### Title
Using Nova to Construct Agent-Based Models for Epidemiological Teaching and Research




### ISBN
9781467397438



### Authors
Getz, Wayne M
Salter, Richard M
Sippl-Swezey, Nicolas



### Publication Date
2015-12-01

### DOI
10.1109/wsc.2015.7408509

Peer reviewed

*Proceedings of the 2015 Winter Simulation Conference*
*L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.*

# USING NOVA TO CONSTRUCT AGENT-BASED MODELS FOR EPIDEMIOLOGICAL TEACHING AND RESEARCH

Wayne M. Getz

Department of Environmental Science
Policy and Management
University of California at Berkeley
Berkeley, CA 94720-3114, US

School of Mathematical Sciences
University of KwaZulu-Natal
PB X54001, Durban 4000, South Africa

Richard M. Salter

Computer Science Department
Oberlin College
Oberlin, Ohio, OH 44074, USA

Nicolas Sippl-Swezey

Francis I. Proctor Foundation
University of California at San Francisco
San Francisco, CA 94143-0412, USA

## ABSTRACT

Epidemic modeling is dominated by systems models—so-called SIR models—that describe the spatio-temporal and network dynamics of disease outbreaks. Reed-Frost, discrete-time, stochastic transmission-chain models have also been important; but, increasingly, epidemiological modelers are turning to agent-based (ABM) approaches that permit the inclusion of individual-specific characters, which may relate to the genetics of hosts or pathogens, host exposure histories, co-infections or other general health correlates. Here we introduce Nova, a graphically driven computational modeling platform for creating and running both dynamical systems and ABM models that have application both in teaching and research. Because Nova is based on the JavaScript language, all Nova models are easily transformed into Nova Online web apps. In the teaching arena, our presentation features our "SIR circle games"; in the research arena we discuss the application of Nova to modeling outbreaks of Ebola and measles.

## 1 INTRODUCTION

Graphically oriented, modeling building and simulation software, such as the Stella (Smith et al. 2005), Berkeley Madonna (Krause and Lowe 2014), Insightmaker (Fortman-Roe 2014), Simile (Morales et al. 2003, Smith et al. 2005) and AnyLogic (Borshchev et al. 2002) platforms have been a boon to applied scientists who require simulation modeling tools in the practice of their professions, but do not have well-developed coding skills. This is particularly true of the field of applied computational population biology, were computationally complex dynamical systems and agent-based models are used to advance research and implement policy in number of areas, including sustainable resource exploitation, and ecosystems and disease systems management. A growing research endeavor is the construction of computational population models that include behavioral, physiological, genetic, ecological and evolutionary processes, and take place over spatially-detailed landscape. It is, of course, facilitated by the power of cloud computing, with access requiring either considerable coding fluency and skills or software that minimizes the need for such fluency and skills. Eliminating the need for former and enabling is a challenge (Getz 2013) that we focus on here in the context of epidemiological thinking and modeling. We also feature the Nova software platform (Salter 2013) in this context as well; although, in the next section, Nova is introduced as a general purpose platform for both dynamical systems and agent-based approaches to modeling.

The prevailing paradigm for modeling epidemics is to use systems of deterministic or stochastic difference or differential equations that divide the population into disease classes (e.g. susceptible, infected,

recovered, immune) (Hethcote 2000): so-called SIR models. This paradigm, for example, has been applied to modeling both measles (Duncan et al. 1997, Bjørnstad et al. 2002, Keeling and Grenfell 1997) and Ebola viral disease (EVD) (Legrand et al. 2007, Chowell et al. 2004, Lekone and Finkenstädt 2006). It has also been applied to modeling systems where demographic (e.g. age and sex (Cross and Getz 2006)) and various types of behavioral classes (e.g. sexually active individuals (Bellan et al. 2013)) have been included. An alternative method for modeling epidemics, originally proposed by Reed and Frost, is to follow transmission chains (incidence and offspring distributions, transmission trees and branches), as semi-Markov branching process (Barbour and Utev 2004). The first paradigm is most useful for large-scale epidemics involving infection of a significant fraction of the susceptible population ($> 1\%$)—which, for example, is the case for influenza, HIV, tuberculosis, and measles prior to widespread vaccination. The Reed-Frost approach is more appropriate for emerging diseases (Jones et al. 2008) when the proportion of infected individuals is often very small ($< 0.1\%$)—which includes inter alia, SARS (Lloyd-Smith et al. 2003), EVD (Getz et al. 2015), and the recent measles outbreak in the US (McCarthy 2015).

For this reason, the analysis we report here is based on a Markov chain—more specifically, a Galton-Watson branching process (Murai et al. 2013)—approach, as discussed in Chowell et al. (2004); but modified to allow for evaluation of the implications and efficacy of key interventions, such as explicitly including isolation, treatment, and vaccination strategies, as well as incorporating healthcare workers (Lloyd-Smith et al. 2003). We can also include a reduction in transmission rates over the course of the epidemics as healthcare workers mobilize or as individuals change their behavior in response to increased awareness of risky behavior as the epidemic unfolds. Metapopulation structure, as discussed below, can also be included by treating each subpopulation as a node in a network of subpopulations.

## 2 NOVA

Nova is a new Java-based modeling platform that naturally supports the creation of system dynamics, spatial, and agent-based models in a single desktop application. Nova uses a visual language to express model design, and provides automatic conversion for such models to script form for execution. Nova's architecture promotes hierarchical design, code reuse, and extensibility through the use of plug-ins. Nova's expressive power derives from strong design principles: modularity, abstraction and extensibility.

### 2.1 Platform

Nova is fundamentally a dynamic modeling system that is extended through hierarchical design to express spatial and agent-based architectures. A Nova model can be built using the visual language and then, using its capture function, is automatically converted into a runnable script for immediate execution, or possible future network or a supercomputer deployment. Nova focuses on the creation of a modular unit called a capsule. Each capsule is a complete model that interacts with its environment through an interface consisting of input and output channels. The simplest capsule might contain a stock-and-flow model similar to one built in Stella. However, capsule instances (called chips) may appear in other capsules (as long as there is no circularity), communicating with their hosts through their I/O channels. Each chip introduces into its host the functionality of that chip's encapsulated model. Capsules may be exported and reused.

The chip is one type of container. Spatial and agent-based models are constructed using array-like containers called aggregators. The current implementation provides five aggregator types:

- **AgentVectors** are one dimensional arrays of agents; an agent is a capsule extension that includes a representation for location and movement within a two dimensional space. AgentVectors also manage dynamic creation and destruction of agents.
- **CellMatrices** are two dimensional arrays of capsules. They provide a means for representing cellular automata. Cellmatrices may implement either a cartesian or hexagonal topology.
- **NodeNetworks** are an array of graph nodes with weighted directed connections.

- **SimWorlds** combine AgentVectors with CellMatrices, so that agent locations correspond to matrix coordinates. The result is a virtual space of interacting agents and cells.
- **NetWorlds** analogous to SimWorlds substituting a NodeNetwork for a CellMatrix as the space in which the agents operate.

Nova's computational architecture comprise the semantics of NovaScript, a scripting language embedded in Javascript. The NovaScript runtime environment is an extension of the ECMA 1.7 Javascript standard. All Nova simulations are actually NovaScript programs executing on the NovaScript runtime interpreter.

NovaScript takes the form of a set of JSON objects (JavaScript Object Notation; see Crockford, D. (2013)) called *schema*. Each schema acts as a class definition for some component or modular unit. Schema may be nested to define components within components.

### 2.1.1 Nova Features

The Nova Desktop application is equipped with a *Design Canvas* for creating models using a visual language consisting of primitive components and plug-in extensions. The *Dashboard Canvas* contains controls and visualizing components for use at runtime. Among the components available for model building are system dynamic standards such as stocks, flows and terms (analogous to Stella's converters). At runtime these may be treated as modeling continuous processes through the selection of RK2 or RK4 integration. Nova adds a *Sequence* component for use with discrete time simulation.

**Clocked Chips.** One form of model abstraction is accomplished using the aforementioned chip. The latter may be further extended to a *Clocked Chip*, which joins a simulation clock to the abstracted model. Each iteration of the Chip's host capsule produces a complete run of the submodel according to the parameters of the joined simulation clock. This architecture, creating hierarchically synchronized subdivisions of time intervals, is particularly useful for sensitivity analysis and managing multiple runs.

**Code Chips.** System dynamics platforms require model definitions to include code-based expressions that define component values. These are usually limited to simple algebraic and conditional expressions involving numerical values. However, Nova's language for specifying these definitions encompasses all of JavaScript, and so admits loops, function definitions, and all of JavaScript's data structures (numbers, strings, arrays, objects).

Nova goes even further by including a user-coded *Code Chip* component that adds functionality to the primitive component set. Each Code Chip has a well-defined input/output interface; its content consists of a function relating its inputs to its outputs. Inputs and outputs can be of any data type, including objects and functions. Multiple instances of a given Code Chip definition can be deployed, and Code Chips can be exported and shared with other models. Code Chips may be defined with static internal variables that maintain state over the course of simulation.

**Aggregators.** Since Chips are less useful when large numbers of submodels are required, it is more efficient to use some form of array-like container to hold a set of Capsule elements. Like the array, an organizing structure (i.e., an index set) is required to provide a uniform means of access to these constituents.

We take this one step further by adding a set of properties and primitive operators that enforce a topological structure on the Capsules. For example, if we organize the Capsules into a two-dimensional lattice, each represents a single cell in a cellular automaton. In order for this to be of any value, however, each cell must be able to identify its own coordinates and recognize cogent topological structure; for example, cell neighborhoods. This is the role of Nova's aggregating components: 1) organize and provide access to a (possibly large) set of constituent Capsules; and 2) provide a set of properties and primitive operators that create topological structure and foster information transfer among those constituents. Using this fundamental design the four aggregating components currently available with Nova provide different topological organizations for their elements.

**Plug-ins.** Nova provides a plug-in API for Java-based extensions to the platform. Plug-ins are used for input and visualization, but also for extended computations. Among the plug-ins that are included in the

Nova distribution are ones for visualizing the five aggregators; creating histograms; defining graph-based functions; inputing matrices; and implementing a multi-layer, back-propagating neural network. Nova Plug-In API documentation is in progress to promote user-designed plug-ins.

### 2.2 Nova Online

*Nova Online* is a companion technology to the Nova Desktop that creates Web browser-based runtime applications implementing Nova Desktop-designed models.

Although Nova does support deployment as a Java applet, Nova Online uses a fully JavaScript-based implementation of the Nova runtime engine, so that the simulation runs entirely as a Javascript application in the browser. Moreover, the Nova Online environment makes ample use of D3 (Bostock 2013), a powerful tool for manipulating Web-browser documents based on data. Nova Online includes vivid realizations of Nova Desktop's visualizing components, and more are under development. A major goal of our current research is to automate the creation of Nova Online applications directly from their Nova Desktop specifications.

Nova Online relies almost entirely on a data-driven design with two aspects. First, the model schema generated through the capture process directly drive the JavaScript-based Nova Online simulation engine. Second, the HTML elements of the browser page are built using D3 with a common template and a JSON data structure describing the layout and specific input and display components. JavaScript versions of Nova plugins provide the interface between the running simulation and the input and display components, which are mostly specified using D3.

Figure 1 shows a simple agent-based SIR simulation modeling 4 patient states (susceptible, incubating, infectious and resistant). The agent viewer and graphs are configurable D3-based components used repeatedly in these applications. They interface with the running model as NovaScript plug-in components specified through schemas.

## 3 NOVA IN EDUCATION

### 3.1 Concept

A recent US National Academy of Sciences report stresses "using mathematics and computational thinking" in K-12 science classrooms in the future (Committee on a Conceptual Framework for K-12 Science Education 2012). Our response to this was to become involved in building Nova online apps that can be used as tools in teaching systems and computational thinking in science classrooms, particularly the biological sciences.

The educational approach for the Nova platform is to facilitate increasing levels of model abstraction while maintaining coherent and cognitively maintainable foundation components. The platform emphasizes composition of components, while exposing powerful tools for abstraction. An apt metaphor here is a ladder, where the Nova platform allows the user to climb upward to new levels of abstraction, while still allowing the user to "look below," inspect and modify downward to lower levels of model components. The foundation of the model presented here, or first rung on the ladder, is a single agent with a state that is either infectious or susceptible.

The second rung on the ladder, is to abstract this single agent into multiple connected agents. At this level of abstraction, agents now interact with a local neighborhood of agents, where disease can be transmitted between agents in the same neighborhood. This level of abstraction allows the student to explore the effect of contact neighborhood size and probability of transmission on the rate of disease transmission. We continue up the ladder to a third level of abstraction: multiple simulations. Because these are stochastic agent-based models, each simulation, even with the same parameters, generates a different epidemic trajectory. The general trend can only be discerned from multiple simulations, and the variability of epidemic trajectory given the same input parameters can be surprising. The fourth level of abstraction is to abstract across fundamental disease processes. The initial model only allows for susceptible and infectious(SI) states. By introducing recovery to a susceptible state(SIS) students can test hypothesis

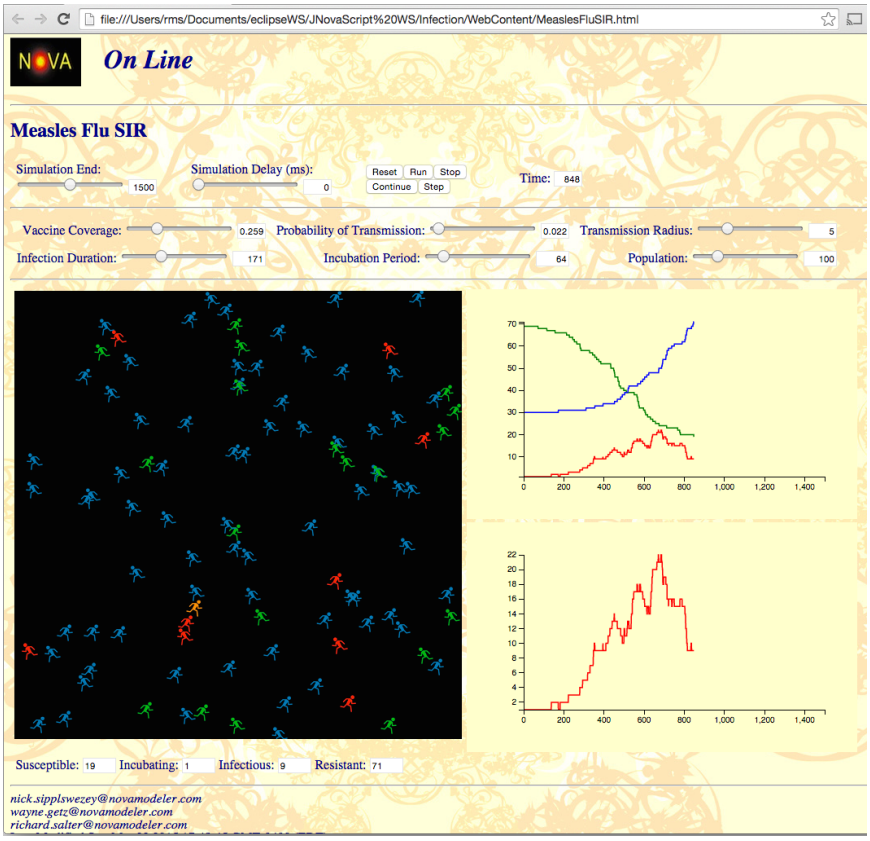


Figure 1: A Nova online agent-based SIR movement model in the plane with specifiable population size, radius of transmission (mass-action), probability of transmission when two individuals are within this radius, incubation period, duration of infection and level of vaccine coverage.

regarding the effect of recovery, multiple parameters, and using data from multiple simulations. Similarly they can compare results to a model that involves recovery to a resistant state(SIR)

The educational track allows for two powerful learning paradigms that are rarely paired effectively: rapid iteration and maintenance of complexity. The student can iteratively change properties of the model at any level of abstraction, while not needing to reconstruct the whole composition. The complexity of the model is maintained, as well as the ability to iterate rapidly. This pedagogical "ladder" approach to modeling complex systems has yielded positive results in teaching principles of infectious disease transmission to master level high school instructors, and to hundreds of students in the San Francisco Bay Area.

### 3.2 Epidemiology Example

The suite of apps that have been used to teach concepts in epidemiology in the San Francisco Bay area and, also, in support of a course on "An Introduction to Mathematical Modeling of Epidemics" taught by Dr. Lauren Riva at St. Mark's School, Southborough, Massachusetts.

Beyond providing students with a deeper understanding of infectious disease, the goal of the lessons prepared was to provide students with experience in computational modeling and an understanding of what 'modeling' is. This goal was achieved seating students in a circle and then asking them to play a game that simulated or modeled a disease process. Specifically, one student in the circle started out infectious, with the remaining susceptible (and with some resistant, in variants of the game) and then going around the circle with students in term generating random numbers (e.g. using dice or drawing numbers out of a hat) that would determine whether or not they became infected, based on how many other individuals

in their immediate neighborhood (which could vary in size) were infected. They would then be shown a NOVA app of the game (cf. Fig. 2), played according to the same rules, thereby making the point that the game itself is the model of a process, and the NOVA app is a digital implementation of the game they had just played.

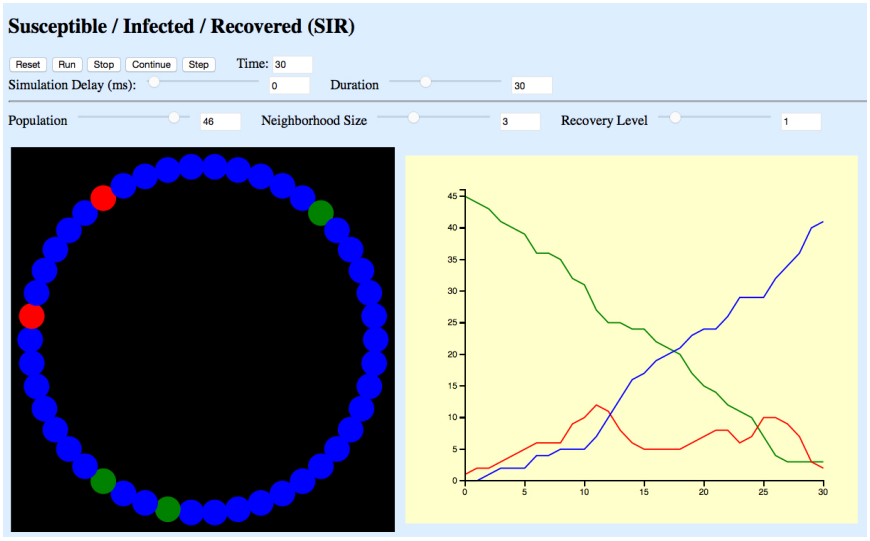


Figure 2: A cropped screen shot of a 46-player version of our Nova Online SIR circle game. The right-side shows the number of susceptible (S: green), infectious (I: red) and recovered with immunity (R: blue) over time, while the state of the circle (left side) is after 30 times steps. The neighborhood size on either side of the focal individual and the recovery rate can be selected using sliders.

Once the students had absorbed the notions of disease transmission and recovery without (SIS game) or with immunity (SIR game) by, first, physically and, then, digitally playing the various circle games, they then moved onto SIR simulations in settings (i.e. in the plane, and on networks). Depending on how many hours instruction are devoted to discussing disease transmission, and on the level of the group—high school, undergraduate, graduate classes in a school of public health, or workshops involving healthcare professionals—Nova apps can be used to introduce the concepts on the spread of disease over random networks (Hladish et al. 2012, Newman 2002) or networks regularized in some fashion, or structured into dense local clusters sparsely linked by long range connections (i.e. small world networks—cf. Watts and Strogatz (1998)), as illustrated in Fig. 3.

## 4 NOVA IN RESEARCH

In this section, we first outline our approach to formulating SIR agent-based models (SIR-ABM) in NOVA and then discuss their application to building models to explore aspects of the recent measles outbreak in the US and Ebola viral disease (EVD) in west Africa.

### 4.1 Agent-based models

Stochastic methods in epidemiological modeling fall within one of three paradigms: discrete-time compartmental models, continuous time-to-the-next-event models, or individual-based (more generally agent-based) models (Getz and Lloyd-Smith 2006, Cauchemez and Ferguson 2008, Vynnycky and White 2010). Nova facilitates coding models within any of these three paradigms; or, indeed, using a mixture of paradigms. Here, we only disscuss the ABM approach, because it is the only one that allows us to directly compute transmission chain lengths. Other approaches, though, have approximate methods for estimating the dis-

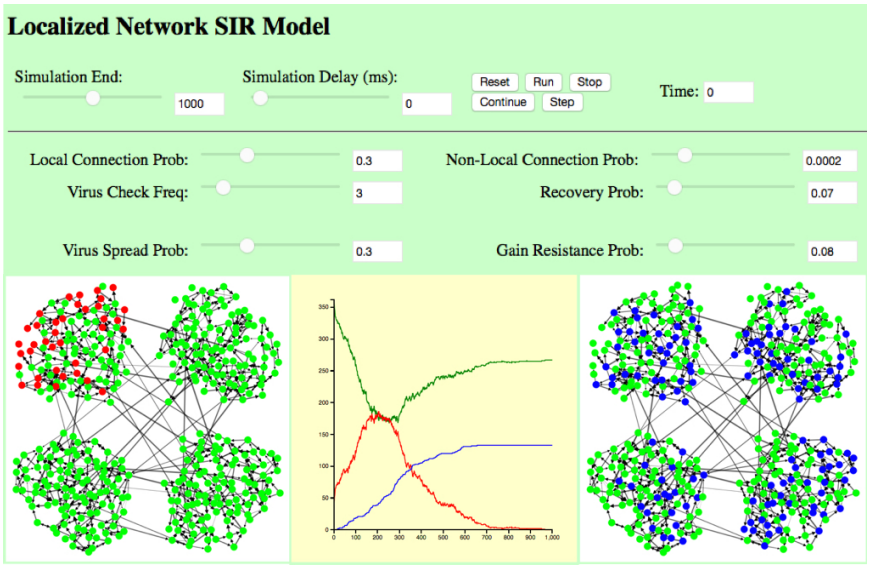


Figure 3: A collage of visual components of a Nova Online 'SIR epidemic on small-world networks' online app. Sliders in the top panel permit the density of connections within local clusters and the sparsity of connections among these clusters to be manipulated, as well as setting parameters that control rates of transmission, recovery and levels of resistance after recovery to be set. The lower left and right panels illustrate the initial and final (in this case after 1000 time iterations) states of the simulation (S: green; I: red; R: resistant) while the course of the epidemic is plotted (same color codes) over time.

tribution of transmission chain lengths using stochastic process theory (Barbour and Utev 2004). A further advantage of an ABM approach is that it also allows us to consider the specific characteristics and histories of individuals, should we want models to include host immunological factors (e.g. history of responses to different doses of the pathogen and possible cross-resistance to related pathogens or different strains of the same pathogen), genetic factors or physiological condition. SIR-ABMs can also be used to trace genetic changes within pathogen strains transmitted along a particular host chains (Magiorkinis et al. 2013).

Agent-based methods are conceptually simple (for an introduction see: Vynnycky and White (2010)). Essentially, each agent can be in one of a discrete number of states at time $t$. These states could be as simple as susceptible (S), incubating (E), infectious (I), or resistant (R), or could include that duration of the infection (i.e. individual in state $I_\tau$, $\tau = 1, ..., \tau_{\max}$). As time progresses, a Markov chain transition matrix is used to compute the probability that an individual makes the transition from state to state. This matrix is typically sparse, since most transitions are directional (e.g time since infection). These probabilities can either be constant or functions of the number of individuals in the population in particular states, but the matrix itself must be stochastic (i.e. all its columns add to 1).

In the simplest of all processes, an individual can either be in state S or state I and, under a random mixing, mass-interaction assumption, the probability that an individual makes the transition from state S to state I over the time interval $[t, t+1]$, for some transmission rate parameter $\beta > 0$, is given by (Getz and Lloyd-Smith 2006)

$$p_t = \text{Prob}\{\text{Individual S} \to \text{I on interval } [t, t+1]\} = 1 - e^{-\beta \times (\text{total number of I's at time } t)}$$

In more complicated processes individuals can either be community members or health-care workers, while susceptible can be either vaccinated or unvaccinated (in which case their susceptibility infection is reduced, but is not necessarily 0). Additionally, infectious individuals can be identified as such and then isolated and treated, or they can be in the community at large transmitting disease. In most real epidemics, individuals are not under going random mass interactions, but have contact rates somewhat independent of population

densities (e.g. sexually transmitted disease) or influenced by their movement among population clusters (metapopulation processes). We will discuss the specifics of such elaborations in the context of our Nova measles and Ebola viral disease models presented below.

In the Nova SIR-ABM formulations discussed here, we keep track only of individuals that pick up the pathogen and then pass through various disease stages (e.g. for Ebola: incubating, infectious in the community, isolated under treatment, and recovered or dead, as well as healthcare workers in these some stages). Susceptible individuals are collectively assumed to be either in essentially infinite pools (rates of infection are not limited by the supply of susceptible individuals) or in finite pools of specified size (rates of infection decrease as susceptible individuals are transformed to other states). As each infected individual is generated in our Nova model, he or she is given a unique agent number and all infected individuals, apart from the index case (source or root of the epidemic) is tagged with the number of the agent from whom they got the disease. This tagging allows us to construct both the next-generation distribution (cf. Fig. 5 below)—which is a discrete probability distribution of the proportion of infected individuals that passed on the disease to 0, 1, 2, ..., other individuals—as well as chain-length distributions that can only be generated at the end of the epidemic from a branching tree representation of the epidemic (e.g. the root individual has a chain length equal to the size of the epidemic, while individuals that do not pass the disease on to any others are terminals or leaves of the tree—see Klinkenberg et al. (2006) for a discussion of trees in the context of the utility of contact tracing data in emerging epidemics).

At selected times during the epidemic, a next-generation distribution can be constructed for all individuals acquiring the disease over specified intervals time. For example, if individuals can only transmit the disease within three weeks of becoming infected themselves, then six weeks into the epidemic, a next-generation distribution can be constructed based on the number of individuals that each infectious individual (including the index case) infected during the first three weeks of the epidemic. The mean of these next generation distributions—i.e. the expected number of infections that each infectious individual will generate during the course of the epidemic—is referred to as $R_0$ ("R zero") if it is computed near the start of the epidemic and $R_{\text{eff}}$ ("R effective") if it is generated later into the epidemic. The epidemic, on average, grows as long as $R_{\text{eff}} > 1$ and, on average, declines as long as $R_{\text{eff}} < 1$. Over the total epidemic (i.e. a distribution generated from all cases), if the epidemic is sufficiently large ($\geq 50$) then we can expect $R_{\text{eff}} \approx 1$ (Getz et al. 2015).

### 4.2 Measles

Measles is a classic, highly infectious SEIR process, with near life-long immunity for those who recover. Widespread application of the trivalent MMR (measles, mumps and rubella) vaccination program since the early 1980s has, through the phenomenon of herd immunity, dramatically reduced $R_0$ for measles in Europe and North America: specifically from values in the range 4 to 30 (estimates are highly variable, as discussed in Bjørnstad et al. (2002) and Mossong and Muller (2000)) to $R_0 << 1$ in communities where vaccination coverage exceeds 95% and $R_0 \approx 1$ in communities with coverage around 85-90%.

Over the past two decades, fueled by a now retracted 1998 study that linked the MMR vaccine to autism (Editors of The Lancet 2010), clusters of children with vaccination rates below herd-immunity levels are arising in the US (Lieu et al. 2015). Because of this precarious situation, beginning January 2015, an outbreak of measles occurred in California, fueled by high transmission rates at an amusement park acting as a "super-spreading hotspot." We built a Nova SIR-ABM model and online app designed to explore the role that under-vaccinated clusters (below herd immunity levels) play in fueling outbreaks in otherwise adequately vaccinated larger communities. In each homogeneous cluster, the latent period (E state) was set to 7 days, infectious period to 3 days (after which we assumed patients are isolated at home), and assumed removed individuals are either dead or fully immune. We selected an infection rate parameter that corresponded to a Poisson transmission process with mean $\lambda = 2.2$ infections per day. This produced an epidemic with $R_0 = 6.6$, and a herd immunity level at 84% vaccination coverage (i.e. any individual selected at random is only susceptible to infection with a probably of 0.16).

We next built a model that involved five different groups or regions: Regions 1 and 3 were large (during outbreaks $I << S$) homogeneous communities with 95% and 85% vaccination coverage respectively. Groups 2 and 4 were school communities attached to groups 1 and 3 respectively. Each of these groups had a limited number (400) of susceptible individuals, and these individuals spent 1/3 of their time in groups 2 or 4 and the remainder in their parent community. Individuals in the community also spend 2% of their time at superspreading centers—treated as Group 5—that had a low rate of infectious individuals entering from outside the five group system. We also assumed that $\lambda = 2.2$ in groups 1 and 3, but was five times this value in the schools (groups 2 and 4) and 15 times this value in the collective superspreading centers (group 5), where individuals are at high densities and in close contact with one another respectively (Fig. 4).

We ran this measles model under two scenarios: 1.) basic scenario (described above); 2.) basic scenario plus individuals not vaccinated in Region 3 spent 80% less time at school. When the model was run 100 times, the following results were obtained: i) Scenario 1: average epidemic sizes in Regions 1 and 3 were 348 and 2.4 individuals respectively; ii) Scenario 2: average epidemic sizes in Regions 1 and 3 were 42 and 1.7 individuals respectively. From these results, the strategy of sending unvaccinated children home from school during the course of an outbreak appears to be highly effective strategy in containing the outbreak.

### 4.3 Ebola

Although EVD is also an SEIR process, our Nova SIR-ABM model of this epidemic had important intervention components not considered in the measles model. First, we assumed that the behavior of individuals changes with the progression of time in a way that reduces risk, as community members become more informed regarding measures that can be taken to reduce exposure (e.g. refraining from washing the bodies of family members after death). Second, as the epidemic progresses the healthcare infrastructure gears-up to more rapidly detect and more effectively isolate infectious individuals. Third, healthcare workers play a central role in the epidemic, so they should be treated differently in the model from other members of the community. Fourth, we thought it useful to assess levels of vaccination, should an effective vaccine become available, to extinguish the epidemic or provide herd immunity.

Details of the model and our results are reported in Getz et al. (2015), so we only summarize some results here. Not surprisingly, since $R_0$ for EVD in the recent west African outbreak appears to be in the range $[1.5, 2.5]$, herd immunity levels are low (e.g. vaccination levels around 20% are expected to be very effective, unless more virulent strains of Ebola evolve) and moderate behavioral and health-care infrastructural responses taken together can be very effective. What is interesting to consider here, in the context of SIR-ABM models, is the production of the offspring distributions mentioned in Sub-section 4.1 above, for which Fig. 5 provides evidence of how these change during the course of the epidemic.

## 5 CONCLUSION

Beyond the epidemiological field, or any other field of computational population biology, Nova has application in all fields of the natural and social sciences, engineering and operations research. Since Nova ultimately generates an extended JavaScript program that runs in any Java real-time run environment, the front and backends of Nova are completely separable, and models built on a tablet can be run in the cloud. Thus our long-term goal is to improve both the portability and supercomputerability of Nova so that it is as equally at home in schools and high-powered research environments.

## ACKNOWLEDGMENTS

The development of NOVA has been supported in part by NSF grant CNS-0939153 to Oberlin College with Richard Salter as PI. We thank John Pataki of Logical Laboratories for his considerable help and input into creating and supporting the Nova Modeler website

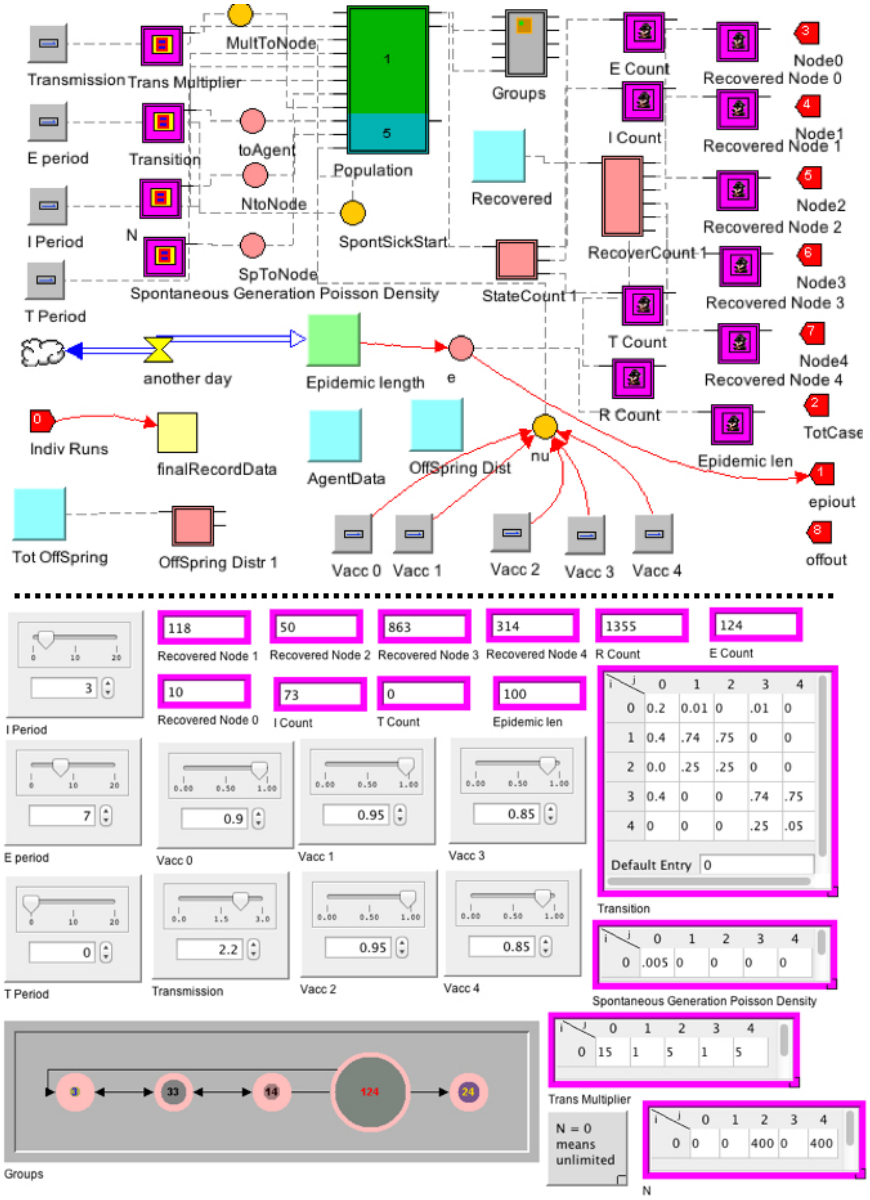


Figure 4: The Nova graphics canvas (top) and dashboard (bottom) for the multi-run level of the Nova SIR-ABM measles model.

## REFERENCES

Barbour, A., and S. Utev. 2004. "Approximating the Reed-Frost epidemic process". *Stochastic Processes and their Applications* 113 (2): 173 – 197.

Bellan, S. E., K. J. Fiorella, D. Y. Melesse, W. M. Getz, B. G. Williams, and J. Dushoff. 2013. "Extra-couple HIV transmission in sub-Saharan Africa: a mathematical modelling study of survey data". *The Lancet* 381 (9877): 1561 – 1569.

Bjørnstad, O. N., B. F. Finkenstädt, and B. T. Grenfell. 2002. "Dynamics of Measles Epidemics: Estimating Scaling of Transmission Rates Using a Time Series SIR Model". *Ecological Monographs* 72 (2): pp. 169–184.

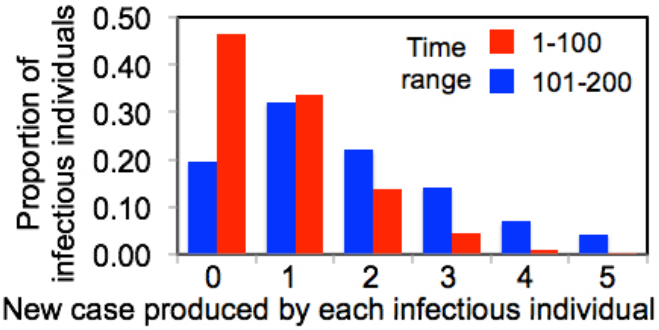


Figure 5: Two offspring distributions generated from one particular run of our Ebola model: the red and blue histograms were generated from cases transmitted by individuals infected during the first 100 (mean = 1.74) and the second 100 (mean = 0.77) days (time steps) of the epidemic.

Borshchev, A., Y. Karpov, and V. Kharitonov. 2002. "Distributed simulation of hybrid systems with AnyLogic and HLA". *Future Generation Computer Systems* 18 (6): 829–839.

Bostock, M. 2013. "D3 – Data Drive Documents". http://d3js.org/.

Cauchemez, S., and N. M. Ferguson. 2008. "Likelihood-based estimation of continuous-time epidemic models from time-series data: application to measles transmission in London". *Journal of The Royal Society Interface* 5 (25): 885–897.

Chowell, G., N. Hengartner, C. Castillo-Chavez, P. Fenimore, and J. Hyman. 2004. "The basic reproductive number of Ebola and the effects of public health measures: the cases of Congo and Uganda". *Journal of Theoretical Biology* 229 (1): 119 – 126.

Committee on a Conceptual Framework for K-12 Science Education, . 2012. *A Framework for K-12 Science Education: Practices, Crosscutting Concepts, and Core Ideas*. The National Academies Press.

Crockford, D. 2013. "Introducing JSON". http://www.json.org/.

Cross, P. C., and W. M. Getz. 2006. "Assessing vaccination as a control strategy in an ongoing epidemic: Bovine tuberculosis in African buffalo". *Ecological Modelling* 196 (34): 494 – 504.

Duncan, C., S. Duncan, and S. Scott. 1997. "The Dynamics of Measles Epidemics". *Theoretical Population Biology* 52 (2): 155 – 163.

Editors of The Lancet, . 2010. "Retraction—Ileal-lymphoid-nodular hyperplasia, non-specific colitis, and pervasive developmental disorder in children". *The Lancet* 375 (9713): 445.

Fortman-Roe, S. 2014. "Insight Maker: A general-purpose tool for web-based modeling and simulation". *Simulation Modelling Practice and Theory* 47 (September): 28–45.

Getz, W., and J. Lloyd-Smith. 2006. "Basic Methods for Modeling the Invasion and Spread of Infectious Diseases". *DIMACS Series in Discrete Mathematics and Theoretical Computer Science* 71:1–23.

Getz, W. M. 2013. "Computational population biology: linking the inner and outer worlds of organisms". *Israel Journal of Ecology and Evolution* 59 (1): 2–16.

Getz, W. M., J. P. Gonzalez, R. Salter, J. Bangura, C. Carlson, M. Coomber, E. Dougherty, D. Kargbo, N. D. Wolfe, and N. Wauquier. 2015. "Tactics and strategies for managing Ebola outbreaks and the salience of immunization". *Computational and Mathematical Methods in Medicine* Article ID 736507:Article ID 736507.

Hethcote, H. W. 2000. "The Mathematics of Infectious Diseases". *SIAM Review* 42 (4): pp. 599–653.

Hladish, T., E. Melamud, L. Barrera, A. Galvani, and L. Meyers. 2012. "EpiFire: An open source C++ library and application for contact network epidemiology". *BMC Bioinformatics* 13 (1): 76.

Jones, K. E., N. G. Patel, M. A. Levy, A. Storeygard, D. Balk, J. L. Gittleman, and P. Daszak. 2008, 02. "Global trends in emerging infectious diseases". *Nature* 451 (7181): 990–993.

Keeling, M. J., and B. T. Grenfell. 1997. "Disease Extinction and Community Size: Modeling the Persistence of Measles". *Science* 275 (5296): 65–67.

Klinkenberg, D., C. Fraser, and H. Heesterbeek. 2006, 12. "The Effectiveness of Contact Tracing in Emerging Epidemics". *PLoS ONE* 1 (1): e12.

Krause, A., and P. Lowe. 2014. "Visualization and Communication of Pharmacometric Models With Berkeley Madonna". *CPT: Pharmacometrics and Systems Pharmacology* 3 (5): 1–20.

Legrand, J., R. F. Grais, P. Y. Boelle, A. J. Valleron, and A. Flahault. 2007, 5. "Understanding the dynamics of Ebola epidemics". *Epidemiology and Infection* 135:610–621.

Lekone, P. E., and B. F. Finkenstädt. 2006. "Statistical Inference in a Stochastic Epidemic SEIR Model with Control Intervention: Ebola as a Case Study". *Biometrics* 62 (4): pp. 1170–1177.

Lieu, T. A., G. T. Ray, N. P. Klein, C. Chung, and M. Kulldorff. 2015. "Geographic Clusters in Underimmunization and Vaccine Refusal". *Pediatrics* 135 (2): 280–289.

Lloyd-Smith, J. O., A. P. Galvani, and W. M. Getz. 2003. "Curtailing Transmission of Severe Acute Respiratory Syndrome within a Community and Its Hospital". *Proceedings: Biological Sciences* 270 (1528): pp. 1979–1989.

Magiorkinis, G., V. Sypsa, E. Magiorkinis, D. Paraskevis, A. Katsoulidou, R. Belshaw, C. Fraser, O. G. Pybus, and A. Hatzakis. 2013, 01. "Integrating Phylodynamics and Epidemiology to Estimate Transmission Diversity in Viral Epidemics". *PLoS Computational Biology* 9 (1): e1002876.

McCarthy, M. 2015. "Measles cases exceed 100 in US outbreak". *BMJ* 350:H622–H622.

Morales, J. M., D. Fortin, J. L. Frair, and E. H. Merrill. 2003. "The Simile visual modelling environment". *European Journal of Agronomy* 18 (3-4): 345–358.

Mossong, J., and C. P. Muller. 2000, 4. "Estimation of the basic reproduction number of measles during an outbreak in a partially vaccinated population". *Epidemiology and Infection* 124:273–278.

Murai, F., B. Ribeiro, D. Towsley, and K. Gile. 2013. "Characterizing branching processes from sampled data". In *Proceedings of the 22nd international conference on World Wide Web companion*, 805–812. International World Wide Web Conferences Steering Committee.

Newman, M. 2002. "Spread of epidemic disease on networks". *Phys Rev E* 66:016128.

Salter, R. M. 2013. "Nova: A modern platform for system dynamics, spatial, and agent-based modeling". *Procedia Computer Science* 18:1784–1793.

Smith, F. P., D. P. Holzworth, and M. J. Robertson. 2005. "Linking icon-based models to code-based models: a case study with the agricultural production systems simulator". *Agricultural Systems* 83 (2): 135–151.

Vynnycky, E., and R. G. White. 2010. *Infectious Disease Modeling*. First ed. Oxford University Press.

Watts, D., and S. Strogatz. 1998. "Collective dynamics of "small-world" networks". *Nature* 393:440–442.

## AUTHOR BIOGRAPHIES

**WAYNE GETZ** (wgetz@berkeley.edu) is A. Starker Leopold Professor of Wildlife Ecology at the University of California, Berkeley and an Extraordinary Professor in the Mathematical Sciences at the University of KwaZulu-Natal. He received a Ph.D. degree in Applied Mathematics from the University of the Witwatersrand, South Africa, and a D.Sc. in Zoology from the University of Cape Town, South Africa. Beyond his 250 plus research publications, he has coauthored a monograph on Population Harvesting and a textbook on Calculus for the Life Sciences.

**RICHARD SALTER** (rms@cs.oberlin.edu) is Professor of Computer Science at Oberlin College. He received his Ph.D. in Mathematics from Indiana University. He is the author of more than 30 research and educational publications, and the designer of many software applications for industry and education. His research has been supported by the National Science Foundation and the Office of Naval Research.

**NICOLAS SIPPL-SWEZEY** (nsipplswezey@gmail.com) is a Hacker in Residence at Hack Reactor, San Francisco and a Research Specialist at the University of California, San Francisco. His research has been supported by the NIH Models in Infectious Disease Agent Study (MIDAS) network and the NIH Office of Science Education.