
Contents

Introduction	ix
Arnaud BANOS, Christophe LANG and Nicolas MARILLEAU	
Chapter 1. NetLogo, an Open Simulation Environment	1
Benoît GAUDOU, Christophe LANG, Nicolas MARILLEAU, Guilhelm SAVIN, Sébastien REY COYREHOURCQ and Jean-Marc NICOD	
1.1. Introduction to extensions in NetLogo	1
1.1.1. Examples of typical NetLogo extensions	2
1.1.2. Installing and using extensions in models	3
1.2. Designing and developing extensions	5
1.2.1. Environment for compiling extensions	5
1.2.2. Notes on type conversion between NetLogo and Java/Scala	7
1.2.3. Commentary of an example extension	9
1.2.4. Minimum content of an extension	9
1.2.5. Snapshot of a primitive	11
1.2.6. Future versions of the NetLogo API	15
1.2.7. Extending the graphical interface	15
1.2.8. Example: the RungeKutta extension	20
1.3. Using NetLogo from other platforms	22
1.3.1. Using NetLogo from Java	23
1.3.2. Using NetLogo from Python	25
1.3.3. Exploring and analyzing models with R	28
1.3.4. Discussion	30
1.4. Deploying NetLogo models online	30

1.4.1. Netlogo Web	31
1.4.2. Modeling Commons	32
1.4.3. OpenABM	34
1.5. Conclusion	35

Chapter 2. Multiscale Modeling: Application to Traffic Flow

Arnaud BANOS, Nathalie CORSON, Christophe LANG,
Nicolas MARILLEAU and Patrick TAILLANDIER

2.1. Introduction	37
2.1.1. Coupling models together	38
2.2. Two agent-based models: NaSch and Underwood	45
2.2.1. The NaSch model	45
2.2.2. The Underwood model	48
2.3. An equation-based LWR model	51
2.4. Hybrid traffic model	57
2.5. Conclusion and outlook	61

Chapter 3. Macro Models, Micro Models and Network-based Coupling

Arnaud BANOS, Nathalie CORSON, Éric DAUDÉ, Benoit GAUDOU and
Sébastien REY COYREHOURCQ

3.1. Introduction	63
3.2. Description of the equation-based SIR model	64
3.3. Equation-based and agent-based propagation model: EpiSim	65
3.3.1. Distributed and non-spatial SIR model	65
3.3.2. Spatially distributed SIR model with local interactions	67
3.3.3. Spatially distributed model with local neighborhoods and explicit contact between individuals	68
3.3.4. Spatially distributed SIR model with a network of interactions	69
3.4. Coupling SIR models based on networks	72
3.5. SIR coupling without scaling: Metapop model	72
3.5.1. Presentation of the Metapop model	72
3.5.2. Numerical integration of a network of coupled SIR using Runge-Kutta implemented in NetLogo	74
3.5.3. Examples of results	76

3.6. SIR coupling with scaling: MicMac model	77
3.6.1. Model presentation	77
3.6.2. General description of the working principle of the coupling	78
3.6.3. Initialization: calibrating the model	79
3.6.4. Using the RK4 extension to perform numerical integration	80
3.6.5. Switching between the continuous and discrete parts of the model	80
3.6.6. Example results	83
3.7. Conclusion and outlook	84
Chapter 4. Networking, Networks and Dynamic Graphs	85
Stefan BALEV, Antoine DUTOT and Damien OLIVIER	
4.1. Networking	85
4.1.1. Networks: a vague, equivocal and often misused concept	85
4.1.2. From reality to theory	87
4.2. Networks and graphs in NetLogo	89
4.2.1. Generating a random graph	90
4.2.2. Search for the largest connected component	92
4.2.3. Searching for the shortest path	95
4.2.4. Modularity metric	103
4.2.5. Communities	111
Chapter 5. Swarm Problem-Solving	117
Antoine DUTOT and Damien OLIVIER	
5.1. Introduction	117
5.2. Collective approaches	118
5.2.1. Swarm intelligence and collective intelligence	118
5.2.2. Interactions, self-organization and stigmergy	120
5.3. Collective sorting	122
5.3.1. Ant behavior	122
5.3.2. Model analysis	125
5.4. From food sourcing to finding the shortest path	127
5.4.1. From the natural model to the virtual model	129
5.4.2. The traveling salesman and Ant System	138
5.5. The intentions of a swarm	147

5.5.1. Basic model	151
5.5.2. Model with local interactions	161
5.5.3. Applications and discussion	166
5.6. Conclusion	171
Chapter 6. Exploring Complex Models in NetLogo	173
Philippe CAILLOU, Sébastien REY COYREHOURQ, Nicolas MARILLEAU and Arnaud BANOS	
6.1. Introduction	173
6.2. Complex models and simulators	174
6.2.1. Characteristics of complex simulators	174
6.2.2. Inverse problem-solving by exploration: a necessity	175
6.2.3. HPC in simulations	178
6.2.4. The underlying model: NetLogo Ants	179
6.2.5. Calibrating a model by optimization	180
6.2.6. Formulating objectives for the Ants model	182
6.2.7. Adapting the NetLogo Ants model to be used by an optimizer	183
6.2.8. Choosing an optimizer, evolutionary algorithms	186
6.3. Using NetLogo with OpenMOLE	190
6.3.1. Presentation of OpenMOLE	190
6.3.2. Workflow for testing randomness	191
6.3.3. Workflow for calibrating with EA	197
6.4. Analysis and interpretation of results	200
6.4.1. Analysis tools	200
6.4.2. Choosing the graph	201
6.4.3. Interpreting the results	202
6.5. Conclusion	207
Conclusion	209
Bibliography	211
List of Authors	219
Index	221