Proceedings of the 2021 Winter Simulation Conference
S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo, and M. Loper, eds.

A DISCRETE SIMULATION OPTIMIZATION APPROACH TOWARDS CALIBRATION OF AN AGENT-BASED SIMULATION MODEL OF HEPATITIS C VIRUS TRANSMISSION

Soham Das
Varun Ramamohan

Department of Mechanical Engineering
Indian Institute of Technology Delhi
Hauz Khas
New Delhi, 110016, INDIA

Navonil Mustafee

Centre for Simulation, Analytics and Modelling
University of Exeter
Rennes Drive
Exeter, EX4 4ST, UK

ABSTRACT

This study demonstrates the implementation of the stochastic ruler discrete simulation optimization method for calibrating an agent-based model (ABM) developed to simulate hepatitis C virus (HCV) transmission. The ABM simulates HCV transmission between agents interacting in multiple environments relevant for HCV transmission in the Indian context. Key outcomes of the ABM are HCV and injecting drug user (IDU) prevalences among the simulated cohort. Certain input parameters of the ABM need to be calibrated so that simulation outcomes attain values as close as possible to real-world HCV and IDU prevalences. We conceptualize the calibration process as a discrete simulation optimization problem by discretizing the calibration parameter ranges, defining an appropriate objective function, and then applying the stochastic ruler random search method to solve this problem. We also present a method that exploits the monotonic relationship between the simulation outcomes and calibration parameters to yield improved calibration solutions with lesser computational effort.

1 INTRODUCTION AND LITERATURE REVIEW

For many complex simulation models, the data required to estimate every input parameter of the simulation model is often not available. In this situation, one may estimate these input parameters via 'calibration', wherein the input parameter values are set such that the relevant simulation outputs approach as close to observed or known values of the outputs as possible (Law 2015). We refer to the input parameters that are estimated as 'calibration' parameters, and the known values of the simulation outcomes - which are estimated either via data collection from the field, from the published literature, or from domain experts - as 'calibration targets'. In this paper, we demonstrate a discrete simulation optimization approach - specifically the use of the stochastic ruler random search method - towards calibration of a simulation model. The simulation model in question is an agent-based simulation model of the transmission of the hepatitis C virus (HCV) that we have previously developed for the Indian context (Das et al. 2019).

In recent times, the use of simulation optimization methods for simulation model calibration has increased. Genetic algorithms, evolutionary optimization, and their various adaptations have been commonly used for both discrete and continuous calibration parameter search spaces (Voloshin et al. 2015). Stochastic approximation methods such as simultaneous perturbation stochastic approximation have also been used for both discrete and continuous search spaces (Hale et al. 2015). With regard to calibration of agent-based simulaton models, genetic algorithms and evolutionary optimization techniques for calibration of agent-based simulation models have been used in the studies of Fabretti (2013) and Moya et al. (2021), respectively. Fabretti (2013) use a genetic algorithm and an adaptation of the Nelder-Mead simplex algorithm for calibration of an agent-based model for financial markets. Other than these techniques, Johnson et al.

978-1-6654-3311-2/21/$31.00 ©2021 IEEE

(2009) use a Latin hypercube design for the calibration parameter search space for the calibration of an agent-based simulation model developed to model interactions within a refugee camp. We refer readers to Pietzsch et al. (2020) for a detailed review of metamodels used in calibration of agent-based simulation models.

We now discuss the calibration of simulation models in the healthcare space. Kong et al. (2009) employed genetic algorithm and simulated annealing for their calibration problem. Taylor et al. (2010) and Karnon and Vanni (2011) compare random search methods against the use of other simulation optimization techniques such as manual search and Nelder-Mead simplex methods, and gradient-based search. Bicher et al. (2017) do not use simulation optimization to calibrate their agent based simulation of rehospitalization of psychiatric patients - instead, they use the law of the iterated logarithm. Example techniques used to calibrate infectious disease transmission models include the grid search method (Luo et al. 2018), Latin hypercube modelling (Shrestha et al. 2017), and the genetic algorithm (Reiker and Penny 2021). Hazelbag et al. (2020) provides a useful reference for studies that document the use of optimization techniques to calibrate agent-based disease transmission simulation models.

More generally, our search of the literature did not yield studies wherein conventional discrete simulation optimization methods such as the stochastic ruler method (Yan and Mukai 1992), probabilistic branch and bound method (Norkin et al. 1998) and COMPASS (Hong and Nelson 2006) have been used for calibration of either discrete-event or agent-based simulations - in the healthcare area or in other application areas. In this study, we demonstrate the conceptualization of the calibration process as a simulation optimization process that can be solved via the stochastic ruler random search method.

The stochastic ruler method is one of the first discrete simulation optimization methods with provable asymptotic convergence to the global optimum, and is relatively straightforward to implement. Modifications of the stochastic ruler method have been developed to improve the convergence of the method (Alrefaei and Andradóttir 2005); however, we implement the original version by Yan and Mukai (1992) as an initial proof-of-concept of this approach towards calibrating a simulation model. Further, we also demonstrate a method for using information regarding the relationship between the calibration parameters and the calibration targets to reduce the size of the search space. Applying the stochastic ruler method on this truncated calibration parameter search can yield, as we demonstrate, improved solutions while incurring a potentially lower computational cost in comparison with applying the stochastic ruler method on the original solution space. Our search of the literature did not yield another study that truncated the solution space in this manner for discrete simulation optimization problems.

We now describe the agent-based simulation of HCV transmission that we use to demonstrate our model calibration approach.

2 THE HCV TRANSMISSION MODEL

The agent-based model of HCV transmission that we develop is comprehensive, in that we incorporate all key modes of transmission of HCV in the Indian context. In India, HCV is a significant public health concern in the state of Punjab, where the prevalence of the disease is substantially higher than the average prevalence in India (3.6% in Punjab versus 1% in India) (Sood et al. 2018). Because of this, and the consequent fact that data regarding HCV transmission modes (e.g., via injecting drug use) and epidemiology for Punjab is available more widely than for other states, we select Punjab as the geographical area of interest for the transmission model (Chakravarti et al. 2013; Ambekar and Tripathi. 2008).

The key modes of transmission of HCV spread in India are medical procedures (blood transfusions, surgeries, injections and dental procedures), injecting drug use, and to a lesser extent, tattooing (Chakravarti et al. 2013). In our agent-based simulation, we create representative environments for disease transmission through each of these modes. For modeling spread of HCV through medical procedures, we create a medical environment. A social interaction environment models the transmission of HCV through injecting drug use. For this, there are two processes included within the environment- conversion of non-injecting-drug-users (non-IDUs) into IDUs, and transmission of HCV due to sharing of needles between IDUs. Nearly 75%

of IDUs are in the age group of 18-29 years (Ambekar and Tripathi. 2008), and a large proportion of these are in the age group of 18 and 24 years. In Punjab, 20% of people in the age group 18-24 years avail of higher education (Ministry of Home Affairs 2016). Thus, to model spread among young people availing higher education, we incorporate a higher education environment. Though sexual transmission contributes very less to HCV transmissions (Chakravarti et al. 2013), we nevertheless incorporate this mode so that the simulation model can later be adapted for hepatitis B virus transmission or to study HCV/HIV comorbidities. Our ABM is the first simulation model to explicitly model the mechanisms or sub-processes of transmission of HCV through all of the above modes.

The calibration targets for this simulation model are key prevalence outcomes of the simulation - HCV antibody, HCV RNA and IDU prevalence values - for which reliable estimates are available in the literature. The HCV antibody and HCV RNA observed values (which we collectively call HCV prevalence values) were taken from the study of Sood et al. (2018). This study documented a large cross-sectional epidemiological survey conducted in Punjab in 2014 to determine the prevalence of HCV. The IDU prevalence was estimated from the study of Ambekar and Tripathi. (2008). The calibration process involves running the simulation for 50 years of simulation time, with daily time steps. This calibration or burn-in period was chosen as it yielded rates of increase of HCV prevalence values which were deemed suitable by our collaborating clinical expert (Das et al. 2019).

All agents in the models are placed into groups (a proxy for a 'family'), where each group consists of an older pair (aged 48 years and above), young pairs (aged between 23-48 years), and children and young adults (below the age of 23 years). Note that while disease transmission through sexual interaction between pairs is included in the model, this mode is of limited interest from the calibration point of view. This is because, given the very low per-event probability of transmission of HCV (Osmond et al. 1993), this mode of transmission has a significantly lower contribution to the prevalence of HCV (Chakravarti et al. 2013; Das et al. 2019) when compared to other transmission modes. Further, the parameter of interest - the per-event probability of transmission via this mode - is also reliably estimated.

We now briefly describe the specifics of HCV transmission in each environment.

2.1 HCV Transmission in the Medical Environment

An agent in the simulation can visit the medical environment for a blood transfusion, a surgery, a dental procedure or to receive an injection, per the key modes of transmission documented in Chakravarti et al. (2013). Transmission in this environment occurs as follows.

a. Any agent in the model can visit the medical environment on a given day with probability p_1 calculated using (1).

$$p_1 = \frac{N_{inj} + N_{bt} + N_{sur} + N_{dp}}{360} \tag{1}$$

N_{inj}, N_{bt}, N_{sur} and N_{dp} represent the average number of injections, average number of blood transfusions, average number of surgeries and average number of dental procedures, respectively, that an Indian person undergoes on an annual basis (that is, in 360 days).

b. At the medical environment, the number of medical professionals is estimated to be 40, based on the fact that there is approximately one doctor per 1,800 population in India (1:1800) (Das et al. 2019). The average number of agents considered across our simulation time horizon is 75,000.

c. Based on a survey of dental clinics of India (details in Das et al. (2019)), we assume that 50% of medical professionals do not implement medically acceptable decontamination protocols in their workspaces. Hence, we randomly assign 20 medical professionals out of 40 as those who work in 'contaminated' environments.

d. If an infected agent visits a medical professional working in a 'contaminated' environment, then every uninfected agent visiting the professional after the infected agent has a probability p_2 of getting infected. The value of p_2 is found using (2).

$$p_2 = \frac{N_{inj} \times p_{inj} + N_{bt} \times p_{bt} + N_{sur} \times p_{sur} + N_{dp} \times p_{dp}}{N_{inj} + N_{bt} + N_{sur} + N_{dp}} \tag{2}$$

Here, p_{inj}, p_{bt}, p_{sur} and p_{dp} are the per-event probabilities of getting infected through each of the four modes of transmission within this environment.

While we were able to obtain reliable estimates of p_{inj}, p_{bt}, and p_{sur}, we were unable to do so for p_{dp}. Thus, p_2 in effect becomes a calibration parameter, even though we can obtain a reasonable initial estimate of p_2 for the calibration process by assuming p_{dp} to take a value between that of p_{inj} and p_{sur}. We make this assumption because, given the nature of dental procedures, it is likely that the transmission risk is likely to be greater than that from a needle-stick injury that occurs during an injection, and likely to be lesser the transmission risk from a significantly more invasive procedure such as a surgery. This reasoning was validated by our collaborating clinical expert, as documented in Das et al. (2019).

2.2 HCV Transmission in the Social Interaction Environment

Two types of interactions occur in this environment: interactions between IDUs, and interactions between IDUs and non-IDUs. These are described below.

a. We calculate the probability of an IDU visiting the social interaction environment, p_3, using (3) below.

$$p_3 = \frac{\sum\limits_{districts} N_{district} \times f_{district}}{\sum\limits_{districts} N_{district}} \tag{3}$$

Ambekar and Tripathi. (2008) studied IDU characteristics in certain districts of Punjab, and reported the weekly frequency of injecting drugs for IDUs in these districts. $N_{district}$ refers to the population of the districts of Punjab studied in Ambekar and Tripathi. (2008), and $f_{district}$ refers to the weekly frequency of injecting drugs for IDUs in these districts.

b. The daily probability of a non-IDU going to the social interaction environment was assumed to be $\frac{1}{7}$ (i.e., once a week).

c. Each group in the model is assigned to one of three geographical clusters. In the social interaction environment, agents in the same cluster interact with each other. Based on IDU demographic data from Ambekar and Tripathi. (2008), we impose the condition that only agents between the ages of 18 and 32 years can be IDUs, and the maximum duration for which an agent engages in injecting drug use is 3 years.

d. An interaction between a non-IDU and an IDU can lead to a non-IDU becoming an IDU with probability p_{inf}. redAs documented in Ambekar and Tripathi. (2008), the proportion unemployed among persons aged 18-29 years differs significantly between IDUs and non-IDUs (p_{ue}^{IDU} and p_{ue}^{gen} in the equations below). Incorporating this factor may allow simulation-based design of interventions for IDUs that take their employment status into consideration. Therefore, as given in (4) below, we decided to calculate p_{inf} as a weighted average of the probabilities of influence for employed (p_{inf}^{e}) and unemployed persons (p_{inf}^{ue}) respectively.

$$p_{inf}^{ue} \times p_{ue}^{gen} \; + \; p_{inf}^{e} \times (1 - p_{ue}^{gen}) = p_{inf} \tag{4}$$

As discussed in Das et al. (2019), we assume that p_{inf}^{ue} and p_{inf}^{e} are related to each other per the ratio of the proportions of unemployed persons among IDUs and the general population respectively, as expressed below in (5).

$$\frac{p_{inf}^{ue}}{p_{inf}^{e}} = \frac{p_{ue}^{IDU}}{p_{ue}^{gen}} \tag{5}$$

We could not find literature that reported estimates for p_{inf}, p_{inf}^{ue}, and p_{inf}^{e}. Therefore, we considered p_{inf} to be our second calibration parameter, and calculated p_{inf}^{ue} and p_{inf}^{e} from (4) and (5) once the value of p_{inf} is estimated via the calibration process.

e. Azim et al. (2008) reported that IDUs interact in groups of 1-2.8 persons. We took the group size of interaction for IDUs to be 3 because we do not explicitly model HCV transmission through tattooing,

which is a factor for HCV transmission in the region concerning our study (Chakravarti et al. 2013). Given that HCV transmission through tattooing is also via infected needles, similar to HCV transmission via injecting drug use, we assumed an IDU injecting group size of 3 (approximately the upper limit of the reported range) to implicitly include the effects of tattooing on HCV transmission.

f. Ambekar and Tripathi. (2008) also provided data regarding the proportion of IDUs that reported sharing needles at least once for the surveyed districts. We denote this as $s_{district}$ for a given district. Thus, we found the proportion of IDUs who reported sharing needles at least using the weighted average of the $s_{district}$ values for all surveyed districts, weighted using the population of each district. We obtained this value as 50.4%. Thus, the probability of sharing needles during a given injecting drug use event is likely to be less than this value. This yielded a reasonable initial estimate for estimating the per-injecting-event needle sharing probability, which is the third calibration parameter. Note that if there is one infected agent in a network of IDUs engaging in needle sharing, then the per-event infection probability through injections determines whether an uninfected agent gets infected.

2.3 HCV Transmission in the Education Environment

This environment was included in the simulation to incorporate IDU-based interactions regardless of geographical considerations (interactions in the social interaction environment are assumed to occur only between agents in the same cluster), and to also facilitate future research on the effects of an awareness campaign conducted in educational environments on HCV epidemiology. This environment also incorporates the processes of conversion of non-IDUs to IDUs and of uninfected IDUs into infected IDUs. Although we remove geography-based restrictions on HCV transmission in this environment, the contribution of this environment to HCV prevalences is found to be very low as only 20% of agents between the ages of 18 and 24 years avail of higher education in Punjab (Ministry of Home Affairs 2016).

3 MODEL CALIBRATION AND THE STOCHASTIC RULER METHOD

We now describe the conceptualization of the calibration process as a simulation optimization problem. We conducted preliminary experimentation to determine the calibration parameter search space. To determine the search space for the per-event infection probability in the medical environment, we considered parameter estimates between the estimate of per-event infection probability through injections and per-event infection probability through surgeries. Similarly, adjustments were made to the per-event influence probability so that the IDU prevalence moves towards its calibration target.

We define the calibration problem in a general sense first, and then describe its application to our case. Let m be the number of calibration variables and n be the number of outcome variables. We define x, the m-tuple of calibration parameters as $x = (x_1, x_2, ..., x_m)$. The n-tuple of simulation outcomes to be calibrated to the calibration targets is defined as $y = (y_1, y_2, ..., y_n)$. Note that each $y_i = f_i(x)$, $i = 1$ to n, where f_i represents the relationship between the i^{th} simulation outcome and the calibration parameters x, implicitly given by the simulation. Correspondingly, we define the n-tuple of the calibration targets as $y^0 = (y_1^0, y_2^0, ..., y_n^0)$.

A discrete simulation optimization problem typically takes the form:

$$\min_{x \in \mathbb{S}} E[g(x)] \tag{6}$$

Here $g(x)$ is the output of a single replication of the simulation, x are the decision variables, and $\mathbb{S}$ is the discrete solution space. In our case, we construct the following function that measures the sum of the absolute values of the distances between the simulation outcomes y and the calibration targets y^0.

$$h(y) = \sum_{i=1}^{n} \left| 1 - \frac{y_i}{y_i^0} \right| \tag{7}$$

$h(y)$ is a function of the random variable y, and given that each of the $y_i = f_i(x)$ ($i = 1$ to n), $h(y)$ in turn becomes a function of the calibration parameters x - that is, $h(y) = h(y_1 = f_1(x), y_2 = f_2(x), \ldots, y_n = f_n(x))$. A single replicate output of the simulation, which in (6) is represented by $g(x)$, in our case is given by the median of, say, k values of $h(y)$. We denote the median of the k values of $h(y)$ as $\hat{h}(y)$. We choose to define $g(x)$ in this manner (i.e., instead of setting $h(y)$ directly equal to $g(x)$) because of the high variance of the y_i. Further, we denote the simulation outcomes corresponding to $\hat{h}(y)$ as $\hat{y}$. Also, note that the $\hat{y}$ are functions of the calibration parameters x; that is, we can denote the simulation outcomes as $\hat{y}(x)$.

For the HCV simulation model, we define x_1 to be the per-event infection probability in the medical environment, x_2 to be the per-event needle sharing probability, and x_3 to be the per-event influence probability. y_1 is the HCV antibody prevalence, y_2 is the HCV RNA prevalence and y_3 is the IDU prevalence. Thus, in our model, given that we estimate y from the simulation outcomes corresponding to $\hat{h}(y)$ (i.e., $\hat{y}$), we can write $\hat{y}$ = ($\hat{y}_1$,$\hat{y}_2$,$\hat{y}_3$).

For the HCV transmission simulation, we observe that as the per-event infection probability in the medical environment (x_1) and the per-event needle sharing probability (x_2) increase (decrease), the estimated values of the HCV antibody prevalence ($\hat{y}_1$) and HCV RNA prevalence ($\hat{y}_2$) also increase (decrease), but do not impact the expected value of the IDU prevalence ($\hat{y}_3$). As the influence probability (x_3) increases (decreases), $\hat{y}_3$ increases (decreases). Thus, we have a non-decreasing (monotonic) relationship between the decision variables x and the estimated simulation outcomes $\hat{y}$.

This is seen in the results of our preliminary experimentation (Table 1 below) to determine the range of possible values for the calibration parameters as well. Table 1 contains the lower and higher bounds on the calibration parameters that we determined through these preliminary experiments.

Table 1: Results of preliminary experimentation.

$x = (x_1, x_2, x_3)$	$\hat{y} = (\hat{y_1}, \hat{y_2}, \hat{y_3})$
(0.035, 0.2, 1.9 $\times 10^{-5}$)	(1.17%, 0.934%, 0.087%)
(0.037, 0.4, 2.3 $\times 10^{-5}$)	(5.01%, 4.0%, 0.13%)

Note that the values of $\hat{y}^0$ (estimates of calibration targets) for our model are (3.6%, 2.6%, 0.1%), as obtained from Sood et al. (2018). If we define $x = (0.035, 0.2, 1.9 \times 10^{-5})$ as x^l and $x = (0.037, 0.4, 2.3 \times 10^{-5})$ as x^r, and the corresponding $\hat{y}$ values as $\hat{y}^l$ and $\hat{y}^r$, then the 'optimal' values of x, given by $x^* = (x_1^*, x_2^*, x_3^*)$ must satisfy the following relationship:

$$x_1^l \leq x_1^* \leq x_1^r \;\; \& \;\; x_2^l \leq x_2^* \leq x_2^r \;\; \& \;\; x_3^l \leq x_3^* \leq x_3^r$$

Now, for each calibration parameter, we define:

$$S(x_1) = \{x_1^1, x_1^2, ..., x_1^{k1}\} \;\; \& \;\; S(x_2) = \{x_2^1, x_2^2, ..., x_2^{k2}\} \;\; \& \;\; \ldots \;\; S(x_m) = \{x_m^1, x_m^2, ..., x_m^{km}\}$$

Here, $S(x_i)$ represents the set of possible values the i^{th} ($i = 1$ to m) calibration parameters can take, with ki representing the cardinality of $S(x_i)$, $i = 1$ to m. The solution space $\mathbb{S}$ is formed by the Cartesian product of the $S(x_i)$.

For our model, we define:

$$S(x_1) = \{0.035, 0.03525, 0.0355, 0.03575, 0.036, 0.03625, 0.0365, 0.03675, 0.037\}$$
$$S(x_2) = \{0.2, 0.25, 0.3, 0.35, 0.4\}$$
$$S(x_3) = \{1.9 \times 10^{-5}, 2.0 \times 10^{-5}, 2.1 \times 10^{-5}, 2.2 \times 10^{-5}, 2.3 \times 10^{-5}\}$$

Note that the elements in each of the above $S(x_i)$ are arranged in increasing order. Without loss of generality, we shall assume henceforth that $\forall\, x_i^j, x_i^l \in S(x_i)$ ($j \neq l$ and $j, l \in \{1, 2, \ldots, ki\}$, $i = 1$ to m), $x_i^j < x_i^l$ if $j < l$. Note also that $k1 = 9$, $k2 = 5$ and $km = k3 = 5$, implying $\mathbb{S}$ is a set with cardinality 225.

We now describe the application of the stochastic ruler method to calibrate the HCV transmission simulation. We shall not explain the stochastic ruler algorithm in detail due to space limitations, and we refer the author to Yan and Mukai (1992) for a detailed description of the algorithm and its underlying definitions and assumptions. The algorithm derives its name from the stochastic ruler θ against which a replicate output from the simulation is compared. We recall here that the output of a single replication of the simulation, $g(x)$, is set to be the median of, say, k replicate values of $h(y)$, as defined in (7), which we denote by $g(x) = \hat{h}(y)$. Note that the ideal (optimal) value of $E[\hat{h}(y)]$ is 0, implying that the expected value of the simulation outcomes are equal to the calibration targets in this case. Thus a value of $\hat{h}(y)$ equal to, for example, 0.3 implies an average percentage deviation of 10% for a simulation outcome from its calibration target. Thus the lower limit a of the stochastic ruler can be set as any value of $\hat{h}(y)$ below the value of $\hat{h}(y)$ corresponding to the maximum allowable average fractional deviation of the simulation outcomes from their calibration targets. We refer to this particular value of $\hat{h}(y)$ as the 'threshold' of interest, and denote it by the symbol δ. We experiment with four such thresholds - 0.45, 0.375, 0.3 and 0.2. We set a to be 0.1, lower than the smallest threshold explored in this analysis.

For determining the upper limit b of the stochastic ruler, we run the simulation at the two extreme x values x^l and x^r, and set b equal to the maximum of the $\hat{h}$ values corresponding to x^l and x^r. In other words, $b = \max(\hat{h}(y^l), \hat{h}(y^r))$. We obtained 1.446 as the value of $\hat{h}(y^l)$ and 1.229 as the value of $\hat{h}(y^r)$, and therefore b = 1.446.

Next, we construct the neighborhood structure for each candidate solution x. We first define a neighbor set $N(x_i^j)$ for each $x_i^j \in S(x_i)$ ($j = 1$ to ki, $i = 1$ to m), in the following manner.

$$
\begin{aligned}
N(x_i^j) &= \{x_i^{j-1}, x_i^j, x_i^{j+1}\}, \quad 2 \leq j \leq (ki-1) \\
N(x_i^j) &= \{x_i^{ki}, x_i^1, x_i^2\}, \quad j = 1 \\
N(x_i^j) &= \{x_i^{ki-1}, x_i^{ki}, x_i^1\}, \quad j = ki
\end{aligned}
$$

Then the collection of solutions forming the neighborhood of any solution $x = (x_1^j, x_2^j, \ldots, x_m^j)$ is given by the Cartesian product of the $N(x_i^j)$ excluding x. In other words, $N(x) = \prod_{i=1}^{m} N(x_i^j) - x$.

We take the example of per-event infection probability in the medical facility, x_1, to illustrate the construction of the above neighborhood structure. For x_1^3 = 0.0355, $N(x_1^3)$ = $\{0.03525, 0.0355, 0.03575\}$; for x_1^1 = 0.035, $N(x_1^1)$ = $\{0.037, 0.035, 0.03525\}$; and for $x_1^{k1} = x_1^9$ = 0.037, $N(x_1^9)$ = $\{0.03675, 0.037, 0.035\}$.

Therefore, for any x, we have 26 neighbors. For example, if $x = (0.0355, 0.3, 2.0 \times 10^{-9})$, then $N(x_1)$ = $\{0.03525, 0.0355, 0.03575\}$, $N(x_2)$ = $\{0.25, 0.3, 0.35\}$, $N(x_3)$ = $\{1.9 \times 10^{-9}, 2.0 \times 10^{-9}, 2.1 \times 10^{-9}\}$, and $N(x) = N(x_1) \times N(x_2) \times N(x_3) - x$.

We initiated the stochastic ruler method with x^l. The method involves selection of an appropriate candidate solution for the next iteration, where an iteration t is defined as the random sampling of a neighbor (candidate solution) of the current solution and checking whether it can be set as the next estimate of the solution to the optimization problem. A candidate solution z, sampled from the neighborhood $N(x)$ of the current solution x with probability $\frac{1}{|N(x)|}$, is selected as the next estimate of the optimal solution if all M_t tests against samples from the stochastic ruler $\theta(a,b)$ 'succeed'. For a neighbor to be selected as the system at the next iteration $t+1$, it has to pass M_t number of tests, where each test involves generating a replicate value of $g(z) = \hat{h}(y_z)$ and a sample $\theta \sim \theta(a,b)$, and then checking whether $\hat{h}(y_z) \leq \theta$. If $\hat{h}(y_z) \leq \theta$ (i.e., a 'success'), then another test is conducted with new samples $\hat{h}(y_z)$ and θ until one of the tests is unsuccessful or all M_t tests are successful. If $\hat{h}(y_z) > \theta$ (an unsuccessful test), then $t := t+1$, and the solution at iteration t is retained as the estimate of the solution at iteration $t+1$. Note that if all tests succeed, then $z \in N(x)$ is taken as the next estimate of the optimal solution, and $t := t+1$. M_t must be selected such that it must be an non-decreasing function of t. We set M_t equal to $\lceil log(t+10) - log(5) \rceil$.

The method stops if $t > T$ or $\hat{h}(y) < \delta$, where T is the computational budget set in terms of the number of iterations.

The results of applying the stochastic ruler method are depicted in Table 2 below. We used a computational budget of 40 iterations, given that generating a single realization of $\hat{h}(y)$ requires approximately 3 hours on a Intel *i*7 workstation with 3.3 GHz clock speed and 32 GB memory. Given this computational budget, we see that solutions that yield average deviations from the calibration targets that are less than the maximum allowable average fractional deviations of 10% and 6.67% (corresponding to δ values of 0.3 and 0.2) are not found. However, solutions that yield average percentage deviations less than threshold values of 15% and 12.5% are found in 15 iterations. Note that the same solution found by the SR method yields an $\hat{h}(y)$ that is less than both δ thresholds of 0.45 and 0.375 (i.e., it yields an average fractional deviation less than both 0.15 and 0.125 from the calibration targets), and hence the rows of Table 2 appear mostly identical.

Table 2: Model calibration via the stochastic ruler method: results. Notes: δ_{avg} denotes the maximum allowable average fractional deviation, $\delta_{avg(o)}$ denotes the average fractional deviation obtained from the calibration process, and t_f denotes the number of iterations to termination.

δ (δ_{avg})	$\hat{h}(y)$ ($\delta_{avg(o)}$)	Prevalence outcomes (% of population)			Calibration parameters			t_f
		HCV Antibody	HCV RNA	IDU	X_1	X_2	X_2	
0.45 (0.15)	0.360 (0.120)	2.98	2.42	0.112	0.03525	0.25	2.3×10^{-5}	15
0.375 (0.125)	0.360 (0.120)	2.98	2.42	0.112	0.03525	0.25	2.3×10^{-5}	15

4 SOLUTION SPACE TRUNCATION METHOD

In this section, we describe a method to reduce the size of the solution space - i.e., truncate it - by exploiting the monotonic relationship between the estimated simulation outcomes $\hat{y} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n)$ and the calibration parameters x_j ($j = 1$ to m). We refer to this approach as the solution space truncation (*SST*) approach. We note here that when used in the stochastic ruler method, $\hat{y}$ represents the outcome of a single 'replication' of the simulation (even though it actually represents the simulation outcomes corresponding to the median of k replicate values of $\hat{h}(y)$). For the application of the SST approach, while we generate $\hat{y}$ in the same manner as for the stochastic ruler method, we also we utilize the fact that $\hat{y}$ is the median of, say, k replicate values of the y. This is seen in Assumption 1 below, where we formalize the monotonic relationship between the $\hat{y}$ and the x_j ($j = 1$ to m).

Assumption 1 If $x_j^1 \leq x_j^2$, then $\hat{y}^1 \leq \hat{y}^2$, for $j = 1$ to m. Here, x_j^1 and $x_j^2 \in S(x_j)$ ($j = 1$ to m) and $\hat{y}^1$ and $\hat{y}^2$ are simulation outcomes corresponding to $\hat{h}(y^1)$ and $\hat{h}(y^2)$.

We begin the SST approach with x^l, and then systematically move to other solutions x^t, where $x^t \in \mathbb{S}$ and $t \in \{1, 2, .., |\mathbb{S}|\}$. We remind readers here that: (a) $|\mathbb{S}| = \prod_{j=1}^{m} |S(x_j)|$, and (b) we assume that the elements $x_j^1, x_j^2, \ldots, x_j^{kj}$ of each $S(x_j)$ are indexed in ascending order of magnitude; that is, if $l < k$ then $x_j^l < x_j^k$ for $l \neq k$ and $j = 1$ to m. Further $kj = |S(x_j)|$.

x^{t+1} is obtained by incrementing the index j of each component x_j of the current x^t by one. For example, if $x^t = (x_1^t, x_2^t, ..., x_m^t)$, then $x^{t+1} = (x_1^{t+1}, x_2^{t+1}, ..., x_m^{t+1})$. For each solution $x^t \in \mathbb{S}$, we generate the corresponding $\hat{y}^t$, evaluate $\hat{y}^t$ using a criterion that we develop below, and move on to x^{t+1} if $\hat{y}^t$ satisfies said criterion. Starting from x^l, if a given x^t does not satisfy this criterion, we stop, and set $x^l = x^{t-1}$. We then start the second part of the SST approach by beginning with $x^t = x^r$ and decreasing the indices of the components of x^t by one and evaluating the corresponding $\hat{y}^t$. We terminate the SST approach when $\hat{y}^t$ does not satisfy the evaluation criterion, and set $x^r = x^{t+1}$. This yields a redefinition of the solution

space $\mathbb{S}$ with new 'boundary' solutions x^l and x^r. We describe the criterion for evaluating the $\hat{y}^t$ and the consequent process of redefining $\mathbb{S}$ below.

As part of defining the $\hat{y}^t$ evaluation criterion, we define the optimal solution to the simulation optimization problem representing the calibration process as follows.

Definition. x^* is the optimal solution of formulation (6) if $E[\hat{y}(x^*)]$ equals the calibration targets y^0.

We also make the following assumption regarding the variance of $\hat{y}^t \ \forall \ x^t \in \mathbb{S}$.

Assumption 2 The variance of the simulation outcome estimator $\hat{y}^t$ is small in comparison with the difference between $E[\hat{y}^t]$ and $E[\hat{y}^{t+1}] \ \forall \ 1 \leq t \leq |\mathbb{S}| - 1$.

Assumption 2 is required because we define the evaluation criterion only in terms of $\hat{y}^t$, and do not consider its variance. Determining the impact of the variance of $\hat{y}^t$ on the effectiveness of the SST approach is an important avenue of future research. However, we believe it is a reasonable assumption at this stage because for our case, the variance of $\hat{y}^t$ is small given the median-based definition of $\hat{y}^t$.

Using Assumptions 1 and 2 and the definition of x^*, we are now in a position to state the following proposition that forms the basis for the evaluation criterion used in the SST approach.

Proposition 1 If $\hat{y}^t \prec y^0$ then $x^* \notin \mathbb{B}_t$, where $\mathbb{B}_t = \{x \in \mathbb{S} \mid x \preccurlyeq x^t\}$, where $\prec$ and $\preccurlyeq$ indicate element-wise comparisons. Similarly, if $\hat{y}^t \succ y^0$ then $x^* \notin \mathbb{C}_t$, where $\mathbb{C}_t = \{x \in \mathbb{S} \mid x \succcurlyeq x^t\}$, where $\succ$ and $\succcurlyeq$ indicate element-wise comparisons.

Proof. The proof follows from Assumption 1 and the definitions of x^* and $h(y)$; the definition of $h(y)$ being obtained from (7).

In the first pass of the SST approach, we begin from x^l, and generate the x^t by simultaneously incrementing the indices of x^l by one. For each x^t, we generate the corresponding $\hat{y}^t$ and test whether $\hat{y}^t \prec y^0$. If the condition is satisfied, then we eliminate solutions belonging to the set $\mathbb{B}_t$. This is because, by Assumption 1, we cannot obtain any value of $\hat{y}$ that is closer to y^0 than $\hat{y}^t$ using solutions in set $\mathbb{B}_t$. If $\hat{y}^t \succcurlyeq y^0$ and $\hat{y}^t \neq y^0$, then we move to x^r and begin generating x^t by simultaneously decrementing the indices of the components of x^r by one. For each x^t, we generate the corresponding $\hat{y}^t$ and check whether $\hat{y}^t \succ y^0$. If the condition is satisfied, then we eliminate solutions belonging to the set $\mathbb{C}_t$ following a similar logic to the removal of solutions in the set $\mathbb{B}_t$ in the first pass of the SST approach. If $\hat{y}^t \preccurlyeq y^0$ and $\hat{y}^t \neq y^0$, then we terminate the SST approach.

Note that we also terminate the SST approach (first or second passes) if we exhaust the index sets of any one of the components of the x^t; that is, we terminate the pass if at any point $t > \min(k1, k2, ..., km)$ for that pass. The SST approach is summarized in Algorithms 1 and 2. In order to implement the SST approach computationally, we construct a matrix A comprising all $x_t \in \mathbb{S}$. Each row of A consists of a solution $x^t \in \mathbb{S}$, where $x^t \in \mathbb{R}^m$, and thus A is an $|\mathbb{S} \times m|$ matrix. In our implementation, the first column represented the per-event infection probability in the medical environment, the second represented the per-event needle sharing probability, and third represented the per-event influence probability.

Algorithm 1. Solution space truncation approach: first pass.

```
Initialize with A, y^0
Initialize with x^t = x^l
for t = 1 to min(k1, k2, ..., km) do
   Set x = x^t, that is, x_1 = x^t_1, x_2 = x^t_2, ..., x_m = x^t_m
   Generate ŷ(x)
   if ŷ(x) ≺ y^0 then
      for i = 1 to |S| do
         if A(i,1) ≤ x_1 ∧ A(i,2) ≤ x_2 ∧ ... ∧ A(i,m) ≤ x_m then
            Remove row i from A
         end if
      end for
```

else
BREAK
end if
$x^t = x^{t+1}$; that is: $x_1^t = x_1^{t+1}$, $x_1^t = x_2^{t+1}$,..., $x_m^t = x_m^{t+1}$
end for

Algorithm 2. Solution space truncation approach: second pass.
Initialize with A_l (A generated as output of the first pass), y^0
Initialize with s_{new} = number of rows of A_l
Initialize with $x^t = x^r$
for $t = 1$ to $\min(k1, k2, ..., km)$ **do**
Set $x = x^t$, that is, $x_1 = x_1^t$, $x_2 = x_2^t$, ..., $x_m = x_m^t$
Generate $\hat{y}(x)$
if $\hat{y}(x) \succ y^0$ **then**
for $i = 1$ to s_{new} **do**
if $A_l(i,1) \geq x_1 \wedge A_l(i,2) \geq x_2 \wedge ... \wedge A_l(i,m) \geq x_m$ **then**
Remove row i from A_l
end if
end for
else
BREAK
end if
$x^t = x^{t-1}$; that is: $x_1^t = x_1^{t-1}$, $x_1^t = x_2^{t-1}$,..., $x_m^t = x_m^{t-1}$
end for
Algorithm output: reduced solution space $A_{out} = A_l$

The output of applying Algorithms 1 and 2 on $\mathbb{S}$ (represented by the matrix A in the above algorithms) yields the truncated solution space A_{out}. Applying the stochastic ruler on this truncated solution space yields improved results, as seen in Table 3.

Table 3: Model calibration via the stochastic ruler method and the *SST* approach: results. Notes: δ_{avg} denotes the maximum allowable average fractional deviation, $\delta_{avg(o)}$ denotes the average fractional deviation obtained from the calibration process, and t_f denotes the number of iterations to termination.

δ (δ_{avg})	$\hat{h}(y)$ ($\delta_{avg(o)}$)	Prevalence outcomes (% of population)			Calibration parameters			t_f
		HCV Antibody	HCV RNA	IDU	X_1	X_2	X_2	
0.45 (0.15)	0.444 (0.148)	2.74	2.23	0.094	0.03625	0.3	2.1×10^{-5}	1
0.375 (0.125)	0.2788 (0.092)	3.37	2.75	0.116	0.03625	0.35	2.2×10^{-5}	6
0.3 (0.1)	0.2788 (0.092)	3.37	2.75	0.116	0.03625	0.35	2.2×10^{-5}	6
0.2 (0.067)	0.1464 (0.05)	3.34	2.70	0.104	0.03575	0.35	1.9×10^{-5}	29

We see that when the stochastic ruler is applied on the solution space represented by A_{out}, solutions that yield average deviations from the calibration targets that are less than all desired threshold average fractional deviations - 15%, 12.5%, 10%, and 6.67% - are found. Further, we see that for all threshold average fractional deviations save 6.67%, the number of iterations until termination reduce substantially. However, we note here that a certain amount of computational effort must be expended in the application of the SST approach itself, and hence the tradeoff in achieving improved calibration accuracy at the cost of this computational effort must be evaluated prior to applying the SST approach. In our case, the stochastic ruler method on the original solution space required approximately 7 days and 6 hours until termination (exhaustion of the computational budget). In comparison, application of the *SST* algorithm required 9 hours, and as can be seen from Table 3, finding solutions that yielded average fractional deviations less than 10% from the calibration targets were found in less than half the number of iterations than when working with the original solution space.

5 CONCLUSIONS

In this work, we present a discrete simulation optimization approach for estimating key parameters of a complex agent-based simulation of HCV transmission via a calibration process. We apply the stochastic ruler method to solve the simulation optimization conceptualization of this model calibration process, and within a prespecified computational budget, find solutions that achieve acceptable average deviations of the simulation outcomes from their calibration targets. However, upon applying a method that we develop to reduce the solution space size using the monotonicity of the relationship between the simulation outcomes and the calibration parameters, we find improved solutions at lesser computational expense.

The HCV transmission agent-based model that we use here to demonstrate our approach towards model calibration has high variance in its key simulation outcomes. This necessitated the use of an aggregate measure of simulation outcomes in the application of the stochastic ruler method, which added to the significant computational overhead of the calibration process. However, we anticipate that this approach towards simulation model calibration can be used for other simulations (e.g., discrete-event, Monte Carlo, or other agent-based simulations) in different contexts, which may be subject to less variance in their outcomes. In such situations, the computational overhead of this approach may be substantially lower. Hence, an avenue of future work involves exploring the use of this approach for other simulations in different application settings.

REFERENCES

Alrefaei, M. H., and S. Andradóttir. 2005. "Discrete Stochastic Optimization Using Variants of the Stochastic Ruler Method". *Naval Research Logistics* 52(4):344–360.

Ambekar, A., and B. M. Tripathi. 2008. "Size Estimation of Injecting Drug Use in Punjab and Haryana". https://www.researchgate.net/publication/279849518_Size_estimation_of_Injecting_Drug_Use_in_Punjab_and_Haryana, accessed 25th February 2021.

Azim, T., S. I. Khan, F. Haseen, N. L. Huq, L. Henning, M. M. Pervez, M. E. Chowdhury, and I. Sarafian. 2008. "HIV and AIDS in Bangladesh". *Journal of Health, Population, and Nutrition* 26(3):311.

Bicher, M., C. Urach, G. Zauner, C. Rippinger, and N. Popper. 2017. "Calibration of a Stochastic Agent-based Model for Re-hospitalization Numbers of Psychiatric Patients". In *Proceedings of the 2017 Winter Simulation Conference (WSC)*, edited by V. W. K. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. H. Page, 2940–2951. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Chakravarti, A., A. Ashraf, S. Malik et al. 2013. "A Study of Changing Trends of Prevalence and Genotypic Distribution of Hepatitis C Virus among High Risk Groups in North India". *Indian Journal of Medical Microbiology* 31(4):354.

Das, S., D. Sen, V. Ramamohan, and A. Sood. 2019. "An Agent-based Model of Hepatitis C Virus Transmission Dynamics in India". In *Proceedings of the 2019 Winter Simulation Conference (WSC)*, edited by N. Mustafee, K. G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y. Son, 984–995. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Fabretti, A. 2013. "On the Problem of Calibrating an Agent Based Model for Financial Markets". *Journal of Economic Interaction and Coordination* 8(2):277–293.

Hale, D. K., C. Antoniou, M. Brackstone, D. Michalaka, A. T. Moreno, and K. Parikh. 2015. "Optimization-based Assisted Calibration of Traffic Simulation Models". *Transportation Research Part C: Emerging Technologies* 55:100–115.

Hazelbag, C. M., J. Dushoff, E. M. Dominic, Z. E. Mthombothi, and W. Delva. 2020. "Calibration of Individual-based Models to Epidemiological Data: A Systematic Review". *PLoS Computational Biology* 16(5):e1007893.

Hong, L. J., and B. L. Nelson. 2006. "Discrete Optimization via Simulation Using COMPASS". *Operations Research* 54(1):115–129.

Johnson, R. T., T. A. Lampe, and S. Seichter. 2009. "Calibration of an Agent-based Simulation Model Depicting a Refugee Camp Scenario". In *Proceedings of the 2009 Winter Simulation Conference (WSC)*, edited by M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, 1778–1786. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Karnon, J., and T. Vanni. 2011. "Calibrating Models in Economic Evaluation". *Pharmacoeconomics* 29(1):51–62.

Kong, C. Y., P. M. McMahon, and G. S. Gazelle. 2009. "Calibration of Disease Simulation Model Using an Engineering Approach". *Value in Health* 12(4):521–529.

Law, A. M. 2015. *Simulation Modeling and Analysis*. 5th ed. New York: McGraw-Hill, Inc.

Luo, W., D. A. Katz, D. T. Hamilton, J. McKenney, S. M. Jenness, S. M. Goodreau, J. D. Stekler, E. S. Rosenberg, P. S. Sullivan, and S. Cassels. 2018. "Development of an Agent-based Model to Investigate the Impact of HIV Self-testing Programs on Men who have Sex with Men in Atlanta and Seattle". *JMIR Public Health and Surveillance* 4(2):e58.

Ministry of Home Affairs, Government of India. 2016. "SRS Statistical Report 2016". https://censusindia.gov.in/vital_statistics/SRS_Reports__2016.html, accessed 26th February 2021.

Moya, I., M. Chica, and Ó. Cordón. 2021. "Evolutionary Multiobjective Optimization for Automatic Agent-Based Model Calibration: A Comparative Study". *IEEE Access* 9:55284–55299.

Norkin, V. I., G. C. Pflug, and A. Ruszczyński. 1998. "A Branch and Bound Method for Stochastic Global Optimization". *Mathematical Programming* 83(1):425–450.

Osmond, D. H., N. S. Padian, H. W. Sheppard, S. Glass, S. C. Shiboski, and A. Reingold. 1993. "Risk Factors for Hepatitis C Virus Seropositivity in Heterosexual Couples". *Journal of the American Medical Association* 269(3):361–365.

Pietzsch, B., S. Fiedler, K. G. Mertens, M. Richter, C. Scherer, K. Widyastuti, M.-C. Wimmler, L. Zakharova, and U. Berger. 2020. "Metamodels for Evaluating, Calibrating and Applying Agent-based Models: a Review". *Journal of Artificial Societies and Social Simulation* 23(2).

Reiker, T., M. Golumbeanu, A. Shattock, L. Burgert, T. A. Smith, S. Filippi, E. Cameron, and M. A. Penny 2021. "Machine Learning Approaches to Calibrate Individual-based Infectious Disease Models". https://www.medrxiv.org/content/10.1101/2021.01.27.21250484v3, accessed 8th March 2021.

Shrestha, S., A. N. Hill, S. M. Marks, and D. W. Dowdy. 2017. "Comparing Drivers and Dynamics of Tuberculosis in California, Florida, New York, and Texas". *American Journal of Respiratory and Critical Care Medicine* 196(8):1050–1059.

Sood, A., A. Suryaprasad, A. Trickey, S. Kanchi, V. Midha, M. Foster, E. Bennett, S. Kamili, F. Alvarez-Bognar, S. Shadaker et al. 2018. "The Burden of Hepatitis C Virus Infection in Punjab, India: a Population-based Serosurvey". *PLoS ONE* 13(7):e0200461.

Taylor, D. C., V. Pawar, D. Kruzikas, K. E. Gilmore, A. Pandya, R. Iskandar, and M. C. Weinstein. 2010. "Methods of Model Calibration". *Pharmacoeconomics* 28(11):995–1000.

Voloshin, D., D. Rybokonenko, and V. Karbovskii. 2015. "Optimization-based Calibration for Micro-level Agent-based Simulation of Pedestrian Behavior in Public Spaces". *Procedia Computer Science* 66:372–381.

Yan, D., and H. Mukai. 1992. "Stochastic Discrete Optimization". *SIAM Journal on Control and Optimization* 30(3):594–612.

AUTHOR BIOGRAPHIES

SOHAM DAS is a Ph.D. student in the Department of Mechanical Engineering at the Indian Institute of Technology (IIT) Delhi. He holds an M.Tech. degree in Industrial Engineering from IIT Delhi. His email address is 2019MEZ8426@mech.iitd.ac.in.

NAVONIL MUSTAFEE is a Professor at the University of Exeter Business School. He holds a Ph.D. in Information Systems and Computing from Brunel University. His email address is n.mustafee@exeter.ac.uk.

VARUN RAMAMOHAN is an Assistant Professor in the Department of Mechanical Engineering at the Indian Institute of Technology Delhi. He holds a PhD in Industrial Engineering from Purdue University. His email address is varunr@mech.iitd.ac.in.