

Automatic data collection in a DataFrame at desired intervals.

Batch running and batch data collection.

Visualizing agent distributions on grids.

OPEN TO COLLABORATION?

?

SOURCE

Suggestion

LICENSED OR 3D

ANYLOGIC PEDESTRIAN SIMULATION ***

Promising, responsive to emails. Draw back may be cost for license. Contact Stuart Rossiter (Head of Technology & Training, DSE Consulting, <stuart@dseconsulting.co.uk>).

URL

<https://www.anylogic.com/airports-stations-shopping-malls/>

	Rating (/3 [*])
Documentation:	?
Complexity:	***
Accessibility:	**
Usefulness:	***
Modularity:	***
Maintenance:	***

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

Java

2D/3D?

2D and 3D

TUTORIALS?

Examples:

<https://www.anylogic.com/features/>

<https://www.anylogic.com/resources/libraries/pedestrian-library/>

<https://www.anylogic.com/resources/case-studies/>

INTENDED USE

Pedestrian simulations (primarily industrial applications: evacuation, shopping mall, airport, transport terminal etc. See: <https://cloud.anylogic.com/model/90677dc1-15c5-4bb9-917e-7f6547feceb7?mode=SETTINGS&tab=GENERAL>).

MODELS IMPLEMENTED

Helbing & Molnar (<https://arxiv.org/abs/cond-mat/9805244>)

COMMAND LINE/GUI?

GUI

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

Yes

FEATURES

Can 'enrich' your pedestrian agents with any extra explicitly-designed behaviour you want (and/or move pedestrians in and out of the pedestrian-dynamics physics model where you don't need the detail/accuracy of that movement).

> Is the source code accessible/is it possible to write modules?

AnyLogic is a completely extensible Java-based platform. Elements (agents, etc.) you design visually are used to generate and compile Java (which you can view the source code of, though this is not generally useful and this is not editable). Your visually-designed code is still the real 'source code' for the model, but you can add any Java libraries and use those, plus add 'raw' Java classes and interfaces to your models. Agents also have low-level Java constructs (fields, functions/methods, etc.) you can add visually (or as raw Java) for more 'algorithmic' parts of your model logic. (There are visual constructs such as statecharts which are commonly used for modelling many types of agent-based behaviour, including message-based inter-agent communication, plus built-in support for things like agent networks and communication across those networks.)

> Is it possible to intervene during the simulation and do things such as change parameters/add agents in arbitrary locations?

Yes you typically have full control to programmatically adjust things during run-time. Some things are harder than others (i.e., require more low-level Java), such as dynamically changing space markup (where you can't directly change it, but you can create new markup dynamically and switch agents to using that). And AnyLogic has some capabilities to more elegantly handle some areas here (e.g., you can dynamically change model parameters, with mechanisms to run 'on change' logic as needed to propagate necessary changes to other parts of the model).

Stuart Rossiter (Head of Technology & Training, DSE Consulting)

OPEN TO COLLABORATION?

SOURCE

Google

INCONTROL SIMULATION SOFTWARE **

Licensed pedestrian dynamics software. Free trial. Company proud of fact that Keith Still from MMU endorses it. Pedestrian Dynamics software built on top of the Enterprise Dynamics platform.

“With Enterprise Dynamics itself we have already established many projects/implementation in which communication is taking place by means of SAP connection, OPC, TCP/IP, Databases, XML, etc. The Pedestrian Dynamics customer starts to be interested in this as well. For example, they want to detect how many people are present in certain areas and change the model accordingly. An other example is that they want to see cameras in the 3D model and be able to double click on it to see the actual footage. We are currently looking at implementing a system to accommodate REST API’s.”

Contact: Fred Jansma <Fred.Jansma@incontrolsim.com>

URL

<https://www.incontrolsim.com/software/pedestrian-dynamics/>

	Rating (/3 [*])
Documentation:	?
Complexity:	***
Accessibility:	*
Usefulness:	***
Modularity:	?
Maintenance:	***

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

?

2D/3D?

2D and 3D

TUTORIALS?

?

INTENDED USE

Pedestrian simulation environment (Public Transport Hubs, Commercial Venues, Stadiums & Arenas, Public Events, Cities & Infrastructures)

MODELS IMPLEMENTED

“All kinds of academic research is incorporated in the product.”

To model the infrastructure you can model it manually, or be smart and use a CAD or BIM to generate the infrastructure of the model.

COMMAND LINE/GUI

GUI

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

Yes. That is one of the strengths of Pedestrian Dynamics. You also don't have to worry about OD-matrices. We don't use them. We use goal locations and the agent will figure out how to get there. Pedestrian Dynamics is capable of large amounts of pedestrians in one single model over various 3D levels. We have a proven track record of some of the largest stadiums in the world.

FEATURES

The simulation platform itself is propriety software. Everything you develop on top of it is readable source code for you and you have the ability to change or add code. You are even able to develop completely new simulation objects/modules. We also have the ability to link DLL's that uses your own code.

OPEN TO COLLABORATION?

Potentially – currently working on similar projects.

SOURCE

Google

OASYS MASSMOTION *

Seems to be commercial modelling software for industrial applications (modelling transport ports etc). Claims to be most advanced software available anywhere so should be able useful for any application. Emailed but no response. May be difficult to adapt as is proprietary software.

URL

<https://www.oasys-software.com/products/pedestrian-simulation/massmotion/>

Rating (/3 *)

Documentation: ***

Complexity: ***

Accessibility: *

Usefulness: ***

Modularity: ?

Maintenance: *

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

?

2D/3D?

Combine 3D and 2D assets – Import BIM files and use their geometry directly or import your 2D CAD files and then develop them into 3D models using MassMotion’s powerful polygon modelling tools.

TUTORIALS?

?

INTENDED USE

Simulations and analysis of large complex models,

MODELS IMPLEMENTED

?

COMMAND LINE/GUI?

GUI

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

Claims to be most advanced software available anywhere, should be capable.

FEATURES

Advanced pedestrian simulation and crowd analysis tools. Simulate hundreds of thousands of people within a matter of hours

Combine 3D and 2D assets

Enables intelligent agents (virtual pedestrians) to respond dynamically to evolving operational conditions and user-defined triggers

MassMotion Software Development Kit - Provides direct access to the MassMotion crowd engine for custom behaviours or connections to other software tools.

OPEN TO COLLABORATION?

?

SOURCE

SourceForge

QUADSTONE PARAMICS *

Licensed software, both pedestrian and traffic simulation. Not explored in detail. Emailed but no response.

“Quadstone Paramics is a leading microscopic traffic and pedestrian simulation software used by planning professionals to design efficient, economical, driver and pedestrian friendly transportation infrastructure allowing operational assessment for current and future year traffic conditions, detailed reporting of key MOE’s and high definition presentations to non-technical stake holders.”

URL

<http://www.paramics-online.com/>

	Rating (/3 [*])
Documentation:	?
Complexity:	?
Accessibility:	?
Usefulness:	?
Modularity:	?
Maintenance:	?

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

?

2D/3D?

3D

TUTORIALS?

?

INTENDED USE

Microscopic pedestrian/traffic simulation

MODELS IMPLEMENTED

?

COMMAND LINE/GUI?

GUI

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

?

OPEN TO COLLABORATION?

?

SOURCE

Wikipedia

UCROWDS

3D engine. Presumably licensed. Software library made in C++, with an API written in C. Has 3D plugin for Unity.

URL

<https://www.ucrowds.com/>

Rating
(/3^{*})

Documentation: ** [Click here](#)

Complexity: **

Accessibility: *

Usefulness: ?

Modularity: ?

Maintenance: ?

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

C++ and C

2D/3D?

2D, 3D plugin for Unity

TUTORIALS?

?

INTENDED USE

Software solution for simulating crowds in big infrastructures, events or virtual worlds.

MODELS IMPLEMENTED

All components described in published scientific papers:

https://www.staff.science.uu.nl/~gerae101/UU_crowd_simulation_publications.html

COMMAND LINE/GUI?

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

Features such as social groups, dynamic re-planning when obstacles appear or disappear, density-based crowds, and flow-based movements in crowded situations.

OPEN TO COLLABORATION?

?

SOURCE

Google

CROWDSIM3D [NE]

A crowd and traffic simulations add-on for Blender. Looks to be used primarily for visual effects. Not explored since 3D excessive (not particularly relevant for our project). Licensed, free demo complete but limited to 25 agents.

URL

<https://www.crowdsim3d.com/#>

Rating (/3^{*})

Documentation:	***	https://www.crowdsim3d.com/docs/
Complexity:	?	
Accessibility:	**	
Usefulness:	?	
Modularity:	?	
Maintenance:	***	

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

Blender addon

2D/3D?

3D

TUTORIALS?

<https://www.youtube.com/crowdsim3d>

INTENDED USE

Crowd and traffic simulations/special effects.

MODELS IMPLEMENTED

“Various algorithms.”

COMMAND LINE/GUI?

Blender addon (GUI)

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

Place characters in different formations in your scene. Configure the distance, randomness, pattern, area and more.

Let agents navigate the scene, avoid obstacles and other agents. CrowdSim3D supports various algorithms.

Automatic animation merging based on the movement of the agent

Visually program and debug your crowd

OPEN TO COLLABORATION?

?

SOURCE

Suggestion

CROWDMASTER [NE]

Crowd simulator for Blender 3D. Looks to be used primarily for visual effects. Not explored since 3D excessive.

URL

<http://crowdmaster.org/>

<https://github.com/johnroper100/CrowdMaster>

Rating (/3^{*})

Documentation: *** <http://crowdmaster.org/tutorials>

Complexity: ?

Accessibility: ?

Usefulness: ?

Modularity: ?

Maintenance: ?

SUCCESSFUL INSTALLATION/TRIAL RUN?

Not attempted

COMPATIBLE OPERATING SYSTEMS

?

LANGUAGE?

Blender addon

2D/3D?

3D

TUTORIALS?

<http://crowdmaster.org/tutorials>

INTENDED USE

3D crowd simulation addon for Blender 3D.

MODELS IMPLEMENTED

?

COMMAND LINE/GUI?

Blender addon (GUI)

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

?

FEATURES

Node-based interface

Customizable agent generation

Powerful animation tools

OPEN TO COLLABORATION?

?

SOURCE

GitHub

CROWD.LAB [NE]

Lightweight crowd simulator engine for Javascript. Not explored since 3D excessive. Example can be viewed in browser. Blender addon. Crowd.js is a lightweight crowd simulator engine for javascript. It is an direct port of a university algorithm in C++ to JavaScript made by Samuel Girardin (<http://www.visualiser.fr>) using Emscripten.

URL

<http://lo-th.github.io/Crowd.lab/#basic>

<https://github.com/lo-th/Crowd.lab>

SOURCE

GitHub

URHO3D [NE]

Urho3D is a free lightweight, cross-platform 2D and 3D game engine implemented in C++ and released under the MIT license. Greatly inspired by OGRE and Horde3D. Not explored since appears to be primarily for gaming.

URL

<https://urho3d.github.io/>

SOURCE

SourceForge

GOLAEM CROWD [NE]

Crowd simulation for special effects/visuals (e.g. used in Game of Thrones). Licensed software, free version for schools/universities. Not explored as it seemed excessive/not appropriate. Plug-in for Autodesk Maya (licensed, expensive).

URL

<http://golaem.com/>

SOURCE

Wikipedia

MASSIVE [NE]

Crowd simulation for special effects (e.g. used in Lord of the Rings). Not explored as it seemed excessive/not appropriate. Also used for architectural modelling

URL

<http://www.massivesoftware.com/>

SOURCE

Wikipedia

PTV VISSIM [NE]

For traffic simulation, not pedestrian so not explored in detail. Free demo available.

URL

<https://www.ptvgroup.com/en/solutions/products/ptv-vissim/>

SOURCE

Wikipedia

CROWDVISION [NE]

Automated passenger analytics and insights company. Not explored as arrangements to meet them had already been made.

URL

<https://www.crowdvision.com/>

SOURCE

Suggestion from Jon Ward

CROWD DYNAMICS [NE]

Not explored as arrangements to meet them had already been made.

URL

<https://www.crowddynamics.com/>

SOURCE

Suggestion from Jon Ward

MODELLING ENVIRONMENTS/ENGINES

BLENDER [NE]

Blender is the free and open source 3D creation suite. It supports the entirety of the 3D pipeline— modeling, rigging, animation, simulation, rendering, compositing and motion tracking, video editing and 2D animation pipeline.

URL

<https://www.blender.org/>

UNITY [NE]

Real-time 3D development platform.

URL

<https://unity.com/>

NETLOGO [NE]

Simulation software.

URL

<https://ccl.northwestern.edu/netlogo/>

SOURCE

Mesa documentation

REPAST [NE]

Simulation software.

URL

<https://repast.github.io/>

SOURCE

Mesa documentation

MASON [NE]

Simulation software.

URL

<https://cs.gmu.edu/~eclab/projects/mason/>

SOURCE

Mesa documentation

STEERSUITE [NE]

A suite of test cases, tools, and libraries for steering behaviors. Includes social forces model (<http://steersuite.eecs.yorku.ca/>)

URL

<https://github.com/SteerSuite/Release>

SOURCE

GitHub

CROWD NAV [NE]

Not explored as not updated for 3 years. Runs on SUMO (https://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/).

URL

<https://github.com/Starofall/CrowdNav>

SOURCE

GitHub

A CROWD SIMULATION USING OPENCL™ AND BASED ON THE CONTINUUM APPROACH [NE]

May not have been updated since 2011. OpenCL, runs on GPU, doesn't use ABMs. From Masters thesis.

URL

<https://github.com/hduregger/crowd>

SOURCE

GitHub

TEMPLATE ***[SL][NE]

Description

URL

Rating (/3^{*})

Documentation: *** www.docURL.com

Complexity: ***

Accessibility: ***

Usefulness: ***

Modularity: ***

Maintenance: ***

SUCCESSFUL INSTALLATION/TRIAL RUN?

COMPATIBLE OPERATING SYSTEMS

LANGUAGE?

2D/3D?

TUTORIALS?

INTENDED USE

MODELS IMPLEMENTED

COMMAND LINE/GUI?

INTERVENE/CHANGE PARAMETERS DURING SIMULATION?

FEATURES

OPEN TO COLLABORATION?

SOURCE