

My Project

Generated by Doxygen 1.8.9.1

Tue May 5 2015 15:51:30

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	Class Documentation	5
3.1	ChainBinomial_Sim Class Reference	5
3.2	compTime Class Reference	6
3.3	Deterministic_Network_SIR_Sim Class Reference	6
3.4	DiffEq_Sim Class Reference	7
3.5	Edge Class Reference	7
3.6	Event Class Reference	8
3.7	Gillespie_MassAction_Sim Class Reference	9
3.8	Gillespie_Network_SEIRS_Sim Class Reference	9
3.9	Intervention Class Reference	10
3.10	MTRand Class Reference	10
3.11	Network Class Reference	11
3.12	Node Class Reference	13
3.13	Opinion_formation Class Reference	14
3.14	Percolation_Sim Class Reference	14
3.15	RPlot Class Reference	15
3.16	Simulator Class Reference	16
3.17	SIR Class Reference	16
3.18	Trigger Class Reference	17
3.19	Yseries Class Reference	17
	Index	19

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

compTime	6
DiffEq_Sim	7
Deterministic_Network_SIR_Sim	6
SIR	16
Edge	7
Event	8
Gillespie_MassAction_Sim	9
Gillespie_Network_SEIRS_Sim	9
Intervention	10
MTRand	10
Network	11
Node	13
RPlot	15
Simulator	16
ChainBinomial_Sim	5
Opinion_formation	14
Percolation_Sim	14
Trigger	17
Yseries	17

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

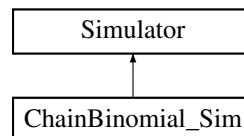
ChainBinomial_Sim	5
compTime	6
Deterministic_Network_SIR_Sim	6
DiffEq_Sim	7
Edge	7
Event	8
Gillespie_MassAction_Sim	9
Gillespie_Network_SEIRS_Sim	9
Intervention	10
MTRand	10
Network	11
Node	13
Opinion_formation	14
Percolation_Sim	14
RPlot	15
Simulator	16
SIR	16
Trigger	17
Yseries	17

Chapter 3

Class Documentation

3.1 ChainBinomial_Sim Class Reference

Inheritance diagram for ChainBinomial_Sim:



Public Member Functions

- **ChainBinomial_Sim** ([Network](#) *net, int infectious_period, double T)
- void **set_network** ([Network](#) *net)
- void **set_infectious_period** (int d)
- void **set_transmissibility** (double t)
- vector< double > **define_time_dist** ()
- int **get_infectious_period** ()
- double **get_transmissibility** ()
- vector< [Node](#) * > **rand_infect** (int n)
- void **infect_node** ([Node](#) *node)
- void **step_simulation** ()
- void **run_simulation** ()
- void **add_event** ([Node](#) *sink_node, int time, [Node](#) *source_node)
- int **count_infected** ()
- int **epidemic_size** ()
- vector< int > **get_epi_curve** ()
- vector< int > **get_prevalence_curve** ()
- vector< pair< int, [Node](#) * > > **get_detailed_epi_curve** ()
- void **reset** ()
- void **summary** ()

Public Attributes

- double **T**
- int **infectious_period**

Protected Attributes

- list< [Node](#) * > **infected**
- vector< [Node](#) * > **recovered**
- vector< double > **time_dist**
- bool **update_time_dist**
- priority_queue< [Event](#), vector< [Event](#) >, [compTime](#) > **transmissionQ**
- vector< int > **epi_curve**
- vector< pair< int, [Node](#) * > > **detailed_epi_curve**

The documentation for this class was generated from the following file:

- ChainBinomial_Sim.h

3.2 compTime Class Reference

Public Member Functions

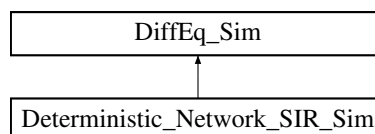
- bool **operator()** (const [Event](#) *lhs, const [Event](#) *rhs) const
- bool **operator()** (const [Event](#) &lhs, const [Event](#) &rhs) const
- bool **operator()** (const [Event](#) *lhs, const [Event](#) *rhs) const
- bool **operator()** (const [Event](#) &lhs, const [Event](#) &rhs) const
- bool **operator()** (const [Event](#) *lhs, const [Event](#) *rhs) const
- bool **operator()** (const [Event](#) &lhs, const [Event](#) &rhs) const

The documentation for this class was generated from the following files:

- ChainBinomial_Sim.h
- Gillespie_MassAction_Sim.h
- Gillespie_Network_SEIRS_Sim.h

3.3 Deterministic_Network_SIR_Sim Class Reference

Inheritance diagram for Deterministic_Network_SIR_Sim:



Public Member Functions

- **Deterministic_Network_SIR_Sim** (double r_param, double mu_param, vector< double > deg_dist_param)
- void **initialize** (double theta, double pS, double pI, double I)
- double **current_susceptible** ()
- double **current_infectious** ()
- double **current_recovered** ()
- double **g** (double theta)
- double **dg** (double theta)
- double **ddg** (double theta)
- void **derivative** (double const y[], double dydt[])

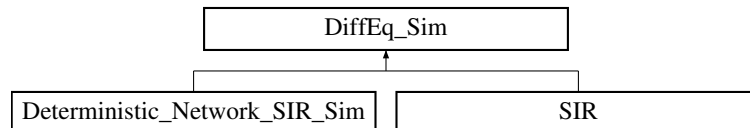
Additional Inherited Members

The documentation for this class was generated from the following file:

- Deterministic_Network_SIR_Sim.h

3.4 DiffEq_Sim Class Reference

Inheritance diagram for DiffEq_Sim:



Public Member Functions

- void **printY** ()
- vector< double > **get_state** ()
- double **get_time** ()
- virtual void **initialize** ()
- virtual void **derivative** (const double y[], double dydt[])
- int **run_simulation** ()
- int **step_simulation** (double stepsize)

Static Public Member Functions

- static int **function** (double t, double const y[], double dydt[], void *params)

Public Attributes

- int **nbins**
- double * **y**

The documentation for this class was generated from the following file:

- DiffEq_Sim.h

3.5 Edge Class Reference

Public Member Functions

- void **delete_edge** ()
- void **disconnect_nodes** ()
- int **get_id** ()
- double **get_cost** ()
- [Node](#) * **get_start** ()
- [Node](#) * **get_end** ()
- [Network](#) * **get_network** ()

- void **set_cost** (double c)
- [Edge](#) * **get_complement** ()
- void **swap_ends** ([Edge](#) *other_edge)
- void **break_end** ()
- void **define_end** ([Node](#) *end_node)
- bool **is_stub** ()
- bool **operator==** (const [Edge](#) &e2)
- void **dumper** ()

Friends

- class **Network**
- class **Node**
- ostream & **operator<<** (ostream &out, [Edge](#) *edge)

The documentation for this class was generated from the following files:

- Network.h
- Network.cpp

3.6 Event Class Reference

Public Member Functions

- **Event** (const [Event](#) &o)
- **Event** ([Node](#) *sink, int t, [Node](#) *source)
- [Event](#) & **operator=** (const [Event](#) &o)
- **Event** (const [Event](#) &o)
- **Event** (double t, char e)
- [Event](#) & **operator=** (const [Event](#) &o)
- **Event** (const [Event](#) &o)
- **Event** (double t, char e, [Node](#) *n)
- [Event](#) & **operator=** (const [Event](#) &o)

Public Attributes

- [Node](#) * **sink_node**
- int **time**
- [Node](#) * **source_node**
- double **time**
- char **type**
- [Node](#) * **node**

The documentation for this class was generated from the following files:

- ChainBinomial_Sim.h
- Gillespie_MassAction_Sim.h
- Gillespie_Network_SEIRS_Sim.h

3.7 Gillespie_MassAction_Sim Class Reference

Public Member Functions

- **Gillespie_MassAction_Sim** (int n, double gamma, double beta)
- void **run_simulation** ()
- int **epidemic_size** ()
- int **reset** ()
- void **rand_infect** (int k)
- void **infect** ()
- bool **is_susceptible** (int x)
- int **next_event** ()
- void **add_event** (double time, char type)

Public Attributes

- int **N**
- double **GAMMA**
- double **BETA**
- priority_queue< [Event](#), vector< [Event](#) >, [compTime](#) > **EventQ**
- vector< int > **Compartments**
- double **Now**
- [MTRand](#) **mtrand**

The documentation for this class was generated from the following file:

- Gillespie_MassAction_Sim.h

3.8 Gillespie_Network_SEIRS_Sim Class Reference

Public Types

- enum **stateType** {
SUSCEPTIBLE, **EXPOSED**, **INFECTIOUS**, **RESISTANT**,
STATE_SIZE }

Public Member Functions

- **Gillespie_Network_SEIRS_Sim** ([Network](#) *net, double m, double b, double g, double im_dur)
- void **run_simulation** (double duration)
- int **current_epidemic_size** ()
- int **reset** ()
- vector< [Node](#) * > **rand_choose_nodes** (int n)
- void **rand_infect** (int n)
- void **infect** ([Node](#) *node)
- int **next_event** ()
- void **add_event** (double time, char type, [Node](#) *node)

Public Attributes

- [Network](#) * **network**
- double **mu**
- double **beta**
- double **gamma**
- double **immunity_duration**
- priority_queue< [Event](#), vector< [Event](#) >, [compTime](#) > **EventQ**
- vector< int > **state_counts**
- double **Now**
- [MTRand](#) **mtrand**

The documentation for this class was generated from the following file:

- Gillespie_Network_SEIRS_Sim.h

3.9 Intervention Class Reference

The documentation for this class was generated from the following file:

- Intervention.h

3.10 MTRand Class Reference

Public Types

- enum { **N** = 624 }
- enum { **SAVE** = N + 1 }
- typedef unsigned long **uint32**

Public Member Functions

- **MTRand** (const uint32 &oneSeed)
- **MTRand** (uint32 *const bigSeed, uint32 const seedLength=N)
- double **rand** ()
- double **rand** (const double &n)
- double **randExc** ()
- double **randExc** (const double &n)
- double **randDbExc** ()
- double **randDbExc** (const double &n)
- uint32 **randInt** ()
- uint32 **randInt** (const uint32 &n)
- double **operator()** ()
- double **rand53** ()
- double **randNorm** (const double &mean=0.0, const double &std_dev=1.0)
- void **seed** (const uint32 oneSeed)
- void **seed** (uint32 *const bigSeed, const uint32 seedLength=N)
- void **seed** ()
- void **save** (uint32 *saveArray) const
- void **load** (uint32 *const loadArray)

Protected Types

- enum { **M** = 397 }

Protected Member Functions

- void **initialize** (const uint32 oneSeed)
- void **reload** ()
- uint32 **hiBit** (const uint32 &u) const
- uint32 **loBit** (const uint32 &u) const
- uint32 **loBits** (const uint32 &u) const
- uint32 **mixBits** (const uint32 &u, const uint32 &v) const
- uint32 **twist** (const uint32 &m, const uint32 &s0, const uint32 &s1) const

Static Protected Member Functions

- static uint32 **hash** (time_t t, clock_t c)

Protected Attributes

- uint32 **state** [N]
- uint32 * **pNext**
- int **left**

Friends

- std::ostream & **operator**<< (std::ostream &os, const [MTRand](#) &mtrand)
- std::istream & **operator**>> (std::istream &is, [MTRand](#) &mtrand)

The documentation for this class was generated from the following file:

- MersenneTwister.h

3.11 Network Class Reference

Public Types

- enum **netType** { **Undirected** =0, **Directed** =1 }
- enum **outputType** { **NodeNames** =0, **NodeIDs** =1 }

Public Member Functions

- **Network** (string name, netType directed)
- **Network** (const [Network](#) &net)
- [Network](#) * **duplicate** ()
- bool **operator==** (const [Network](#) &n2)
- int **get_id** ()
- string **get_name** ()
- int **size** ()
- bool **has_unit_edges** ()
- bool **is_directed** ()

- **MTRand** * **get_rng** ()
- vector< **Node** * > **get_nodes** ()
- **Node** * **get_node** (int node_id)
- **Node** * **get_node_by_name** (string node_name)
- **Node** * **get_rand_node** ()
- vector< **Edge** * > **get_edges** ()
- **Edge** * **get_edge** (int id)
- vector< stateType > **get_node_states** ()
- void **get_bad_edges** (vector< **Edge** * > &self_loops, vector< **Edge** * > &multiedges)
- vector< **Node** * > **get_component** (**Node** *node)
- vector< vector< **Node** * > > **get_components** ()
- vector< **Node** * > **get_biggest_component** ()
- bool **topology_altered** ()
- **Node** * **add_new_node** ()
- void **populate** (int n)
- void **add_node** (**Node** *node)
- void **delete_node** (**Node** *node)
- bool **all_to_all_coupling** ()
- bool **erdos_renyi** (double lambda)
- bool **sparse_random_graph** (double lambda)
- bool **fast_random_graph** (double lambda)
- bool **ring_lattice** (int N, int K)
- bool **square_lattice** (int R, int C, bool diag)
- bool **small_world** (int N, int K, double beta)
- bool **rand_connect_poisson** (double lambda)
- bool **rand_connect_powerlaw** (double alpha, double kappa)
- bool **rand_connect_exponential** (double lambda)
- bool **rand_connect_user** (vector< double > dist)
- bool **rand_connect_explicit** (vector< int > deg_series)
- bool **rand_connect_stubs** (vector< **Edge** * > stubs)
- bool **lose_loops** ()
- void **clear_nodes** ()
- void **clear_edges** ()
- void **disconnect_edges** ()
- bool **shuffle_edges** (double frac)
- void **set_node_states** (vector< stateType > &states)
- void **initialize** (double mean_coupling, double var_coupling, double mean_preference, double var_↵ preference, double mean_initial_state)
- void **set_topology_altered** (bool flag)
- void **read_edgelist** (string filename, char sep=',')
- void **write_edgelist** (string filename, outputType names_or_ids, char sep=',')
- void **graphviz** (string filename)
- void **dumper** ()
- bool **gen_deg_series** (vector< int > °_series)
- vector< stateType > **get_states** ()
- vector< vector< stateType > > **get_states_by_degree** ()
- double **get_coarse_state** ()
- bool **validate** ()
- vector< int > **get_deg_series** ()
- vector< int > **get_deg_dist** ()
- vector< double > **get_gen_deg_dist** ()
- double **mean_deg** ()
- map< **Node** *, int > **k_shell_decomposition** ()
- double **transitivity** ()
- double **transitivity** (vector< **Node** * > node_set)

- bool **is_weighted** ()
- double **mean_dist** (vector< [Node](#) * > node_set)
- void **calculate_distances** (vector< [Node](#) * > &destinations, vector< vector< double > > &distances)
- void **print_distances** (vector< [Node](#) * > &full_node_set)
- void **stop_processing** ()
- void **reset_processing_flag** ()

Friends

- class **Node**
- class **Edge**

The documentation for this class was generated from the following files:

- Network.h
- Network.cpp

3.12 Node Class Reference

Public Member Functions

- bool **is_stopped** ()
- void **delete_node** ()
- void **set_network** ([Network](#) *network)
- int **get_id** ()
- string **get_name** ()
- [Network](#) * **get_network** ()
- vector< [Edge](#) * > **get_edges_in** ()
- vector< [Edge](#) * > **get_edges_out** ()
- vector< double > **get_loc** ()
- stateType **get_state** ()
- double **get_coupling** ()
- double **get_preference** ()
- double **get_utility** ()
- void **set_loc** (const vector< double > &newloc)
- void **set_state** (stateType s)
- void **set_coupling** (double c)
- void **set_preference** (double p)
- double **mean_min_path** ()
- vector< double > **min_paths** (vector< [Node](#) * > &node_set)
- void **add_stubs** (int deg)
- [Edge](#) * **get_rand_edge** ()
- vector< [Node](#) * > **get_neighbors** ()
- double **get_neighbors_state** ()
- void **update_utility_function** ()
- bool **is_neighbor** ([Node](#) *node2)
- void **connect_to** ([Node](#) *end)
- bool **change_neighbors** ([Node](#) *old_neighbor, [Node](#) *new_neighbor)
- bool **operator==** (const [Node](#) &n2)
- void **dumper** ()
- double **min_path** ([Node](#) *dest)
- [Edge](#) * **add_stub_out** ()
- string **get_name_or_id** ()
- int **deg** ()

Friends

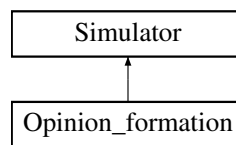
- class **Network**
- class **Edge**
- ostream & **operator**<< (ostream &out, [Node](#) *node)

The documentation for this class was generated from the following files:

- Network.h
- Network.cpp

3.13 Opinion_formation Class Reference

Inheritance diagram for Opinion_formation:



Public Member Functions

- **Opinion_formation** ([Network](#) *net, string &filename)
- **Opinion_formation** (vector< [Network](#) * > net_list, string &filename)
- void **time_step** ()
- void **run_simulation** (int max_time)
- void **reset** ()
- void **summary** ()

Public Attributes

- ofstream **outfile**

Protected Attributes

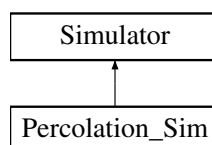
- double **beta** =10

The documentation for this class was generated from the following file:

- Opinion_formation.h

3.14 Percolation_Sim Class Reference

Inheritance diagram for Percolation_Sim:



Public Types

- enum **stateType** { **S** =0, **I** =1, **R** =-1 }

Public Member Functions

- **Percolation_Sim** ([Network](#) *net)
- void **set_transmissibility** (double t)
- double **expected_R0** ()
- vector< [Node](#) * > **rand_infect** (int n)
- void **step_simulation** ()
- void **run_simulation** ()
- int **count_infected** ()
- int **epidemic_size** ()
- void **reset** ()
- void **summary** ()

Public Attributes

- float **T**

Protected Attributes

- vector< [Node](#) * > **infected**
- vector< [Node](#) * > **recovered**

The documentation for this class was generated from the following file:

- Percolation_Sim.h

3.15 RPlot Class Reference

Public Member Functions

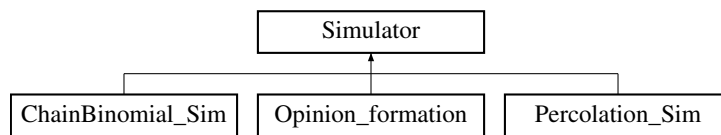
- void **define_header** (string h)
- void **define_footer** (string f)
- void **define** (string par, double val)
- void **define** (string par, string val)
- void **set_x** (vector< double > x)
- void **set_y** (vector< double > y, string color="1", string pch="1", string type="p")
- void **add_y** (vector< double > y, string color="1", string pch="1", string type="p")
- int **pdf** (string filename, double width=10, double height=7.5)
- int **png** (string filename, double width=1000, double height=750)
- int **_plotter** (string plot_type, string filename, double width, double height)
- void **write_datafile** (string filename)
- string **xlim** (vector< double > X, vector< [Yseries](#) * > Y)
- string **ylim** (vector< double > X, vector< [Yseries](#) * > Y)
- vector< double > **determine_limits** (vector< [Yseries](#) * > Y)

The documentation for this class was generated from the following file:

- RPlot.h

3.16 Simulator Class Reference

Inheritance diagram for Simulator:



Public Member Functions

- **Simulator** ([Network](#) *net, string fname)
- **Simulator** (vector< [Network](#) * > net_list, string fname)
- void **set_network** ([Network](#) *net)
- [Network](#) * **network** ()
- int **get_time** ()
- void **reset_time** ()
- void **set_all_nodes_to_state** (stateType s)
- void **set_these_nodes_to_state** (vector< [Node](#) * > nodes, stateType s)
- vector< [Node](#) * > **rand_choose_nodes** (int n)
- vector< [Node](#) * > **rand_set_nodes_to_state** (int n, stateType state)
- virtual void **time_step** ()
- virtual void **run_simulation** ()
- virtual int **count_infected** ()
- virtual void **reset** ()

Public Attributes

- int **time**
- [Network](#) * **net**
- [MTRand](#) * **mtrand**
- vector< [Network](#) * > **net_list**
- string **fname**

Protected Attributes

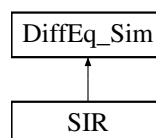
- time_t **starttime**

The documentation for this class was generated from the following file:

- Simulator.h

3.17 SIR Class Reference

Inheritance diagram for SIR:



Public Member Functions

- **SIR** (double b, double g)
- void **initialize** (double S, double I, double R)
- void **derivative** (double const y[], double dydt[])

Additional Inherited Members

The documentation for this class was generated from the following file:

- SIR_Sim.h

3.18 Trigger Class Reference

The documentation for this class was generated from the following file:

- Intervention.h

3.19 Yseries Class Reference

Public Member Functions

- **Yseries** (vector< double > d, string c, string p, string t)
- int **size** ()
- vector< double > **data** ()
- double **data** (int i)
- string **col** ()
- string **pch** ()
- string **type** ()

Public Attributes

- vector< double > **D**
- string **C**
- string **P**
- string **T**

The documentation for this class was generated from the following file:

- RPlot.h

Index

ChainBinomial_Sim, [5](#)
compTime, [6](#)

Deterministic_Network_SIR_Sim, [6](#)
DiffEq_Sim, [7](#)

Edge, [7](#)
Event, [8](#)

Gillespie_MassAction_Sim, [9](#)
Gillespie_Network_SEIRS_Sim, [9](#)

Intervention, [10](#)

MTRand, [10](#)

Network, [11](#)
Node, [13](#)

Opinion_formation, [14](#)

Percolation_Sim, [14](#)

RPlot, [15](#)

SIR, [16](#)
Simulator, [16](#)

Trigger, [17](#)

Yseries, [17](#)