GLEAM-AI: Neural Surrogate for Accelerated Epidemic Analytics and Forecasting

Mohammadmehdi (Mehdi) Zahedi
Network Science Institute
Northeastern University
Boston, MA 02115,
The Roux Institute
Northeastern University
Portland, ME 04101,
zahedi.m@northeastern.edu

Dongxia Wu
University of California, San Diego
La Jolla, CA 92093
dowu@ucsd.edu

Jessica T. Davis
Network Science Institute
Northeastern University
Boston, MA 02115,
jes.davis@northeastern.edu

Yi-An Ma
University of California, San Diego
La Jolla, CA 92093
yianma@ucsd.edu

Alessandro Vespignani
Science Institute
Northeastern University
Boston, MA 02115
A.Vespignani@northeastern.edu

Rose Yu
University of California, San Diego
La Jolla, CA 92093
roseyu@ucsd.edu

Matteo Chinazzi
Network Science Institute
Northeastern University
Boston, MA 02115,
The Roux Institute
Northeastern University
Portland, ME 04101
m.chinazzi@northeastern.edu

Abstract

Large-scale stochastic mechanistic models are more and more used in recent years to model global epidemic outbreaks and are useful tool for policy and decision makers to project, forecast, and asses the impact of epidemics. However, these models are incredibly demanding from a computational standpoint and time-consuming to run. Here, we present GLEAM-AI, a spatio-temporal probabilistic neural surrogate model to replicate the insights and results of large-scale mechanistic epidemic models and to accelerate stochastic simulations. We show how a surrogate model can efficiently be trained with less than 5.3 thousand simulations from the mechanistic model by utilizing a Bayesian active learning approach. We demonstrate its performance in efficiently replicating the mechanistic dynamics, providing approximately a 200 times speed-up with respect to the original model.

D3S3: Data-driven and Differentiable Simulations, Surrogates, and Solvers @ NeurIPS 2024.

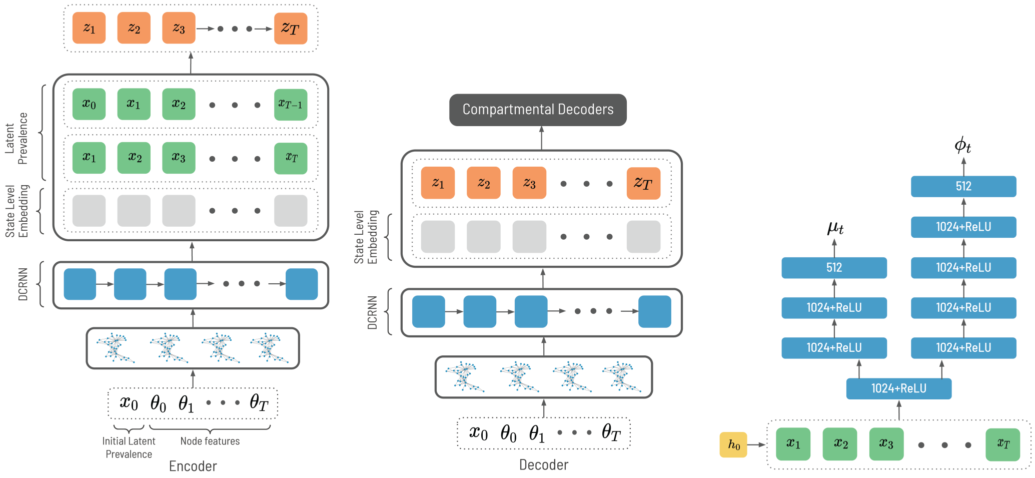
1 Introduction

Large-scale, stochastic, age-structured, meta-population epidemic models are computationally expensive to run, especially when a lot of uncertainty exists on the characteristics of the outbreak under investigation and a large portion of the model's parameter space has to be explored to provide accurate forecasts or scenario analysis projections. In this context, utilizing a deep surrogate model in lieu of a mechanistic model can help accelerate stochastic simulations and reduce the computational cost and time needed to model complex disease dynamics, calibrate large-scale models, or simply provide timely insights on the evolution of an epidemic [1–4]. In this work, we extend a stochastic neural surrogate modeling approach based on spatio-temporal neural processes (STNP) [1] to reproduce the dynamics of the Global Epidemic and Mobility model (GLEAM) [5–8], a data-driven, stochastic, spatial, age-structured, multi-strain, meta-population epidemic model that incorporates high resolution population density data, mobility data, and disease dynamics into a unified framework. In particular, we focus our experiments on replicating the dynamics of a model tailored to reproduce the evolution of seasonal flu outbreaks in the United States. Our preliminary results show that GLEAM-AI can be efficiently trained to reproduce GLEAM model's dynamics using only about 5.3 thousands simulations.

2 Method

Disease dynamics are modeled using a traditional compartmental modeling framework. In particular, in GLEAM, seasonal flu disease dynamics are simulated using a compartmental model that divides the population into: susceptible, latent, infectious, hospitalized, and recovered, where the infectious population is further subdivided into asymptomatic, infectious traveling (e.g. mild cases of the disease), and symptomatic non-traveling (e.g. severe cases). Lastly, all individuals can be either vaccinated or unvaccinated. The details of the compartmental model used are reported in Fig.3, Appendix B. The surrogate model, GLEAM-AI, aims at reproducing the dynamics of two of these compartments, latent and hospitalized. I.e., number of infected individuals and number of hospitalized individuals over time, respectively. Details about the compartmental model and GLEAM are presented in Appendix A.

The proposed surrogate model is based on spatio-temporal neural process (STNP) [1], a generalization of neural process [9–12]. GLEAM-AI comprises of an encoder-decoder architecture, as illustrated in Fig.1. The input to both the encoder and decoder is a mobility graph of the United States where each node represents a state and weighted edges describe long-range traveling patterns across the country. State level node features comprise: basic reproduction number, initial disease prevalence, residual immunity, population, the starting date of the outbreak, expressed as the number of days from January 1$^{\text{st}}$ of the corresponding year, and a seasonality forcing factor. The basic reproduction number R_0 measures how many new cases one infected person is expected to cause in a fully susceptible population. Initial prevalence indicates the initial size of the population in that compartment that has those attributes; for instance, latent prevalence are the number of individuals that are in latent compartment at the start of the simulation. Residual immunity represents the remaining protection against the disease after previous exposure or vaccination. The encoder processes the input graph using a Diffusion Convolutional Recurrent Neural Network (DCRNN) [13]. The resulting graph embeddings are concatenated with the latent prevalence time series and its lagged version, following the approach in neural processes [9], and then fed into a GRU to generate the time-dependent latent variable. The latent prevalence time series in the encoder is normalized using z-score normalization. The decoder uses the temporal graph embedding from DCRNN and latent variable from encoder to generate the samples. The DCRNN, used in both the encoder and decoder, has an input channel of 51 and an output dimension of 256. The latent variable has a dimension of 100. The compartmental decoders are separate structures, each consisting of GRU layers followed by fully connected layers, designed to learn the parameters for each compartment. Fig. 1c shows the compartmental decoder parameterized by a negative binomial distribution. In our experiment, we used a single-layer GRU for each compartmental decoder. The decoder's output has as many heads as there are compartments being generated. For flu, for example, we generate latent incidence, latent prevalence, hospitalized incidence, and hospitalized prevalence, meaning the decoder has four heads. GLEAM-AI extends STNP [1] original framework along several dimensions. First, we redesigned the encoder to improve efficiency by using only the latent prevalence time series, i.e. the number of currently infected individuals at a given time t, to learn the latent embedding space, as that is sufficient, conditional on knowing the initial condition of the model, to learn the disease dynamics encoded in GLEAM. Second, the decoder has been redesigned to map the output to generate all the compartments of interest, such as latent prevalence, latent incidence (i.e. new daily infections), hospitalized prevalence, and hospitalized incidence (i.e. new daily hospital admissions). Third, we designed the decoder to learn the initial states h_0 of the gated recurrent units (GRUs). In GLEAM, only initial values for prevalence are known at the onset of the simulation, i.e. number of people currently members of each specific compartment, while initial values for incidence compartments are not known a priori. By learning initial states of GRU we give the model enough flexibility to capture dynamic of compartments not used by the encoder. Fourth, we explicitly provide as a time-varying node parameters, a temporal embedding describing the physics of the seasonal forcing dynamics that modulate disease transmission during the year by rescaling the effective reproductive number, $R(t)$ [14]. Here, we adopted the same functional form for the seasonality forcing factor as used in GLEAM (see Eq. 1). Lastly, we experimented with different parametric distributions to parameterize the decoder, including the Gaussian distribution and the negative binomial distribution. Since the time series of each output compartment consists of count data, the negative binomial distribution is a natural choice as it can capture both the mean and dispersion simultaneously. So instead of point estimates, the decoder outputs distribution $p(\Phi_t|\Phi_{(t-1):1}, z_{(t-1):1})$, where Φ_t is any parameter of the distribution conditioned on their past values and the latent variables.



(a) (b) (c)

Figure 1: GLEAM-AI architecture. (a) encoder, (b) decoder, (c) compartmental decoder. The input to both the encoder and decoder is a mobility graph of the United States where each node represents a state and weighted edges describe long-range traveling patterns across the country. State-level node features comprise: basic reproduction number, initial disease prevalence, residual immunity, population, the starting date of the outbreak, and a seasonality forcing factor.

We employ Bayesian active learning [15–17, 1] to train GLEAM-AI in a sample-efficient way. Define $S^{(i)}$ as train dataset at iteration i. First, we generate 120 random simulations based on random inputs to serve as the initial training dataset $S^{(1)}$, followed by alternating between acquiring new samples and training the model for 120 epochs. At every training phase, we use early stopping based on train loss to terminate the training. To select the input parameters for generating additional simulations, we predict $\hat{x}_{1:T}$ on a predefined range of input parameters and using the acquisition function $r(.)$ to compute the expected score $E_{p(x_{1:T}|z_{1:T},\theta)}[r(\hat{x}_{1:T}|z_{1:T}, \theta)]$ for each input. Based on these scores, we query the simulator $F(\theta^{(i+1)}; \xi)$, where ξ represents the internal states of the simulator, to generate new samples. For the acquisition function, we used the maximum mean standard deviation [16] and query 60 samples from simulator at each acquisition phase. The loss function used in the model, derived from variational inference principles [1], is composed of the reconstruction loss and KL divergence, with the negative binomial likelihood used as the reconstruction loss. The algorithm is shown in Appending C.

3 Experiments

Here, we present some preliminary results validating the model by comparing the distributions generated by GLEAM and GLEAM-AI. In Fig. 2, we compare the ensembles of more than 13,000 stochastic realizations for different ranges of parameters for a sample of representative states.

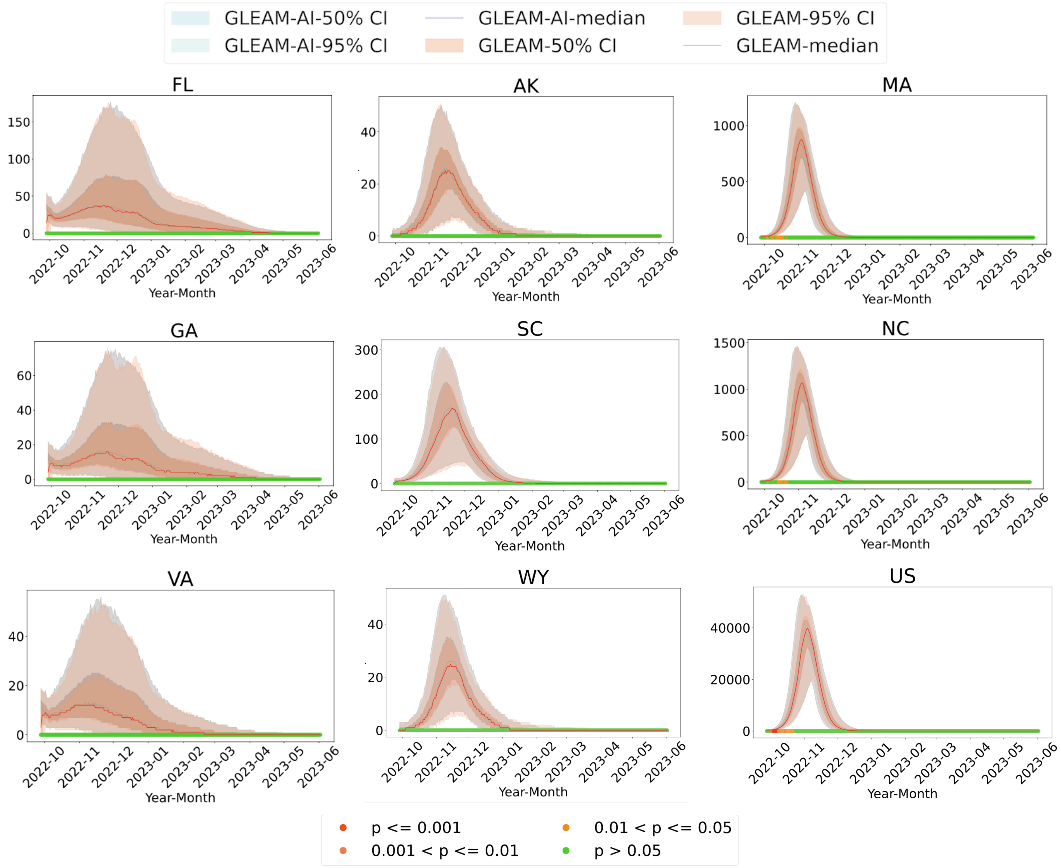


Figure 2: GLEAM vs GLEAM-AI: ensembles of the number of new daily flu hospital admissions (y-axis) over time for Alaska (AK), Florida (FA), Georgia (GA), Massachusetts (MA), North Carolina (NC), South Carolina (SC), Virginia (VA), Wyoming (WY) and United States (US). The solid lines are median, dark shades are 50% and light shades are 95% CI, respectively. The value of R_0 for the first column is $[1.75, 2.0]$, second column is $[2.25, 2.5]$, third column $[2.75, 3.0]$. The ranges for the other parameters are kept fixed across experiments and they are: seasonality forcing in $[0.8, 0.85]$, initial disease prevalence is $140/1000000$, starting dates of the outbreak in [2022-09-28, 2022-10-02], and residual immunity between 25%–30%. KS-test color-coded p-values comparing GLEAM and GLEAM-AI added at the bottom of each subfigure.

Despite some discrepancies in distribution for some parameter ranges, the speed gains in generating samples outweighs the drawbacks. Generating a synthetic epidemic outbreak from GLEAM-AI is approximately 200 times faster than generating the same simulation in GLEAM. As a reference, running 100 replicates of the compartmental model used here takes approximately 3 hours using GLEAM and less than a minute using GLEAM-AI. This increased efficiency in generating data samples constitutes an advantage during real-world applications that typically involve running a large number (e.g. hundreds of thousands) of computationally expensive simulations to then select those that best match observational data. Indeed, thanks to the speed of execution of GLEAM-AI, we can use Approximate Bayesian Computation coupled with Sequential Monte Carlo (ABC-SMC) to efficiently calibrate the surrogate model to observed surveillance data, rather than a simpler ABC approach, and therefore estimate the posterior distribution of the inputs parameters.

4 Conclusion

In this work, we introduced GLEAM-AI, a spatio-temporal neural surrogate model designed to accelerate simulation of large-scale epidemic models. Our approach utilizes a probabilistic framework that integrates spatio-temporal neural process and Bayesian active learning to reduce the computation cost and number of queries from a mechanistic model. Our results demonstrate that GLEAM-AI is a viable model for real-time forecasting and pandemic response.

References

[1] Dongxia Wu, Ruijia Niu, Matteo Chinazzi, Alessandro Vespignani, Yi-An Ma, and Rose Yu. Deep bayesian active learning for accelerating stochastic simulation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2559–2569, 2023.

[2] Reza Alizadeh, Janet K. Allen, and Farrokh Mistree. Managing computational complexity using surrogate models: a critical review. *Research in Engineering Design*, 31(3):275–298, April 2020. ISSN 1435-6066. doi: 10.1007/s00163-020-00336-7. URL `http://dx.doi.org/10.1007/s00163-020-00336-7`.

[3] Giovanni Charles, Timothy M. Wolock, Peter Winskill, Azra Ghani, Samir Bhatt, and Seth Flaxman. Seq2seq surrogates of epidemic models to facilitate bayesian inference. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(12):14170–14177, June 2023. ISSN 2159-5399. doi: 10.1609/aaai.v37i12.26658. URL `http://dx.doi.org/10.1609/aaai.v37i12.26658`.

[4] Theresa Reiker, Monica Golumbeanu, Andrew Shattock, Lydia Burgert, Thomas A. Smith, Sarah Filippi, Ewan Cameron, and Melissa A. Penny. Emulator-based bayesian optimization for efficient multi-objective calibration of an individual-based model of malaria. *Nature Communications*, 12(1), December 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-27486-z. URL `http://dx.doi.org/10.1038/s41467-021-27486-z`.

[5] Duygu Balcan, Bruno Gonçalves, Hao Hu, José J. Ramasco, Vittoria Colizza, and Alessandro Vespignani. Modeling the spatial spread of infectious diseases: The global epidemic and mobility computational model. *Journal of computational science*, 1 3:132–145, 2010. URL `https://api.semanticscholar.org/CorpusID:5102920`.

[6] Matteo Chinazzi, Jessica T. Davis, Marco Ajelli, Corrado Gioannini, Maria Litvinova, Stefano Merler, Ana Pastore y Piontti, Kunpeng Mu, Luca Rossi, Kaiyuan Sun, Cécile Viboud, Xinyue Xiong, Hongjie Yu, M. Elizabeth Halloran, Ira M. Longini, and Alessandro Vespignani. The effect of travel restrictions on the spread of the 2019 novel coronavirus (covid-19) outbreak. *Science*, 368(6489):395–400, 2020. doi: 10.1126/science.aba9757. URL `https://www.science.org/doi/abs/10.1126/science.aba9757`.

[7] Jessica T Davis, Matteo Chinazzi, Nicola Perra, Kunpeng Mu, Ana Pastore Y Piontti, Marco Ajelli, Natalie E Dean, Corrado Gioannini, Maria Litvinova, Stefano Merler, Luca Rossi, Kaiyuan Sun, Xinyue Xiong, Ira M Longini, Jr, M Elizabeth Halloran, Cécile Viboud, and Alessandro Vespignani. Cryptic transmission of SARS-CoV-2 and the first COVID-19 wave. *Nature*, 600(7887):127–132, December 2021.

[8] Matteo Chinazzi, Jessica T Davis, Ana Pastore Y Piontti, Kunpeng Mu, Nicolò Gozzi, Marco Ajelli, Nicola Perra, and Alessandro Vespignani. A multiscale modeling framework for scenario modeling: Characterizing the heterogeneity of the COVID-19 epidemic in the US. *Epidemics*, 47(100757):100757, June 2024.

[9] Marta Garnelo, Jonathan Schwarz, Dan Rosenbaum, Fabio Viola, Danilo J Rezende, SM Eslami, and Yee Whye Teh. Neural processes. *arXiv preprint arXiv:1807.01622*, 2018.

[10] Hyunjik Kim, Andriy Mnih, Jonathan Schwarz, Marta Garnelo, Ali Eslami, Dan Rosenbaum, Oriol Vinyals, and Yee Whye Teh. Attentive neural processes. *arXiv preprint arXiv:1901.05761*, 2019.

[11] Christos Louizos, Xiahan Shi, Klamer Schutte, and Max Welling. The functional neural process. *Advances in Neural Information Processing Systems*, 32, 2019.

[12] Gautam Singh, Jaesik Yoon, Youngsung Son, and Sungjin Ahn. Sequential neural processes. *Advances in Neural Information Processing Systems*, 32, 2019.

[13] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting, 2018.

[14] Duygu Balcan, Hao Hu, Bruno Goncalves, Paolo Bajardi, Chiara Poletto, Jose J Ramasco, Daniela Paolotti, Nicola Perra, Michele Tizzoni, Wouter Van den Broeck, Vittoria Colizza, and Alessandro Vespignani. Seasonal transmission potential and activity peaks of the new influenza A(H1N1): a monte carlo likelihood analysis based on human mobility. *BMC Med.*, 7(1):45, September 2009.

[15] Adam Foster, Desi R Ivanova, Ilyas Malik, and Tom Rainforth. Deep adaptive design: Amortizing sequential bayesian experimental design. In *International conference on machine learning*, pages 3384–3395. PMLR, 2021.

[16] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International conference on machine learning*, pages 1183–1192. PMLR, 2017.

[17] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.

A Global Epidemic and Mobility model

The Global Epidemic and Mobility (GLEAM) [5–8] is a stochastic, age-structured, meta-population epidemic model that divides the world into more than 3200 sub-populations across nearly 190 countries. Each subpopulation is centered around a major transportation hub, such as airports. Sub-populations are connected by the daily movement of individuals, capturing both short-range and long-range traveling. Long-range traveling captures international and domestic air traveling collected from the Official Aviation Guide database. Short-range traveling, in the form of ground mobility and commuting flows, are modeled from data collected from statistics offices of 30 countries on 5 continents. Lastly, the effects of seasonality are explicitly modeled by introducing a time-varying rescaling of the effective reproduction number defined as:

$$s_i(t) = \frac{1}{2}\left[\left(1 - \frac{\alpha_{\text{min}}}{\alpha_{\text{max}}}\right)\sin\left(\frac{2\pi}{365}(t - t_{\text{max},i}) + \frac{\pi}{2}\right) + 1 + \frac{\alpha_{\text{min}}}{\alpha_{\text{max}}}\right], \tag{1}$$

where i refers to the hemisphere considered, and $t_{\text{max},i}$ is the time corresponding to the maximum of the sinusoidal function. For the northern hemisphere it is fixed to January 15th.
Within each sub-population, the disease dynamics are simulated using a compartmental model.

B Compartmental Model

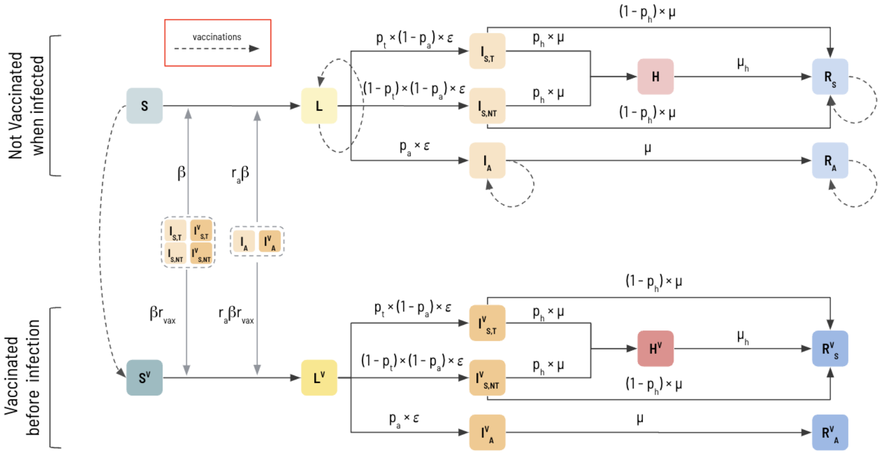


Figure 3: Compartmental model. Disease dynamics unfold as follows, people who are at risk of getting the disease start in the susceptible compartment (S). Infected individuals move into the latent compartment (L). On average, after a specified latent period of ε^{-1} days, individuals in the latent state become infectious. After an average infectious period of μ^{-1} days the fraction of population that is in infectious compartments transition to either the hospitalized compartment or to removed compartment. The other parameters are defined as follows: β denotes the transmissibility parameter; p_a denotes the probability of having a symptomatic infectious individual; conditional on an individual being symptomatic, p_t denotes the probability of the individual being marked as *traveling* in the simulations, instead with probability $1 - p_t$ individuals will be marked as *non-traveling* due to the severity of their illness; μ_h^{-1} denotes the average number of days spent in a hospital; r and r_a a transmissibility reduction factors applied to asymptomatic and vaccinated individuals, respectively.

C Bayesian Active Learning

Algorithm 1 Bayesian Active Learning

```
Input: Initial simulation dataset $S^1$
Train the model NP$^1(S^1)$
for  doi=1,2,···
    Learn $(z_1, z_2, , z_T) \sim q^{(i)}(z_{1:T}|x_{1:T}, \theta, S_i)$;
    Predict $(\hat{x}_1, \hat{x}_2, , \hat{x}_T) \sim p^{(i)}(x_{1:T}|z_{1:T}, \theta, S_i)$
    Select a batch $\theta_{(i+1)} \leftarrow$ argmax$_\theta$ $E_{p(x_{1:T}|z_{1:T},\theta)}[r(\hat{x}_{1:T}|z_{1:T},\theta)]$
    Simulate $x^{(i+1)}_{1:t} \leftarrow$ Query the simulator $F(\theta^{(i+1)};\xi)$
    Augment training set $S^{i+1} \leftarrow S^i \bigcup\{\theta^{(i+1)}, x^{(i+1)}_{1:T}\}$
    Update the model NP$^{(i+1)}(S^{i+1})$
end for
```