Cell-DEVS for Modeling Epidemiology: Dengue and Hepatitis B

Hoda Khalil

Carleton University, hoda.khalil@carleton.ca

Alonso Inostrosa-Psijas

Universidad de Valparaíso, alonso.inostrosa@uv.cl

Gabriel Wainer

Carleton University, gwainer@sce.carleton.ca

Using simulation models to predict future states in a field like epidemiology is essential in the absence of sufficient data and the difficulty of performing physical experimentation. However, simulation results cannot be trusted if the used modeling and simulation methodology assumes simplistic environments, does not provide formal mechanisms to reflect the complex reality, or does not offer enough supporting implementation and visualization software tools. To deal with this issues we use Cell-DEVS as a trustworthy, transparent, and well-supported modeling and simulation methodology. We show that Cell-DEVS is reliable for modeling and simulating epidemics and their complex dynamics, and apply the methodology to model the spread of two different diseases (Dengue and Hepatitis B). In each case, we provide a basic model and elaborate variations of the model to predict the behavior when the environment changes for reasons such as imposing control measures to limit the spread of diseases, and different immune system responses to a virus, among others. Throughout the process, we employ available theoretical capabilities of the Cell-DEVS formalism, as well as the supporting software implementation and visualization tools. By talking through previous epidemiology simulation work, providing examples of how Cell-DEVS overcomes some shortcomings of other methods, presenting the case studies, and discussing their results, we put forward Cell-DEVS formalism and its tools as an applicable package for researchers and policymakers for simulating and predicting epidemics behavior in complex environments.

CCS CONCEPTS • methodologies~Modeling and simulation • methodologies~Modeling and simulation~Simulation types and techniques~Discrete-event simulation • Applied computing~Life and medical sciences

Additional Keywords and Phrases: Epidemiology, Cell-DEVS, Dengue, HBV

ACM Reference Format:
First Author's Name, Initials, and Last Name, Second Author's Name, Initials, and Last Name, and Third Author's Name, Initials, and Last Name. 2018. The Title of the Paper: ACM Conference Proceedings Manuscript Submission Template: This is the subtitle of the paper, this document both explains and embodies the submission format for authors using Word. In Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY. ACM, New York, NY, USA, 10 pages. NOTE: This block will be automatically generated when manuscripts are processed after acceptance.

1 INTRODUCTION

The SARS-CoV-2 (COVID-19) outbreak in 2019 took the world by storm while its guard was down. The disease spread worldwide in less than four months. One year later, in March 2021, there were more than 121 million cases globally, with a death toll surpassing 2.6 million [1]. However, COVID-19 was neither the first nor the most fatal pandemic that attacked humankind. The Spanish flu was responsible for the death of 17.4 million people [2], the Dengue Virus (DENV) - that was discovered in the 1780s - kills 40,000 people every year, and the Hepatitis B Virus (HBV) is responsible for the death of 820,000 people every year [3]. The loss of lives and the spread of diseases led government officials, researchers, and healthcare providers to search for ways to prevent the expansion of diseases and eventually control them. While physical experimentation in the field of epidemiology was always constrained by ethical and practical limitations, Modeling and Simulation (M&S) methodologies were a viable and practical solutions. As such, trusted M&S methods and tools that are suitable for the epidemiology field are essential to understanding the dynamics of contagious diseases and forecasting the time course and behavior of contagions spread in a population. This makes them a critical tool for developing preparedness programs as they can be used for predicting the outcomes of different health policies to control contagious disease spreading [4-6]. M&S methodologies have been developed and used to model disease spread. Each methodology has its advantages, limitations, assumptions, and fields to which it is applicable. To trust the simulation results of a M&S methodology, one must evaluate its applicability to the specific field of study. First, a model whose assumptions approximately correspond to reality is trustworthy and will produce more valid predictions than other models whose assumptions are unrealistic (robustness thesis) [6]. Second, it is essential for the M&S methodology to have supporting development and visualization tools to be able to transparently examine the assumptions and the progress of the simulation. These tools facilitate testing the underlying model and comparing its progress to real-world observations.

Motivated by the need to offer an M&S methodology that avoids unrealistic assumptions and provides the necessary tools to guarantee M&S transparency, the objective of this manuscript is to prove the suitability of Cellular Discrete EVent Simulation (Cell-DEVS) [REF] for modeling and simulating epidemic dynamics. Cell-DEVS has unique characteristics that overcome the limitations of other M&S methodologies which strengthens its applicability to the epidemiology field. Cell-DEVS is an extension to DEVS [REF] that allows defining cell spaces with explicit timing delays. In the case of epidemiological diseases, the formalism allows modelers to define a nonhomogeneous population in its cellular space that naturally mimics real-world spaces. It also updates the state of the cells in the model's lattice asynchronously over a continuous time frame, which produces more precise results in less execution time and supports modeling complex timing behavior. Along with other advantages listed in section 2, this allows Cell-DEVS to avoid making assumptions that deviate from real-life scenarios. Furthermore, it has a complete suite of supporting tools that facilitate the development, simulation, visualization, and testing of the models and their simulation. As such, it provides a means to transparently examine the models and compare their progress to expected results. By this, Cell-DEVS formalism and its tools provide a package that is suitable for epidemiology modeling.

In section 2 of this manuscript, we review some of the related work using three main M&S methodologies that simulate the dynamics of contagious diseases. We discuss their underlying assumptions, advantages, and limitations. We also discuss Cell-DEVS and how it overcomes the limitations of other methods. In section 3, we present two case studies where Cell-DEVS is used to model and simulate the spread of two different diseases with various characteristics. We illustrate how Cell-DEVS can be used to evaluate the effect of different factors

on the spread of these diseases and how different epidemic rules can be incorporated into the models. We use the available suite of Cell-DEVS development, simulation, and visualization tools to present the results. Each model's results are discussed. Models, scripts, and videos produced by visualization tools are provided. Finally, the conclusion is provided in section 4.

2 BACKGROUND AND RELATED WORK

Many of the most recent works focusing on the dynamics of contagious diseases were mainly based on Agent-Based Simulations (ABS), Differential Equations (DE), or Cellular Automata-based (CA). Each of these approaches has its advantages, disadvantages, and theoretical basis.

ABS is a computationally expensive methodology that models disease contagion based solely on the rules governing the individual interaction among agents or with their environment. Many researchers have examined ABS to model contagious diseases. For example, Perez and Dragicevic have developed an ABS approach that simulates the spread of communicable diseases due to individuals' geospatial interactions [7]. The authors apply the approach to a case study of a measles outbreak that occurred in Vancouver, British Colombia in 1997. Each individual in the model is represented as an agent associated with places where they interact with other agents. However, the agents are mobile since they can move between places via a transportation network. Theoretically, the approach is useful to simulate various scenarios of outbreaks in realistic geospatial environments with modeled interactions between the population. However, the model has not been validated due to the lack of data on the states of the individuals on the micro level (e.g., interactions and disease transmission). Furthermore, representing each individual as an agent introduces computation complexities and requires extensive resources. Similar to the ABS framework developed by Perez and Dragicevic, Merini et al. have proposed a framework that considers transportation where each agent (i.e., the individual in the population) is associated with a space or a group of spaces [8]. Marini et al have also used agents by integrating two models into an ABS framework to predict influenza infections in Australia in 2050. The first one is a stochastic model that simulates the daily activities of all individuals in a population while the second is an epidemic model that simulates the transmission of influenza from an infected to a susceptible agent. The simulation results are validated against actual data collected for the influenza seasons of 2015/2016, 2016/2017, and 2017/2018. While the model predictions for the first two seasons only differ from the actual data by approximately 1% to 3%, the model predictions for the later season differ by approximately 13% from the actual data [8]. With a different focus, Cuevas also developed an ABS model to simulate disease transmission. The goal of the model is to evaluate COVID-19 transmission risk in facilities. Their simple agents represent individuals interacting inside a facility where each agent profile defines its social characteristics and health conditions that determine the behavior of the agent. Although several hypothetical scenarios have been used to test the model, the scenarios are not validated against real-life data [9].

Unlike classical mathematical models that assume the homogeneity of the population, ABS models can model individuals with distinct characteristics to provide more realistic results. However, while including all the rules that affect the spread of the disease may obscure the connections between the agents' characteristics and the resulting spread of the disease, simplifying the rules limits the model's ability to provide a realistic representation of the population. Furthermore, the ABS approach of fine-grain population modeling makes it challenging to simulate large-scale populations due to high computing demands.

Contrary to ABS which deals with individuals at the micro level, Differential Equations assume a homogeneous and well-mixed population [6]. When simulating epidemics, the individuals are classified into different categories (compartments) related to the infection stages of the contagious disease being modeled. These equation-based approaches are called compartmental models [10]. The classical susceptible-infectious-recovered model (SIR) [12] is the most widely adopted compartmental model for characterizing the spread of disease. The basic SIR model considers the population collectively, dividing it into compartments associated with the disease stages through which individuals in a population must pass. Consequently, the population consists of Susceptible (S), Infected (I), or Recovered (R) individuals. Each class of individuals is described by an Ordinary Differential Equation (ODE) or a Partial Differential Equation (PDE) (i.e., S(t), I(t), and R(t)) to calculate their respective fractions in the populations at time t. Other SIR-based models add features allowing representation of the disease's latency, effects of quarantine, vaccination policies, and population isolation enforcement [13].

The popularity of the compartmental models has encouraged others to use different variations of such models to simulate the epidemic's spread. SEIR models are one type of such extension, which adds a new class to consider Exposed (E) individuals for modeling diseases that, upon the infection, have an incubation period during which individuals pose no risk of infecting others [11]. Beira and Sebastião used compartmental models to simulate the spread of COVID-19 in Portugal. The availability of real-time data allowed the authors to calibrate and validate their models and hence make them more trustworthy for predicting future spreads. As in any other compartmental model, the authors box the population into different categories and model the disease spread based on those categories. The categories used are based on the previously developed PSEIRD (Protected, Susceptible, Exposed, Infected, Recovered, Deaths) model with some modifications such as considering vaccination [14].

Other studies research and compare different SIR-based models. For example, Yang et al. investigate the forecasting abilities of five ODE models, seven empirical functions, and four statistical inference methods. The different forecasting methods are compared based on COVID-19 data collected from seven provinces/cities in China. In addition to the SIR model and the SEIR model, the authors study other compartmental models. Those models are the SEIR-QD, SEIR-PO, and SEIR-AHQ. While the SEIR-QD adds a quarantined state introduced between infectious and recovered in the SEIR model, the SEIR-PO model considers public opinion by adding two states (Unconscious and Conscious) to model the individual's knowledge of epidemics and self-protection. As for the SEIR-AHQ, it stratifies the population as susceptible, exposed, infected but not yet symptomatic (A), infections with symptoms (I), hospitalized (H), recovered (R), quarantined susceptible (S_q), and isolated exposed (E_q). When comparing ODE-based models, the authors conclude that due to the trade-off between complexity and fitting accuracy, the SEIR-QD and SEIR-PO models perform better than other models (i.e., SIR, SEIR, and SEIR-AHQ). However, they note that in general, the RMSE of the model prediction decreases exponentially with the size of the training data [15].

The model developed by Wang and Yamamoto to simulate the spread of COVID-19 uses PDEs instead of ODEs to facilitate including geographical and social characteristics in Arizona in modeling the spread of COVID-19. The social characteristics are based on data derived from the Google Community Reports. Personal habits such as wearing facial masks and practicing social distancing are considered. The prediction results accuracy of the model is quite high (94%). Because of the lack of data on the granular level from different geographical areas, the authors base their modeling on regions of counties in Arizona [16]. Therefore, the lack of data is still

an obstacle when predicting the spread of new outbreaks. As such, any of the studied models may have overestimated or underestimated the pandemic development unpredictably. Hupper and Cartiel explain how the prediction success of SIR-based compartmental models is highly dependable on certain assumptions that simplify reality. When those assumptions are not valid, we cannot have faith in the prediction. Due to the unrealistic nature of those assumptions, adapting the SIR model to shape policies is faced with skepticism. For example, the SIR assumes a well-mixed homogeneous population and an exponentially distributed function of infection which is not applicable in many cases [6].

Hunter, Namee, and Kelleher compare compartmental and agent based models by simulating measles outbreaks in 33 Irish towns and measuring the simulation results. They have created an agent-based model that follows the SEIR (susceptible, exposed, infected, and recovered) rules with the agents transitioning between the four states. The synthetic data used to create the model is derived from real-life data available for an Irish town. As for the compartmental model, the population is categorized into 29 age groups where each group has two equations (susceptible but not vaccinated and susceptible vaccinated). The authors conclude that although the ABS models are more computationally intensive, they capture more information than the more popular compartmental models and thus may be more accurate [11].

Generally speaking, the main weakness of compartmental models is that they assume a homogeneous and well-mixed population, ignoring the individuals' variable infection susceptibility. Also, these models do not appropriately consider the spatial-geographical aspects of disease transmission [17]. On the other hand, CA models overcome these shortcomings and have been successfully used to model contagious disease-spreading dynamics - including COVID-19 - in combination with different variations of the compartmental models [6, 17, 18, 19, 20].

Dai et al. presented a CA model for the spread of COVID-19. The model considers a simplified social community with different parameters such as sex, age, and population movement using data from New York City and Iowa to validate the model. The authors then use the model to simulate different control scenarios to aid decision-makers in forming pandemic policies [21]. Another study that applies control measures considerations on a COVID-19 CA model is the one by Ghosh and Bhattacharya. The authors propose a probabilistic CA model while considering social distancing and its effect on the spread of the disease. The research considers different factors like the probability of exposure, probability of testing, and imposing lockdown [22]. The use of CA allows the authors of both studies to incorporate spatial factors such as realistic motion behaviors and demographic features into the dynamics of disease spread. Another study that uses CA to overcome the homogeneity assumption of SIR models is carried on by White, Rey, and Sanchez. They use a CA model that considers the three classes of population (S, I, and R) in each cell. However, the portion of each class of population varies from one cell to the other. The authors also consider the vaccination effects. The study does not provide a validation for the theoretical model [17].

While giving the modeler the flexibility of modeling a nonhomogeneous population, basic CA modeling has some shortcomings. Cell-DEVS overcomes some of those shortcomings while maintaining the same modeling flexibility. First, in CA simulation, the state transitions in the cells are computed synchronously which deteriorates the precision of the simulation. Cell-DEVS overcomes this by allowing asynchronous state updates over continuous time which produces more precise results in less execution time. Second, by allowing different delay functions for each cell, complex timing behavior can be represented in Cell-DEVS models, unlike CA. Third, contrary to CA, Cell-DEVS provides the means to integrate CA with other modeling approaches to model

more complex behavior [23]. A Cell-DEVS model is composed of an n-dimensional lattice of cells where every cell is an atomic model that can interact with other cells through the model's interface ports [33]. In addition to the advantages of Cell-DEVS as a formal modeling technique, many available tools support implementing the formalism. As such, Cell-DEVS is suitable for the requirements of epidemiology M&S.

The CA model proposed by White, Rey, and Sanchez [17] was used as a base for a Cell-DEVS model calibrated by Cárdenas and Wainer using data from the spread of SARS-CoV-2 in South Korea. The authors provided a Cell-DEVS model that allows for more accurate timing since, unlike CA, time advances in a continuous timeline when events happen [24]. In another study, Cárdenas, Inostrosa-Psijas, and Wainer [25] presented a Cell-DEVS SIRDS model for prototyping COVID-19-related scenarios. The authors used the model to study the effect of reduced mobility on the spread of the disease [26]. While this study used the SIRDS model in the Cell-DEVS space, others integrate natural phenomena such as viral particle spread and carbon dioxide dispersion into Cell-DEVS to model and simulate the spread of the disease under different conditions [27]. Other studies focus on the effect of certain measures on the spread of the disease such as the usage of different types of masks [28]. That said, Cell-DEVS has been used to study the proliferation of the COVID-19 pandemic specifically in several research works [25-28]. Nevertheless, the advantages of Cell-DEVS and its applicability for modeling and simulating epidemics in general, have not been experimented yet. In this research, we build conduct detailed models (DENV and HBV) to illustrate the applicability of Cell-DEVS formalism and associated tools to model and simulate all aspects and the relevant complexities of different epidemics.

3 CELL-DEVS EPIDEMIOLOGY MODELS

In this section, we present two Cell-DEVS models to simulate the spread of viruses with different natures and characteristics. For each case, we discuss the Cell-DEVS model states, neighborhoods and the different factors can were incorporated in the models as parameters to help understand the effect of those factors on the spread of the viruses and hence on controlling the epidemics.

3.1 Dengue Virus Modeling

The Dengue virus (DENV) is a mosquito-borne viral infection. The transmission of the virus can be caused by one or more of any of the four DENV serotypes (i.e., groups of single species of microorganisms that share a distinctive structure [29]). In other words, one could get infected by DENV four times each by a different one of the four closely related serotypes. The estimate reported by the World Health Organization (WHO) is that 100-400 million DENV infections occur each year, mostly in tropical and subtropical climates. Some of the infected cases are mild, whereas other cases are lethal. The largest reported number of DENV infections took place in 2019, with more than 3.1 million cases in the American region alone [30]. DENV infections can happen in both directions between mosquitos and humans, but the transmission between humans is low. However, vertical transmission from a pregnant mother to the infant is possible and may result in serious consequences (e.g., fetal distress and pre-term birth) [31]. Temperature and Dengue Vaccine (Dengvaxia) are two of many factors that affect the rate of spread of the disease [32]. Such factors are not easy to experiment with within the physical environment. Therefore, M&S provides a "virtual lab" for experimenting with such factors and measuring their effects. We use Cell-DEVS to model and simulate the spread of DENV and the different factors that may affect it (e.g., temperature) to find measures to control the spread and reduce the death rate caused by DENV. In this section, we explore the conceptual DENV model, the Cell-DEVS models, and the simulation results.

3.1.1 *Basic Model*

We start with a basic conceptual model with a few assumptions. We adapt the basic conceptual CA SEIR model validated by Gagliardi, da Silva, and Alves [32]. In the model, the population is divided into the human population and the mosquito population. Infection between members of the same population is not possible except through an intermediate agent from the other population. Humans can get exposed to infection when bitten by an infected mosquito whereas a mosquito is exposed to infection when biting an infected human. However, mosquitos do not get infected by interacting with another mosquito neither a human can get infected through an interaction with another human. In the basic model, this infection happens through one serotype only. Also, humans recover from the disease after a while whereas mosquitos stay infected and do not recover until the end of the simulation. Deaths and reproduction are not considered in this version of the model. As for movement, we limit the effect of the individuals' mobility to the neighborhood part of the lattice only.

We first specify the explained conceptual model using a state machine. Each individual in the two populations (humans and mosquitos) is defined, at any point in time, by one of several states (the I variable in the model definition). Figure 1 is the state diagram representing the basic Cell-DEVS model. A human cell in the model can be in one of four states: Susceptible Human (SH), Exposed Human (EH), Infected Human (IH), or Recovered Human (RH).

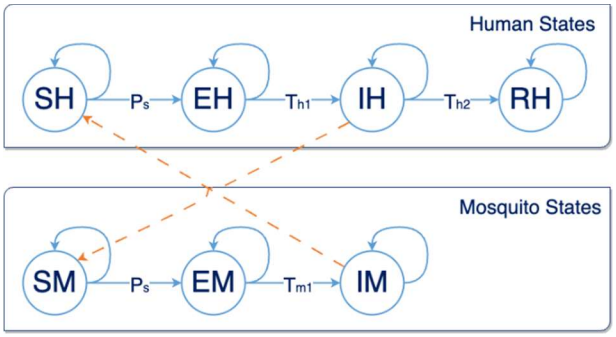


Figure 1: DENV Model I States: A member of the human population can be in one of four states while a mosquito can be in one of five states only. An individual transitions from susceptible to infected when interacting with an individual from the other population. The switching between the rest of the states is based on given periods. Note that the red dotted arrows are no transitions between the states but are external events that affect the state of the cell.

A mosquito can be in one of three states Susceptible Mosquito (SM), Exposed Mosquito (EM), or Infected Mosquito (IM). A Susceptible individual in any of the two populations is healthy but vulnerable to infection whereas an exposed individual is exposed if they have incurred the infection but are not capable of spreading the virus yet. After an incubation period, the exposed individual becomes infected and capable of spreading the virus as well. Finally, a recovered individual is a human that has been infected, has recovered from the disease, and is now immune to the virus. A recovered individual does not lose immunity and therefore remains in the same state until the end of the simulation. All the cells are initially susceptible and have a probability P_S of transitioning to the exposed state, after a specified period, due to exposure to infection from an infected neighbor.

The probability P_S of transitioning, as specified by the reference model [32], from the susceptible state to the exposed state is calculated based on the total number of infected individuals in the cells defining the neighborhood of the current cell using the following equation:

$$P_S = \Gamma P_G + \Lambda P_L \quad (1)$$

Where P_G is the probability of getting infected by any member of the other population that is not in the neighborhood, whereas P_L is the probability of getting infected by members of the other population who are present in the neighborhood of the individual only. As for Γ and Λ, those are the weight parameters where $\Gamma + \Lambda \leq 1$ to keep the probability in the *[0,1]* interval. Since we exclude the motion from this version of the model, we focus only on P_L which is defined as follows for the human population:

$$P_L = 1 - (1 - \lambda_m)^{n_{Im}} \quad (2)$$

As for the mosquito population, P_L is defined as follows:

$$P_L = 1 - (1 - \lambda_h)^{n_{Ih}} \quad (3)$$

Where λ_h and λ_m are the probabilities for a human and a mosquito, respectively, to get infected by a single infected individual, in the neighborhood, from the other population. n_{ih} and n_{im} are the numbers of infected individuals in the cell's neighborhood that belong to the human population and the mosquito population, respectively, at time t. In the Cell-DEVS representation of the basic model, we use a cell space (*C) that is composed of* two 30 X 30 ($\{t_1, \ldots, t_n\}$) cell lattice layers where humans reside in the top layer, whereas the mosquito population is in the bottom layer. The individuals of each population reside in one cell and are affected by a Moore neighborhood of a radius of 1 from the other population. Figure 2 illustrates the neighborhood cells that can contribute to the changes in the state of a mosquito cell (referred to as *M* in Figure 2 (a)), the neighborhood for a human cell (referred to as *H* in Figure 2 (b)), and the number representation of the possible states of any cell (Figure 2 (c)). As shown in Figure, each cell's neighborhood (η) consists of 10 cells; one is from the same population while the remaining nine are from the other population.

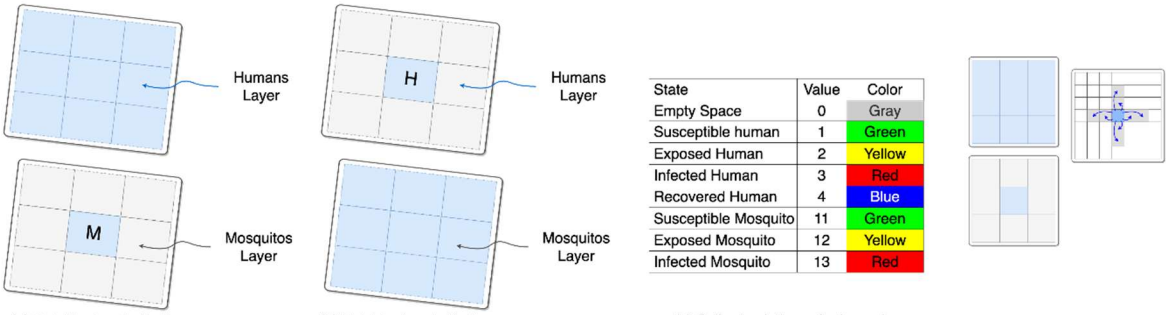


Figure 1: (a) The shaded neighborhood cells can affect the new state of a mosquito cell. (b) The shaded neighborhood cells can affect the new state of a human cell. (c) The table shows the color scheme used to represent the states of the cells in

the simulation results of the basic model. A grey cell is an empty cell. The different states of members of the human population are numbered 1-4, whereas the different states of members of the mosquito population are numbered 11-13.

The transitions between the different states shown in Figure 1, calculated based on the defined neighborhood, are represented in CD++ using the following rules:

DENV Model I: Basic Dengue Model Rules

```
1   zone : mosquito-rule { (0,0,0)..(30,30,0) }
2   zone : human-rule { (0,0,1)..(30,30,1) }
    %% Mosquito Lattice
3   [mosquito-rule]
4   rule : { 11 } { $s := 11; } 1000 { (0,0,0) = 0 }
5   rule : { 12 } { $s := 12; } 1000 { (0,0,0) = 11 and
                                       random <= (0.8 * (1 - power(0.06, (statecount(3))))))}
6   rule : { 11 } { $s := 11; } 1000 { (0,0,0) = 11 }
7   rule : { 13 } { $s := 13; } 1000 { (0,0,0) = 12 }
8   rule : { 13 } { $s := 13; } 8000 { (0,0,0) = 13 }
9   [human-rule]
10  rule : { 1 } { $s := 1; } 1000  { (0,0,0) = 0 }
11  rule : { 2 } { $s := 2; } 1000 { (0,0,0) = 1 and
12                                   random <= (0.8 * (1 - power(0.06, (statecount(13))))))}
13  rule : { 1 } { $s := 1; } 1000  { (0,0,0) = 1 }
14  rule : { 3 } { $s := 3; } 3000 { (0,0,0) = 2 }
15  rule : { 4 } { $s := 4; } 8000 { (0,0,0) = 3 }
16  rule : { 4 } { $s := 4; } 1000  { (0,0,0) = 4 }0 100 { t }
17  [epidemic-rule]
18  rule : { $s } 100 { t }
```

In lines 1 and 2, we use the *zone* feature provided by CD++. Each zone has its own transition rules that are different than the rules of the other zone. We define two zones: the mosquito zone (the bottom 30x30 cells layer of the model) and the human zone (the top 30x30 cells lattice of the model). The first 5 rules (lines 4-8) in the script Model Rules I define the transitions for the mosquito population, whereas the following 6 rules (lines 10-15) define the transitions for the human population. Rule 1 is applied at the beginning of the simulation when the default cell state (*s)* is 0. In this case, the cell is switched to be a susceptible mosquito since the model assumes that all individuals are susceptible unless indicated otherwise by the .val file. A similar rule is stated for the human population but with a different numerical representation for the states (line 10). Note that (*x,y,z)* indicates the value at the outport of a cell with the coordinates x, y, and z whereas $s indicates the value of the state variable *s* of the cell. The second rule (line 5) checks the precondition for switching a cell from the susceptible state to the exposed state. The precondition consists of two logical expressions that must coexist for the switch to happen. First, the cell must be in the susceptible state (SH with numerical value 1). Second, we use the CD++ *random* function to generate a random number between 0 and 1. This is to represent the probability. If this generated random number is less than or equal to the probability expressed in (2), the cell switches. We choose λ to be 0.06 and Λ to be 0.8 for this version of the model, but domain knowledge experts

can easily change those values based on available statistics. The CD++ function *statecount* is used to calculate the number of infected humans in the neighborhood illustrated in Figure 2 (a). This rule also has an equivalent rule in the human zone (line 11). In case the second precondition is not satisfied, the cell remains in the susceptible state (line 6 and line 13). The transition from exposed to infected happens after a specified time (lines 7 and 14). An infected mosquito remains infected for the simulation lifetime (line 8) whereas an infected individual recovers eventually and stays recovered for the rest of the simulation (lines 15 and 16). Finally, for a cell in any of the two populations where none of the preconditions of all the rules has been satisfied, the epidemic rule (line 17) applies. The value of the current state *(*s*)* of the cell is sent to the output port.

Figure 3 shows the simulation results of running the rules listed in the Basic Dengue Model. At the beginning of the simulation, all humans are susceptible while some mosquitos are infected. The shown results are visualized using DEVS Simulation Viewer [33].

Figure 2: DENV Model I Simulation Results: (a) After one cycle, some human cells transition from susceptible to exposed due to exposure to infected cells from the mosquitos in the neighborhood cells. (b) After ten cycles, the spread of the disease is evident in both populations even if the human population started the simulation in a healthy but susceptible state. (c) After 20 simulation cycles, humans started to recover and block the spread of the disease in the human population. (d) Most humans have recovered, and the infected mosquitos remain infected until the end of the simulation.

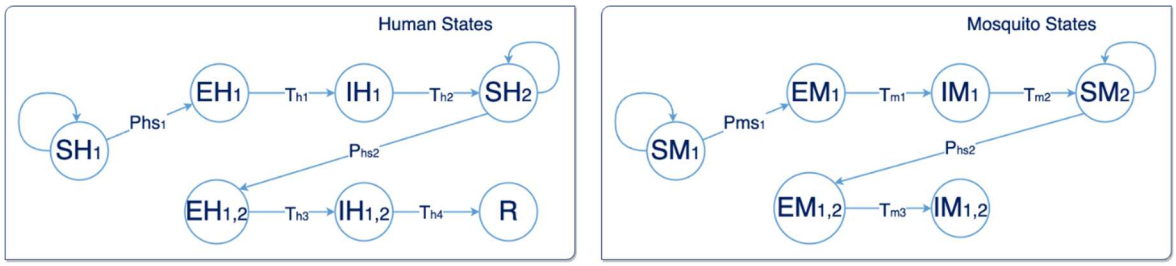
After one simulation cycle, some humans (right side of Figure 3 (a)) are shown to be exposed due to having infected mosquito cells in the neighborhood in the bottom loner (shown on the left side of Figure 3 (a)). Figure 3 (b) shows the simulation results after 10 cycles where the disease started spreading rapidly to the human population and back to the mosquito population. After ten more cycles, some humans started recovering and thus stopped spreading the disease to the mosquito population and stopped changing states (Figure 4 (c)). As

a result, by the end of the simulation (Figure 4 (d)) most humans have recovered while most mosquitos are infected as they never recover from the infection state.

3.1.2 Vaccine, Temperature, and Two Stereotypes Dengue Model

In DENV Model II, we incorporate different factors that may affect the spread of DENV in a population. Temperature is one of the factors we consider since it has been proven to alter the rate of the spread of DENV [35, 36]. Studies have observed a higher transmission rate of DENV-2 with higher temperatures. Other studies have reported that mosquitos cannot spread the virus in environments with a temperature that is less than or equal to 18°C [36]. Another factor included in the second version of the Dengue model is vaccination. There have been several efforts to develop a Dengue vaccine. In 2019, The U.S. Food and Drug Administration approved *Dengvaxia* as a vaccine to prevent the spread of DENV where the virus is endemic, and only to youth aged 19-16 who have been previously infected with the virus [38].

In addition to the temperature and the vaccine, we consider two serotypes in this version of the model. Figure 4 (a) and Figure 4(b) show the different states and the transitions between the states for the Humans population and the Mosquitos populations respectively for the second DENV model. A mosquito population can be in one of 6 states, whereas a human can be in one of 7 states due to the additional recovered state. Due to the presence of more than one serotype, we assume that an infected individual can become susceptible again. For example, a human who has been infected by serotype 1 and thus in the state Infected Human 1 (IH1) becomes susceptible to serotype 2 (SH2) and can get exposed to serotype 2 (EH2). This happens when that human is bitten by a serotype 2 infected mosquito (SM2). Note that the assumption here is that serotype 1 appears in the population first followed by the second serotype. However, other scenarios can be modeled using different state diagrams. The interaction between the two populations is omitted from Figure 4 for simplicity. Any member of the population in a susceptible state remains in that state until receiving an external event (bite) from the other population. Hence, susceptible states in both populations have self-transitions.



(a) Humans' state in second DENV model. (b) Mosquitos' state in second DENV model.

Figure 3 DENV Model II: (a) There are 7 states for a human cell whereas (b) a mosquito can be in one of 6 possible states. The probability of getting infected due to serotype 1 is Ps1, whereas the probability of getting infected due to serotype 2 is Ps2.

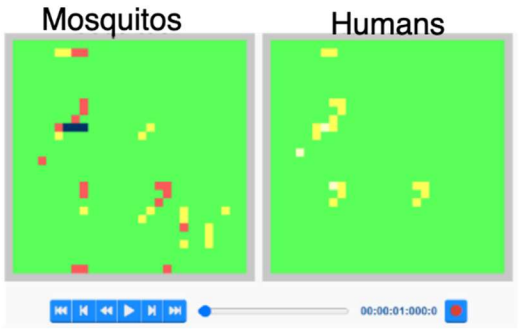
For DENV Model II, we use the same basic Cell-DEVS representation used in the basic model. We use two girds, each with dimensions of 30X30 cells. However, we add rules to handle the additional states and transitions required for incorporating the second serotype. For example, the mosquito zone rule in line 9 of

DENV Model below I is replaced by the first two rules listed below, whereas the human zone rule in line 15 of the DENV Model I is replaced by the rules in lines 3 and 4. All the models' code, initial value files, and execution instructions are available on our GitHub repository [39].

DENV Model II: Examples of Additional Rules for Handling 2 Serotypes

```
1  rule : { 14 } { $s := 14; } 4000 { (0,0,0) = 13 and random < (0.8 * (1 - power(0.06,
                                    (statecount(6)))))}
2  rule : { 13 } { $s := 13; } 4000 { (0,0,0) = 13 }
3  rule : { 4  } { $s := 4; } 4000 { (0,0,0) = 3 and random < (0.8 * (1 - power(0.06,
                                     (statecount(16)))))}
4  rule : { 3  } { $s := 3; } 4000 { (0,0,0) = 3 }
```

Figure 5 shows the simulation results of the test case scenario of DENV Model II. The simulation reflects the newly added rules. Figures 5(a) and 5 (b) visualize the simulation results at the beginning and after 23 cycles respectively, whereas Figure 5 (c) shows the numerical value of each state and the color legend used in the simulation visualization. Initially, only 3 mosquitos are infected with serotype 2 while around 20 are infected with serotype 1 (Figure 5 (a)). After 23 cycles, one can observe from Figure 5 (b) that most individuals in both populations have already been infected with serotype 1 and are now susceptible only to the second serotype (states 4 and 14), whereas some individuals in the human population have caught the virus from the 3 mosquitos initially infected with serotype 2. Then, those infected humans are spreading the virus back to the mosquito population (states 6 and 16). We also notice that an individual in the human population has already recovered from serotypes 1 and 2 (state 7) and will remain as such until the end of the simulation. The color legend of Figure 5 (c) will be used to visualize the simulation results for the rest of the DENV models presented in this paper.



(a) At the beginning of the simulation.

(b) After 23 cycles.

State	Value	Color
Empty Space	0	Gray
Susceptible human 1	1	Green
Exposed Human 1	2	Yellow
Infected Human 1	3	Red
Susceptible human 2	4	Beige
Exposed Human 2	5	Cyan
Infected Human 2	6	Navy
Recovered Human	7	Blue
Susceptible Mosquito 1	11	Green
Exposed Mosquito 1	12	Yellow
Infected Mosquito 1	13	Red
Susceptible Mosquito 2	14	Beige
Exposed Mosquito 2	15	Cyan
Infected Mosquito 2	16	Navy

(c) Cells simulation color legend.

Figure 4: DENV Model II Simulation: (a) At the beginning of the simulation, only 3 mosquitoes are infected with serotype 2 (state 16). (b) After 23 cycles, the simulation shows that most individuals infected with serotype 1 have recovered from serotype 1 (states 4 and 14), but the human population started contracting the virus through serotype 2 (state 6) and spreading it back to the mosquitos.

To consider the temperature, we have assumed a temperature range of 18ºC to 50ºC since studies have shown that the virus does not spread in lower temperatures [35]. We add to the model a state variable t to represent the temperature and scale the probability of virus spread by the range of temperatures. Figure 6 shows the simulation results of the model when the environment temperature is considered. When comparing Figure 6 (a) and Figure 6 (b), it is obvious that the high temperature has accelerated the infection rate which shows as soon as 30 cycles from the start of the simulation. In the case of hot temperatures, most members of both populations are already infected with serotype 1 and susceptible to serotype 2. In addition, some humans have been infected by both serotypes; some have recovered, and others are still spreading serotype 2 (Figure 5 (a)). In the case of a lower temperature, after 30 cycles most individuals are still susceptible to stereotypy 1 which shows a much slower rate of virus spread (Figure 5 (c)). At the end of the simulation in the case of hot weather, all individuals get infected with both serotypes (Figure 5 (b)), whereas in the case of cooler temperatures, many individuals do not get exposed to any of the two serotypes while most of the populations have been infected by one serotype only (Figure 5 (d)).

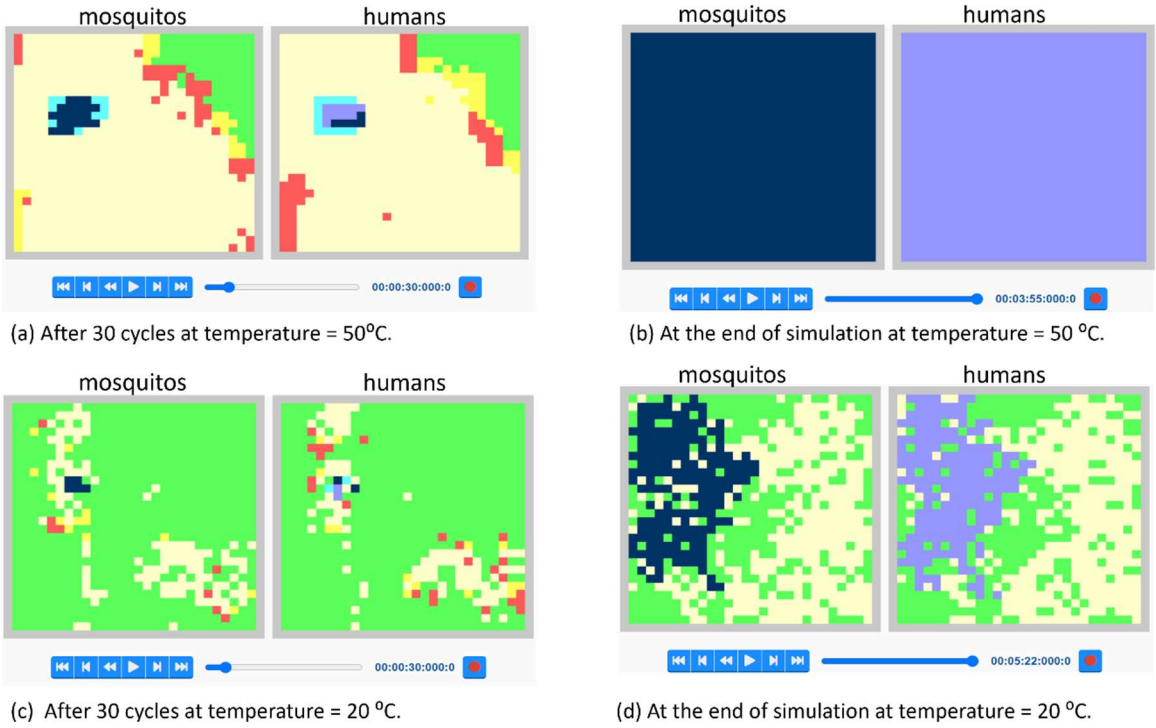


Figure 5 Simulating the Effect of Temperature on DENV Spread: (a) In hot weather, the virus spreads faster. Therefore, most members are already infected by serotype 1 after 30 cycles. (b) By the end of the simulation, all members have contracted the virus through both serotypes. (c) After 30 cycles in the cooler temperature, we can see that, in contrast to (a), most individuals have not yet been infected by any of the two serotypes. (d) At the end of the simulation in cooler weather, approximately only one-third of the human population have contracted the virus through 2 serotypes while many individuals did not get infected at all.

In Figure 7, we visualize the effect of vaccination on the human population. We assume that vaccines make the recovered individuals immune to both serotypes and thus never change state throughout the simulation lifecycle. In other words, a vaccinated human may get exposed to the virus, but they do not transition from the exposed state to the infected state. Therefore, they do not spread the virus back to mosquitos when bitten. However, a probability of transitioning from an exposed state to an infected state can be added to model each vaccine depending on its effectiveness as determined by the domain knowledge experts. All simulations shown in Figure 7 consider a temperature of 50ºC. In Figure 7 (a) and Figure 7 (b), almost the full population (≈90%) is vaccinated and therefore the spread of the virus is minimal. On the contrary, in Figure 7 (c) and Figure 7(d), around 50% of the human population is randomly chosen, through a CD++ rule, and vaccinated. Hence, around half of the population contracted DENV through both serotypes by the end of the simulation (Figure 7(d)). The viral spread in Figures 7 (c) and (d) is vicious when compared to Figure 7 (a) and Figure (b) because of the lower vaccination rate. However, the spread is much lower than the visualized spread in Figure 6 (a) and Figure 6 (b) where there is a complete absence of vaccination.

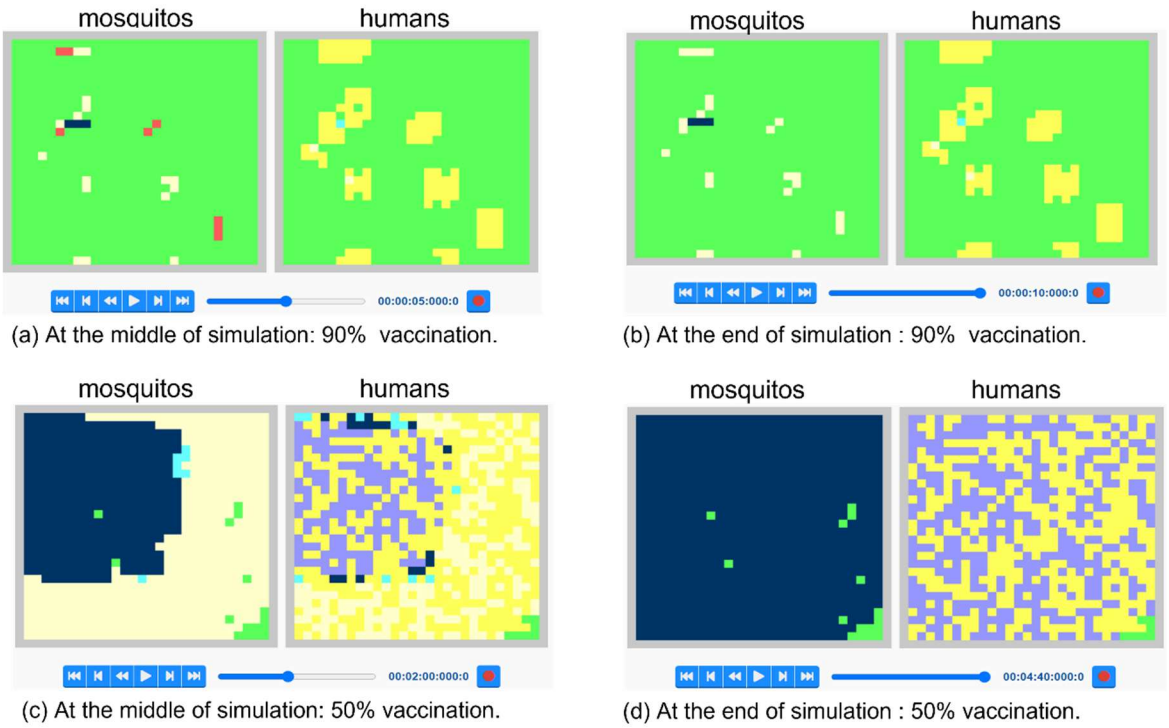


Figure 6 Simulating the Effect of Vaccines on DENV Spread: (a) When almost all humans are vaccinated, the spread of the virus is minimal. (b) By the end of the simulation, few members have contracted the virus in both serotypes (state 4). (c) DENV spread is vicious, compared to (a) when only half the population is vaccinated. (d) Around half the population has contracted the virus through the 2 serotypes and recovered.

3.1.3 Discussion

We have discussed in this section how Cell-DEVS and its supporting tools (e.g., CD++ and DEVS simulation visualizer) [32] can be used to model the spread of DENV. The models presented consider different factors such as the spread of the virus through more than one serotype, the effect of vaccination, and the effect of different temperatures on the viral spread. Noteworthy, we are concerned with showcasing that Cell-DEVS as a formal specification method and its supporting tools can be used for modeling a viral spread of a disease such as Dengue. However, changes to the model rules details such as the probability of infection and the effectiveness of vaccines can be easily adjusted based on domain knowledge expertise and the evolution of research in the area of epidemiology. All the used models are available on GitHub [39] and videos of the simulation results are available through the ARSLab YouTube channel [40].

3.2 Hepatitis B Virus

The second model we present here is modeling the spread of Hepatitis B. Hepatitis is an inflammation of the liver caused mainly by a viral contagion or a toxic element. There are several types of viral-based hepatitis, each named by a letter (going from A to F). In this work, we focus on the modeling and simulation of the Hepatitis B Virus (HBV) [41]. According to the WHO, HBV is a significant global health issue, with over 296 million

individuals carrying the virus. At a global level, HBV is most common in Asia and Africa, where more than 116 million and 81 million individuals (respectively) are chronic patients. Estimates are that approximately 1.5 million individuals contract the virus each year. In 2019, over 800,000 people died of cirrhosis or cancer developed from chronic HBV [42-44]. About 10% of infected adults will develop chronic liver disease; in approximately 30% of them, HBV will evolve into chronic hepatitis [45]. Most people with acute cases will recover from the illness on a short-term basis, while few might develop it as a long-term chronic disease, leading to cirrhosis or even cancer. Age plays a crucial role in the development of HBV long-term infection effects. Almost 90% of individuals infected before age 6 develop chronic infection. But the risk almost fades completely for individuals over six years old; nearly all fully recover. Among individuals, HBV is transmitted in different ways; However, the most common cause of transmission is horizontally through the exchange of body fluids. It can also be passed vertically by an infected mother to her child during pregnancy or at birth (perinatal contagion) [41, 42]. When an individual contracts the disease, a series of interactions occur between its immune system and the virus. Eventually, within the liver, HVB spreads in a hepatocyte-to-hepatocyte fashion.

Again, modeling and simulation allow for experimentation in a safe and controlled environment while regarding ethical limits. Simulation allows experimenting with changes to different factors and strategies to measure their effects on the model. In this section, we use Cell-DEVS to model and simulate the spread of HBV among cells within the liver, to study the factors that may affect the seriousness of this illness, according to a CA model defined by Xiao, Shao, and Chou [6]. In this section, we explore the conceptual HBV model, the corresponding Cell-DEVS model, and the simulation results.

3.2.1 Basic Model

We adapted the probability CA compartmental model developed by Xiao, Shao, and Chou [41] for the modeling and simulation with Cell-DEVS. The hepatocytes are comprised of two subsets of cells, namely, R cells (which are resistant to the virus) and S cells (which are susceptible to the virus and are its primary focus for infection [46]). Both types of cells respond differently to the virus depending on their level of maturity as specified by the London-Blumberg model [47]. Nevertheless, both R and S cells can acquire the virus by the contact of infected neighboring cells. While the lifetime of infected cells is shorter than that of a healthy one, infected cells can become healthy again. Cell reproduction occurs by the division of R cells, representing cell replenishment. New cells can transform from R to S to maintain their ratio. However, the number of cells in the model remains constant.

In the model, the cells of the CA represent the hepatocytes. The liver is described as a two-dimensional lattice where each cell corresponds to a R or S hepatocyte. Also, in the model, R cells can divide (or reproduce) into R or S cells as needed. On the other hand, S cells represent fully grown and mature cells.

The infection process is represented by cell-to-cell spreading. Cells at any state can die due to lifetime expiration but can revive, describing their replenishment due to R cell division. The model presented in [41] has six compartments to represent the state of each hepatocyte: (1) Susceptible R cell, (2) Susceptible S cell, (3) Infected R cell, (4) Infected S cell, (5) Defectively infected (R or S) cell, and (6) Dead cell. A cell (of any type) can die by being killed by immune cells or surpassing its lifetime. The amount of hepatocytes remains the same, although they can change type (R to S), and they can change state according to their stage regarding the disease.

We first specify the model of [41] using a state machine. In it, at any given time, a cell can be in one of several states. As depicted in Figure 8, the model has six compartments for representing each cell: Susceptible R cell (SR), Susceptible S cell (SS), Infected R cell (IR), Infected S cell (IS), Defectively infected cell (DI), and Dead cell (D). A susceptible cell – of either type - is healthy but vulnerable to infection, so they can contract the virus when interacting with infected cells. Infected cells can recover or become defectively infected at different rates. A defectively infected cell cannot transmit the virus anymore and will die soon. A cell of any kind dies when it surpasses its lifetime.

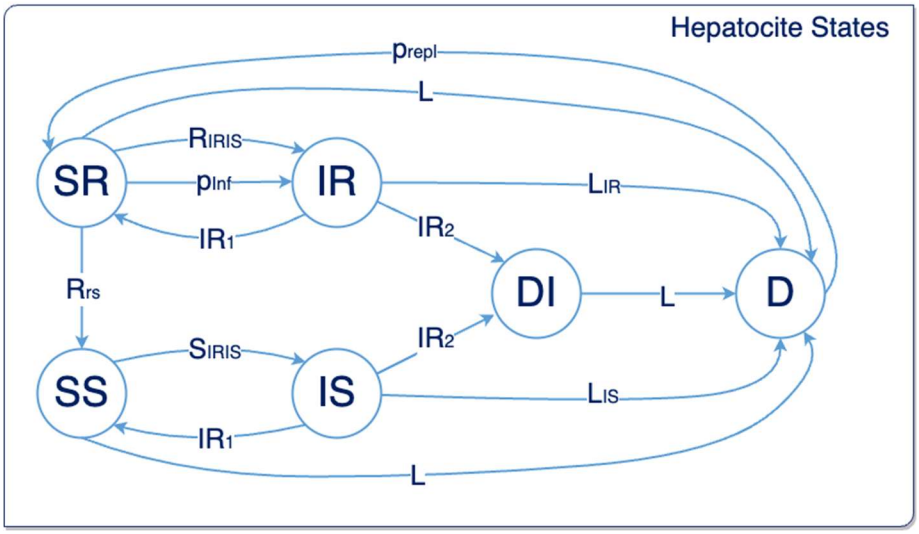
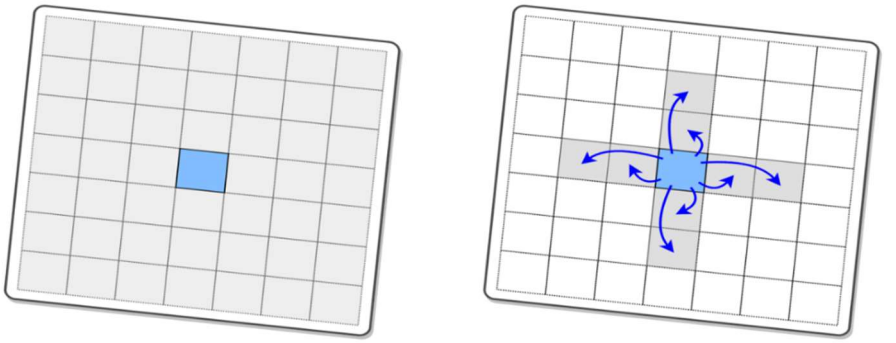


Figure 8 HBV Model States: A hepatocyte can transition to infected when interacting with infected neighbor cells, while maintaining its type. Infected cells can recover back to the corresponding susceptible state or become defectively infected. Replenishment of cells occurs randomly, after a dead cell becomes a new R cell. Transitions to the dead state are based on time periods [41].

In the Cell-DEVS representation of the HBV model, we use a 10x10 cell lattice for representing the hepatocytes (although it can be modified as needed). As observed in Figure 8, the transition among states for a hepatocyte can occur either probabilistically (based on its state and that of its neighbors) or by the expiration of its lifetime. Thus, two state variables are defined to store the current cell's state and another to keep track of the moment of its "birth" for age calculation. A set of 9 rules describes the cell model's state transitions [41]: Rule 1 describes that susceptible cells die once their lifetime reaches a period defined by *L* (transition SR or SS to D). Rule 2 states that defectively infected cells die after their lifetime reaches a period defined by *L* (transition DI to D). Rule 3 describes that susceptible cells become infected with probability defined as R_{IRIS} = S_{IRIS} = (INFECT_RATE_R x n + INFECT_RATE_S x m)/8, where n and m are the amounts of neighboring infected R and S cells, respectively (transitions SR to IR and SS to IS). Such a neighborhood consists of the four nearest neighbors to the cell and the next nearest four neighbors. Rule 4 indicates that healthy R cells can transform into susceptible S to maintain an overall ratio of R to S cells in the liver given by R_{rs}. Rule 5 allows the transition from state I to D, indicating that infected R or S cells die after a period defined (correspondingly) by L_{IR} or L_{IS}. Rule 6 describes how infected R or S cells recover from the virus and become susceptible again

at an inverse rate IR_1 (transitions from IR to SS and IS to SS). Rule 7 defines that an infected cell becomes defectively infected at an inverse rate IR_2 (transitions IR/IS to DI). This kind of cell cannot recover nor infect others. Rule 8 represents the repopulation of cells through the division of R cells according to a probability P^{repl} (transition D to SR). Finally, rule 9 states that a new R cell can become infected with a probability P_{inf} (transition SR to IR).



(a) Neighborhood to maintain the ratio of all R cells to all S cells almost invariable.

(b) Neighborhood affecting a given cell of HBV contagion.

Figure 9 HBV Model Neighborhoods: (a) The neighborhood for allowing computation of rule 8. (b) Neighbor cells to evaluate rule 1.

An excerpt of the HBV model implemented in CD++ is shown in "HBV Model I: Model Rules" shown below:

HBV Model I: Model Rules

```
1   StateVariables : cell_date_birth cell_type
2   localtransition : hbv-rules
3   [hbv-rules]
4   rule : { 1 } { $cell_type:=1; $cell_date_birth := -1 * ( round(uniform(0,48)) +
                                  round(time/1000) ); } 100 { (0,0) = -1 }
5   rule : { (0,0) } { $cell_type:=(0,0); $cell_date_birth := -1 * ( round(uniform(0,48))
                                   + round(time/1000) ); } 100 { $cell_date_birth = -1 }
6   rule : { 6 } { $cell_type:=6; } 1000 { (0,0) = 1 and ( ( (time/1000) -
                                                          $cell_date_birth ) > 48 ) }
7   rule : { 6 } { $cell_type:=6; } 1000 { (0,0) = 2 and ( ( (time/1000) -
                                                         $cell_date_birth )  > 48 ) }
8   rule : { 6 } {$cell_type:=6; } 1000 { (0,0) = 5 and ( ( (time/1000) -
                                                         $cell_date_birth ) > 48 ) }
9   rule : { 3 } { $cell_type:=3; } 1000 { (0,0) = 1 and
                  ( uniform(0,1) < ( ( 0.5 * (#Macro(INFECTED_R_CELLS_NEIGHBORS))/8 ) +
                                     ( 0.6 * #Macro(INFECTED_S_CELLS_NEIGHBORS))/8 ) ) }
10  rule : { 4 } { $cell_type:=4; } 1000 { (0,0) = 2 and
                  ( uniform(0,1) < ( ( 0.5 * (#Macro(INFECTED_R_CELLS_NEIGHBORS))/8) +
                  ( 0.6 * #Macro(INFECTED_S_CELLS_NEIGHBORS))/8 ) ) }
```

```
11  rule : { 2 }{ $cell_type:=2; $cell_date_birth:=round(time/1000); } 1000 { (0,0) = 1
                and ( ( stateCount(1) + stateCount(3) )/(10*10) >= 0.05 )
                and ( random < ( ( stateCount(1)+stateCount(3) - ( 10 * 10 * 0.05 ) ) /
                                 (stateCount(1)+stateCount(3)) ) ) }
12  rule : { 6 } { $cell_type:=6; } 1000 { (0,0) = 3 and
                                       ( ( ( time/1000) - $cell_date_birth ) > 6 ) }
13  rule : { 6 } { $cell_type:=6; } 1000 { (0,0) = 4 and
                                       ( ( ( time/1000) - $cell_date_birth ) > 4 ) }
14  rule : { 1 } { $cell_type:=1; } 1000 { (0,0) = 3 and ( uniform(0,1) <= 0.001 ) }
15  rule : { 2 } { $cell_type:=2; } 1000 { (0,0) = 4 and ( uniform(0,1) <= 0.001 ) }
16  rule : { 5 } { $cell_type:=5; } 1000 { ( (0,0) = 3 or (0,0) = 4 ) and
                                         ( uniform(0,1) <= 0.001 ) }
17  rule : { 1 }{ $cell_type := 1; $cell_date_birth := round(time/1000); } 1000
                                          { (0,0) = 6 and ( uniform(0,1) <=  0.99 ) }
18  rule : { 3 }{ $cell_type := 3; }  1000 { ( (0,0) = 1 and
                                           ( (time/1000) - $cell_date_birth ) = 0 and
                                           (uniform(0,1) <= 0.001 ) ) }
19  rule : { (0,0) } 100 { t }
```

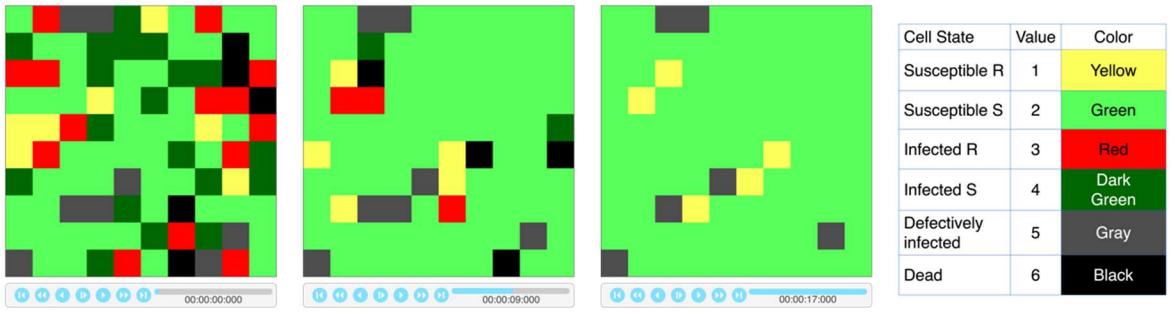
Line 1 indicates that each cell has two state variables – *cell_date_birth* and *cell_type* – to store the moment of birth and the cell type of each cell (according to the compartmental model of Figure 8). The idea behind using two state variables instead of just one and the cell value is to provide a more straightforward model that is easier to read by the user. The moment of birth permits calculating the current age of a cell for computing rules related to the lifespan of each cell. Initially, the cell and state variables values are set to -1; however, the values of each cell can also be loaded from a .val file. The first two rules are applied only at the beginning of the simulation and enable initializing cells and their state variables. The rule implemented in line 4 allows the definition of values for cells and state variables (as R cells) for those not specified in the file. The value of the cell corresponds to the same as the *cell_type* state variable. The rule of line 5 is applied to cells whose initial value is defined in the file. These two rules also allow to initialize the moment of birth of a cell randomly. Since the age of a cell must be between 0 and 48 (the lifespan of a cell), the CD++ function *uniform(0,48)* allows the definition of its date of birth, and the value is multiplied by -1 to represent that they were generated before the beginning of the simulation. Additionally, the rule also sets the value of the *cell_type* state variable the same as the cell value. Lines 6-19 in the script of HBV Model I are the actual rules of the HBV model defined in Table 1. Rule 1, defined in lines 6 and 7, allows the transition of susceptible R and S cells (respectively) to the Death state when surpassing their lifetime. For the precondition of this rule to be evaluated as true, the age of the cell must exceed its lifetime *L* (48 in the model). In CD++, simulation time is equivalent to a clock time. Thus, we consider one simulated second equivalent to a week to simplify age cell-related calculations. As such, the age of a cell is defined as the difference between its date of birth (stored in the *cell_date_birth* state variable) and the current simulation time (divided by 1000 as time granularity is milliseconds). Rule 2 is the same as Rule 1, but for defectively infected cells, that is, those in DI state transiting to state D. When this rule evaluates to true, the cell state (and its state variable *cell_type*) changes its value to 6 (representing the D state). Rule 3 is implemented as two different rules for R and S cells, in lines 9 and 10, for ease of reading. These rules describe the infection process of susceptible R and S cells (cells with values 1 or 2, respectively). The procedure of

counting the *n* infected R cells and the *m* infected S cells in the neighborhood – that, for this particular rule, consists of the nearest four cells and the following four nearest cells (as depicted in Figure 9(b)) – is defined in the macros *INFECTED_R_CELLS_NEIGHBORS* and *INFECTED_S_CELLS_NEIGHBORS* respectively. At each macro, we check the value of each cell (according to the corresponding neighborhood) using an *if* statement that returns 1 if the cell contains the value 3 (infected R cells). Then, we add the returned values to calculate the value of *n*. Analogously, but with the value 4 (infected S cells), it is made to compute the value of *m*. Rule 4, specified in line 11, maintains a given ratio of R to S cells in the lattice by randomly changing R cells into S as needed. The definition of such a rule is not trivial in a CA (or any related formalisms such as Cell-DEVS) since applying a rule to a cell considers its previous state and the previous state of its neighborhood. Still, this particular rule requires knowledge of the state of every cell in the model and, simultaneously, of its concurrent application to other cells in the current timestep; otherwise, it would turn every R into an S cell. Therefore, computing this rule (in CD++) is defined in two parts. First, the R to S cells ratio is computed by counting R cells, considering the complete lattice as the neighborhood. Then, if the number of R cells in the current state of the model exceeds the given ratio, changing an R cell to S is calculated cell-to-cell using the amount needed to achieve the ratio as a probability. The new amount of S cells will not necessarily be exact as the specified ratio, but the more cells compose the model, the closer will be the resulting ratio. For implementing this rule, we define a precondition consisting of three logical expressions applied to two AND operators. The first part (1) only checks that the cell corresponds to an R cell. The second part (2) computes the current ratio of R to S cells by counting the number of R cells (whose value is 1 or 3, using CD++ built-in function *statecount*) and dividing such amount by the number of cells (in this case, 10x10), checking if it exceeds the predefined ratio R_{rs} (5%). The third part (3) calculates the percentage of R cells that exceeds the given ratio R_{rs} of their total. Then, we generate a random number between 0 and 1 (using the *random* function provided by CD++) and compare it to be less than the percentage of exceeding R cells. If parts 1, 2 and 3 are true, the postcondition turns the R cell to an S, and the date of birth of the new S cell is reset to the current simulation time. Rule 5**,** implemented in lines 12 and 13, computes when an infected R or S cell dies after their age reaches its lifetime L_{IR} or L_{IS}, respectively. The age is computed as the difference between the current simulation time (divided by 1000 since it is expressed in milliseconds) and the date of birth (an integer). Rule 6, defined in lines 14 and 15, is specified as two separate rules for ease of reading. Allow the transition from IR to SR or IS to SS, respectively. This means that infected (R and S) cells can recover, becoming susceptible again at a rate IR_1. Thus, the precondition of the rule is defined as a random number between 0 and 1 (using the CD++ built-in *uniform* function) that must be lower than 0.001 (the value specified for the parameter IR_1) for applying the postcondition, changing the cell - and state variable *cell_type* - to SR (1) or SS (2), as appropriate. Rule 7, specified in line 16, corresponds to when an infected cell (IR or IS) becomes defectively infected (DI) at an inverse rate IR_2, losing its ability to recover or infect other cells. Therefore, the precondition of the rule is defined as computing a random number between 0 and 1 (using the uniform function provided by CD++) that must be less or equal to 0.001 (the value set for IR2) to evaluate as true in order to apply the postcondition turning the cell's value and its state variable cell_type to DI (or 5, in the model). Rule 8**,** implemented in line 17, defines the transition from D to SR, representing R cell replenishment as an effect of healthy R cell division. When a randomly generated number is less or equal to the probability P_{repl} (0.99 in the model), its precondition evaluates to true. The postcondition replaces the dead cell (whose value, and the value of its *cell_type* state variable, is 6) with a new susceptible R cell by changing its value (and the value of its *cell_type state* variable) to 1. Since it is a new cell, the value

of its state variable *cell_date_birth* is reset to the current simulation time. Finally, rule 9 – implemented in line 18 – allows the transition of newly added cells from the SR state to IR. New R cells (those whose age is zero) become infected with a probability P_{inf}. As such, its precondition checks if the age of the cell is zero and if the probability P_{inf} is met. The age is calculated as the difference of the value of its state variable *cell_date_birth* with the current simulation time equal to 0 (meaning its age is zero), and the probability is computed as a random number that must be less than P_{inf} (0.001 in the model). The postcondition changes the cell and the *cell_type* state variable values to 3 (the IR state). The default rule is implemented in line 19.

Table 1 - Model parameters defining a healthy adult [12].

Parameter	**Parameter description**	**Default value**	**Rule in which it is used**
L	Lifetime of a cell	48 weeks	1 and 7
INFECT_RATE_R	Infect rate for R cells	0.5	2
INFECT_RATE_S	Infect rate for S cells	0.6	2
R_{rs}	Overall ratio of R to S cells	0.05	3
L_{IR}	Lifetime of an infected R cell (IR)	6 weeks	4
L_{IS}	Lifetime of an infected S cell (IS)	4 weeks	4
IR_1	Inverse rate (recovery rate) for cells to become susceptible again	0.001	5
IR_2	Inverse rate (infection rate) at which cells become defectively infected	0.001	6
P_{repl}	Probability of a dead cell to transform into a new R cell	0.99	8
P_{inf}	Probability of a new R cell of becoming randomly infected	0.001	9



(a) At the beginning of the simulation. (b) After nine cycles (weeks). (c) After 17 cycles (weeks). (d) Cells simulation color legend.

Figure 10: HBV Model simulation: (a) Initially, the virus has already spread. There are 33% of infected R and S cells, another 8% of defectively infected cells, and 5% are dead cells. The rest corresponds to susceptible cells. (b) After nine weeks, the virus has diminished almost completely, with only 5% of infected cells, 7% remain defectively infected, and 4% of other cells have died. The rest corresponds to susceptible cells. (c) After 17 weeks, the virus has been eradicated and hepatocytes have recovered almost completely, also the ratio of R to S cells is near normal at this point. (d) Cells simulation color legend.

3.2.2 Results for Healthy Adults

Figure 10 displays simulation results for a scenario executed with parameters corresponding to a healthy adult (as described in Table 2). At the beginning of the simulation (see Figure 10(a)), the values of each cell (along with their *cell_type* state variable) are defined randomly, describing a state of the model where the disease has

already spread among the hepatocytes. In that state, 33% correspond to infected R and S cells, 8% are defectively infected cells, 5% are dead cells, and the rest correspond to susceptible (R and S) cells. Figure 10(b) depicts the model state after nine weeks. At this time, most infected cells from the beginning of the simulation have died (rule 5) since the lifespan of infected cells is only four weeks and have transformed into new R cells (rule 8), as the probability of cell replenishment is 99%. Few infected cells have directly recovered (rule 6) as the recovery rate (IR_1 in Table *1*) is set only at 0.1%; analogously, few new R cells (if any) have become infected (rule 9) since the probability of infection is defined as 0.1% (P_{inf} in Table 2). The virus has indeed spread (rule 3), but the process of recovery of cells has been faster than the infection. Thus, the virus remains active in only 5% of the cells. Figure 10(c) depicts the model's state after 17 cycles, showing that 94% of cells have recovered, and 5% correspond to R cells, satisfying the R to S defined ratio thanks to rule 4. Also, 6% of cells are defectively infected, and new healthy cells will replace them within a few more cycles (through the continuous execution of rules 5, 8, and 4). At this point, the hepatocytes have recovered, and the liver can be considered free of the virus since defectively infected cells cannot spread HBV to others. Figure 10(d) describes each cell color and the corresponding cell value (and *cell_type* state variable value). It is worth noting that the initial set of values for the model contained a considerable amount (33%) of already infected cells, which impacts the amount of time (cycles) needed to achieve a full recovery. Nevertheless, short-term recovery is achieved when the simulation is performed considering model parameters for a healthy adult.

3.2.3 Results for the Immunity-Deprived

It is known that HBV infection can become a long-term disease for some patients. Approximately 3% of adults and 90% of children (up to 6 years old) who acquire the virus will develop a chronic infection [42, 43]. Model parameters most related to immunity issues correspond to infection rates and lifespan of infected cells. In persons with less immunity, infection rates are expected to be larger. The same applies to the lifespan of infected cells since the immune system will take longer to destroy those cells. As such, without performing any modifications to the model and, according to the model definition and its parameters, we adjusted the values for the rates of infection and the lifespan of infected cells, simulating the model independently.

Figures 11 and 12 show simulation results for two scenarios: i) Increased infectiousness by considering higher infection rates, and ii) Increased resistance by considering a larger lifespan for infected cells. The same initial values for the cells (obtained from the scenario for healthy adults) are used for both scenarios. Parameter values for each case are described in Table 2. In both scenarios, it can be observed that the number of infected and susceptible cells oscillates (as one increases, the other decreases) in a cyclic fashion. It can also be noticed that the liver immune system cannot eliminate the virus, which persists (at least) during 110 and 162 weeks, respectively, indicating that HBV has become a chronic disease, as indicated by health organizations and the literature [42, 43]. Setting smaller parameter values than those in Table 2 are not sufficient to describe a chronic infection, however, may take a few more cycles for a complete recovery.

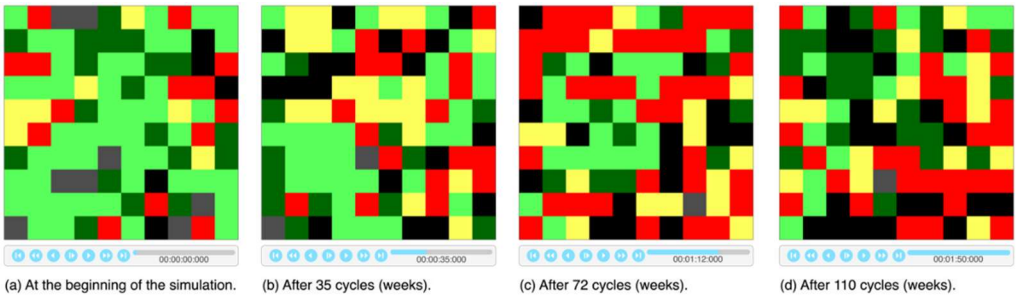


Figure 11: HBV scenario with increased infectiousness (higher infection rate): (a) Simulation starts with the state of the scenario for healthy adults. (b) After 35 cycles, infected cells remain approximately the same (30%), while susceptible cells have slightly dropped from 54% to 50%. (c) After 72 cycles, infected cells have risen again to 58%, and susceptible cells have reduced to 25%. (d) After 110 cycles, infected cells remain, but reducing to 49% while 29% remaining susceptible.

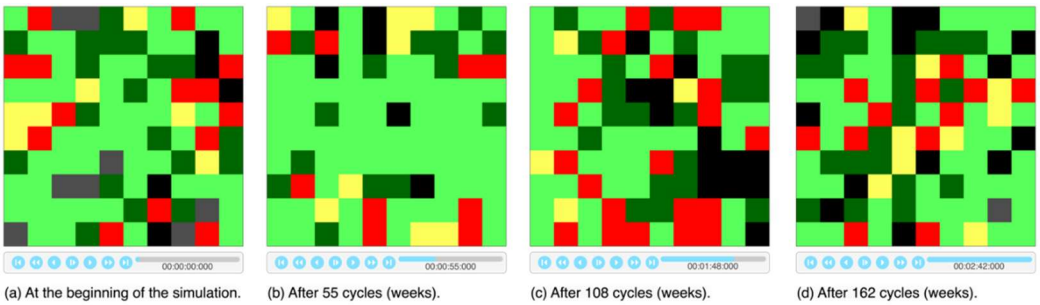


Figure 12: HBV scenario with increased resistance to human immunity (higher lifespan of infected cells): (a) Simulation starts with the state as for the scenario for healthy adults. (b) The number of infected cells has reduced to 23% while susceptible cells increased to 71%. (c) 40% infected 57% susceptible (d) 32% infected 53% susceptible.

Table 2 - Model parameters for the definition of the immunity-deprived.

Case	**Parameters changed from base case**	**Parameter description**	**New value**	**Rule in which it is used**
Increased infectiousness	*INFECT_RATE_R*	Infect rate for R cells	0.8	2
	INFECT_RATE_S	The infect rate for S cells	0.85	2
Increased resistance	L_{IR}	Lifetime of an infected R cell (IR)	7 weeks	4
	L_{IS}	Lifetime of an infected S cell (IS)	5 weeks	4

3.2.4 Discussion

In this section, we have described how Cell-DEVS can be used to model the spread of HBV among hepatocytes. Implementing the model was challenging compared to others due to a single rule needing knowledge of every cell's state, implying a rather unorthodox neighborhood definition. Nevertheless, the model is flexible enough to be extended for adding more compartments and transitions, as domain experts may require. The model can be used to explore different scenarios with ease (as defined in the previous section) simply by adjusting its parameters, allowing modelers to predict – for example – the behavior of a strain of the virus more resistant to the immune system or to consider smaller contagion rates supported by vaccination. In this study, we have

simulated three scenarios, all with results agreeing with the literature. The first corresponds to a scenario with parameters set for a healthy adult. Results show that a complete recovery is achieved within a few weeks. For the other two scenarios, we chose two sets of parameters describing a person that has difficulties recovering from HBV. One set of the modified parameters increased the values for the infection rates of R and S cells, describing a scenario of acute virus contagion. The other set extended their lifespans, representing a scenario where the immune system takes longer to kill infected hepatocytes. Both scenarios showed that the virus continued infecting hepatocytes for more than 110 and 162 weeks, respectively, where the first resulted in a more extended virus infection than the latter. Nevertheless, results of both scenarios suggest the development of chronic hepatitis. The Cell-DEVS model is available in [48] and videos corresponding to the three simulated scenarios are available on the ARSLab YouTube channel [49].

4 CONCLUSION

In the recent few years, the world has faced several outbreaks of diseases at different geographical locations and on various scales. The research community as well as the public experienced the need and usefulness of different M&S methodologies to understand the spread of such diseases and predict their behavior. Decision-makers have been using simulation models to measure the consequences of applying control and safety measures (e.g., vaccinations, population confinement and social distancing) during epidemics and pandemics. Some simulation models are more trustworthy than others. Motivated by the need to increase the trustworthiness of simulation models, we propose Cell-DEVS as a reliable methodology to model and simulate disease spread.

We present two case studies to illustrate the applicability of Cell-DEVS for modeling and simulating epidemics and pandemics. In each case, we present a basic model and apply different variations to that model to show the effect of various factors (e.g., vaccination, temperature, and immunity) on the spread of the disease. We practically illustrate how one can create models at different levels of complexity to incorporate more variables depending on the nature of the disease. We use the available Cell-DEVS throughout the M&S process. By this, we prove that for the following reasons: (1) Being based on CA, makes it inherently suitable for creating spatial models of geographical areas and other spaces, (2) It avoids many of the unrealistic assumptions (e.g., homogeneity of the population) taken by other M&S methodologies, (3) It allows for asynchronous execution over continuous time which results in more accurate simulations, and (4) it has a full suite of supporting tool that provides transparency for understanding the behavior of the model and visualizing the simulation results.

ACKNOWLEDGMENTS

This work has been partially supported by NSERC, Canada and Fondecyt de Iniciación 11230961 from ANID, Chile.

REFERENCES

[1] Huadong Du, Lauren Gardner, and Ensheng Dong. 2020. An interactive web-based dashboard to track COVID-19 in real-time. Infect. Dis. (The Lancet) 20, 5 (M ay 2020), 533-534. https://doi.org/10.1016/S1473-3099(20)30120-1

[2] Max Roser. 2020. The Spanish flu: The global impact of the largest influenza pandemic in history. Retrieved October 2023 from https://ourworldindata.org/spanish-flu-largest-influenza-pandemic-in-history

[3] Vanessa Haye. 2023. Hepatitis B mortality rate: Stats and more. Medical News Today. Retrieved October 2023 from https://www.medicalnewstoday.com/articles/mortality-rate-of-hepatitis-b

[4] Julien Arino, Chris Bauch, Fred Brauer, S. Michelle Driedger, Amy L. Greer, S.M. Moghadas, Nick J. Pizzi, Beate Sander, Ashleigh Tuite, P.

van den Driessche, James Watmough, Jianhong Wu, and Ping Yan. 2011. Pandemic influenza: Modelling and public health perspectives. Math. Biosci. Eng. 8, 1 (2011), 1-20. https://doi.org/10.3934/mbe.2011.8.1

[5] Julien Arino, Fred Brauer, P. van den Driessche, James Watmough, and Jianhong Wu. 2008. A model for influenza with vaccination and antiviral treatment. J. Theor. Biol. 253, 1 (July 2008), 118-130. https://doi.org/10.1016/j.jtbi.2008.02.026

[6] Amit Huppert, Guy Katriel. 2013. Mathematical modelling and prediction in infectious disease epidemiology. Clin. Microbiol. Infect. 19, 11 (2013), 999-1005. https://doi.org/10.1111/1469-0691.12308

[7] Liliana Perez and Suzana Dragicevic. 2009. An Agent-Based Approach for Modeling Dynamics of Contagious Disease Spread. Int. J. Health Geogr 8, 50. https://doi.org/10.1186/1476-072X-8-50

[8] Marcello Marini, Cyril Brunner, Ndaona Chokani, and Reza S. Abhari. 2020. Enhancing Response Preparedness to Influenza Epidemics: Agent-Based Study of 2050 Influenza Season in Switzerland. Simul. Model. Pract. Theory 103, 102091. https://doi.org/10.1016/j.simpat.2020.102091

[9] Erik Cuevas. 2020. An agent-based model to evaluate the COVID-19 transmission risks in facilities. Comput. Biol. Med. 121 (May 2020), 103827. https://doi.org/10.1016/j.compbiomed.2020.103827

[10] William Ogilvy Kermack and A. G. McKendrick. 1927. A contribution to the mathematical theory of epidemics. Proceedings of the Royal Society of London. Series A, 115: 700-721. http://doi.org/10.1098/rspa.1927.0118

[11] Elizabeth Hunter, Brian Mac Namee, and John D. Kelleher. 2018. A Comparison of Agent-Based Models and Equation Based Models for Infectious Disease Epidemiology. In Proceedings of the 26th AIAI Irish Conference on Artificial Intelligence and Cognitive Science (AICS '18). DOI: https://doi.org/10.21427/rtq2-hs52

[12] Michael Small and Chi K Tse. 2005. Small World And Scale Free Model Of Transmission Of Sars. Int. J. Bifurc. Chaos 15, 5 (2005), 1745-1755. https://doi.org/10.1142/S0218127405012776

[13] Joan L Aron and Ira B Schwartz. 1984. Seasonality and period-doubling bifurcations in an epidemic model. J. Theor. Biol. 110, 4 (1984), 665-679. https://doi.org/10.1016/0022-5193(84)90131-8

[14] Maria Jardim Beira, Anant Kumar, L Perfeito, J Gonçalves-Sá, Pedro Sebastião. 2020. A data-driven epidemiological model to explain the Covid-19 pandemic in multiple countries and help in choosing mitigation strategies. medRxiv (2020), https://doi.org/10.1101/2020.08.15.20175588

[15] Wuyue Yang, Dongyan Zhang, Liangrong Peng, Changjing Zhuge, and Liu Hong. 2021. Rational evaluation of various epidemic models based on the COVID-19 data of China. Epidemics 37: 100501. https://doi.org/10.1016/j.epidem.2021.100501

[16] Hai Yan Wang and Nao Yamamoto. 2020. Using a partial differential equation with Google Mobility data to predict COVID-19 in Arizona. Math. Biosci. Eng. 17, 5 (July 2020), 4891-4904. https://doi.org/10.3934/mbe.2020266

[17] S. Hoya White, Ángel Martín del Rey, Gerardo Rodríguez Sánchez. 2007. Modeling epidemics using cellular automata. Appl. Math. Comput. 186, 1 (2007), 193-202. https://doi.org/10.1016/j.amc.2006.06.126

[18] M.A. Fuentes and Marcelo N. Kuperman. 1999. Cellular automata and epidemiological models with spatial dependence. Physica A: Stat. Mech. Appl. 267, 3–4 (1999), 471-486. https://doi.org/10.1016/S0378-4371(99)00027-8

[19] Behrouz Afshar-Nadjafi, Seyed Taghi Akhavan Niaki. 2021. Seesaw Scenarios of Lockdown for COVID-19 Pandemic: Simulation and Failure Analysis. Sustain. Cities Soc. 73:103108. https://doi.org/10.1016/j.scs.2021.103108.

[20] Chaolin Huang, Yeming Wang, Xingwang Li, Lili Ren, Jianping Zhao, Yi Hu, Li Zhang, Guohui Fan, Jiuyang Xu, Xiaoying Gu, Zhenshun Cheng, Ting Yu, Jiaan Xia, Yuan Wei, Wenjuan Wu, Xuelei Xie, Wen Yin, Hui Li, Min Liu, Yan Xiao, Hong Gao, Li Guo, Jungang Xie, Guangfa Wang, Rongmeng Jiang, Zhancheng Gao, Qi Jin, Jianwei Wang and Bin Cao. 2020. Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. The Lancet 395 (10223), 497-506. https://doi.org/10.1016/S0140-6736(20)30183-51

[21] Jindong Dai, Chi Zhai, Jiali Ai, Jiaying Ma, Jingde Wang, and Wei Sun. 2021. Modeling the Spread of Epidemics Based on Cellular Automata. Processes 9, 1 (December 2021), 55. https://doi.org/10.3390/pr9010055

[22] Sayantari Ghosh and Saumik Bhattacharya. 2021. Computational Model on COVID-19 Pandemic Using Probabilistic Cellular Automata. SN Comput. Sci. 2, 230 (April 2021). https://doi.org/10.1007/s42979-021-00619-3

[23] Gabriel A. Wainer and Norbert Giambiasi. 2005. Cell-DEVS/GDEVS for Complex Continuous Systems. Simulation 81, 2 (February 2005). https://doi.org/10.1177/003754970505223

[24] Román Cárdenas and Gabriel Wainer. 2022. Asymmetric Cell-DEVS models with the Cadmium simulator. Simul. Model. Pract. Theory 121, 102649 (2022). https://doi.org/10.1016/j.simpat.2022.102649

[25] Román Cárdenas, Alonso Inostrosa-Psijas, and Gabriel Wainer. 2021. A Modeling and Simulation Platform for Space-Based Compartmental Modeling of Pandemic Spread. In 2021 Annual Modeling and Simulation Conference (ANNSIM). IEEE, Fairfax, VA, USA. https://doi.org/10.23919/ANNSIM52504.2021.9552046

[26] Román Cárdenas, Kevin Henares, Cristina Ruiz-Martín, and Gabriel Wainer. 2022. Cell-DEVS Models for the Spread of COVID-19. LNTCS 2021, 12599. https://doi.org/10.1007/978-3-030-69480-7_24

[27] Taghreed Altamimi, Hoda Khalil, Vinu Subashini Rajus, Ryan Carriere, and Gabriel A. Wainer. 2021. Cell-DEVS models with BIM integration for airborne transmission of COVID-19 indoors

[28] Zein Hajj-Ali and Gabriel Wainer. 2021. Spatial Models and Masks in Indoor Analysis for the Spread of COVID-19. In Proceedings of the 2021 Winter Simulation Conference (WSC), Phoenix, AZ, USA, pp. 1-12. https://doi.org/10.1109/WSC52266.2021.9715334

[29] Muhammad Taslim, Arsunan Arsin, Hasnah Ishak, Sitti Nasir, and Andi Nilawati Usman. 2018. Diversity of Dengue Virus Serotype in

Endemic Region of South Sulawesi Province. J. Trop. Med. 2018, 9682784 (April 2018). https://doi.org/10.1155/2018/9682784

[30] WHO. 2023. Dengue and severe dengue. Retrieved on October 29, 2023 from https://www.who.int/news-room/fact-sheets/detail/dengue-and-severe-dengue

[31] Sawyer H Pouliot 1, Xu Xiong, Emily Harville, Valerie Paz-Soldan, Kay M Tomashek, Gerard Breart, and Pierre Buekens. 2010. Maternal Dengue and Pregnancy Outcomes: a Systematic Review. Obstet Gynecol Surv. 65, 2 (February 2010), 107-118. https://pubmed.ncbi.nlm.nih.gov/20100360/

[32] Henrique F. Gagliardi, Fabrício A. B. da Silva, and Domingos Alves. 2006. Automata network simulator applied to the epidemiology of urban dengue fever. In Proceedings of the 6th international conference on Computational Science - Volume Part III (ICCS'06). Springer-Verlag, Berlin, Heidelberg, 297–304. https://doi.org/10.1007/11758532_41

[33] Bruno St-Aubin, Eli Yammine, Majed Nayef, and Gabriel Wainer. 2019. Analytics and visualization of spatial models as a service. In Proceedings of the 2019 Summer Simulation Conference (SummerSim '19). Society for Computer Simulation International, San Diego, CA, USA, 39, 1–12. https://doi.org/10.5555/3374138.3374177

[34] Chiril Chidisiuc and Gabriel A. Wainer. 2007. CD++Builder: an eclipse-based IDE for DEVS modeling. In Proceedings of the 2007 spring simulation multiconference - Volume 2 (SpringSim '07). Society for Computer Simulation International, San Diego, CA, USA, 235–240.

[35] Liu Z, Zhang Z, Lai Z, Zhou T, Jia Z, Gu J, Wu K, Chen XG. 2017. Temperature Increase Enhances Aedes albopictus Competence to Transmit Dengue Virus. Front Microbiol. 2017. Front Microbiol. 8, 2337 (December 2017). https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5717519/

[36] Shen JC, Luo L, Li L, Jing QL, Ou CQ, Yang ZC, and Chen XG. 2015. The Impacts of Mosquito Density and Meteorological Factors on Dengue Fever Epidemics in Guangzhou, China, 2006-2014: a Time-series Analysis. Biomed Environ Sci. 28, 4 (May 2015), 321-9. https://doi.org/10.3967/bes2015.046. PMID: 26055559.

[37] Chiril Chidisiuc and Gabriel A. Wainer. 2007. CD++Builder: an eclipse-based IDE for DEVS modeling. In Proceedings of the 2007 spring simulation multiconference - Volume 2 (SpringSim '07). Society for Computer Simulation International, San Diego, CA, USA, 235–240.

[38] CDC. 2021. Serotypes and the Importance of Serotyping Salmonella. Retrieved on October 29, 2023 from https://www.cdc.gov/salmonella/reportspubs/salmonella-atlas/serotyping-importance.html

[39] Hoda Khalil and Gabriel Wainer. 2021. Cell-DEVS Dengue Models. GitHub. (21 September 2021). Retrieved September 21, 2021 from https://github.com/SimulationEverywhere-Models/CDPP-DengueModels

[40] Hoda Khalil and Gabriel Wainer. 2021. Cell-DEVS for Modeling Epidemics: Dengue Example. Video. (21 September 2021). Retrieved September 21, 2021 from https://www.youtube.com/watch?v=m6wvYKiw8P0

[41] Xuan Xiao, Shi-Huang Shao, and Kuo-Chen Chou. 2006. A probability cellular automaton model for hepatitis B viral infections. Biochem. Biophys. Res. Commun. 342, 2 (April 2006), 605–610. DOI: https://doi.org/10.1016/j.bbrc.2006.01.166

[42] CDC 2023. Hepatitis B. Retrieved October 13, 2023 from https://www.cdc.gov/hepatitis/hbv/index.htm

[43] WHO 2023. Hepatitis B Fact Sheets. Retrieved October 13, 2023 from https://www.who.int/news-room/fact-sheets/detail/hepatitis-b

[44] Yao-Chun Hsu, Daniel Q. Huang, and Mindie H. Nguyen. 2023. Global burden of hepatitis B virus: current status, missed opportunities and a call for action. Nat. Rev. Gastroenterol. Hepatol. 20 (April 2023), 524–537. DOI: https://doi.org/10.1038/s41575-023-00760-9

[45] Marco Picardi, Fabrizio Pane, Concetta Quintarelli, Amalia De Renzo, Annalisa Del Giudice, Bianca De Divitiis, Marcello Persico, Rosanna Ciancia, Francesco Salvatore, Bruno Rotoli. 2003. Hepatitis B virus reactivation after fludarabine-based regimens for indolent non-Hodgkin's lymphomas: high prevalence of acquired viral genomic mutations. Haematologica 88, 11 (2003), 1296–1303.

[46] Robert J. H. Payne, Martin A. Nowak, and Baruch S. Blumberg. 1996. The Dynamics of Hepatitis B Virus Infection. Proc. Natl. Acad. Sci. U.S.A. 93, 13 (1996), 6542–46. http://www.jstor.org/stable/39343

[47] W. Thomas London and Baruch S. Blumberg. 2000. A Cellular Model of the Role of Hepatitis B Virus in the Pathogenesis of Primary Hepatocellular Carcinoma. World Sci. Ser. 20th Cent. Biol., pp. 332-336 (2000). https://doi.org/10.1142/9789812813688_0030

[48] Alonso Inostrosa-Psijas and Gabriel Wainer. 2023. CDPP-HepatitisB-Virus. GitHub. (24 October 2023). Retrieved October 24, 2023 from https://github.com/SimulationEverywhere-Models/CDPP-HepatitisB-Virus

[49] Alonso Inostrosa-Psijas and Gabriel Wainer. 2023. Modeling Epidemiology Using Cell-DEVS: HBV. Video. (24 October 2023). Retrieved October 24, 2023 from https://www.youtube.com/watch?v=WGy8BO4dbs8