npj Complexity

https://doi.org/10.1038/s44260-026-00094-y

Article in Press

EFECT enables reliable and reproducible stochastic simulations across disciplines

Received: 23 December 2025

Accepted: 28 July 2026

Published online: 05 August 2026

Cite this article as: Sego, T.J., König, M., Fonseca, L.L. *et al.* EFECT enables reliable and reproducible stochastic simulations across disciplines. *npj Complex* (2026). https://doi.org/10.1038/s44260-026-00094-y

T. J. Sego, Matthias König, Luis L. Fonseca, Dilan Pathirana, Frank T. Bergmann, Hernán E. Grecco, Mauro Silberberg, Subhasis Ray, Baylor Fain, Adam C. Knapp, Krishna Tiwari, Henning Hermjakob, Herbert M. Sauro, James A. Glazier, Reinhard C. Laubenbacher & Rahuman S. Malik-Sheriff

EFECT enables reliable and reproducible stochastic simulations across disciplines

T.J. Sego[1*], Matthias König[2,3], Luis L. Fonseca[1], Dilan Pathirana[4], Frank T. Bergmann[5], Hernán E. Grecco[6], Mauro Silberberg[6], Subhasis Ray[7], Baylor Fain[1], Adam C. Knapp[1], Krishna Tiwari[8], Henning Hermjakob[8], Herbert M. Sauro[9], James A. Glazier[10], Reinhard C. Laubenbacher[1], Rahuman S. Malik-Sheriff[1,8,11†]

[1]Department of Medicine, University of Florida, Gainesville, FL, USA

[2]Institute for Theoretical Biology, Humboldt University Berlin, Berlin, Germany

[3]University Hospital Schleswig-Holstein, Campus Lübeck, First Department of Medicine, 23538 Lübeck, Germany

[4]Life and Medical Sciences Institute & Bonn Center for Mathematical Life Sciences, University of Bonn, Bonn 53115, Germany

[5]BioQUANT, Heidelberg University, 69120 Heidelberg, Germany

[6]Universidad de Buenos Aires, Facultad de Ciencias Exactas y Naturales, Departamento de Física and CONICET - Universidad de Buenos Aires, Instituto de Física de Buenos Aires (IFIBA). Buenos Aires 1426, Argentina

[7]Centre for High Impact Neuroscience and Translational Applications (CHINTA), TCG CREST, Kolkata, India

[8]European Molecular Biology Laboratory, European Bioinformatics Institute (EMBL-EBI), Hinxton, Cambridge, UK

[9]Department of Bioengineering, University of Washington, Seattle, WA, USA

[10]Department of Intelligent Systems Engineering and Biocomplexity Institute, Indiana University, Bloomington, IN, USA

[11]Department of Surgery & Cancer, Faculty of Medicine, Imperial College London, London, UK

* timothy.sego@medicine.ufl.edu

† sheriff@ebi.ac.uk

Abstract

Stochastic simulations underpin computational research in fields from systems biology and epidemiology to finance and physics-informed machine learning, yet their reproducibility is hard to quantify because each run yields different outcomes. Existing practices to improve computational reproducibility focus on sharing code, simulation seeds, data, and software environments but do not ensure independent reproduction of results, limiting scientific rigor. Here we introduce the Empirical Characteristic Function Equality Convergence Test (EFECT), a universal computational framework for quantifying the statistical reproducibility of stochastic simulations based on empirical characteristic functions. EFECT defines a normalized EFECT error that measures distributional differences between two sets of simulation outputs over a

model- and scale-independent range, and an EFECT convergence point that specifies the minimum sample size required to achieve a desired reproducibility threshold at a chosen significance level. EFECT applies to any simulation whose results can be represented as bounded, real-valued data, regardless of modelling formalism or source of stochasticity. To facilitate widespread adoption and data exchange, we implemented EFECT in an open-source software library and a compact, machine-readable standardized report. We rigorously evaluated the framework across more than 40 test cases spanning stochastic differential equations, agent-based models, Boolean networks, and partial differential equations. Applying EFECT to published models in pandemic epidemiology, financial stochastic volatility, and physics-informed neural networks reveals that commonly used sample sizes are often underpowered: substantial parameter differences can go undetected unless thousands of replicates are simulated. By providing a domain-agnostic statistical reproducibility metric, reusable software libraries, and a standardized report format, EFECT offers a practical framework for embedding quantitative reproducibility and replicability checks into stochastic simulation workflows across data-intensive disciplines. Thus, EFECT delivers the capability to make scientific inferences and policy decisions on the basis of reliable, reproducible stochastic simulations.

Introduction

Various disciplines have acknowledged a crisis of reproducibility in science (1), to which computational research is not immune (2). Reproducibility has been argued to be the minimum standard for assessing scientific claims in computational science (3). For stochastic modeling studies, reproducibility is essential to ensure reliability in critical predictions, from environmental assessments to financial forecasts. In deterministic computational research, reproducibility can be quantitatively assessed. In contrast, stochastic simulation studies lack a general framework for reproducibility evaluation, leaving the true extent of the problem unknown and likely underestimated due to the intrinsic variability of such systems and the absence of standardized assessment methods.

In life sciences, significant efforts are made through the development of an ecosystem of community infrastructures and standards to improve model credibility by ensuring simulation tools yield consistent outputs for identical inputs (4). These efforts include resources like BioModels, the world's largest repository of curated biological models (5), and BioSimulators, a free online registry of biological model simulators (6), both of which leverage community standards such as the Systems Biology Ontology (SBO, (7)), Systems Biology Markup Language (SBML, (8)), and Simulation Experiment Description Markup Language (SED-ML, (9)), to promote reusability and reproducibility of biological models in diverse fields including neuroscience (10), quantitative systems pharmacology (11), and drug discovery (12). A 2021 study showed that, despite these efforts, the simulation results of nearly half of 455 analyzed deterministic ordinary differential equation model from peer-reviewed articles cannot be reproduced (13). Given that stochastic simulation yields varying results with every run, it was expected that their reproducibility would be worse.

Best practices for reproducible computational research emphasize that researchers must, at minimum, reproduce their own results (14). In systems biology, guidelines recommend storing all raw simulation data, sharing data underlying graphs and tables, and automating model verification and validation (15), with some advocating for journal policies mandating raw data publication (16). Additional recommendations include ensuring statistical repeatability of simulation results and expanding workflows to validate reproducibility (17). Most guidelines

focus on deterministic models, yet advanced research in systems biology, finance, and environmental sciences increasingly uses stochastic models to capture random processes that deterministic approaches cannot (18–20).

Emerging technologies like digital twins, which will likely use stochastic modeling, need rigorous verification and validation of simulation results (21). There is a growing emphasis on analysis of sample distributions of simulation results to ensure model credibility (22). Reproducibility in stochastic modeling is crucial for reliable environmental predictions, such as climate models, where variability and uncertainty must be systematically addressed (23). Terminology concerning reproducibility, replicability, and repeatability differs across scientific communities. Often, reproducibility refers to obtaining consistent results from independent studies under similar experimental conditions (16,24). In this study, we use “statistical reproducibility” specifically to describe whether independently generated stochastic simulation samples are sufficiently consistent in distribution under equivalent model and input conditions. Stochastic simulation results, influenced by intrinsic variability or input sampling (e.g., parameters, initial conditions), vary with each run. This inherent variability of results necessitates robust metrics and standards to quantify reproducibility and guarantee that outcomes will not be different when a study is repeated. This is essential to ensure findings are reliable and consistently validated across studies, especially when findings influence decisions pertaining to large social and economic impact.

Reproducing individual stochastic simulation trajectories of a study is feasible using the seed from pseudorandom number generators, though there are challenges due to software implementation differences (17). Moreover, reproducing a trajectory using a fixed seed does not establish that independently generated stochastic samples are statistically consistent. Furthermore, this approach also has two major limitations: (i) it overlooks underpowered or selective model simulation sampling in the study, weakening assessment, and (ii) storing and sharing data for all individual trajectories does not scale up to community-level practice. Alternatively, comparing stochastic results via distributional statistics requires assumptions (e.g., normality) that limit applicability outside specific contexts. For instance, the SBML Test Suite (25) compares stochastic results using summary statistics. However, summary statistics like mean and standard deviation are not unique and so are unreliable for general identification of differences in stochastic model results (*e.g.*, when initial conditions have different distributions with the same summary statistics, Source Code S3.20). Previous work developed a test and heuristics to quantify aleatoric uncertainty for determining a minimum sample size that sufficiently represents a stochastic model (26,27) but did not provide a means to support standardized reproduction of results.

In this work, we developed a method and metrics to assess the statistical reproducibility of stochastic simulation results across the model lifecycle. Our method evaluates whether independently generated simulation samples are consistent in distribution through statistically principled comparisons of output distributions. Specifically, it tests whether simulation results converge to the same unique, unknown distribution using empirical characteristic functions. Hence, we call our method the Empirical Characteristic Function Equality Convergence Test (EFECT). EFECT is therefore intended as a statistical reproducibility-oriented method for stochastic simulation data, rather than as a universal definition or formalization of reproducibility or replicability, which varies across scientific domains. Hence, it can be applied to both scientific replicability and reproducibility to assess statistical consistency, agnostic to the definition chosen by different scientific communities.

We showed the broad relevance of our method by demonstrating its application with multiple stochastic modeling applications, modeling methodologies, and BioModels entries. Our method quantifies differences in stochastic simulation results independent of modeling method and with universal meaning, such that outcomes from reproducing results are generally interpretable. We designed a workflow built on our method to accommodate data exchange within modeling communities for stochastic model reproducibility through outlets like online repositories and journal publications.

Results

EFECT-based reproduction of stochastic simulation results consists of two components: first, generation of reliable and statistically reproducible stochastic simulation results by the modeler; second, testing whether the stochastic simulation results have been sufficiently reproduced using a well-defined quantitative metric by a model curator. Throughout, we employ the following definitions to describe results:

- *Reproduced results*: results for which another set of sufficiently similar results have been independently produced.
- *Reproducible results*: results that can be reproduced.

We refer to the individual who produces the results as the Modeler and to the one who independently analyzes the model and reproduces the results as the Curator. We require that EFECT assesses similarity of results using a quantitative metric, called the *EFECT error*, with meaning that is both independent of a model's mathematical construct and scale of results, and robust to support a broad range of models and sources of results' stochasticity. In particular, we require that EFECT does not rely on any knowledge about the model from which results were produced. Likewise, we require that the EFECT error supports results with variance produced by simulation stochasticity and/or stochastic selection of model parameters and initial conditions. EFECT assesses the similarity of stochastic simulation results by testing for equality in distribution of results through comparison of empirical characteristic functions (ECFs) generated from results distributions. Throughout, we refer to a set results collected from one or more executions of a stochastic simulation as a *simulation sample*, and likewise to the number of executions as the *sample size*. The EFECT error quantifies differences between simulation samples over a model- and scale-independent range, where an EFECT error of zero demonstrates equality of simulation samples, and an error of two demonstrates maximum difference. Using EFECT, a Modeler performs a *test for reproducibility* to test whether the distribution of their simulation sample is reproducible by randomly dividing their simulation sample into two subsamples of equal size and measuring the EFECT error of the subsamples. The Modeler improves the reproducibility of their simulation sample by repeating this process while increasing sample size until the distribution of the sampled EFECT error satisfies a reproducibility criterion, called the *EFECT convergence point*. We refer to the minimum simulation sample size to satisfy the EFECT convergent point as the *EFECT sample size*. Given the size of the subsamples, the ECFs generated from one of them, and some additional information (described below) accompanying a reported model, a Curator tests whether they can reproduce a simulation sample by comparing to ECFs from their own implementation of the reported model. A Curator tests a reported simulation sample to their own for equality in distribution according to the null hypothesis that samples of the EFECT error when comparing simulation samples and when quantifying reproducibility are taken from the same unknown distribution. A Curator assesses whether they reproduce a simulation sample by rejecting the

null hypothesis at an *a priori* significance level according to Chebyshev's inequality for unknown population mean and variance. For details, see *Methods*. For a summary for all test scenarios, see Table S1. For details on test scenarios and locating their supplementary materials, see Table S2.

EFECT Accurately Quantifies Similarity in Stochastic Simulation Results

To evaluate the robustness and generality of EFECT, we rigorously tested it across more than 40 diverse use cases, including stochastic differential equations, agent-based models, and Boolean networks, encompassing both intrinsic stochasticity and variability introduced by input sampling. A full list of these test scenarios is provided in Table S2. Below, we summarize these use cases and highlight EFECT's ability to quantify reproducibility across a range of modeling approaches, simulation platforms, and sources of stochasticity.

Stochastic Algorithms and Sampling of Parameters and Initial Conditions

We first tested the capability of EFECT to quantify the similarity of simulation samples using a variety of ordinary differential equation (ODE) models with different sources of stochasticity (Figure 1). Test models consisted of either deterministic simulation while sampling model parameters and/or initial conditions using a normal distribution for one, two, and ten parameters and/or initial conditions, or using a uniform distribution sampling for one or ten parameters and/or initial conditions; or of stochastic simulation using the Gillespie Stochastic Simulation Algorithm (28,29), including two BioModels entries (30,31). For all test models subject to a source of stochasticity, we increased sample size and quantified reproducibility of the simulation sample by evaluating the similarity of equally sized subsamples as previously described for our Modeler's test for reproducibility. Similarity of simulation samples was quantified using the EFECT error defined in [3]. We found that the EFECT error decreased with increasing sample size for all modeler's test cases according to a power law. A representative result of this general observation is shown in Figure 1D. EFECT reliably quantifies reproducibility of stochastic results from ODE simulations, with reproducibility improving systematically as sample size increases.

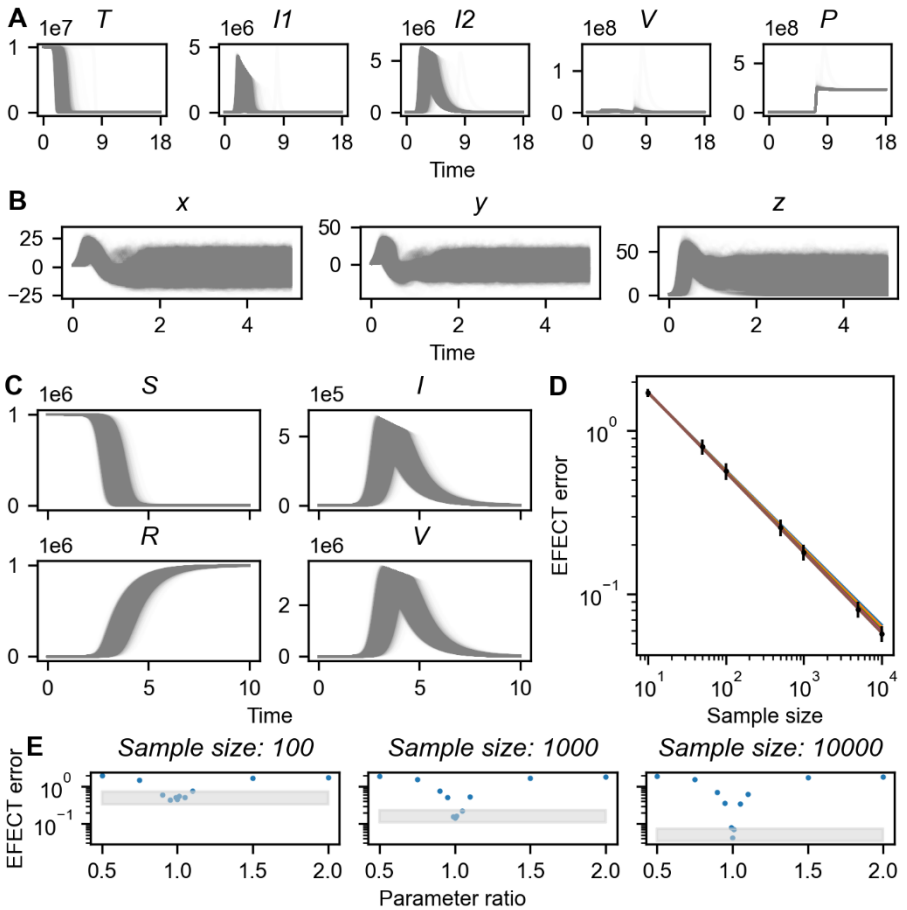

Figure 1. Testing similarity of stochastic biological simulation results from ordinary differential equation models with different sources of stochasticity. A: Trajectories while randomly sampling a parameter of a model of viral-bacterial coinfection with cell populations *T*, *I1*, *I2*, and viral and bacterial loads *V* and *P* (Source Code S3.1). Sample size of 10k trajectories shown. B: Trajectories while simulating a Lorenz system with the Gillespie stochastic simulation algorithm (Source Code S3.9). Sample size of 10k trajectories shown. C: Trajectories while randomly sampling a parameter of a model of viral infection. Sample size of 10k trajectories shown. D: EFECT error quantifying reproducibility of a sample from the model shown in panel C as a function of sample size. An EFECT error of zero demonstrates equality, and of two demonstrates complete dissimilarity. Lines show fits of a typical power law relationship, where similarity increases with increasing sample size. Dots show mean evaluations of the EFECT error. Bars show one standard deviation of EFECT error evaluations upward and downward. E: EFECT error per sample size from the model shown in panel C while comparing samples from models of the same structure but a difference in parameter (as measured by the parameter ratio). EFECT error ranges from 0 (identical distributions) to 2 (maximally different distributions). Shaded regions show the mean ± three standard deviations of the EFECT error when testing for reproducibility.

Detecting Parametric and Structural Differences

We then tested the capability of EFECT to detect structural and parametric differences in SDE models by comparing simulation samples generated from them. We tested detection of structural differences by comparing simulation samples of various sample sizes generated from different models. When comparing simulation samples from models with structural differences, we found that the EFECT error does not decrease with increasing sample size but instead tends to near maximum difference (Source Code S3.13). We tested detection of parametric differences by comparing simulation samples of various sizes generated from the same model but with a difference in the distribution from which a parameter value was drawn. Specifically, we assessed the likelihood of a false-positive according to [6] when testing for equality in

distribution for a given sample size and difference in parameter distribution. Generally, EFECT can detect a difference if it is unlikely to yield a false-positive. In this test, we perform the following procedure: Step 1: generate a simulation sample; Step 2: quantify reproducibility of the simulation sample; Step 3: extract a subsample of half the size (as would be done when reporting a model); Step 4: generate a second simulation sample of the same half size and with a modified parameter distribution; and Step 5: test how well the simulation sample from Step 4 reproduces the simulation sample from Step 3.

For the model shown in Figure 1C, we scaled the mean and standard deviation of the normal distribution for a single parameter by factors ranging from 0.5 to 2 (Source Code S3.14). Using a significance level of 0.05, we found that EFECT could detect differences in distributions for a sample size of 10,000 when scaling downward or upward by as little as 5% to 1%, respectively. The same test and model with a sample size of 1,000 could detect 5% scaling differences. With a sample size of 100, the same test and model could detect 25% downward or 50% upward scaling differences. See Table S1 for computed p-values. The same analysis was repeated the following: with different ODE models (detected 5% or more scaling differences for sample size 10,000, Source Code S3.17); with constant parameters with a scaling difference and stochastic simulation (detected 5% or more scaling differences for sample size 10,000, Source Code S3.18); or while individually scaling distribution mean (detected 5% or more scaling differences for sample sizes 5,000 and 10,000, Source Code S3.16) or variance (detected scaling differences of 25% or more downward, and 10% or more upward, for a sample size of 10,000, Source Code S3.19). In all tests, upward and downward scaling differences of between 5% and 50% were detectable with a significance level of 0.05 when the reproducibility test EFECT error mean was below 0.1. Hence, EFECT can detect even a single parametric difference between models by comparing samples from them, and increasingly so with decreasing EFECT error.

Reproducibility with Stochastic Agent-Based and Boolean Network Models

To demonstrate whether EFECT supports reproducibility of stochastic results from methods other than SDEs, we tested whether we could perform the same analysis with three implementations of an agent-based model (ABM, Figure 2). We considered the Grass-Sheep-Wolf (GSW) ABM implemented in NetLogo ((32), Figure 2A), and implementations derived from it in MATLAB and Python (Source Code S4). We generated samples with sizes up to 20,000 (Figure 2B) and again found a decrease in EFECT error with increasing sample size according to a power law when testing for reproducibility (Figure 2C). Comparing results from different implementations, EFECT showed that no two implementations produce results that are equal in distribution (Figure 2D). Thus, we found that results from the three ABM implementations were reproducible, and that none of them produced the same results, according to EFECT. We also tested a stochastic Boolean network model of cell fate (33) implemented in MaBoSS (34) and found that EFECT supports evaluation of the reproducibility of stochastic Boolean network model results (Source Code S2.1, S3.26). EFECT distinguishes between different implementations of the same agent-based model and quantified reproducibility of the stochastic Boolean network model, demonstrating its utility for verifying reproducibility independent of modeling approach.

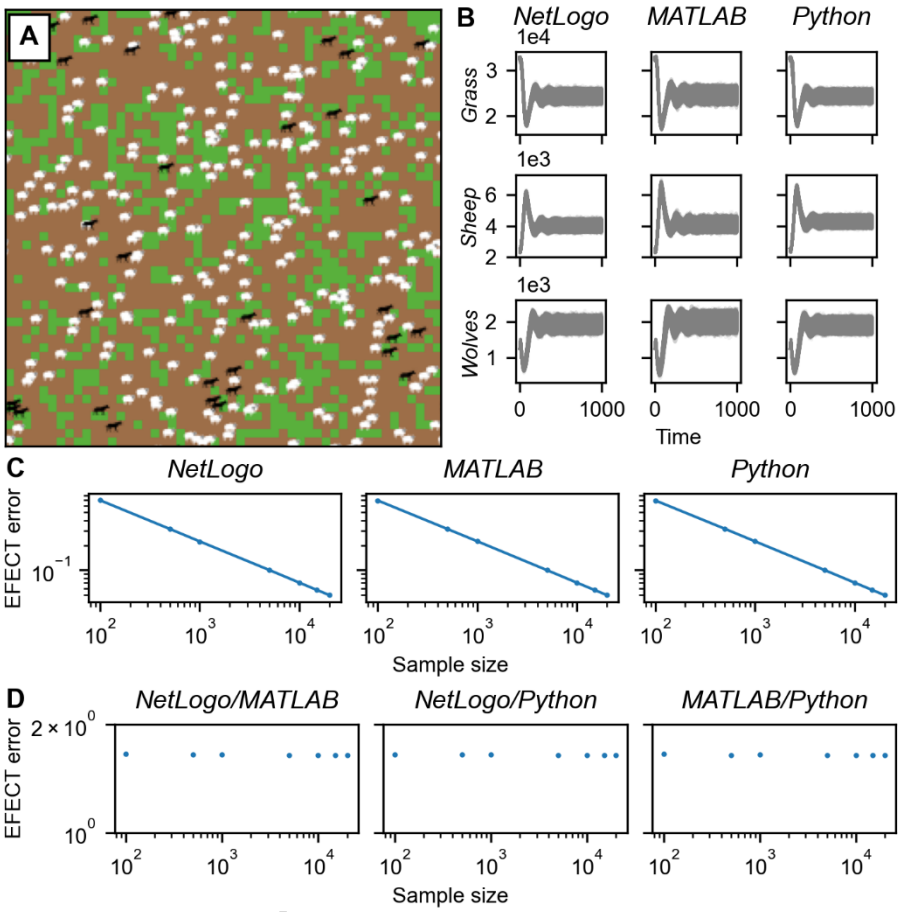

Figure 2. Testing reproducibility of results from three implementations of a Grass-Sheep-Wolf agent-based model. A: Screenshot of the original agent-based model implemented in NetLogo. B: Trajectories of model Grass (top), Sheep (middle), and Wolves (bottom) from 20,000 executions of the NetLogo (left, Source Code S4.2), MATLAB (center, Source Code S4.1), and Python (right, Source Code S4.3) implementations. Distributions of trajectories have subtle differences across simulators that are difficult to detect by inspection (e.g., slightly different oscillations). C: Testing for reproducibility of results distributions from each implementation (Source Code S3.21). Results from all implementations converged in distribution. D: Testing for equality in distribution of pairs of samples from all implementations. No two implementations produced samples that were equal in distribution, as demonstrated by non-converging EFECT errors.

Robustness of EFECT to Model Complexity and Scale

Finally, further evaluation showed that testing for reproducibility with EFECT is generally insensitive to the number of model variables (Source Code S3.30, Figure S1), significant differences in scale between model variables (Source Code S3.27, Figure S2), or order of magnitude of standard deviation (Source Code S3.31). This demonstrates that EFECT is a robust method to test model reproducibility. We also found that the runtime of our implementation of EFECT increases linearly with the number of model variables (runtime of 305 $\pm$ 86 s, 401 $\pm$ 119 s, 469 $\pm$ 134 s, 642 $\pm$ 232 s, and 1,261 $\pm$ 484 s when testing reproducibility for 2, 3, 4, 5, and 10 model variables on an Intel i9, Source Code S2.2). EFECT maintains consistent performance across models with varying numbers of variables, scales, and variances, highlighting its robustness and generalizability.

Workflows for Reproducibility of Stochastic Models

After testing that EFECT supports quantifying whether simulation results are reproducible and reproduced, we developed a Stochastic Simulation Reproducibility (SSR) workflow that defines the steps required by all stakeholders (*i.e.*, the Modeler and Curator) over the lifetime of a model to generate stochastic results that are independently reproduced. Figure 3 graphically demonstrates these steps, the details of which are described in Methods. Generally, the workflow involves the Modeler producing a published model and supporting data, which the Curator uses to test and report whether they can reproduce results according to an *a priori* significance level. Generally, neither stakeholder is required to use the same simulator that implements the model (though it may be beneficial to use different simulators). In this workflow, the Modeler's test evaluates the internal statistical reproducibility of simulation samples generated under a specified implementation and set of inputs. The Curator's test evaluates whether an independently generated sample is statistically consistent with the Modeler's reported sample. Depending on the terminology adopted by a particular community, the latter activity may form part of a broader reproducibility or replicability assessment, but EFECT itself evaluates only the distributional consistency of the stochastic outputs.

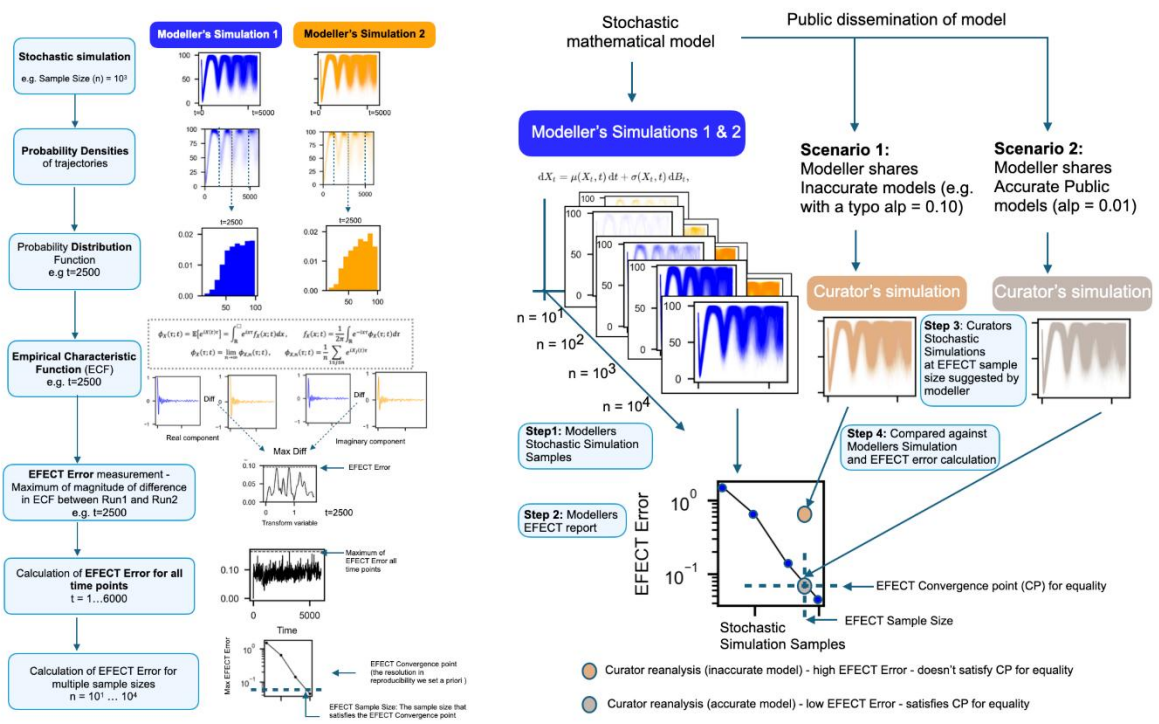

Figure 3. Graphical representation of the steps involved in stochastic simulation results comparison using EFECT. Left: Demonstration of each step in the process of modelers' internal reproducibility exercise, comparing two independent stochastic simulation results using EFECT. A modeler samples a stochastic simulation and produces empirical characteristic functions (ECFs) of them for testing whether their sample is equal in distribution to another sample by comparison of ECFs. The modeler uses the EFECT Error to compare the ECFs and quantify reproducibility of their results for each sample size. The modeller increases sample and continues the comparison until a threshold, the EFECT Convergence Point, is achieved. Right: Two possible scenarios during a reproducibility study using EFECT, where differences in results are detected (Scenario 1) and results are verified as reproduced (Scenario 2). Having achieved an established EFECT Convergence Point, results can be compared in a statistical test of whether two given results samples are realizations of the same stochastic process. Disproving the hypothesis with an *a priori* significance indicates that results are not reproduced.

A Modeler iteratively adds simulation results to a sample until results are sufficiently reproducible. The Modeler evaluates reproducibility by randomly generating two equally sized subsamples of their current sample and calculating the EFECT error when comparing ECFs of the subsamples. Evaluations of the EFECT error are collected as a random variable and the process of randomly selecting and evaluating subsamples is repeated until the absolute relative change in the mean of EFECT error evaluations converges below a threshold (here tested with a threshold of 0.1%). Reproducibility is determined according to a threshold for the statistics of the EFECT error. Here we assume that the modeler, curator, and others in their field have defined such a threshold, an *EFECT convergence point*, through consensus (here accepted when the mean plus three standard deviations is below 0.075). When a sample is not reproducible, the Modeler generates more results, adds them to the sample, and repeats the process. When a sample is reproducible, we recommend that the Modeler generates minimal supporting data, called an *EFECT report*, to allow others to reproduce a subsample. Excluding relevant information specific to a model, simulation, or data formatting other than what our workflow requires, the contents of an EFECT report is as follows (see Table S3 for examples):

- *Stochastic variable names*: A list of all named stochastic variables in reported results.
- *Simulation times*: A list of all simulation times at which results are reported.
- *EFECT error statistics*. The accepted mean and standard deviation of the EFECT error during the test for reproducibility.
- *EFECT sample size*: The size of an accepted sample when testing for reproducibility.
- *ECF evaluations*: Evaluations of ECFs generated from a subsample of an accepted sample for each named stochastic variable at each simulation time.
- *ECF domains*: Maximum transform variable value for all reported ECF evaluations according to [5].
- *Significant figures*: The significant figures of results produced by the employed simulator(s) when testing for reproducibility.

A Curator develops their own implementation of a published model and generates results distributions according to information provided by the EFECT report accompanying the model. Specifically, the sample generated by the Curator will be of a size equal to the reported sample size and record results for all named stochastic variables at all reported simulation times. The Curator performs the test for reproducibility and generates their own EFECT error statistics. Finally, the Curator randomly selects a subsample, computes the EFECT error for comparing the generated ECFs to those reported in the EFECT report, and computes the probability that the EFECT error when comparing ECFs was taken from the (unknown) distribution of the EFECT error during the Curator's test for reproducibility. The Curator determines an outcome according to whether the computed probability is above (reproduced) or below (not reproduced) a significance level. A process diagram graphically illustrates these steps in Figure S3.

To test whether our SSR workflow and EFECT report sufficiently support a lifecycle of model development that includes reproducing stochastic simulation results, we simulated the processes defined for a Modeler and Curator and exchange of information between them for an ODE model of viral infection (35) defined in SBML (Figure 4). In these simulations, a Modeler reports results from an ODE model with random parameter sampling as generated by libRoadRunner (36), and a Curator reproduces results using COPASI (37). In our simulation of the processes performed by the Modeler (*i.e.*, the "Modeler Workflow" in Figure 3), the Modeler

begins by iteratively increasing the size of their sample and testing for reproducibility until a power law could be fit to EFECT error evaluations. The Modeler rounds simulation results to six significant figures for compatibility with COPASI default output (neglecting this step results in failure to reproduce, Source Code S3.25). The fit is used to estimate the required sample size to achieve an EFECT convergence point, here defined *ad hoc* as an EFECT error with mean plus three standard deviations below 0.075, when testing for reproducibility (EFECT error of $54.4\times10^{-3} \pm 6.40\times10^{-3}$, Figure 4A, B, C), the final size of which was 9,644. An EFECT report is then generated and stored to file (Source Code S3.22). In our simulation of the processes performed by the Curator (*i.e.*, the "Curator Workflow" in Figure 3), the Curator implements the same model in COPASI, generates a sample of the reported size for all reported stochastic model variables and simulation times, and performs the test for reproducibility (EFECT Error of $53.5\times10^{-3} \pm 6.59\times10^{-3}$). The Curator then compares the ECFs generated from a randomly selected subsample to those reported in the EFECT report. Our simulation showed that the Curator accepts that results are reproduced with EFECT error 0.0568 and p-value 1 (Figure 4D, Source Code S3.23) and can readily detect differences in model parameter distributions (Source Code S3.24).

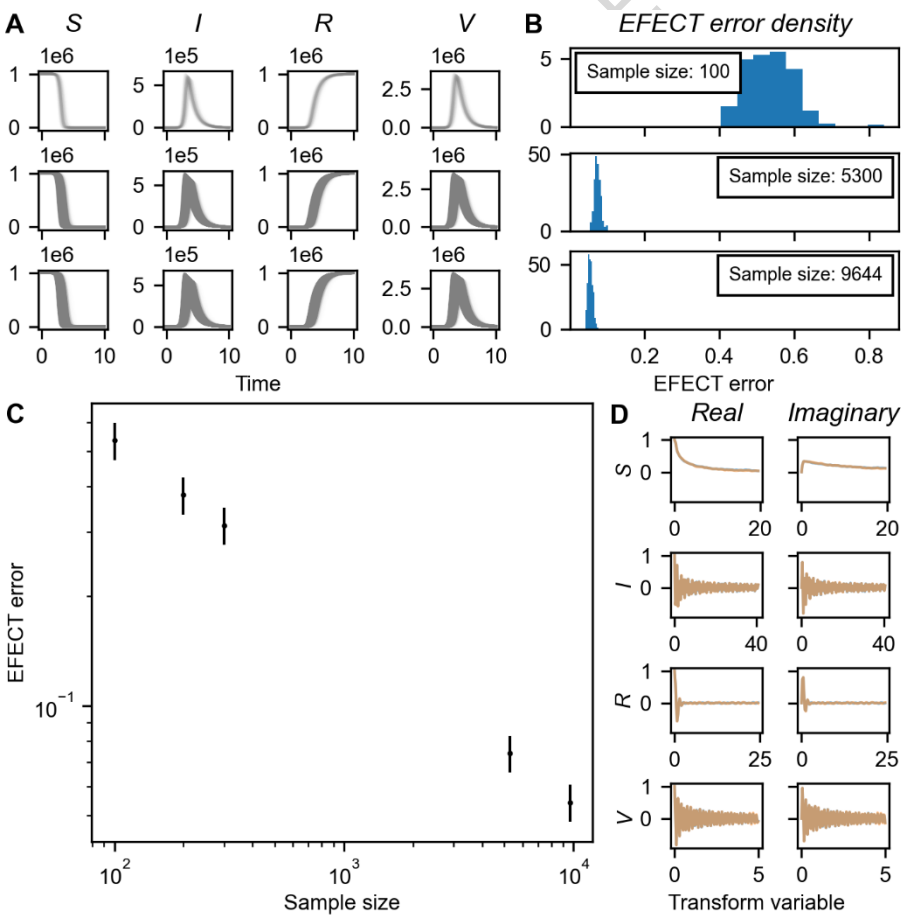

Figure 4. Simulation of the workflow for reproducing stochastic simulation results using different simulators. A, B, C: Modeler workflow using libRoadRunner. Trajectories (panel A) and corresponding EFECT error sampling (panel B) per select sample size are shown when testing for reproducibility (panel C). The distributions in panel B are from sampling the EFECT error when testing for reproducibility of samples shown in panel A. A sample was considered reproducible when the EFECT error plus three standard deviations was below a threshold (here 0.075). Error bars show three standard deviations. D: Comparison of empirical characteristic functions during the Curator workflow.

Curator results were generated using COPASI to generate empirical characteristic functions (orange) for comparison to those reported from the Modeler workflow (blue). The two sets of empirical characteristic functions cannot be individually distinguished because they are nearly identical. Transform variable values are multiplied by a factor of 10^5.

Reproducible Stochastic Simulation in Select Domains of Interest

Finally, we performed a preliminary survey of the reproducibility of stochastic simulation results in select modeling domains of broad interest (pandemic epidemiology, finance, and physics-informed neural networks). In all cases, we first attempted to qualitatively reproduce reported results, and then assessed the extent to which reported results can be reliably reproduced according to the EFECT error.

We selected six published stochastic differential equation models of the COVID-19 pandemic (38–43) and attempted to reproduce 14 reported figures (Table S2, Source Code S3.32-S.37). Of these, we could qualitatively reproduce four figures and another four figures with corrections but could not reproduce the other six. Only one figure reported results from a sample of sufficient size to produce an EFECT error on the order of our *ad hoc* EFECT convergence point of 0.075 (sample size 50,000 reported, computed EFECT error less than 0.05 for all experiments with a sample size of 10,000). For all attempted figures, we were able to generate reproducible results with EFECT error around 0.075 or less with a sample size of 5,000. One study (42) reported details on parameter fitting to data with a sample size of 20. When evaluating the reproducibility of results using the model and fitted parameters from this study, we found that sample sizes as great as 1,000 allow selecting any value for one of the fitted parameters within its bounds without producing detectable differences. We also found that a sample size of 10,000 would have made parametric differences of at least 35% detectable.

We assessed the reproducibility of results from demonstrations of financial model parameterization using Maximum Likelihood estimation with Efficient Importance Sampling (44). In summary, the approach uses adapted Markov Chain Monte Carlo (MC) sampling to estimate marginal parameter distributions for a given model and target data, a common procedure in Bayesian inference (45). We were able to qualitatively reproduce results from two Stochastic Volatility models (Source Code S3.38). We then evaluated the reproducibility of results using fitted parameters for both models, which were determined using a MC sample size of 50. With this sample size, we found that we could increase or decrease each fitted parameter by two to three standard errors without producing a detectable difference in results, whereas sample sizes of 5,000 to 10,000 would have made all parametric differences within one standard error detectable.

Last, we assessed the ability to adapt EFECT for evaluating reproducibility of partial differential equation (PDE) results during stochastic sampling when performing uncertainty quantification in physics-informed neural networks (PINNs, (46)). In summary, PINNs use deep neural networks to efficiently solve PDEs, uncertainty quantification of which requires training on results from stochastic sampling of PDE model parameters. EFECT can support such approaches for PDEs that are discretized using a fixed grid, where the PDE measurement at each grid point is treated like a model variable. We could qualitatively reproduce results in (46) and computed an EFECT error of 0.170 ± 0.0156 for the reported sample size of 1,000 used in uncertainty quantification, and of 0.0538 ± 0.00516 for a sample size of 10,000 (Source Code S3.39). We tested whether we could detect differences in four model parameters and found that the reported sample size allows decreasing one model parameter by 50%, or increasing it by

100%, without producing detectable differences in results, whereas a sample size of 10,000 allows detection of both differences.

Discussion

EFECT is agnostic to modeling methodology, broadly supports analysis of stochastic simulation results, and is readily computable. Its methodological constraints are that evaluated results must be representable as vectors of evaluations of bounded, real-valued random variables. The interpretability of the EFECT error is independent of modeling methodology, ensuring that reproducibility metrics are interpretable and comparable across diverse model types. Our workflow produces a minimal data set supporting reproducible simulation without requiring storage of all simulation data (which would be required for comparing distributions via the Kolmogorov-Smirnov statistic and comparable), a major advantage for promoting reproducible stochastic simulation as a broadly accepted hallmark of credible stochastic modeling. While SBML and SED-ML standardize model structure and simulation procedures, EFECT uniquely addresses statistical reproducibility of simulation outcomes. Our SSR workflow with EFECT enables testing for reproduced simulation results according to a significance level, which could be determined *a priori* through consensus in various modeling communities with different expectations concerning tradeoffs in accuracy and computational cost.

We evaluated the capability of EFECT to detect differences in stochastic simulation results due to a variety of causes. Among those tested, a structural difference between models is easiest to detect, and a single parametric difference is hardest to detect. The degree to which a parametric difference can be detected in stochastic simulation results depends on the sensitivity of the assessed model outputs to the parameter. This observation is deeply connected to the structural identifiability of a model and its parameters. It naturally follows that structurally nonidentifiable models permit scenarios where different parameter sets can produce indistinguishable outputs (47). Being a data-driven approach, EFECT cannot address identifiability issues, which are better reserved for model analysis techniques (48,49). Rather, EFECT provides a statistical foundation and workflow for modeling communities to develop standards and best practices for reproducible stochastic simulation results.

The observed power law of EFECT error vs. sample size when testing reproducibility (Figure 1D) is an expected consequence of empirical-characteristic-function sampling rather than an artefact of the selected ECF domain. Since the ECF is the average of a bounded complex-valued random variable (defined in [2] below), its fluctuation amplitude when sampling from the same distribution shrinks at a rate proportional to the inverse square root of the sample size (by the Central Limit Theorem). Hence, a reproducibility test with EFECT can have arbitrary statistical power at the expense of computational cost. While this cost may not be a limitation when using an inexpensive modeling methodology like ODEs, EFECT quantifies distributions at specified times, which excludes simulation methods that samples time. We consider such outputs as outside of the scope of EFECT, which is concerned with quantify differences in distributions at given measurement times.

SBML Test Suite has served the SBML community for verifying implementations of the Gillespie Stochastic Simulation Algorithm for nearly twenty years. EFECT can serve as the foundation for similar verification of simulators implementing other stochastic simulation algorithms. To this end, the EFECT report can be incorporated into the SED-ML standard. For example, in reference to SED-ML Level 1 Version 4, data for each model variable can be taken

from DataSet entries specified in SEDBase::Output::Report::ListOfDataSets, where variable objects of corresponding data generators are scalar and output data includes simulation times. Sample size can be specified through a RepeatedTask. Parameter sampling (if any) can be specified through a RepeatedTask (*i.e.*, via SetValue) for all distributions supported by SED-ML (*i.e.*, those supported by MathML). Thus, specifying a SED-ML script for data generation during a Curator's workflow naturally follows and could be automated using SED-ML library implementations (e.g., libSEDML in C++, Java, Python, and other programming languages). It would also naturally follow that EFECT could be made broadly available as a public service for stochastic simulation verification and reproducibility in BioSimulators.

The computational cost of EFECT can be non-negligible when testing for reproducibility, depending on sample size and variance of results. With the prototype implementation in Python used in this work, tests for reproducibility in all test cases ranged over orders of seconds to hours. Furthermore, the sample sizes required to produce reproducible results may be prohibitive for computationally expensive simulations, though various modeling communities can decide what level of accuracy (*i.e.*, a significance level) is appropriate for the typical computational cost of producing simulation results. Future work should develop infrastructure supporting usage of EFECT through our SSR workflow, including a markup language, publicly distributed libraries in popular scientific programming languages (*e.g.*, C++, Python, R, Julia), hardware-accelerated algorithms, and online web services. Related strategies have been previously demonstrated for accelerating ECF computations (50). To this end, we have started such a software project, called "libSSR" (github.com/tjsego/libSSR), which is intended to provide such infrastructure in anticipation of future standardization efforts on stochastic simulation reproducibility. Current work is providing support in Python. Early work in application with libSSR has demonstrated supporting reproducibility studies (Source Code S3.40, S3.41, S3.42) with COPASI (51), libRoadRunner (36), the Multiscale Object Oriented Simulation Environment (52), and SimBio (53). We have developed an additional package, "mkstd" (github.com/dilpath/mkstd), which we used to implement the EFECT report. This implementation enables the reading, writing, and validation of EFECT reports in any of the HDF5, JSON, XML, and YAML file formats, independently of "mkstd", "libSSR", or Python. Future work should also define the computations and minimal necessary data to recover the probability distribution functions encoded in ECFs according to our standard data, which could provide an efficient means to share statistical model results using an enhanced form of the EFECT Report.

Lastly, future work should promote community adoption of EFECT as a statistical reproducibility metric for stochastic simulation data by tools, public repositories, journals, modeling communities, and regulatory agencies. Importantly, EFECT is not intended to formalize all existing informal or formal uses of "reproducibility" or "replicability" across domains. Rather, it provides communities with capability to rigorously assess the statistical reproducibility of stochastic simulation results. EFECT, by using distributional statistics, offers an operational strategy for evaluating whether independent implementations produce consistent results from the same model specification and inputs, or whether independent input data produce consistent results from the same model specification and implementation. Hence, EFECT is applicable across different definitions of "reproducibility" and "replicability", extending beyond the traditional reproducibility assessment that requires using the same random number generator seed.

Simulation tools could provide built-in features that perform our Modeler's workflow. Modeling communities could develop standard test cases like those performed here to develop

consensus on EFECT convergence points that define sufficiently reproduced results. Such was how we chose the *ad hoc* convergence point used in this work, which produced marginal false positives (e.g., Figure 1E) in a reasonable amount of time for the modeling methodologies we primarily used. Communities that use expensive or exceptionally stochastic simulations may be inclined to adopt more forgiving criteria for reproducibility. Public repositories could curate stochastic models according to EFECT convergence points and report EFECT errors from testing for reproduced results. Journals could promote or even require including our standard data as supplementary information accompanying publication of stochastic models (3,16). Regulatory agencies could define what EFECT convergence point must be met, or proven, for stochastic simulation results to be considered valid evidence supporting applications and guidance on policymaking.

Methods

We test for equality in distribution of two stochastic processes $X(t)$ and $Y(t)$ by testing whether the two stochastic processes have the same unknown probability distribution $f(x;\ t)$ (*i.e.*, $X(t) \overset{\mathcal{D}}{=} Y(t)$ if and only if $X, Y \sim f$). Our test is performed over an arbitrary set of simulation times on samples produced from repeated execution of stochastic simulation without regard for determining, or performing any characterization of, the underlying distribution(s). Rather, our method relies on the limit behavior of empirical distributions. Specifically, our method relies characteristic functions (CFs). For a stochastic process $X(t)$, the CF $\phi_X(\tau; t)$ is defined as

$$\phi_X(\tau; t) = \mathbb{E}\left[e^{iX(t)\tau}\right]. \qquad [1]$$

Here τ is a transform variable and i is the imaginary unit. We built our method around CFs because of specific useful properties of a CF: 1. The magnitude of all CFs is in [0,1], which allows assessment of model variables without case-specific decision-making to handle their statistical features (*e.g.*, means at different orders of magnitude); 2. A CF supports all bounded, real-valued random variables, whether discrete or continuous, whereas the probability distribution function only supports continuous random variables; and 3. The empirical characteristic function (ECF) is naturally computed from a sample of stochastic simulation results (54). For a sample $\{X_i(t)\}_{i=1}^n$ of size n of $X(t)$ at time t, the ECF $\phi_{X,n}(\tau; t)$ is defined as

$$\phi_{X,n}(\tau; t) = \frac{1}{n} \sum_{1 \le j \le n} e^{iX_j(t)\tau} \qquad [2]$$

We use Lévy's Continuity Theorem to test for equality in distribution of simulation results: for samples $\{X_i(t)\}_{i=1}^n$ of $X(t)$ and $\{Y_i(t)\}_{i=1}^n$ of $Y(t)$, $\{X_i(t)\}_{i=1}^n \overset{\mathcal{D}}{\to} X(t)$ if and only if $\lim_{n\to\infty} \phi_{X_n}(t) = \phi_X(t)$, and likewise for $\{Y_i(t)\}_{i=1}^n$ and $Y(t)$. For two sampled random processes $X(t)$ and $Y(t)$ over $t \in \mathcal{T}$, we test for pointwise convergence of the characteristic functions $\phi_X(\tau; t)$ and $\phi_Y(\tau; t)$ over $\tau \in \mathcal{t}$ to the same unknown distribution using the EFECT error δ,

$$\delta\left(\phi_{X,n}(\tau; t), \phi_{Y,n}(\tau; t)\right) = \sup\left\{\left|\phi_{X,n}(\tau; t) - \phi_{Y,n}(\tau; t)\right| : \tau \in \mathcal{t}, t \in \mathcal{T}\right\}. \qquad [3]$$

It naturally follows that $X(t) \overset{\mathcal{D}}{=} Y(t)$ if and only if $\delta\left(\phi_{X,n}(\tau; t), \phi_{Y,n}(\tau; t)\right) \to 0$ as $n \to \infty$. When

comparing two samples of a model, we consider the maximum EFECT error of all stochastic variables of the model as the EFECT error of the two samples.

Methodological Details

Our test for reproducibility evaluates the likelihood that a sample would be assessed as reproduced by another sample of the same model according to comparison of generated ECFs. A sample of size $2N$ is tested for reproducibility by sampling the EFECT error [3] via random selection and evaluation of two $N-$ combinations. If the sample produces statistics of the EFECT error that satisfy some *a priori* criterion, an EFECT convergence point, then the sample is considered sufficiently reproducible. While we intend in this work to enable consensus building as to what criterion should be considered broadly acceptable, generally an EFECT error closer to zero produces more reliably reproducible results. Tests thus far suggest that requiring an EFECT error mean below 0.05 may be unnecessarily demanding to detect differences in samples that may not be significant to some modeling communities (Figure 1).

The CF has support for the transform variable (*i.e.*, τ in [1] and [2]) over the real line. Since the intent of the EFECT report is to encode sufficient information to test whether the sample represented by the EFECT report is sufficiently reproduced by another sample, we define a parameterization of the domain of ECFs to produce representative evaluations to test for equality. Specifically, choosing an excessively small domain produces evaluations that are all approximately equal to one, while choosing an excessively large domain can produce evaluations that are equal to either one (*i.e.*, when $\tau = 0$) or zero elsewhere (since, by Riemann-Lebesgue theorem, $\phi_X(\tau;t) \to 0$ as $|\tau| \to \infty$ if X has a density, of which $\phi_X(\tau;t)$ is the Fourier dual). We rewrite [1] to produce a form for parameterizing the domain over which to evaluate an ECF,

$$\phi_X(\tau;t) = \theta(\tau;t)\sqrt{\left(\mathbb{E}\left(\cos\left(Z(t)\sigma\left(X(t)\right)\tau\right)\right)\right)^2 + \left(\mathbb{E}\left(\sin\left(Z(t)\sigma\left(X(t)\right)\tau\right)\right)\right)^2}, \qquad [4]$$
$$\sigma\left(X(t)\right)Z(t) = X(t) - \mu\left(X(t)\right).$$

Here $\theta(\tau;t)$ is the phase of $\phi_X(\tau;t)$, the details of which are unimportant here except to note that $|\theta(\tau;t)| = 1$, and $Z(t)$ is standardized $X(t)$ with mean $\mu\left(X(t)\right)$ and standard deviation $\sigma\left(X(t)\right)$. Oscillations in the amplitude of $\phi_X(\tau;t)$ are expected to occur on the scale of periods inversely proportional to the standard deviation of $X(t)$ (since $Z(t)$ is standardized), hence we evaluate an ECF over m periods inversely proportional to the standard deviation,

$$\tau \in \left[0, \frac{2\pi m}{\sigma\left(X(t)\right)}\right], \qquad m \geq 1. \qquad [5]$$

In application, we evaluate ECFs over a uniform distribution of the transform variable according to [5] for a given value of m. All results in this work were performed with a value of m equal to 3 and 100 uniformly distributed values of the transform variable, which were determined *ad hoc* by inspection of results. Generally, greater values of m for the same number of evaluations risks loss of sufficient encoding, and greater numbers of evaluations increases the storage requirements of the EFECT report. We make no formal recommendation here as to what values for these parameters should be broadly acceptable except to note that preliminary tests suggest that a value of m equal to one may be sufficient for a number of evaluations on the order of 100,

whereas increasing the number of evaluations nearer to 1,000 may provide marginal improvements to the quality of encoding for at least m equal to three (Source Code S3.28, S3.29).

Testing for reproduced results is performed using the null hypothesis that the EFECT error from comparing Modeler and Curator ECFs was taken from the same distribution as the EFECT error from the Curator's test for reproducibility. Assessment of whether results have been reproduced is performed according to an *a priori* significance level using Chebyshev's inequality for unknown population mean and variance (55). For Curator's and Modeler's stochastic processes $X(t)$ and $Y(t)$, respectively, a Curator performs the test for reproducibility with their model implementation and generates a sample of EFECT error δ_{XX} with sample size n, mean $\bar{\delta}_{XX}$, and variance $\tilde{\delta}_{XX}$. The Curator then generates an EFECT error δ_{XY} by comparing the ECF(s) of their sample to that(those) reported by the Modeler and calculates the probability that δ_{XY} was taken from the same distribution as δ_{XX},

$$\Pr\left(\left|\delta_{XX}-\bar{\delta}_{XX}\right| \geq \left|\delta_{XY}-\bar{\delta}_{XX}\right|\right) = \frac{1}{n+1}\left\lfloor \frac{n+1}{n}\left(\frac{n^2-1}{n}\frac{\tilde{\delta}_{XX}}{\left(\delta_{XY}-\bar{\delta}_{XX}\right)^2}+1\right)\right\rfloor. \qquad [6]$$

Hence, the Curator concludes whether they have reproduced the Modeler's results via testing of the null hypothesis for a significance level α,

$$X \overset{\mathcal{D}}{=} Y \leftrightarrow \Pr\left(\left|\delta_{XX}-\bar{\delta}_{XX}\right| \geq \left|\delta_{XY}-\bar{\delta}_{XX}\right|\right) \geq \alpha. \qquad [7]$$

Implementation Details

ODEs were executed with libRoadRunner v2.4.0, except those described as being executed in COPASI v4.42. Stochastic ODE simulation with libRoadRunner was performed using the libRoadRunner implementation of the Gillespie stochastic simulation algorithm. ODEs executed in libRoadRunner were specified in Antimony v2.13.4 (56) to generate SBML specification, except those imported from BioModels. ODEs executed in COPASI were specified using generated SBML from Antimony specification. Parameter sampling for ODE simulation with libRoadRunner was performed using methods provided in SciPy v1.11.3 (57). The GSW ABM NetLogo implementation was executed in NetLogo v6.1.1 (September 26, 2019; Source Code S4.2); the MATLAB implementation was executed in MATLAB 2020b (Source Code S4.1); the Python implementation was executed in Python v3.10.13 (Source Code S4.3). All ECF generation and analysis was performed using library functions available in NumPy v1.26.0 (58). The stochastic Boolean network model was executed in the MaBoSS deployment in CompuCell3D v4.5.0 (59). Source code to perform all tests and analyses in Python and simulations with libRoadRunner and MaBoSS are available in Source Code S2. Source code to perform data generation with COPASI is available in Source Code S5. Source code to install all Python dependencies is available in Source Code S1. All project source code for this work is available at https://github.com/tjsego/ssr_project_2024. Instructions to install Python dependencies and execute Python source code and Jupyter Notebooks are available in Instruction S1. The executable Jupyter notebooks are also available in the public repository's notebook directory; rendered versions are included in the supplementary file.

Acknowledgments

TJS acknowledges funding from National Institutes of Health grant number U24EB028887 and National Science Foundation grant number 2000281. LLF acknowledges funding from National Institutes of Health grants number R01 GM127909 and R01 AI135128, and Defense Advanced Research Projects Agency grant number HR00112220038. ACK acknowledges funding from National Science Foundation grant number 2325776. KT, HH and RMS acknowledge EMBL Core funding. HMS acknowledges funding from National Institutes of Health grant numbers U24EB028887 and P41EB023912. RCL acknowledges funding from National Institutes of Health grants number R01 GM127909, R01 AI135128, and R01 HL169974, and Defense Advanced Research Projects Agency grant number HR00112220038. SR acknowledges funding from the Kavli Foundation under CZI EOSS Cycle-6 grant number LS-2024-GR-34-2950. HEG and MS acknowledge funding from the Universidad de. Buenos Aire and CONICET. FTB was supported LIBIS, and by de.NBI, the German Network for Bioinformatics Infrastructure (W-de.NBI-016). DP acknowledges funding from the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under the project ID 432325352 – SFB 1454, and from the University of Bonn, Germany (via the Schlegel Professorship of Prof. Dr. Jan Hasenauer).

The authors would like to thank Lutz Brusch, Randy Heiland, Jürgen Pahle, Sven Sahle, and Lucian Smith for their insightful feedback during reproducibility workshop discussions at the COMBINE 2023 (University of Connecticut), HARMONY 2024 (University College London), COMBINE 2024 (University of Stuttgart), and HARMONY 2025 (Katholieke Universiteit Leuven) events.

Competing Interests

The authors declare no competing financial or non-financial interests.

Data Availability

All reported data are included as supplementary materials.

Code Availability

All reported software are included as supplementary materials.

References

1. Baker M. 1,500 scientists lift the lid on reproducibility. Nature. 2016 May 26;533(7604):452-454 (GSM 2024 h5-index 488).

2. Stodden V, Seiler J, Ma Z. An empirical analysis of journal policy effectiveness for computational reproducibility. Proc Natl Acad Sci USA. 2018 Mar 13;115(11):2584–9.

3. Peng RD. Reproducible research in computational science. Science. 2011 Dec 2;334(6060):1226–7.

4. Bergmann FT, Sauro HM. Comparing simulation results of SBML capable simulators. Bioinformatics. 2008 Sep 1;24(17):1963–5.

5. Malik-Sheriff RS, Glont M, Nguyen TVN, Tiwari K, Roberts MG, Xavier A, et al. BioModels-15 years of sharing computational models in life science. Nucleic Acids Res. 2020 Jan 8;48(D1):D407–15.

6. Shaikh B, Smith LP, Vasilescu D, Marupilla G, Wilson M, Agmon E, et al. BioSimulators: a central registry of simulation engines and services for recommending specific tools. Nucleic Acids Res. 2022 Jul 5;50(W1):W108–14.

7. Courtot M, Juty N, Knüpfer C, Waltemath D, Zhukova A, Dräger A, et al. Controlled vocabularies and semantics in systems biology. Mol Syst Biol. 2011 Oct 25;7:543.

8. Keating SM, Waltemath D, König M, Zhang F, Dräger A, Chaouiya C, et al. SBML Level 3: an extensible format for the exchange and reuse of biological models. Mol Syst Biol. 2020 Aug;16(8):e9110.

9. Waltemath D, Adams R, Bergmann FT, Hucka M, Kolpakov F, Miller AK, et al. Reproducible computational biology experiments with SED-ML--the Simulation Experiment Description Markup Language. BMC Syst Biol. 2011 Dec 15;5:198.

10. McDougal RA, Bulanova AS, Lytton WW. Reproducibility in computational neuroscience models and simulations. IEEE Trans Biomed Eng. 2016 Mar 8;63(10):2021–35.

11. Kirouac DC, Cicali B, Schmidt S. Reproducibility of quantitative systems pharmacology models: current challenges and future opportunities. CPT Pharmacometrics Syst Pharmacol. 2019 Apr;8(4):205–10.

12. Schaduangrat N, Lampa S, Simeon S, Gleeson MP, Spjuth O, Nantasenamat C. Towards reproducible computational drug discovery. J Cheminform. 2020 Jan 28;12(1):9.

13. Tiwari K, Kananathan S, Roberts MG, Meyer JP, Sharif Shohan MU, Xavier A, et al. Reproducibility in systems biology modelling. Mol Syst Biol. 2021 Feb;17(2):e9982.

14. Sandve GK, Nekrutenko A, Taylor J, Hovig E. Ten simple rules for reproducible computational research. PLoS Comput Biol. 2013 Oct 24;9(10):e1003285.

15. Porubsky VL, Goldberg AP, Rampadarath AK, Nickerson DP, Karr JR, Sauro HM. Best practices for making reproducible biochemical models. Cell Syst. 2020 Aug 26;11(2):109–20.

16. Mendes P. Reproducible research using biomodels. Bull Math Biol. 2018 Dec;80(12):3081–7.

17. Medley JK, Goldberg AP, Karr JR. Guidelines for reproducibly building and simulating systems biology models. IEEE Trans Biomed Eng. 2016 Oct;63(10):2015–20.

18. Goldenfeld N, Kadanoff LP. Simple lessons from complexity. Science. 1999 Apr 2;284(5411):87–9.

19. Casdagli M. Chaos and Deterministic *Versus* Stochastic Non-Linear Modelling. Journal of the Royal Statistical Society Series B: Statistical Methodology. 1992 Jan 1;54(2):303–28.

20. Friedrich R, Peinke J, Sahimi M, Reza Rahimi Tabar M. Approaching complexity by stochastic methods: From biological systems to turbulence. Physics Reports. 2011 Sep;506(5):87–162.

21. Fuller A, Fan Z, Day C, Barlow C. Digital twin: enabling technologies, challenges and open research. IEEE Access. 2020;8:108952–71.

22. Read MN, Alden K, Timmis J, Andrews PS. Strategies for calibrating models of biology. Brief Bioinformatics. 2020 Jan 17;21(1):24–35.

23. Stainforth DA, Aina T, Christensen C, Collins M, Faull N, Frame DJ, et al. Uncertainty in predictions of the climate response to rising levels of greenhouse gases. Nature. 2005 Jan 27;433(7024):403–6.

24. Goodman SN, Fanelli D, Ioannidis JP. What does research reproducibility mean? Sci Transl Med. 2016 Jun 1;8(341):1–6.

25. Evans TW, Gillespie CS, Wilkinson DJ. The SBML discrete stochastic models test suite. Bioinformatics. 2008 Jan 15;24(2):285–6.

26. Vargha A, Delaney HD. A Critique and Improvement of the*CL* Common Language Effect Size Statistics of McGraw and Wong. Journal of Educational and Behavioral Statistics. 2000 Jun;25(2):101–32.

27. Reiczigel J, Zakariás I, Rózsa L. A bootstrap test of stochastic equality of two populations. The American Statistician. 2005 May;59(2):156–61.

28. Wearing HJ, Rohani P, Keeling MJ. Appropriate models for the management of infectious diseases. PLoS Med. 2005 Jul 26;2(7):e174.

29. Gillespie DT. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J Comput Phys. 1976 Dec;22(4):403–34.

30. Liu Z, Pu Y, Li F, Shaffer CA, Hoops S, Tyson JJ, et al. Hybrid modeling and simulation of stochastic effects on progression through the eukaryotic cell cycle. J Chem Phys. 2012 Jan 21;136(3):034105.

31. Lo K, Denney WS, Diamond SL. Stochastic modeling of blood coagulation initiation. Pathophysiol Haemost Thromb. 2005;34(2–3):80–90.

32. Wilensky U, Reisman K. Thinking Like a Wolf, a Sheep, or a Firefly: Learning Biology Through Constructing and Testing Computational Theories—An Embodied Modeling Approach. Cogn Instr. 2006 Jun;24(2):171–209.

33. Calzone L, Tournier L, Fourquet S, Thieffry D, Zhivotovsky B, Barillot E, et al. Mathematical modelling of cell-fate decision in response to death receptor engagement. PLoS Comput Biol. 2010 Mar 5;6(3):e1000702.

34. Stoll G, Caron B, Viara E, Dugourd A, Zinovyev A, Naldi A, et al. MaBoSS 2.0: an environment for stochastic Boolean modeling. Bioinformatics. 2017 Jul 15;33(14):2226–8.

35. Kermack WO, McKendrick AG. A Contribution to the Mathematical Theory of Epidemics. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences. 1927 Aug 1;115(772):700–21.

36. Welsh C, Xu J, Smith L, König M, Choi K, Sauro HM. libRoadRunner 2.0: a high performance SBML simulation and analysis library. Bioinformatics. 2023 Jan 1;39(1).

37. Hoops S, Sahle S, Gauges R, Lee C, Pahle J, Simus N, et al. COPASI--a COmplex PAthway SImulator. Bioinformatics. 2006 Dec 15;22(24):3067–74.

38. Adak D, Majumder A, Bairagi N. Mathematical perspective of Covid-19 pandemic: Disease extinction criteria in deterministic and stochastic models. Chaos Solitons Fractals. 2021 Jan;142:110381.

39. Din A, Khan A, Baleanu D. Stationary distribution and extinction of stochastic coronavirus (COVID-19) epidemic model. Chaos Solitons Fractals. 2020 Oct;139:110036.

40. Faranda D, Alberti T. Modeling the second wave of COVID-19 infections in France and Italy via a stochastic SEIR model. Chaos. 2020 Nov;30(11):111101.

41. Mamis K, Farazmand M. Stochastic compartmental models of the COVID-19 pandemic must have temporally correlated uncertainties. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences. 2023 Jan;479(2269).

42. Niño-Torres D, Ríos-Gutiérrez A, Arunachalam V, Ohajunwa C, Seshaiyer P. Stochastic modeling, analysis, and simulation of the COVID-19 pandemic with explicit behavioral changes in Bogotá: A case study. Infect Dis Model. 2022 Mar;7(1):199–211.

43. Tesfaye AW, Satana TS. Stochastic model of the transmission dynamics of COVID-19 pandemic. Adv Differ Equ. 2021 Oct 18;2021(1):457.

44. Liesenfeld R, Richard J-F. Univariate and multivariate stochastic volatility models: estimation and diagnostics. Journal of Empirical Finance. 2003 Sep;10(4):505–31.

45. Shirley K. Inference from Simulations and Monitoring Convergence. In: Brooks S, Gelman A, Jones G, Meng X-L, editors. Handbook of markov chain monte carlo. Chapman and Hall/CRC; 2011.

46. Zhang D, Lu L, Guo L, Karniadakis GE. Quantifying total uncertainty in physics-informed neural networks for solving forward and inverse stochastic problems. J Comput Phys. 2019 Nov;397:108850.

47. Wieland F-G, Hauber AL, Rosenblatt M, Tönsing C, Timmer J. On structural and practical identifiability. Current Opinion in Systems Biology. 2021 Mar;

48. Doherty J, Hunt RJ. Two statistics for evaluating parameter identifiability and error reduction. J Hydrol (Amst). 2009 Mar;366(1–4):119–27.

49. Hines KE, Middendorf TR, Aldrich RW. Determination of parameter identifiability in nonlinear biophysical models: A Bayesian approach. J Gen Physiol. 2014 Mar;143(3):401–16.

50. O'Brien TA, Collins WD, Rauscher SA, Ringler TD. Reducing the computational cost of the ECF using a nuFFT: A fast and objective probability density estimation method. Comput Stat Data Anal. 2014 Nov;79:222–34.

51. Bergmann FT. BASICO: A simplified Python interface to COPASI. JOSS. 2023 Oct 15;8(90):5553.

52. Ray S, Deshpande R, Dudani N, Bhalla US. A general biological simulator: the multiscale object oriented simulation environment, MOOSE. BMC Neurosci. 2008;9(Suppl 1):P93.

53. Silberberg M, Hermjakob H, Malik-Sheriff RS, Grecco HE. Poincaré and SimBio: a versatile and extensible Python ecosystem for modeling systems. Bioinformatics. 2024 Aug 2;40(8).

54. Lukacs E. A survey of the theory of characteristic functions. Adv Appl Probab. 1972 Apr;4(1):1–37.

55. Kabán A. Non-parametric detection of meaningless distances in high dimensional data. Stat Comput. 2012 Mar;22(2):375–85.

56. Smith LP, Bergmann FT, Chandran D, Sauro HM. Antimony: a modular model definition language. Bioinformatics. 2009 Sep 15;25(18):2452–4.

57. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat Methods. 2020 Mar;17(3):261–72.

58. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. Array programming with NumPy. Nature. 2020 Sep 16;585(7825):357–62.

59. Swat MH, Thomas GL, Belmonte JM, Shirinifard A, Hmeljak D, Glazier JA. Multi-scale modeling of tissues using CompuCell3D. Methods Cell Biol. 2012;110:325–66.