https://doi.org/10.1007/s11222-023-10222-6

ORIGINAL PAPER

Learning-based importance sampling via stochastic optimal control for stochastic reaction networks

Chiheb Ben Hammouda[1] · Nadhir Ben Rached[2] · Raúl Tempone[3,4] · Sophia Wiechert[1]

Received: 21 December 2021 / Accepted: 4 February 2023 / Published online: 28 March 2023

Abstract

We explore efficient estimation of statistical quantities, particularly rare event probabilities, for stochastic reaction networks. Consequently, we propose an importance sampling (IS) approach to improve the Monte Carlo (MC) estimator efficiency based on an approximate tau-leap scheme. The crucial step in the IS framework is choosing an appropriate change of probability measure to achieve substantial variance reduction. This task is typically challenging and often requires insights into the underlying problem. Therefore, we propose an automated approach to obtain a highly efficient path-dependent measure change based on an original connection in the stochastic reaction network context between finding optimal IS parameters within a class of probability measures and a stochastic optimal control formulation. Optimal IS parameters are obtained by solving a variance minimization problem. First, we derive an associated dynamic programming equation. Analytically solving this backward equation is challenging, hence we propose an approximate dynamic programming formulation to find near-optimal control parameters. To mitigate the curse of dimensionality, we propose a learning-based method to approximate the value function using a neural network, where the parameters are determined via a stochastic optimization algorithm. Our analysis and numerical experiments verify that the proposed learning-based IS approach substantially reduces MC estimator variance, resulting in a lower computational complexity in the rare event regime, compared with standard tau-leap MC estimators.

Keywords Stochastic reaction networks · Tau-leap · Importance sampling · Stochastic optimal control · Dynamic programming · Rare event

Mathematics Subject Classification 60H35 · 60J75 · 65C05 · 93E20

1 Introduction

We propose an approach to efficiently estimate statistical quantities, particularly rare event probabilities for a particular class of continuous-time Markov chains known as stochastic reaction networks (SRNs). Consequently, we develop a learning-based importance sampling (IS) algorithm to improve the Monte Carlo (MC) estimator efficiency based on an approximate tau-leap (TL) scheme. The automated approach is based on an original connection between optimal IS parameter determination within a class of probability measures and stochastic optimal control (SOC) formulation.

SRNs (see Sect. 1.1 for a short introduction and [9] for more details) describe the time evolution of biochemical reactions, epidemic processes [5,13], and transcription and translation in genomics and virus kinetics [32,48], among other important applications. For the current study, let $\mathbf{X}$ be an SRN that takes values in $\mathbb{N}^d$ and is defined in the time interval $[0, T]$, where $T > 0$ is a user-selected final time. We aim to provide accurate and computationally efficient MC estimations for the expected value $\mathbb{E}[g(\mathbf{X}(T))]$, where $g : \mathbb{N}^d \to \mathbb{R}$ is a scalar observable for $\mathbf{X}$. In particular, we

✉ Sophia Wiechert
wiechert@uq.rwth-aachen.de

[1] Chair of Mathematics for Uncertainty Quantification, RWTH Aachen University, Aachen, Germany

[2] School of Mathematics, University of Leeds, Leeds, UK

[3] Computer, Electrical and Mathematical Sciences and Engineering Division (CEMSE), King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia

[4] Alexander von Humboldt Professor in Mathematics for Uncertainty Quantification, RWTH Aachen University, Aachen, Germany

study estimating rare event probabilities with $g(\mathbf{x}) = \mathbf{1}_{\{\mathbf{x} \in \mathcal{B}\}}$ (i.e., the indicator function for a set $\mathcal{B} \subset \mathbb{R}^d$).

The quantity of interest, $\mathbb{E}[g(\mathbf{X}(T))]$, can be computed by solving the corresponding Kolmogorov backward equations [8]. For most SRNs, deriving a closed-form solution for these ordinary differential equations is infeasible, and numerical approximations based on discretized schemes are commonly used. However, the computational cost scales exponentially with the number of species d. Therefore, we are particularly interested in estimating $\mathbb{E}[g(\mathbf{X}(T))]$ using MC methods, an attractive alternative to avoid the curse of dimensionality.

Many schemes have been developed to simulate exact sample paths for SRNs, such as the stochastic simulation algorithm [25] and modified next reaction method [4]. Pathwise exact SRN realizations can incur high computational costs if any reaction channels have high reaction rates. Gillespie [26] and Aparicio and Solari [6] independently proposed the explicit TL method (see Sect. 1.2) to overcome this issue by simulating approximate paths of $\mathbf{X}$, evolving the process with fixed time steps and keeping reaction rates fixed within each time step. Various simulation schemes have been subsequently proposed to deal with situations incorporating well-separated fast and slow time scales [1,2,11,14,40,45].

Various variance reduction techniques have been proposed in the SRN context to reduce the computational work to estimate $\mathbb{E}[g(\mathbf{X}(T))]$. Several multilevel Monte Carlo (MLMC) [21,22] based methods have been proposed to address specific challenges in this context [3,10,11,38,40]. Furthermore, as naive MC and MLMC estimators fail to efficiently and accurately estimate rare event probabilities, different IS approaches [15,16,23,24,36,46,47] have been proposed.

The current paper proposes a path-dependent IS approach based on an approximate TL scheme to improve the MC estimator efficiency, and hence efficiently estimate various statistical quantities for SRNs (particularly rare event probabilities). Our class of probability measure change is based on modifying the Poisson random variable rates used to construct the TL paths. In particular, optimal IS parameters are obtained by minimizing the second moment of the IS estimator (equivalently the variance) which represents the cost function for the associated SOC problem. We show that the corresponding value function solves a dynamic programming relation that is challenging to solve analytically (see Sect. 2.1). We approximate the dynamic programming equation to derive a closed form solution and near-optimal control parameters. The cost to solve the associated backward equation numerically in multi-dimensional settings increases exponentially with respect to the dimension (i.e., the curse of dimensionality). Thus, we propose approximating the resulting value function using a neural network to overcome this issue. Utilizing the optimality criterion for the SOC problem, we obtain a relationship between optimal IS parameters and the value function. Finally, we employ a stochastic optimization algorithm to learn the corresponding neural network parameters. Our analysis and numerical results for different dimensions confirm that the proposed estimator considerably reduces the variance compared with the standard TL-MC method with a negligible additional cost. This allows rare event probabilities to be efficiently computed in a regime where standard TL-MC estimators commonly fail.

The proposed approach is more computationally efficient than previously proposed IS schemes in this context ([15,16,23,24,36,46,47]) because it is based on an approximate TL scheme rather than the exact scheme. In contrast to previous approaches, the change of measure is systematically derived to ensure convergence to the optimal measure within the chosen class of probability measures, minimizing MC estimator variance. The novelty of this work is establishing a connection between IS and SOC in the context of pure jump processes, particularly for SRNs, with an emphasis on related practical and numerical aspects. Note that some previous studies [7,17,20,28–31,33,41,49] have established a similar connection, mainly in the diffusion dynamics context, with less focus on pure jump dynamics. In this work, the proposed methodology is based on an approximate explicit TL scheme, which could and be subsequently extended in future work to continuous-time formulation (exact schemes), and implicit TL schemes which are relevant for systems with fast and slow time scales.

The remainder of this paper is organized as follows. Sections 1.1, 1.2, 1.3 and 1.4 define relevant SRN, TL, MC and IS concepts, respectively. Section 2 establishes the connection between IS and SOC, formulating the SOC problem and defining its main ingredients: controls, cost function, and value function; then presents the dynamic programming solved by the optimal controls. Section 2.3 develops the proposed IS learning-based approach appropriate for multidimensional SRNs. Section 3 provides selected numerical experiments for different dimensions to illustrate the proposed approach's efficiency compared with standard MC approaches. Finally, Sect. 4 summarizes and concludes the work, and discusses possible future research directions.

1.1 Stochastic reaction networks (SRNs)

We are interested in the time evolution for an homogeneously mixed chemical reacting system described by the Markovian pure jump process, $\mathbf{X} : [0, T] \times \Omega \to \mathbb{N}^d$, where $(\Omega, \mathcal{F}, \mathbb{P})$ is a probability space. In this framework, we assume that d different species interact through J reaction channels. The i-th component, $X_i(t)$, describes the abundance of the i-th species present in the chemical system at time t. This work studies the time evolution of the state vector,

$$\mathbf{X}(t) = (X_1(t), \ldots, X_d(t)) \in \mathbb{N}^d. \tag{1.1}$$

Each reaction channel $\mathcal{R}_j$ is a pair $(a_j, \boldsymbol{\nu}_j)$ defined by its propensity function $a_j : \mathbb{R}^d \to \mathbb{R}_+$ and stoichiometric vector $\boldsymbol{\nu}_j = (\nu_{j,1}, \nu_{j,2}, \ldots, \nu_{j,d})^\top$ satisfying

$$\mathbb{P}\left(\mathbf{X}(t+\Delta t) = \mathbf{x}+\boldsymbol{\nu}_j \mid \mathbf{X}(t) = \mathbf{x}\right) = a_j(\mathbf{x})\Delta t + o(\Delta t),\ j = 1, 2, \ldots, J. \tag{1.2}$$

Thus, the probability of observing a jump in the process $\mathbf{X}$ from state $\mathbf{x}$ to state $\mathbf{x}+\boldsymbol{\nu}_j$, a consequence of reaction $\mathcal{R}_j$ firing during the small time interval $(t, t+\Delta t]$, is proportional to the time interval length, Δt, where $a_j(\mathbf{x})$ is the proportionality constant. We set $a_j(\mathbf{x})$=0 for $\mathbf{x}$ such that $\mathbf{x}+\boldsymbol{\nu}_j \notin \mathbb{N}^d$ (i.e., the *non-negativity assumption*: the system can never produce negative population values).

Hence, from (1.2), process $\mathbf{X}$ is a continuous-time, discrete-space Markov chain that can be characterized by Kurtz's random time change representation [19],

$$\mathbf{X}(t) = \mathbf{x}_0 + \sum_{j=1}^{J} Y_j \left(\int_0^t a_j(\mathbf{X}(s))\, \mathrm{d}s \right) \boldsymbol{\nu}_j, \tag{1.3}$$

where $Y_j : \mathbb{R}_+ \times \Omega \to \mathbb{N}$ are independent unit-rate Poisson processes. Conditions on the reaction channels can be imposed to ensure uniqueness [5] and avoid explosions in finite time [18,27,44].

Applying the *stochastic mass-action kinetics* principle, we can assume that the propensity function $a_j(\cdot)$ for reaction channel $\mathcal{R}_j$, represented as[1]

$$\alpha_{j,1} S_1 + \cdots + \alpha_{j,d} S_d \overset{\theta_j}{\to} \beta_{j,1} S_1 + \cdots + \beta_{j,d} S_d \tag{1.4}$$

obeys

$$a_j(\mathbf{x}) := \theta_j \prod_{i=1}^{d} \frac{x_i!}{(x_i - \alpha_{j,i})!} \mathbf{1}_{\{x_i \geq \alpha_{j,i}\}}, \tag{1.5}$$

where $\{\theta_j\}_{j=1}^J$ represents positive constant reaction rates, and x_i is the counting number for species S_i.

1.2 Explicit tau-leap approximation

The explicit-TL scheme is a pathwise approximate method [6,26] to overcome computational drawbacks for exact methods (i.e., when many reactions fire during a short time interval). This scheme can be derived from the random time change representation (1.3) by approximating the integral $\int_{t_i}^{t_{i+1}} a_j(\mathbf{X}(s))\mathrm{d}s$ as $a_j(\mathbf{X}(t_i))\,(t_{i+1} - t_i)$, i.e., using the forward-Euler method with time mesh $\{t_0 = 0, t_1, \ldots, t_N = T\}$ and size $\Delta t = \frac{T}{N}$. Thus, the explicit-TL approximation for $\mathbf{X}$ should satisfy for $k \in \{1, 2, \ldots, N\}$

$$\widehat{\mathbf{X}}_k^{\Delta t} = \mathbf{x}_0 + \sum_{j=1}^{J} Y_j \left(\sum_{i=0}^{k-1} a_j(\widehat{\mathbf{X}}_i^{\Delta t}) \Delta t \right) \boldsymbol{\nu}_j, \tag{1.6}$$

and given $\widehat{\mathbf{X}}_0 := \mathbf{x}_0$, we iteratively simulate a path for $\widehat{\mathbf{X}}^{\Delta t}$ as

$$\widehat{\mathbf{X}}_k^{\Delta t} := \widehat{\mathbf{X}}_{k-1}^{\Delta t} + \sum_{j=1}^{J} \mathcal{P}_{k-1,j}\left(a_j(\widehat{\mathbf{X}}_{k-1}^{\Delta t})\Delta t\right) \boldsymbol{\nu}_j,\ 1 \leq k \leq N, \tag{1.7}$$

where, conditioned on the current state $\widehat{\mathbf{X}}_k^{\Delta t}$, $\{\mathcal{P}_{k,j}(r_{k,j})\}_{\{1 \leq j \leq J\}}$ are independent Poisson random variables with respective rates $r_{k,j} := a_j(\widehat{\mathbf{X}}_k^{\Delta t})\Delta t$.

The explicit-TL path $\widehat{\mathbf{X}}^{\Delta t}$ is defined only at time mesh points, but can be naturally extended to $[0, T]$ as a piecewise constant path. We apply the projection to zero to prevent the process from exiting the lattice (i.e., producing negative values), hence (1.7) becomes

$$\widehat{\mathbf{X}}_k^{\Delta t} := \max\left(\mathbf{0}, \widehat{\mathbf{X}}_{k-1}^{\Delta t} + \sum_{j=1}^{J} \mathcal{P}_{k-1,j}\left(a_j(\widehat{\mathbf{X}}_{k-1}^{\Delta t})\Delta t\right) \boldsymbol{\nu}_j\right), 1 \leq k \leq N, \tag{1.8}$$

where the maximum is applied entry-wise. In this work, we use uniform time steps with length Δt, but the explicit-TL scheme and the proposed IS scheme (see Sect. 2) can also be applied to non-uniform time meshes.

1.3 Biased Monte Carlo estimator

Let $\mathbf{X}$ be a stochastic process and $g : \mathbb{R}^d \to \mathbb{R}$ a scalar observable. We want to approximate $\mathbb{E}\left[g(\mathbf{X}(T))\right]$, but rather than sampling directly from $\mathbf{X}(T)$, we sample from $\overline{\mathbf{X}}^{\Delta t}(T)$, which are random variables generated by a numerical scheme with step size Δt. We assume that variates $\overline{\mathbf{X}}^{\Delta t}(T)$ are generated with an algorithm with weak order, $\mathcal{O}(\Delta t)$, i.e., for sufficiently small Δt,

$$\left| \mathbb{E}\left[g(\mathbf{X}(T)) - g(\overline{\mathbf{X}}^{\Delta t}(T)) \right] \right| \leq C \Delta t \tag{1.9}$$

where $C > 0$.[2]

Let μ_M be the standard MC estimator for $\mathbb{E}\left[g(\overline{\mathbf{X}}^{\Delta t}(T))\right]$,

$$\mu_M := \frac{1}{M} \sum_{m=1}^{M} g(\overline{\mathbf{X}}_{[m]}^{\Delta t}(T)), \tag{1.10}$$

[1] $\alpha_{j,i}$ molecules for species S_i are consumed and $\beta_{j,i}$ are produced. Thus, $(\alpha_{j,i}, \beta_{j,i}) \in \mathbb{N}^2$ but $\beta_{j,i} - \alpha_{j,i}$ can be a negative integer, constituting the vector $\boldsymbol{\nu}_j = \left(\beta_{j,1} - \alpha_{j,1}, \ldots, \beta_{j,d} - \alpha_{j,d}\right) \in \mathbb{Z}^d$.

[2] Refer to [39] for the underlying assumptions and proofs for this statement in the TL scheme context.

where $\{\overline{\mathbf{X}}^{\Delta t}_{[m]}(T)\}_{m=1}^{M}$ are independent and distributed as $\overline{\mathbf{X}}^{\Delta t}(T)$.

The global error for the proposed MC estimator has error decomposition

$$\begin{aligned}&\left|\mathbb{E}\left[g(\mathbf{X}(T))\right]-\mu_M\right|\\&\quad\le\underbrace{\left|\mathbb{E}[g(\mathbf{X}(T))]-\mathbb{E}\left[g(\overline{\mathbf{X}}^{\Delta t}(T))\right]\right|}_{\text{Bias}}+\underbrace{\left|\mathbb{E}\left[g(\overline{\mathbf{X}}^{\Delta t}(T))\right]-\mu_M\right|}_{\text{Statistical Error}}.\end{aligned}\qquad(1.11)$$

To achieve the desired accuracy, TOL, it is sufficient to bound the bias and statistical error equally by $\frac{TOL}{2}$. From (1.9), choosing step size

$$\Delta t(\mathrm{TOL})=\frac{\mathrm{TOL}}{2\cdot C}\qquad(1.12)$$

ensures a bias of $\frac{\mathrm{TOL}}{2}$.

Thus, considering the central limit theorem, the statistical error can be approximated as

$$\left|\mathbb{E}\left[g(\overline{\mathbf{X}}^{\Delta t}(T))\right]-\mu_M\right|\approx C_\alpha\cdot\sqrt{\frac{\mathrm{Var}[g(\overline{\mathbf{X}}^{\Delta t}(T))]}{M}},\qquad(1.13)$$

where constant C_α is the $(1-\frac{\alpha}{2})$−quantile for the standard normal distribution. We choose $C_\alpha = 1.96$ for a 95% confidence level corresponding to $\alpha = 0.05$. Choosing

$$M^*(\mathrm{TOL})=C_\alpha^2\frac{4\cdot\mathrm{Var}\left[g(\overline{\mathbf{X}}^{\Delta t}(T))\right]}{\mathrm{TOL}^2}\qquad(1.14)$$

sample paths ensures the statistical error to be approximately bounded by $\frac{\mathrm{TOL}}{2}$.

Given that the computational cost to simulate a single path is $\mathcal{O}\left(\Delta t^{-1}\right)$, the expected total computational complexity is $\mathcal{O}\left(\mathrm{TOL}^{-3}\right)$; and the complexity scales with $\mathrm{Var}\left[g(\overline{\mathbf{X}}^{\Delta t}(T))\right]$ (see (1.14)).

1.4 Importance sampling

Importance sampling (IS) techniques improve the computational costs for the crude MC estimator by variance reduction when used appropriately. To motivate the use of these techniques, consider estimating rare event probabilities, where the crude MC method is substantially expensive. In particular, consider estimating $q=\mathbb{P}(Y>\gamma)=\mathbb{E}[\mathbf{1}_{\{Y>\gamma\}}]$, where Y is a random variable taking values in $\mathbb{R}$ with probability density function ρ_Y. Let γ be sufficiently large that q becomes sufficiently small. We can approximate q using the MC estimator

$$\widehat{q}=\frac{1}{M}\sum_{i=1}^{M}\mathbf{1}_{\{Y^{(i)}>\gamma\}},\qquad(1.15)$$

where $\{Y^{(i)}\}_{i=1}^{M}$ are independent and identically distributed (i.i.d) realizations sampled according to ρ_Y. The MC estimator variance is

$$Var\left[\mathbf{1}_{\{Y^{(i)}>\gamma\}}\right]=q-q^2.\qquad(1.16)$$

For a sufficiently small q, we can use (1.16) and the central limit theorem to approximate the relative error as

$$\frac{|q-\widehat{q}|}{q}\approx C_\alpha\sqrt{\frac{1}{qM}},\qquad(1.17)$$

where C_α is chosen as in (1.13).

The number of required samples to attain a relative error tolerance TOL_{rel} is $M\approx\frac{C_\alpha^2}{q\cdot TOL_{rel}^2}$. Thus, for q of the order of 10^{-8}, the number of required samples such that $TOL_{rel}=5\%$ is approximately equal to $1.5\cdot10^{11}$.

To demonstrate the IS concept, consider the general problem of estimating $\mathbb{E}[g(Y)]$, where g is a given observable. In the previous example, g was chosen as $g(y)=\mathbf{1}_{\{y>\gamma\}}$. Let $\widehat{\rho}_Z$ be the probability density function for a new real random variable Z, such that $g\cdot\rho_Y$ is dominated by $\widehat{\rho}_Z$, i.e.,

$$\widehat{\rho}_Z(x)=0\implies g(x)\cdot\rho_Y(x)=0\qquad(1.18)$$

for all $x\in\mathbb{R}$. This permits, the quantity of interest to be expressed as

$$\begin{aligned}\mathbb{E}[g(Y)]&=\int_{\mathbb{R}}g(x)\rho_Y(x)dx=\\&=\int_{\mathbb{R}}g(x)\underbrace{\frac{\rho_Y(x)}{\widehat{\rho}_Z(x)}}_{L(x)}\cdot\widehat{\rho}_Z(x)dx=\mathbb{E}[L(Z)\cdot g(Z)],\end{aligned}\qquad(1.19)$$

where $L(\cdot)$ is the likelihood ratio. Hence the expected value under the new measure remains unchanged, but the variance could be reduced due to a different second moment $\mathbb{E}\left[(g(Z)\cdot L(Z))^2\right]$.

The MC estimator under the IS measure is

$$\mu_M^{IS}=\frac{1}{M}\sum_{j=1}^{M}L(Z_{[j]})\cdot g(Z_{[j]})=\frac{1}{M}\sum_{j=1}^{M}\frac{\rho_Y(Z_{[j]})}{\widehat{\rho}_Z(Z_{[j]})}\cdot g(Z_{[j]}),\qquad(1.20)$$

where $Z_{[j]}$ are i.i.d samples from $\widehat{\rho}_Z$ for $j=1,\ldots,M$.

The main challenge when using IS is choosing a new probability measure that substantially reduces the variance compared with the original measure. This step strongly depends on the structure of the problem under consideration. Further, the new measure should be obtained with negligible computational cost to ensure a computational efficient

IS scheme. This is particularly challenging in the present problem, since we are considering path-dependent probability measures. In particular, the aim is to introduce a path-dependent change of probability measure that corresponds to changing the Poisson random variable rates used to construct the TL paths. Section 2.1 shows how the optimal IS parameters can be obtained using a novel connection with SOC.

2 Importance sampling (IS) via stochastic optimal control (SOC)

2.1 Dynamic programming for the importance sample parameters

This section, establishes the connection between optimal IS measure determination within a class of probability measures, and SOC. Let $\mathbf{X}$ be a SRN as defined in Sect. 1.1 and let $\widehat{\mathbf{X}}^{\Delta t}$ denote its TL approximation as given by (1.8). We aim to find a near-optimal IS measure to improve the MC estimator computational performance to estimate $\mathbb{E}\left[g(\mathbf{X}(T))\right]$. Since finding the optimal path-dependent change of measure within all measure classes presents a challenging problem, we limit ourselves to a parameterized class obtained via modifying the Poisson random variable rates of the TL paths. This class of measure change was previously used in [10] to improve the MLMC estimator robustness and performance in this context; we focus on a single-level MC setting, and seek to automate the task to find a near-optimal IS measure within this class.

We introduce the change of measure resulting from changing the Poisson random variable rates in the TL scheme,

$$\overline{P}_{n,j} = \mathcal{P}_{n,j}\left(\delta^{\Delta t}_{n,j}(\overline{\mathbf{X}}^{\Delta t}_n)\Delta t\right), \qquad n = 0, \ldots, N-1,\ j = 1, \ldots, J; \tag{2.1}$$

where $\delta^{\Delta t}_{n,j}(\mathbf{x}) \in \mathcal{A}_{\mathbf{x},j}$ is the control parameter at time step n, under reaction j, and in state $\mathbf{x} \in \mathbb{N}^d$; and conditioned on $\overline{\mathbf{X}}^{\Delta t}_n$, $\mathcal{P}_{n,j}(r_{n,j})$ are independent Poisson random variables with respective rates $r_{n,j} := \delta^{\Delta t}_{n,j}(\overline{\mathbf{X}}^{\Delta t}_n)\Delta t$. The admissible set,

$$\mathcal{A}_{\mathbf{x},j} = \begin{cases} \{0\}, & \text{if } a_j(\mathbf{x}) = 0 \\ \{y \in \mathbb{R} : y > 0\}, & \text{otherwise}, \end{cases} \tag{2.2}$$

is chosen such that (1.18) is fulfilled and to avoid infinite variance for the IS estimator. The control $\delta^{\Delta t}_{n,j}(\mathbf{x}) \in \mathcal{A}_{\mathbf{x},j}$ depends deterministically on the current time step n, reaction channel j, and current state $\mathbf{x} = \overline{\mathbf{X}}^{\Delta t}_n$ for the TL-IS approximation in (2.3).

Therefore, the resulting scheme under the new measure is

$$\begin{aligned} \overline{\mathbf{X}}^{\Delta t}_{n+1} &= \max\left(\mathbf{0}, \overline{\mathbf{X}}^{\Delta t}_n + \sum_{j=1}^{J} \overline{P}_{n,j}\boldsymbol{\nu}_j\right), \quad n = 0, \ldots, N-1, \\ \overline{\mathbf{X}}^{\Delta t}_0 &= \mathbf{x}_0, ; \end{aligned} \tag{2.3}$$

and the likelihood ratio[3] at step n associated with the new IS measure is

$$\begin{aligned} L_n(\overline{\mathbf{P}}_n, \boldsymbol{\delta}^{\Delta t}_n(\overline{\mathbf{X}}^{\Delta t}_n)) &= \prod_{j=1}^{J} \exp\left(-(a_j(\overline{\mathbf{X}}^{\Delta t}_n) - \delta^{\Delta t}_{n,j}(\overline{\mathbf{X}}^{\Delta t}_n))\Delta t\right)\left(\frac{a_j(\overline{\mathbf{X}}^{\Delta t}_n)}{\delta^{\Delta t}_{n,j}(\overline{\mathbf{X}}^{\Delta t}_n)}\right)^{\overline{P}_{n,j}} \\ &= \exp\left(-\left(\sum_{j=1}^{J} a_j(\overline{\mathbf{X}}^{\Delta t}_n) - \delta^{\Delta t}_{n,j}(\overline{\mathbf{X}}^{\Delta t}_n)\right)\Delta t\right) \cdot \prod_{j=1}^{J}\left(\frac{a_j(\overline{\mathbf{X}}^{\Delta t}_n)}{\delta^{\Delta t}_{n,j}(\overline{\mathbf{X}}^{\Delta t}_n)}\right)^{\overline{P}_{n,j}} ; \end{aligned} \tag{2.4}$$

where $\boldsymbol{\delta}^{\Delta t}_n(\mathbf{x}) \in \times_{j=1}^{J}\mathcal{A}_{\mathbf{x},j}$ are the IS parameters with $\left(\boldsymbol{\delta}^{\Delta t}_n(\mathbf{x})\right)_j = \delta^{\Delta t}_{n,j}(\mathbf{x})$ and the Poisson realizations are denoted by $\overline{\mathbf{P}}_n$ with $\left(\overline{\mathbf{P}}_n\right)_j := \overline{P}_{n,j}$ for $j = 1, \ldots, J$. Equation (2.4) uses the convention that $\frac{a_j(\overline{\mathbf{X}}^{\Delta t}_n)}{\delta^{\Delta t}_{n,j}(\overline{\mathbf{X}}^{\Delta t}_n)} = 1$, whenever $a_j(\overline{\mathbf{X}}^{\Delta t}_n) = 0$ and $\delta^{\Delta t}_{n,j}(\overline{\mathbf{X}}^{\Delta t}_n) = 0$. From (2.2), this results in a factor of one in the likelihood ratio for reactions with $a_j(\overline{\mathbf{X}}^{\Delta t}_n) = 0$.

Therefore, the likelihood ratio for $\{\overline{\mathbf{X}}^{\Delta t}_n : n = 0, \ldots, N\}$ across one path is

$$\begin{aligned} &L\left(\left(\overline{\mathbf{P}}_0, \ldots, \overline{\mathbf{P}}_{N-1}\right), \left(\boldsymbol{\delta}^{\Delta t}_0(\overline{\mathbf{X}}^{\Delta t}_0), \ldots, \boldsymbol{\delta}^{\Delta t}_{N-1}(\overline{\mathbf{X}}^{\Delta t}_{N-1})\right)\right) \\ &= \prod_{n=0}^{N-1} L_n\left(\overline{\mathbf{P}}_n, \boldsymbol{\delta}^{\Delta t}_n(\overline{\mathbf{X}}^{\Delta t}_n)\right). \end{aligned} \tag{2.5}$$

This likelihood ratio completes the characterization for the proposed IS approach, and allows the quantity of interest with respect to the new measure to be expressed as

$$\begin{aligned} &\mathbb{E}[g(\widehat{\mathbf{X}}^{\Delta t}_N)] \\ &= \mathbb{E}\left[L\left(\left(\overline{\mathbf{P}}_0, \ldots, \overline{\mathbf{P}}_{N-1}\right), \left(\boldsymbol{\delta}^{\Delta t}_0(\overline{\mathbf{X}}^{\Delta t}_0), \ldots, \boldsymbol{\delta}^{\Delta t}_{N-1}(\overline{\mathbf{X}}^{\Delta t}_{N-1})\right)\right) \cdot g(\overline{\mathbf{X}}^{\Delta t}_N)\right], \end{aligned} \tag{2.6}$$

with the expectation in the right-hand side of (2.6) taken with respect to the dynamics in (2.3).

Hereinafter, we aim to determine optimal parameters $\{\boldsymbol{\delta}^{\Delta t}_n(\mathbf{x})\}_{n=0,\ldots,N-1;\mathbf{x}\in\mathbb{N}^d}$ that minimize the second moment (and hence the variance) for the IS estimator, given that $\overline{\mathbf{X}}^{\Delta t}_0 = \mathbf{x}_0$. To that end, we derive an associated SOC formulation. First we introduce the cost function for the proposed SOC problem in Definition 2.1, then derive a dynamic programming equation in Theorem 2.4 that is satisfied by the

[3] We refer to [10] (Sect. 4.1) for the likelihood factor derivation of a similar IS scheme.

value function $u_{\Delta t}(\cdot,\cdot)$ in Definition 2.3. The proof for Theorem 2.4 is given in “Appendix A”.

Definition 2.1 (*Second moment for the proposed importance sampling estimator*) Let $0 \leq n \leq N$. Given that $\overline{\mathbf{X}}_n^{\Delta t} = \mathbf{x}$, the second moment for the proposed IS estimator can be expressed as

$$\begin{aligned} &C_{n,\mathbf{x}}\left(\boldsymbol{\delta}_n^{\Delta t},\ldots,\boldsymbol{\delta}_{N-1}^{\Delta t}\right) \\ &\quad = \mathbb{E}\left[g^2(\overline{\mathbf{X}}_N^{\Delta t})\prod_{k=n}^{N-1} L_k^2\left(\overline{\mathbf{P}}_k,\boldsymbol{\delta}_k^{\Delta t}(\overline{\mathbf{X}}_k^{\Delta t})\right) | \overline{\mathbf{X}}_n^{\Delta t} = \mathbf{x}\right], \\ &\quad 0 \leq n \leq N-1, \end{aligned} \tag{2.7}$$

with terminal cost $C_{N,\mathbf{x}} = \mathbb{E}\left[g^2\left(\overline{\mathbf{X}}_N^{\Delta t}\right) | \overline{\mathbf{X}}_N^{\Delta t} = \mathbf{x}\right] = g^2(\mathbf{x})$, for any $\mathbf{x} \in \mathbb{N}^d$.

Compared with the classical SOC formulation, (2.7) can be interpreted as the expected total cost; where the main difference is that (2.7) uses a multiplicative cost structure rather than the standard additive one. Therefore, we derive a dynamic programming relation in Theorem 2.4 associated with this cost structure that is fulfilled by the corresponding value function (see Definition 2.3), in the SRN context.

Remark 2.2 (*Structure of the cost function*) One can derive an optimal control formulation with additive structure (similar to [30] in the stochastic differential equation setting) by applying a logarithmic transformation together with Jensen’s inequality to (2.7). This reduces the control problem to a Kullback–Leibler minimization. In [41,42], this Kullback–Leibler minimization problem leads to the same optimal change of measure as the problem of finding the change of measure using a variance minimization approach. However, the previous conclusion needs more investigation in the setting of SRNs, which we leave for future potential work.

Definition 2.3 (*Value function*) The *value function* $u_{\Delta t}(\cdot,\cdot)$ is defined as the optimal (infimum) second moment for the proposed IS estimator. For time step $0 \leq n \leq N$ and state $\mathbf{x} \in \mathbb{N}^d$,

$$\begin{aligned} u_{\Delta t}(n,\mathbf{x}) &:= \inf_{\{\boldsymbol{\delta}_k^{\Delta t}\}_{k=n,\ldots,N-1} \in \mathcal{A}^{N-n}} C_{n,\mathbf{x}}\left(\boldsymbol{\delta}_n^{\Delta t},\ldots,\boldsymbol{\delta}_{N-1}^{\Delta t}\right) \\ &= \inf_{\{\boldsymbol{\delta}_k^{\Delta t}\}_{k=n,\ldots,N-1} \in \mathcal{A}^{N-n}} \\ &\quad \mathbb{E}\left[g^2\left(\overline{\mathbf{X}}_N^{\Delta t}\right)\prod_{k=n}^{N-1} L_k^2\left(\overline{\mathbf{P}}_k,\boldsymbol{\delta}_k^{\Delta t}(\overline{\mathbf{X}}_k^{\Delta t})\right) | \overline{\mathbf{X}}_n^{\Delta t} = \mathbf{x}\right], \end{aligned} \tag{2.8}$$

where $\mathcal{A} = \times_{\mathbf{x}\in\mathbb{N}^d} \times_{j=1}^{J} \mathcal{A}_{\mathbf{x},j} \in \mathbb{R}^{\mathbb{N}^d \times J}$ is the admissible set for the IS parameters; and $u_{\Delta t}(N,\mathbf{x}) = g^2(\mathbf{x})$, for any $\mathbf{x} \in \mathbb{N}^d$.

Theorem 2.4 (Dynamic programming for importance sampling parameters) *For* $\mathbf{x} \in \mathbb{N}^d$, *the value function* $u_{\Delta t}(n,\mathbf{x})$ *fulfills the dynamic programming relation*

$$u_{\Delta t}(N,\mathbf{x}) = g^2(\mathbf{x})$$

$$\text{and for } n = N-1,\ldots,0, \text{ and } \mathcal{A}_{\mathbf{x}} := \bigtimes_{j=1}^{J} \mathcal{A}_{\mathbf{x},j},$$

$$\begin{aligned} u_{\Delta t}(n,\mathbf{x}) = \inf_{\boldsymbol{\delta}_n^{\Delta t}(\mathbf{x}) \in \mathcal{A}_{\mathbf{x}}} &\exp\left(\left(-2\sum_{j=1}^{J} a_j(\mathbf{x}) + \sum_{j=1}^{J} \delta_{n,j}^{\Delta t}(\mathbf{x})\right)\Delta t\right) \\ &\times \sum_{\mathbf{p}\in\mathbb{N}^J}\left(\prod_{j=1}^{J}\frac{(\Delta t \cdot \delta_{n,j}^{\Delta t}(\mathbf{x}))^{p_j}}{p_j!}\left(\frac{a_j(\mathbf{x})}{\delta_{n,j}^{\Delta t}(\mathbf{x})}\right)^{2p_j}\right) \cdot \\ &u_{\Delta t}(n+1,\max(\mathbf{0},\mathbf{x}+\boldsymbol{\nu}\mathbf{p})), \end{aligned} \tag{2.9}$$

where $\boldsymbol{\nu} = (\boldsymbol{\nu}_1,\ldots,\boldsymbol{\nu}_J) \in \mathbb{Z}^{d\times J}$.

Theorem 2.4 breaks down the minimization problem to a simpler optimization that can be solved stepwise backward in time starting from final time T. Solving the minimization problem (2.9) analytically is difficult due to the infinite sum. Section 2.2 shows how to overcome this issue by approximating (2.9) to derive near-optimal parameters for $\{\boldsymbol{\delta}_n^{\Delta t}(\mathbf{x})\}_{n=0,\ldots,N-1;\mathbf{x}\in\mathbb{N}^d}$ for the proposed IS approach.

2.2 Approximate dynamic programming

Theorem 2.4 gives an exact solution for optimal IS parameters resulting from modifying the Poisson random variable rates in the TL paths. However, the infinite sum has to be evaluated in closed form to solve (2.9) analytically, which is generally difficult. Therefore, we propose approximating the value function $u_{\Delta t}(n,\mathbf{x})$ in (2.9) by $\overline{u}_{\Delta t}(n,\mathbf{x})$ for all time steps $n = 0,\ldots,N$, reaction channels $j = 1,\ldots,J$ and states $\mathbf{x} \in \mathbb{N}^d$. First, both $u_{\Delta t}(n,\mathbf{x})$ and $\overline{u}_{\Delta t}(n,\mathbf{x})$ satisfy the same final condition,

$$\overline{u}_{\Delta t}(N,\mathbf{x}) = u_{\Delta t}(N,\mathbf{x}) = g^2(\mathbf{x}). \tag{2.10}$$

Next, to derive the approximate dynamic programming relation for $\overline{u}_{\Delta t}(\cdot,\cdot)$, we presume Assumption 2.5 to hold. This assumption is motivated by the behavior of the original propensities, which are of $\mathcal{O}(1)$ due to the mass-action kinetics principle (refer to (1.5)).

Assumption 2.5 The controls $\{\boldsymbol{\delta}_n^{\Delta t}\}_{n=0,\ldots,N-1}$ are asymptotically constant (i.e., $\delta_{n,j}^{\Delta t}(\mathbf{x}) \rightarrow c_{n,j,\mathbf{x}}$, as $\Delta t \rightarrow 0$, where $c_{n,j,\mathbf{x}}$ are constants for $1 \leq j \leq J$, $0 \leq n \leq N-1$, and $\mathbf{x} \in \mathbb{N}^d$).

Given Assumption 2.5 and that $\{a_j(\cdot)\}_{j=1}^{J}$ are of $\mathcal{O}(1)$, we apply a Taylor expansion around $\Delta t = 0$ to the exponential term in (2.9), then truncate the expression within the infimum such that the remaining terms are $\mathcal{O}(\Delta t)$. This truncates the infinite sum and linearizes the exponential term. Thus, for $\mathbf{x} \in \mathbb{N}^d$ and $n = N-1, \ldots, 0$

$$\begin{aligned}
\overline{u}_{\Delta t}(n, \mathbf{x}) &= \Delta t \inf_{(\delta_1,\ldots,\delta_J)\in\mathcal{A}_{\mathbf{x}}} \Big[\sum_{j=1}^{J} \frac{a_j^2(\mathbf{x})}{\delta_j} \overline{u}_{\Delta t}(n+1, \max(0, \mathbf{x}+\nu_j)) \\
&\quad + \overline{u}_{\Delta t}(n+1, \mathbf{x}) \sum_{j=1}^{J} \delta_j\Big] \\
&\quad + \overline{u}_{\Delta t}(n+1, \mathbf{x}) - 2\Delta t \cdot \overline{u}_{\Delta t}(n+1, \mathbf{x}) \cdot \sum_{j=1}^{J} a_j(\mathbf{x}) \\
&= \Delta t \cdot \sum_{j=1}^{J} \underbrace{\inf_{\delta_j \in \mathcal{A}_{\mathbf{x},j}} \Big[\frac{a_j^2(\mathbf{x})}{\delta_j} \cdot \overline{u}_{\Delta t}(n+1, \max(0, \mathbf{x}+\nu_j)) + \delta_j \cdot \overline{u}_{\Delta t}(n+1, \mathbf{x})\Big]}_{=:Q^{\Delta t}(n,j,\mathbf{x})} \\
&\quad + \left(1 - 2\Delta t \sum_{j=1}^{J} a_j(\mathbf{x})\right) \overline{u}_{\Delta t}(n+1, \mathbf{x}),
\end{aligned} \tag{2.11}$$

where $\delta_j \in \mathcal{A}_{\mathbf{x},j}$, $j = 1, \ldots, J$, are the SOC parameters at state $\mathbf{x}$ for reaction j. The admissible set $\mathcal{A}_{\mathbf{x},j}$ is defined in (2.2). Assumption 2.5 ensures that (i) we can apply the Taylor expansion to the exponential term as Δt decreases, and (ii) we have the exact approximation structure for (2.11) with no further terms scaling with Δt that have order less than Δt^2.

The infimum in (2.11) is attained when

$$\begin{aligned}
&(i)\ \overline{u}_{\Delta t}(n+1, \mathbf{x}) \neq 0, \quad \text{and} \\
&(ii)\ \overline{u}_{\Delta t}(n+1, \max(0, \mathbf{x}+\nu_j)) \neq 0,\ \forall\, 1 \leq j \leq J.
\end{aligned} \tag{2.12}$$

In this case, the approximate optimal SOC parameter $\overline{\delta}_{n,j}^{\Delta t}(\mathbf{x})$ can be analytically determined as

$$\overline{\delta}_{n,j}^{\Delta t}(\mathbf{x}) = \frac{a_j(x)\sqrt{\overline{u}_{\Delta t}(n+1, \max(0, \mathbf{x}+\nu_j))}}{\sqrt{\overline{u}_{\Delta t}(n+1, \mathbf{x})}}, \quad 1 \leq j \leq J. \tag{2.13}$$

Note (2.13) includes the particular case when $a_j(\mathbf{x}) = 0$ for some $j \in \{1, \ldots, J\}$. In such a case, $\overline{\delta}_{n,j}^{\Delta t}(\mathbf{x}) = 0$, which agrees with (2.2).

An important advantage for this numerical approximation, $\overline{u}_{\Delta t}(\cdot, \cdot)$, is that we reduce the complexity of the original optimization problem at each step in (2.9) from a simultaneous optimization over J variables to independent one-dimensional optimization problems that can be solved in parallel using (2.13).

Remark 2.6 (*Assumption* (2.12)) Whether the assumption in (2.12) is generally fulfilled depends on the method employed to solve the dynamic programming principle in (2.11). For example, if we use a direct numerical implementation either some special numerical treatment is required for the cases where (2.12) is violated, or some regularization is required to ensure well-posedness. The proposed approach from Sect. 2.3 avoids that issue since we model $\overline{u}_{\Delta t}(\cdot, \cdot)$ with a strictly positive ansatz function, which guarantees condition (2.12) to hold for any state $\mathbf{x}$ and all time steps n.

Remark 2.7 (*Computational cost for dynamic programming*) To derive a practical numerical algorithm for a finite number of states, we truncate the infinite state space $\mathbb{N}^d$ to $\bigtimes_{i=1}^{d}[0, \overline{S}_i]$, where $\overline{S}_1, \ldots, \overline{S}_d$ is a set of sufficiently large upper bounds. The computational cost to numerically solve the dynamic programming equation (2.11) for step size Δt and state space $\bigtimes_{i=1}^{d}[0, \overline{S}_i]$ can be expressed as

$$W_{\mathrm{dp}}(\overline{\mathbf{S}}, \Delta t) \approx \left(\overline{S}^*\right)^d \cdot \frac{T}{\Delta t} \cdot J, \tag{2.14}$$

where $\overline{S}^* = \max_{i=1,\ldots,d} \overline{S}_i$.

The cost in (2.14) scales exponentially with dimension d. Section 2.3 proposes an alternative approach to address this curse of dimensionality. However, in future work, we aim to combine dimension reduction techniques for SRNs with a direct numerical implementation of dynamic programming.

2.3 Learning-based approach

Using the SOC formulation derived in Sect. 2.2, we propose approximating the value function $\overline{u}_{\Delta t}(\cdot, \cdot)$ with a parameterized ansatz function, $\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta})$.

Remark 2.8 (*Choosing the ansatz function*) The parameterized ansatz function $\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta})$ should consider the final condition of the value function (2.9), and its choice depends on the given SRN and observable $g(\mathbf{x})$. For linear observables, such as $g(\mathbf{x}) = x_i$, we can consider polynomial basis functions as an ansatz. For more complex problems, the ansatz function is a small neural network.

For rare event applications with observable $g(\mathbf{x}) = \mathbf{1}_{\{x_i > \gamma\}}$, we consider a sigmoid with learning parameters $\boldsymbol{\beta} = \left(\boldsymbol{\beta}^{space}, \beta^{time}\right) \in \mathbb{R}^{d+1}$ as the ansatz function

$$\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta}) = \frac{1}{1 + e^{-(1-t)\cdot(\langle \boldsymbol{\beta}^{space}, \mathbf{x}\rangle + \beta^{time}) - b_0 - \beta_0 x_i}}, \tag{2.15}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product, and the time is scaled to one using $t \in [0, 1]$.

Parameters b_0 and β_0 are not learned through optimization but determined by fitting the final condition for Theorem 2.4, which imposes $\widehat{u}(1, \mathbf{x}; \boldsymbol{\beta}) \approx g^2(\mathbf{x}) = \mathbf{1}_{\{x_i > \gamma\}}$. Therefore, the discontinuous indicator function is approximated by a sigmoid, and the fit is characterized by the position of the sigmoid's inflection point and the sharpness of the slope. The position and value of local and global minima with respect to the learned parameters $\boldsymbol{\beta}^{space}$ and β^{time} depend on the choices for b_0 and β_0.

To derive IS parameters from the ansatz function, we use the previous SOC result from (2.13), i.e.,

$$\widehat{\delta}_j^{\Delta t}(n, \mathbf{x}; \boldsymbol{\beta}) = \frac{a_j(\mathbf{x})\sqrt{\widehat{u}\left(\frac{(n+1)\Delta t}{T}, \max(0, \mathbf{x}+\nu_j); \boldsymbol{\beta}\right)}}{\sqrt{\widehat{u}(\frac{(n+1)\Delta t}{T}, \mathbf{x}; \boldsymbol{\beta})}},$$
$$1 \leq j \leq J,\ 0 \leq n \leq N-1,\ \mathbf{x} \in \mathbb{N}^d. \tag{2.16}$$

We define $\widehat{u}(t, \cdot; \cdot)$ in (2.15) as a time-continuous function for $t \in [0, 1]$; whereas the IS controls from $\widehat{\delta}_j^{\Delta t}(n, \cdot; \cdot)$ are discrete in time for $n = 0, \ldots, N-1$, and depend on time step size Δt. Therefore, $\widehat{u}(\cdot, \cdot; \boldsymbol{\beta})$ can be used to derive control parameters for arbitrary Δt in (2.16).

The parameters $\boldsymbol{\beta}$ for the ansatz function are then chosen to minimize the second moment,

$$\inf_{\boldsymbol{\beta} \in \mathbb{R}^{d+1}} \underbrace{\mathbb{E}\left[g^2\left(\overline{\mathbf{X}}_N^{\Delta t, \boldsymbol{\beta}}\right) \prod_{k=0}^{N-1} L_k^2\left(\overline{\mathbf{P}}_k, \widehat{\boldsymbol{\delta}}^{\Delta t}(k, \overline{\mathbf{X}}_k^{\Delta t, \boldsymbol{\beta}}; \boldsymbol{\beta})\right)\right]}_{=:C_{0,\mathbf{x}}\left(\widehat{\boldsymbol{\delta}}_0^{\Delta t}, \ldots, \widehat{\boldsymbol{\delta}}_{N-1}^{\Delta t}; \boldsymbol{\beta}\right)}, \tag{2.17}$$

where $\{\overline{\mathbf{X}}_n^{\Delta t, \boldsymbol{\beta}}\}_{n=1,\ldots,N}$ is the IS path generated using IS parameters from (2.16) and $\left(\widehat{\boldsymbol{\delta}}^{\Delta t}(n, \mathbf{x}; \boldsymbol{\beta})\right)_j = \widehat{\delta}_j^{\Delta t}(n, \mathbf{x}; \boldsymbol{\beta})$ for $1 \leq j \leq J$.

We use a gradient based stochastic optimizer method to solve (2.17), and derive Lemma 2.9 (proof in "Appendix B") for the gradient of the second moment with respect to parameters $\boldsymbol{\beta}$.

Lemma 2.9 *The partial derivatives for the second moment* $C_{0,\mathbf{x}}\left(\widehat{\boldsymbol{\delta}}_0^{\Delta t}, \ldots, \widehat{\boldsymbol{\delta}}_{N-1}^{\Delta t}; \boldsymbol{\beta}\right)$ *in* (2.17) *with respect to* β_l, $l = 1, \ldots, (d+1)$, *are given by*

$$\frac{\partial}{\partial \beta_l} \mathbb{E}\left[\underbrace{g^2\left(\overline{\boldsymbol{X}}_N^{\Delta t, \boldsymbol{\beta}}\right) \prod_{k=0}^{N-1} L_k^2\left(\overline{\boldsymbol{P}}_k, \widehat{\boldsymbol{\delta}}^{\Delta t}(k, \overline{\boldsymbol{X}}_k^{\Delta t, \boldsymbol{\beta}}; \boldsymbol{\beta})\right)}_{=:R(\boldsymbol{x}_0; \boldsymbol{\beta})}\right]$$
$$= \mathbb{E}\left[R(\boldsymbol{x}_0; \boldsymbol{\beta})\left(\sum_{k=1}^{N-1}\sum_{j=1}^{J}\left(\Delta t - \frac{\overline{P}_{k,j}}{\widehat{\delta}_j^{\Delta t}(k, \overline{\boldsymbol{X}}_k^{\Delta t, \boldsymbol{\beta}}; \boldsymbol{\beta})}\right) \cdot \frac{\partial}{\partial \beta_l}\widehat{\delta}_j^{\Delta t}(k, \overline{\boldsymbol{X}}_k^{\Delta t, \boldsymbol{\beta}}; \boldsymbol{\beta})\right)\right], \tag{2.18}$$

where $\{\overline{\mathbf{X}}_n^{\Delta t, \boldsymbol{\beta}}\}_{n=1,\ldots,N}$ *is the IS path generated using the IS parameters from* (2.16) *and*

$$\frac{\partial}{\partial \beta_l}\widehat{\delta}_j^{\Delta t}(k, \boldsymbol{x}; \boldsymbol{\beta})$$
$$= \frac{a_j^2(\boldsymbol{x})}{2\widehat{\delta}_j^{\Delta t}(k, \boldsymbol{x}; \boldsymbol{\beta})} \cdot \left(\frac{\frac{\partial}{\partial \beta_l}\widehat{u}(\frac{(k+1)\Delta t}{T}, \max(\boldsymbol{x}+\nu_j, 0); \boldsymbol{\beta})}{\widehat{u}(\frac{(k+1)\Delta t}{T}, \boldsymbol{x}; \boldsymbol{\beta})}\right.$$
$$\left. - \frac{\widehat{u}(\frac{(k+1)\Delta t}{T}, \max(\boldsymbol{x}+\nu_j, 0); \boldsymbol{\beta})\frac{\partial}{\partial \beta_l}\widehat{u}(\frac{(k+1)\Delta t}{T}, \boldsymbol{x}; \boldsymbol{\beta})}{\widehat{u}^2(\frac{(k+1)\Delta t}{T}, \boldsymbol{x}; \boldsymbol{\beta})}\right). \tag{2.19}$$

Thus, partial derivatives for $\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta})$ for the ansatz (2.15) are

$$\frac{\partial}{\partial \beta_l}\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta})$$
$$= \begin{cases} (1-t)x_i\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta})(1-\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta})) & \text{, if } \beta_l = \left(\boldsymbol{\beta}^{space}\right)_i \\ (1-t)\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta})(1-\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta})) & \text{, if } \beta_l = \beta^{time}, \end{cases} \tag{2.20}$$

where $\left(\boldsymbol{\beta}^{space}\right)_i$ denotes the i-th entry for $\boldsymbol{\beta}^{space}$.

For an ansatz function different from (2.15), the gradient is still given by Lemma 2.9 only the derivation of $\frac{\partial}{\partial \beta_l}\widehat{u}(t, \mathbf{x}; \boldsymbol{\beta})$ in (2.20) changes accordingly.

By estimating the gradient in (2.18) using a MC estimator, we iteratively optimize the parameters $\boldsymbol{\beta}$ to reduce the variance. For this optimization, we use the Adam optimizer with the same parameter values suggested in [34] with the only difference that the step size is tuned to fit our problem setting.

In Sect. 3, we illustrate the potential of our new IS method based on the learning approach numerically in terms of variance reduction. Further theoretical and numerical analysis of this approach is left for future work, particularly the initialization for the learned parameters β^{time} and $\boldsymbol{\beta}^{space}$ in (2.15) and investigations of a stopping rule.

To derive an estimator for $\mathbb{E}[g(\mathbf{X}(T))]$ using the proposed IS change of measure, we first solve the related SOC problem using the approach from this section; then we simulate M paths under the new IS sampling measure. Thus, the MC estimator using the proposed IS change of measure over M paths becomes

$$\mu_{M,\Delta t}^{IS} = \frac{1}{M}\sum_{i=1}^{M} L_i \cdot g(\overline{\mathbf{X}}_{[i],N}^{\Delta t, \boldsymbol{\beta}}), \tag{2.21}$$

where $\overline{\mathbf{X}}_{[i],N}^{\Delta t, \boldsymbol{\beta}}$ is the i-th IS sample path and the corresponding likelihood factor from (2.5) is

$$L_i = L\left(\left(\overline{\mathbf{P}}_0, \ldots, \overline{\mathbf{P}}_{N-1}\right), \left(\widehat{\boldsymbol{\delta}}^{\Delta t}(0, \overline{\mathbf{X}}_{[i],0}^{\Delta t, \boldsymbol{\beta}}; \boldsymbol{\beta}), \ldots, \widehat{\boldsymbol{\delta}}^{\Delta t}(N-1, \overline{\mathbf{X}}_{[i],N-1}^{\Delta t, \boldsymbol{\beta}}; \boldsymbol{\beta})\right)\right). \tag{2.22}$$

Remark 2.10 The explicit pathwise derivatives in Lemma 2.9 have the following advantages compared with the finite difference approach: (i) the explicit pathwise derivatives are unbiased with respect to the TL scheme, resulting in only the MC error for evaluating the expectation (i.e., without additional finite difference error), and (ii) the gradient computation in (2.18) requires the estimation of an expected value with a high relative error because of g being fitted to an indicator function. Using the IS-TL paths we control better the related statistical error.

2.4 Computational cost for the learning-based approach

This section discusses the computational complexity for the learning approach to achieve a prescribed tolerance TOL. Recall that the proposed approach comprises two steps; hence, two types of costs occur: (i) the offline learning cost for the ansatz function parameters $\boldsymbol{\beta}$, and (ii) the online cost to obtaining the MC estimator (2.21) based on M simulated paths using the derived IS measure (see (2.16)).

The offline cost for (i) can be expressed as

$$W_{pl}(I, M_0, \Delta t_{pl}) \approx I \cdot M_0 \cdot \frac{T}{\Delta t_{pl}} \cdot J \cdot (C_{Poi} + C_{grad}),$$

where I is the number of optimizer steps, M_0 is the number of paths needed to derive the estimator of the gradient per optimizer step, C_{Poi} is the cost to generate one Poisson random variable, C_{grad} is the cost for the update of the algebraic evaluation of (2.18), and Δt_{pl} is the step size. In contrast to (2.14), this offline cost does not scale exponentially with dimension d.

The cost for one IS-TL path based on $\widehat{u}(\cdot, \cdot; \boldsymbol{\beta})$ is the same as for a TL path with negligible additional factors $C_{\widehat{\delta}}$ for evaluating (2.16) and C_{lik} for deriving the likelihood update, as given in (2.4),

$$W_{forward}(\Delta t_f) \approx \frac{T}{\Delta t_f} \cdot J \cdot (C_{Poi} + C_{lik} + C_{\widehat{\delta}}),$$

where Δt_f is the step size. Thus, total cost is

$$W_{IS-TL}(M, \Delta t_{pl}, \Delta t_f) \approx W_{pl}(I, M_0, \Delta t_{pl}) + M \cdot W_{forward}(\Delta t_f).$$

Following the same derivation as for (1.11)–(1.14), we choose $\Delta t_f = \frac{\text{TOL}}{2 \cdot C}$, where C is the constant from (1.9), to obtain total computational complexity to derive a prescribed tolerance TOL

$$W_{IS-TL}(\text{TOL}) = W_{pl}(I, M_0, \Delta t_{pl}) + const \cdot \frac{\text{Var}\left[g\left(\overline{\mathbf{X}}_N^{\Delta t, \boldsymbol{\beta}}\right) \cdot L\right]}{\text{TOL}^3}, \qquad (2.23)$$

where L is the likelihood factor corresponding to the IS path $\overline{\mathbf{X}}^{\Delta t, \boldsymbol{\beta}}$ (refer to (2.22)).

Our numerical simulations suggest that the amount of variance reduction achieved with the proposed approach is not related to Δt_{pl} (see Fig. 4). Therefore, we can achieve a low offline parameter learning cost ($W_{pl}(I, M_0, \Delta t_{pl})$) by using $\Delta t_{pl} \gg \Delta t_f$.

For comparison, Sect. 1.3 shows that the standard MC-TL approach has total computational complexity

$$W_{MC-TL}(TOL) = const_{TL} \cdot \frac{Var[g(\widehat{\mathbf{X}}_N^{\Delta t})]}{TOL^3}.$$

The proposed IS approach reduces this cost by variance reduction $\left(\text{Var}[g(\overline{\mathbf{X}}_N^{\Delta t}) \cdot L] \ll Var[g(\widehat{\mathbf{X}}_N^{\Delta t})]\right)$ (refer to Figs. 1, 2, 3). The TL variance becomes increasingly large in the asymptotic regime for very rare event probabilities, such that the additional cost $W_{pl}(I, M_0, \Delta t_{pl})$ for learning $\boldsymbol{\beta}$ in (2.23) becomes negligible. Therefore, we obtain $W_{IS-TL}(TOL) \ll W_{MC-TL}(TOL)$ in the rare event regime.

3 Numerical experiments and results

Through Examples 3.1, 3.2, and 3.3, we demonstrate the advantages for the proposed IS approach compared with the standard MC approach. We numerically show that the proposed approach achieves substantial variance reduction compared with standard MC estimators when applied to SRNs with different dimensions.

Example 3.1 (*Pure decay*) This example considers one species and a single reaction,

$$X \xrightarrow{\theta_1} \emptyset,$$

where $\theta_1 = 1$, and the final time $T = 1$. Thus, the propensity is $a(x) = \theta_1 x$, the stoichiometric vector is $\nu = -1$, and the observable is $g(x) = \mathbf{1}_{\{x>50\}}$ with $X_0 = 100$.

Example 3.2 (*Michaelis–Menten enzyme kinetics*) The Michaelis-Menten enzyme kinetics [43] describe the catalytic conversion of substrate S into a product P through three reactions,

$$E + S \xrightarrow{\theta_1} C, \quad C \xrightarrow{\theta_2} E + S, \quad C \xrightarrow{\theta_3} E + P,$$

where E denotes the enzyme and $\theta = (0.001, 0.005, 0.01)^\top$. We consider the initial state $\mathbf{X}_0 = (E(0), S(0), C(0), P(0))^\top = (100, 100, 0, 0)^\top$ and the final time $T = 1$. The corresponding propensity and the change of the state matrix are

$$a(\mathbf{x}) = \begin{pmatrix} \theta_1 ES \\ \theta_2 C \\ \theta_3 C \end{pmatrix}, \quad \boldsymbol{\nu} = \begin{pmatrix} -1 & 1 & 1 \\ -1 & 1 & 0 \\ 1 & -1 & -1 \\ 0 & 0 & 1 \end{pmatrix}.$$

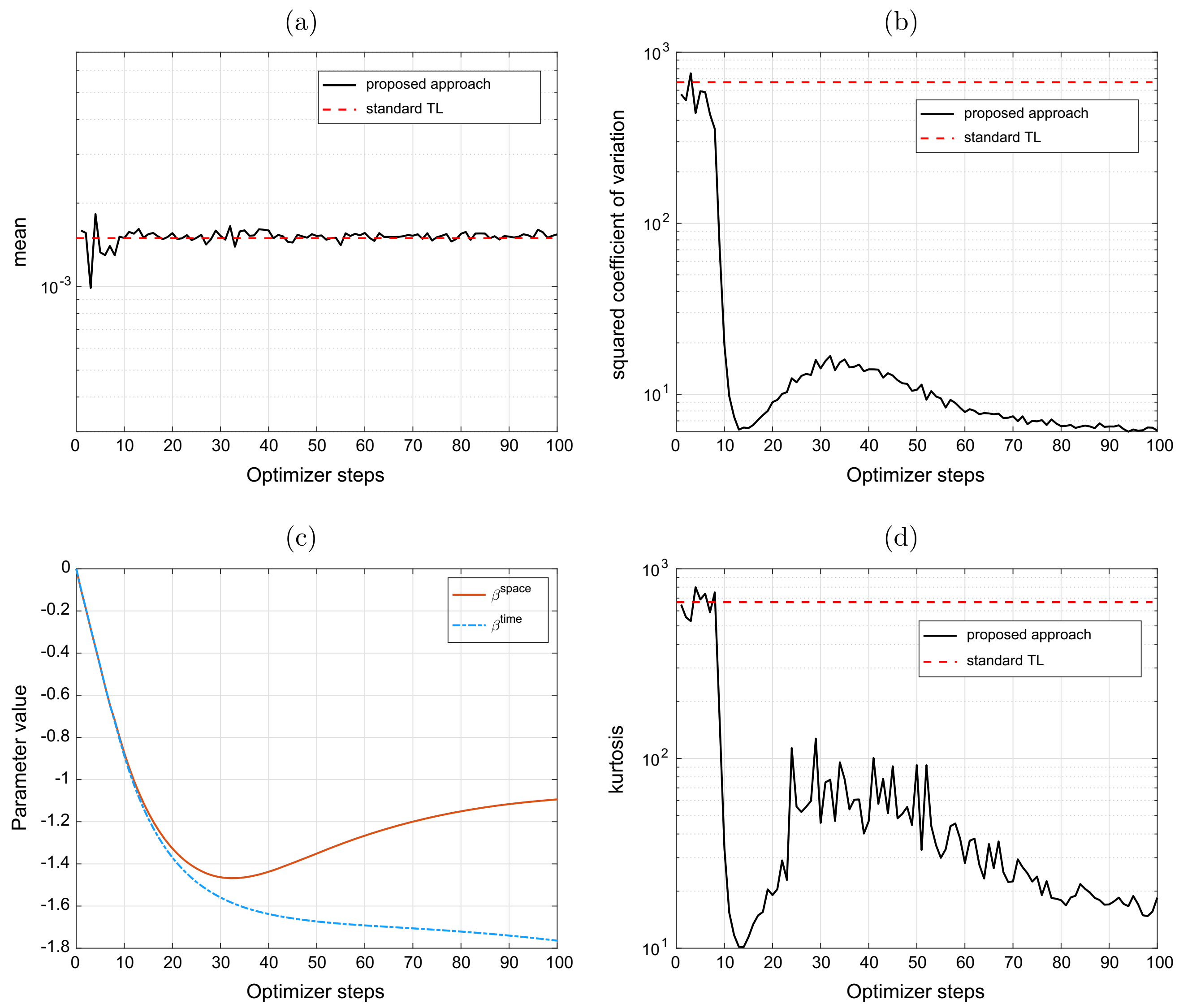

Fig. 1 Example 3.1 with step size $\Delta t_{pl} = \Delta t_f = 1/2^4$ for the proposed IS-MC estimator: **a** sample mean; **b** squared coefficient of variation; **c** parameters; **d** kurtosis for each optimizer step. Adam optimizer gradient, sample variance, and kurtosis were estimated using $M_0 = 10^4$ samples. The reference value for the standard MC-TL approach was derived from a single run with $M = 10^6$ samples and with step size $\Delta t = 1/2^4$

The observable of interest is $g(\mathbf{x}) = \mathbf{1}_{\{x_3>22\}}$.

Example 3.3 (*Enzymatic futile cycle model*) The enzymatic futile cycle [36] describes two instances for the elementary single-substrate enzymatic reaction scheme and can be described by six reactions,

$$
\begin{aligned}
&R_1 : S_1 + S_2 \xrightarrow{\theta_1} S_3, \quad R_2 : S_3 \xrightarrow{\theta_2} S_1 + S_2,\\
&R_3 : S_3 \xrightarrow{\theta_3} S_1 + S_5,\\
&R_4 : S_4 + S_5 \xrightarrow{\theta_4} S_6, \quad R_5 : S_6 \xrightarrow{\theta_5} S_4 + S_5,\\
&R_6 : S_6 \xrightarrow{\theta_6} S_4 + S_2.
\end{aligned}
$$

Initial states are $\mathbf{X}(0) = (S_1(0), \ldots, S_6(0)) = (1, 50, 0, 1, 50, 0)$, and we take the rates as $\theta_1 = \theta_2 = \theta_4 = \theta_5 = 1$, and $\theta_3 = \theta_6 = 0.1$. The propensity $a(\mathbf{x})$ follows the stochastic mass-action kinetics in (1.5) and the final time is $T = 2$. We consider $g(\mathbf{x}) = \mathbf{1}_{\{x_5>60\}}$ as the observable.

Since all three are rare event examples with observable $g(\mathbf{x}) = \mathbf{1}_{\{x_i>\gamma\}}$, we use the ansatz function (2.15) with initial parameters $\boldsymbol{\beta}^{space} = 0$, and $\beta^{time} = 0$. The relative error is more relevant for rare event occurrences than the absolute error, hence we use a relative version of the variance, i.e., the squared coefficient of variation [12,35], which, for a random

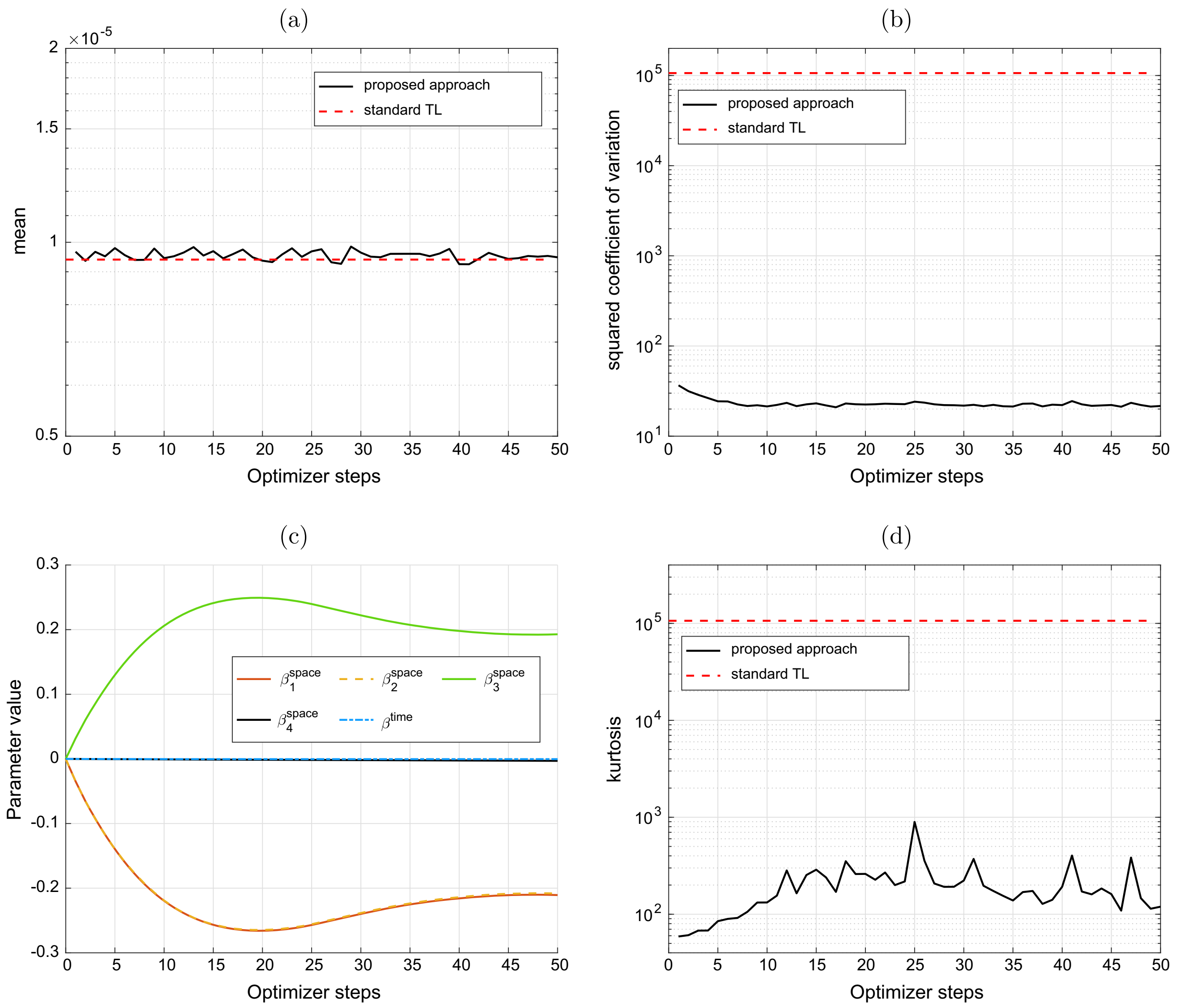

Fig. 2 Example 3.2 with step size $\Delta t_{pl} = \Delta t_f = 1/2^4$ for the proposed IS-MC estimator: **a** sample mean; **b** squared coefficient of variation; **c** parameters; **d** kurtosis for each optimizer step. The gradient for the Adam optimization, the sample variance, and the kurtosis were estimated using $M_0 = 10^5$ samples. Standard MC-TL with step size $\Delta t = 1/2^4$ and $M = 10^7$ samples was used for comparison

variable X, is given by

$$Var_{rel}[X] = \frac{Var[X]}{\mathbb{E}[X]^2}. \tag{3.1}$$

To judge the robustness of our variance estimators, we estimate the kurtosis, $\kappa := \frac{\mathrm{E}\left[(X-E[X])^4\right]}{(\mathrm{Var}[X])^2}$, because the standard deviation of the sample variance [10] is given by

$$\sigma_{\mathcal{S}^2(X)} = \frac{\mathrm{Var}\,[X]}{\sqrt{M}}\sqrt{(\kappa - 1) + \frac{2}{M-1}},$$

where M is the number of samples.

We set the Adam optimizer step size $\alpha = 0.1$ for the simulations.

Figure 1 shows 100 Adam optimization steps for the decay example (Example 3.1) for step size $\Delta t_{pl} = \Delta t_f = 1/2^4$. The quantity of interest is a rare event probability with magnitude 10^{-3}. To estimate the gradient, we use $M = 10^4$ samples per Adam iteration. The squared coefficient of variation is reduced by a factor of 10^2 compared with the standard MC-TL variance after 13 Adam iterations. After reaching this minimum, the squared coefficient of variation increases for the next iteration steps. This behavior might be avoided by

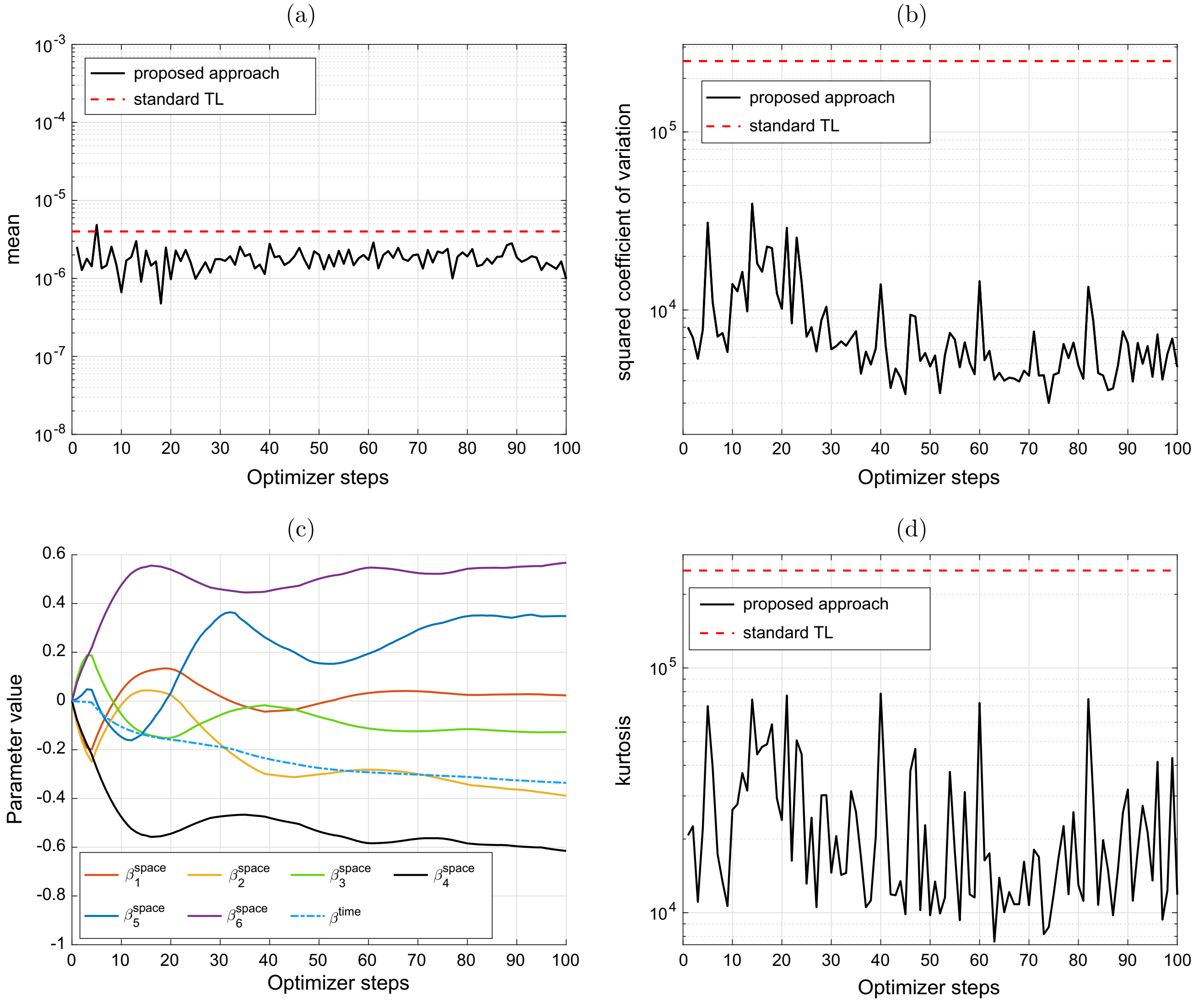

Fig. 3 Example 3.3 with step size $\Delta t_{pl} = \Delta t_f = 1/2^4$ for the proposed IS-MC estimators: **a** sample mean; **b** squared coefficient of variation; **c** parameters; **d** kurtosis for each optimizer step. The gradient for Adam optimization, the sample variance, and the kurtosis were estimated using $M_0 = 10^5$ samples. Standard MC-TL with $M = 10^6$ samples and step size $\Delta t = 1/2^4$ was used for comparison

employing a smaller step size in the Adam algorithm. Figure 1d confirms that the kurtosis is bounded to a level below the standard TL's kurtosis, indicating a robust variance estimator.

For the 4-dimensional stochastic reaction network (Example 3.2), the rare event probability for the event $\{X_3(T) > 22\}$ is of magnitude 10^{-5}. Figure 2b confirms that the proposed learning-based approach reduces the variance by a factor 4×10^3 compared with standard TL for step size $\Delta t_{pl} = \Delta t_f = 1/2^4$. Although Fig. 2c seems to shows that parameters β^{time} and β_4^{space} overlap, this is an artifact from the scale of the y-axis; in fact, the final values are $\beta_{time} = -3.2 \times 10^{-4}$ and $\beta_4^{space} = -3.0 \times 10^{-3}$. The intrinsic structure of Example 3.2 results in similar molecule counts for $E(t)$ and $S(t)$ and hence similar values for β_1^{space} and β_2^{space}. Figure 2d confirms that the kurtosis for the proposed approach is substantially reduced compared with the kurtosis for the standard TL approach.

The 6-dimensional example (Example 3.3) has a rare event probability with magnitude 10^{-6}. Figure 3 shows the Adam optimization results for step size $\Delta t_{pl} = \Delta t_f = 1/2^4$. The TL mean differs from the mean for the proposed approach (Fig. 3a) because the standard MC-TL estimator requires more than 10^6 runs to accurately estimate a probability of order 10^{-6}. The proposed learning-based approach reduces the variance by a factor of more than 50 after 43 iterations. The kurtosis is bounded and lower than the kurtosis for the

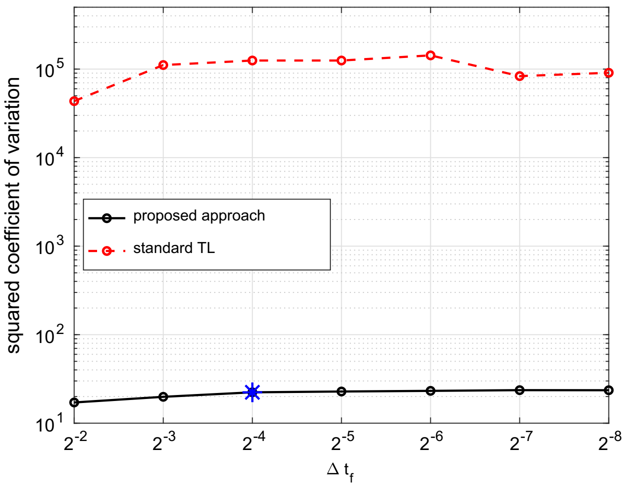

Fig. 4 Example 3.2, parameters $\boldsymbol{\beta}^{space}$ and β^{time} learned with $\Delta t_{pl} = 1/2^4$ (see final optimizer step in Fig. 2) and applied to forward runs with different Δt_f values. The squared coefficient of variation was estimated with $M = 10^6$ sample paths. The standard MC-TL approach is used as reference (*dashed red line*)

TL approach, confirming that the proposed approach results in a robust variance estimator.

Examples 3.2 and 3.3 show that a good choice of the ansatz function in combination with reasonable initial parameters provides substantial variance reduction from the first optimization step. However, we do not expect this behavior in general, particularly for high dimensions, and therefore we performed some optimization iterations.

The examples used step size $\Delta t_{pl} = 1/2^4$ and showed the squared coefficient of variation with respect to the same step size. To demonstrate that the learned parameters $\boldsymbol{\beta}$ can be used for forward runs with smaller step sizes (i.e., $\Delta t_f \ll \Delta t_{pl}$) as claimed in Sect. 2.4, we consider Example 3.2 and the final parameters from Fig. 4 for forward runs with different Δt_f. The results show that the variance reduction is constant with respect to Δt_f, suggesting that a coarse Δt_{pl} is sufficient for parameter learning. The same behavior was observed for other tested examples.

Remark 3.4 We used the ansatz (2.15) based on a single sigmoid for the numerical experiments to demonstrate the potential for the proposed learning-based IS. Further variance reduction may be achieved either by summing several sigmoid functions as ansatz or selecting a different basis function shape. Relevant analyses will be pursued in future work.

4 Conclusions and future work

This work developed an efficient path-dependent IS scheme to estimate statistical quantities for SRN processes, particularly rare event probabilities. Optimal IS parameters were obtained within a pre-selected class of change of measure using the proposed connection to an associated SOC problem, which could be solved via dynamic programming. To mitigate the curse of dimensionality encountered by the dynamic programming relation, we proposed a method for multi-dimensional SRNs based on approximating the value function via an ansatz function (i.e. a neural network), where the parameters were learned using a stochastic optimization algorithm. Numerical examples and subsequent analyses verified that the proposed estimator achieved substantial variance reduction compared with the standard MC method, providing lowered computational complexity in the rare event regime.

Future work will further analyze the proposed learning-based approach and expand it to derive a multilevel MC estimator. We also plan to combine an implementation of the dynamic programming principle as derived in Sect. 2.2 with dimension reduction methods for SRNs.

Acknowledgements This publication is based upon work supported by the King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research (OSR) under Award No. OSR-2019-CRG8-4033. This work was partially performed as part of the Helmholtz School for Data Science in Life, Earth and Energy (HDS-LEE) and received funding from the Helmholtz Association of German Research Centres and the Alexander von Humboldt Foundation.

Funding Open Access funding enabled and organized by Projekt DEAL.

Appendix A: Proof for Theorem 2.4

Proof for Theorem 2.4 To show (2.9), we first reformulate $C_{n,x}(\boldsymbol{\delta}_n^{\Delta t}, \ldots, \boldsymbol{\delta}_{N-1}^{\Delta t})$ using the definition for the likelihood and the notion of conditional expectation,

$$\begin{aligned}
&C_{n,x}(\boldsymbol{\delta}_n^{\Delta t}, \ldots, \boldsymbol{\delta}_{N-1}^{\Delta t}) \\
&\quad = \mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \prod_{k=n}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})) \mid \overline{X}_n^{\Delta t} = x \right] \\
&\quad = \mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \cdot L_n^2(\overline{\mathbf{P}}_n, \boldsymbol{\delta}_n^{\Delta t}(\overline{X}_n^{\Delta t})) \cdot \prod_{k=n+1}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})) \mid \overline{X}_n^{\Delta t} = x \right] \\
&\quad = \mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \cdot \exp\left(-2 \left(\sum_{j=1}^{J} a_j(\overline{X}_n^{\Delta t}) - \delta_{n,j}^{\Delta t}(\overline{X}_n^{\Delta t}) \right) \Delta t \right) \right.
\end{aligned}$$

$$\times \left(\prod_{j=1}^{J} \frac{a_j(\overline{X}_n^{\Delta t})}{\delta_{n,j}^{\Delta t}(\overline{X}_n^{\Delta t})}\right)^{2\overline{P}_{n,j}} \cdot \prod_{k=n+1}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})) \mid \overline{X}_n^{\Delta t} = x\Bigg]. \quad \text{(A.1)}$$

Setting

$$B(\overline{\mathbf{P}}_n) := g^2(\overline{X}_N^{\Delta t}) \exp\left(-2\left(\sum_{j=1}^{J} a_j(\overline{X}_n^{\Delta t}) - \delta_{n,j}^{\Delta t}(\overline{X}_n^{\Delta t})\right)\Delta t\right)$$
$$\left(\prod_{j=1}^{J} \frac{a_j(\overline{X}_n^{\Delta t})}{\delta_{n,j}^{\Delta t}(\overline{X}_n^{\Delta t})}\right)^{2\overline{P}_{n,j}} \prod_{k=n+1}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})),$$

we can reformulate (A.1) and derive

$$C_{n,x}(\boldsymbol{\delta}_n^{\Delta t}, \ldots, \boldsymbol{\delta}_{N-1}^{\Delta t})$$
$$= \sum_{\mathbf{p}\in\mathbb{N}^J} \mathbb{P}\left(\overline{\mathbf{P}}_n = \mathbf{p} \mid \overline{X}_n^{\Delta t} = x\right) \cdot \mathbb{E}\left[B(\overline{\mathbf{P}}_n) \mid \overline{X}_n^{\Delta t} = x, \overline{\mathbf{P}}_n = \mathbf{p}\right]$$
$$= \sum_{\mathbf{p}\in\mathbb{N}^J} \left[\prod_{j=1}^{J} \frac{(\Delta t \cdot \delta_{n,j}^{\Delta t}(x))^{p_j}}{p_j!} \exp(-\Delta t \cdot \sum_{j=0}^{J} \delta_{n,j}^{\Delta t}(x))\right]$$
$$\cdot \exp\left(-2\left(\sum_{j=1}^{J} a_j(x) - \delta_{n,j}^{\Delta t}(x)\right)\Delta t\right) \cdot \left(\prod_{j=1}^{J} \frac{a_j(x)}{\boldsymbol{\delta}_{n,j}^{\Delta t}(x)}\right)^{2p_j}$$
$$\cdot \mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \prod_{k=n+1}^{N-1} L_k^2\left(\overline{\mathbf{P}}_k, \delta_k^{\Delta t}(\overline{X}_k^{\Delta t})\right) \mid \overline{X}_n^{\Delta t} = x, \overline{\mathbf{P}}_n = \mathbf{p}\right]$$
$$= \exp\left(\left(-2\sum_{j=1}^{J} a_j(x) + \sum_{j=1}^{J} \delta_{n,j}^{\Delta t}(x)\right)\Delta t\right)$$
$$\cdot \sum_{\mathbf{p}\in\mathbb{N}^J} \left(\prod_{j=1}^{J} \frac{(\Delta t \cdot \delta_{n,j}^{\Delta t}(x))^{p_j}}{p_j!} (\frac{a_j(x)}{\delta_{n,j}^{\Delta t}(x)})^{2p_j}\right)$$
$$\cdot \mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \prod_{k=n+1}^{N-1} L_k^2\left(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})\right) \mid \overline{X}_{n+1}^{\Delta t} = \max(0, x + \mathbf{p}^T \boldsymbol{\nu})\right].$$

We can prove Theorem 2.4 using the above results. We split the proof into two parts, where the first inequality is obtained by

$$u_{\Delta t}(n, x) = \inf_{\{\boldsymbol{\delta}_i^{\Delta t}\}_{i=n,\ldots,N-1}\in\mathcal{A}^{N-n}}$$
$$\Bigg[\exp\left(\left(-2\sum_{j=1}^{J} a_j(x) + \sum_{j=1}^{J} \delta_{n,j}^{\Delta t}(x)\right)\Delta t\right)$$
$$\sum_{\mathbf{p}\in\mathbb{N}^J} \left(\left(\prod_{j=1}^{J} \frac{(\Delta t \cdot \delta_{n,j}^{\Delta t}(x))^{p_j}}{p_j!} (\frac{a_j(x)}{\delta_{n,j}^{\Delta t}(x)})^{2p_j}\right)\right.$$
$$\left.\times\mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \prod_{k=n+1}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})) \mid \overline{X}_{n+1}^{\Delta t} = \max(0, x + \mathbf{p}^T \nu)\right]\right)\Bigg]$$
$$\geq \inf_{\{\boldsymbol{\delta}_i^{\Delta t}\}_{i=n,\ldots,N-1}\in\mathcal{A}^{N-n}} \Bigg[\exp\left(\left(-2\sum_{j=1}^{J} a_j(x) + \sum_{j=1}^{J} \delta_{n,j}^{\Delta t}(x)\right)\Delta t\right)$$
$$\sum_{\mathbf{p}\in\mathbb{N}^J} \left(\left(\prod_{j=1}^{J} \frac{(\Delta t \cdot \delta_{n,j}^{\Delta t}(x))^{p_j}}{p_j!} (\frac{a_j(x)}{\delta_{n,j}^{\Delta t}(x)})^{2p_j}\right)\right.$$
$$\times \inf_{\{\boldsymbol{\delta}_k^{\Delta t}\}_{k=n+1,\ldots,N-1}\in\mathcal{A}^{N-n-1}} \mathbb{E}$$
$$\left.\left[g^2(\overline{X}_N^{\Delta t}) \prod_{k=n+1}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})) \mid \overline{X}_{n+1}^{\Delta t} = \max(0, x + \mathbf{p}^T \boldsymbol{\nu})\right]\right)\Bigg]$$

$$= \inf_{\boldsymbol{\delta}_n^{\Delta t}(x)\in\mathcal{A}_x} \Bigg[\exp\left(\left(-2\sum_{j=1}^{J} a_j(x) + \sum_{j=1}^{J} \delta_{n,j}^{\Delta t}(x)\right)\Delta t\right)$$
$$\times \sum_{\mathbf{p}\in\mathbb{N}^J} \left(\left(\prod_{j=1}^{J} \frac{(\Delta t \cdot \delta_{n,j}^{\Delta t}(x))^{p_j}}{p_j!} \left(\frac{a_j(x)}{\delta_{n,j}^{\Delta t}(x)}\right)^{2p_j}\right)\right.$$
$$\left.\cdot u_{\Delta t}(n+1, \max(0, x + \mathbf{p}^T \boldsymbol{\nu}))\right)\Bigg]$$

To prove the second inequality, we choose the control at the n-th time step to be an arbitrary $\boldsymbol{\delta}_n^{\Delta t,+} > 0$, and for the remaining controls, we choose the elements of a minimizing sequence of controls such that

$$\lim_{m\to\infty} \mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \prod_{k=n+1}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t,(m)}(\overline{X}_k^{\Delta t})) \mid \overline{X}_{n+1}^{\Delta t} = \max(0, x + \mathbf{p}^T \boldsymbol{\nu})\right]$$
$$= \inf_{\{\boldsymbol{\delta}_k^{\Delta t}\}_{k=n+1,\ldots,N-1}\in\mathcal{A}^{N-n-1}}$$
$$\mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \prod_{k=n+1}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})) \mid \overline{X}_{n+1}^{\Delta t} = \max(0, x + \mathbf{p}^T \boldsymbol{\nu})\right].$$

Therefore,

$$u_{\Delta t}(n, x)$$
$$= \inf_{\{\boldsymbol{\delta}_i^{\Delta t}\}_{i=n,\ldots,N-1}\in\mathcal{A}^{N-n}} \Bigg[\exp\left(\left(-2\sum_{j=1}^{J} a_j(x) + \sum_{j=1}^{J} \delta_{n,j}^{\Delta t}(x)\right)\Delta t\right)$$
$$\sum_{\mathbf{p}\in\mathbb{N}^J} \left(\left(\prod_{j=1}^{J} \frac{(\Delta t \cdot \delta_{n,j}^{\Delta t}(x))^{p_j}}{p_j!} (\frac{a_j(x)}{\delta_{n,j}^{\Delta t}(x)})^{2p_j}\right)\right.$$
$$\left.\times\mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \prod_{k=n+1}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})) \mid \overline{X}_{n+1}^{\Delta t} = \max(0, x + \mathbf{p}^T \boldsymbol{\nu})\right]\right)\Bigg]$$
$$\leq \exp\left(\left(-2\sum_{j=1}^{J} a_j(x) + \sum_{j=1}^{J} \delta_{n,j}^{\Delta t,+}(x)\right)\Delta t\right)$$
$$\sum_{\mathbf{p}\in\mathbb{N}^J} \left(\left(\prod_{j=1}^{J} \frac{(\Delta t \cdot \delta_{n,j}^{\Delta t,+}(x))^{p_j}}{p_j!} \left(\frac{a_j(x)}{\delta_{n,j}^{\Delta t,+}(x)}\right)^{2p_j}\right)\right.$$
$$\times \inf_{\{\boldsymbol{\delta}_k^{\Delta t}\}_{k=n+1,\ldots,N-1}\in\mathcal{A}^{N-n-1}}$$
$$\left.\mathbb{E}\left[g^2(\overline{X}_N^{\Delta t}) \prod_{k=n+1}^{N-1} L_k^2(\overline{\mathbf{P}}_k, \boldsymbol{\delta}_k^{\Delta t}(\overline{X}_k^{\Delta t})) \mid \overline{X}_{n+1}^{\Delta t} = \max(0, x + \mathbf{p}^T \boldsymbol{\nu})\right]\right)$$

$$= \exp\left(\left(-2\sum_{j=1}^{J} a_j(x) + \sum_{j=1}^{J} \delta_{n,j}^{\Delta t,+}(x)\right)\Delta t\right)$$
$$\times \sum_{\mathbf{p}\in\mathbb{N}^J} \left(\prod_{j=1}^{J} \frac{(\Delta t \cdot \delta_{n,j}^{\Delta t,+}(x))^{p_j}}{p_j!} \left(\frac{a_j(x)}{\delta_{n,j}^{\Delta t,+}(x)}\right)^{2p_j}\right.$$
$$\left.\cdot\, u_{\Delta t}(n+1, \max(0, x + \mathbf{p}^T \boldsymbol{\nu})).\right.$$

This inequality holds for any arbitrary $\boldsymbol{\delta}_n^{\Delta t,+} > 0$, and hence

$$u_{\Delta t}(n, x) \leq \inf_{\boldsymbol{\delta}_n^{\Delta t}(x)\in\mathcal{A}_x} \Bigg[\exp\left(\left(-2\sum_{j=1}^{J} a_j(x) + \sum_{j=1}^{J} \delta_{n,j}^{\Delta t}(x)\right)\Delta t\right)$$

$$\sum_{\mathbf{p}\in\mathbb{N}^J}\left(\prod_{j=1}^{J}\frac{(\Delta t\cdot\delta_{n,j}^{\Delta t}(x))^{p_j}}{p_j!}\left(\frac{a_j(x)}{\delta_{n,j}^{\Delta t}(x)}\right)^{2p_j}\right)$$
$$\left.\cdot\, u_{\Delta t}(n+1,\max(0,x+\mathbf{p}^T\boldsymbol{\nu}))\right].$$

This completes the proof. □

Appendix B: Proof for Lemma 2.9

The partial derivatives of the second moment $C_{0,\mathbf{x}}\left(\boldsymbol{\delta}_n^{\Delta t},\ldots,\boldsymbol{\delta}_{N-1}^{\Delta t};\boldsymbol{\beta}\right)$ in (2.7) with respect to β_l, $l=1,\ldots,(d+1)$ can be expressed as

$$\begin{aligned}
&\frac{\partial}{\partial\beta_l}\mathbb{E}\left[g^2\left(\overline{\mathbf{X}}_N^{\Delta t,\boldsymbol{\beta}}\right)\prod_{k=0}^{N-1}L_k^2\left(\overline{\mathbf{P}}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\overline{\mathbf{X}}_k^{\Delta t,\boldsymbol{\beta}};\boldsymbol{\beta})\right)\right]\\
&=\frac{\partial}{\partial\beta_l}\mathbb{E}\left[g^2\left(\widehat{\mathbf{X}}_N^{\Delta t}\right)\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right]\\
&\overset{(1)}{=}\mathbb{E}\left[\frac{\partial}{\partial\beta_l}\left(g^2\left(\widehat{\mathbf{X}}_N^{\Delta t}\right)\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)\right]\\
&\overset{(2)}{=}\mathbb{E}\left[g^2\left(\widehat{\mathbf{X}}_N^{\Delta t}\right)\frac{\partial}{\partial\beta_l}\left(\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)\right],
\end{aligned}\tag{B.1}$$

where the Poisson increments with respect to the TL measure are given in $\mathbf{P}_n$ with $(\mathbf{P}_n)_j := P_{n,j} = \mathcal{P}_{n,j}\left(a_j(\widehat{\mathbf{X}}_k^{\Delta t})\Delta t\right)$ for $j=1,\ldots,J$. In $\overset{(1)}{=}$, we assume that the expected value and the derivative commute (see [37] Assumption A1(1) for sufficient conditions). In $\overset{(2)}{=}$, we consider that $g^2\left(\widehat{\mathbf{X}}_N^{\Delta t}\right)$ is based on the original TL measure and hence is not dependent on β_l.

In (B.1), the term

$$\frac{\partial}{\partial\beta_l}\left(\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)$$

is only deterministically dependent on $\boldsymbol{\beta}$, since $\widehat{\mathbf{X}}_k^{\Delta t}$ is independent of $\boldsymbol{\beta}$, and $P_{k,j}\sim Poi(a_j(\widehat{\mathbf{X}}_k^{\Delta t})\Delta t)$. Thus, the derivative can be computed in a closed form using the identity

$$\frac{\partial}{\partial x}\ln(f(x))=\frac{1}{f(x)}\frac{\partial}{\partial x}f(x)\iff\frac{\partial}{\partial x}f(x)=f(x)\frac{\partial}{\partial x}\ln(f(x)).\tag{B.2}$$

We compute the derivative from (B.2) using the following steps.

1. Apply (B.2),

$$\begin{aligned}
&\frac{\partial}{\partial\beta_l}\left(\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)\\
&=\left(\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)\\
&\quad\frac{\partial}{\partial\beta_l}\ln\left(\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)\\
&=\left(\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)\\
&\quad\sum_{k=0}^{N-1}\frac{\partial}{\partial\beta_l}\ln\left(L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right).
\end{aligned}$$

2. The remaining derivative can be derived by chain rule,

$$\begin{aligned}
&\frac{\partial}{\partial\beta_l}\ln\left(L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)\\
&=\frac{1}{L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)}\frac{\partial}{\partial\beta_l}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right).
\end{aligned}$$

3. Apply a second chain rule,

$$\begin{aligned}
&\frac{\partial}{\partial\beta_l}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\\
&=\frac{\partial}{\partial\beta_l}\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\cdot\nabla_{\boldsymbol{\delta}}L_k(\mathbf{P}_k,\boldsymbol{\delta}).
\end{aligned}\tag{B.3}$$

4. In (B.3), we have from (2.4),

$$\begin{aligned}
L_k\left(\mathbf{P}_k,\boldsymbol{\delta}\right)&=\exp\left(-\left(\sum_{j=1}^{J}a_j(\widehat{\mathbf{X}}_k^{\Delta t})-\delta_j\right)\Delta t\right)\\
&\quad\cdot\prod_{j=1}^{J}\left(\frac{a_j(\widehat{\mathbf{X}}_k^{\Delta t})}{\delta_j}\right)^{P_{k,j}},
\end{aligned}\tag{B.4}$$

hence

$$\begin{aligned}
&\frac{\partial}{\partial\delta_i}L_k\left(\mathbf{P}_k,\boldsymbol{\delta}\right)\\
&=\Delta t\exp\left(-\left(\sum_{j=1}^{J}a_j(\widehat{\mathbf{X}}_k^{\Delta t})-\delta_j\right)\Delta t\right)\cdot\prod_{j=1}^{J}\left(\frac{a_j(\widehat{\mathbf{X}}_k^{\Delta t})}{\delta_j}\right)^{P_{k,j}}\\
&\quad+\exp\left(-\left(\sum_{j=1}^{J}a_j(\widehat{\mathbf{X}}_k^{\Delta t})-\delta_j\right)\Delta t\right)\\
&\quad\cdot(-P_{k,i})\frac{a_i^{P_{k,i}}}{\delta_i^{P_{k,i}+1}}\prod_{j=1,j\neq i}^{J}\left(\frac{a_j(\widehat{\mathbf{X}}_k^{\Delta t})}{\delta_j}\right)^{P_{k,j}}\\
&=\exp\left(-\left(\sum_{j=1}^{J}a_j(\widehat{\mathbf{X}}_k^{\Delta t})-\delta_j\right)\Delta t\right)\\
&\quad\prod_{j=1}^{J}\left(\frac{a_j(\widehat{\mathbf{X}}_k^{\Delta t})}{\delta_j}\right)^{P_{k,j}}\cdot\left(\Delta t-\frac{P_{k,i}}{\delta_i}\right)\\
&=L_k(\mathbf{P}_k,\boldsymbol{\delta})\cdot\left(\Delta t-\frac{P_{k,i}}{\delta_i}\right).
\end{aligned}$$

5. In (B.3), from (2.16), we obtain

$$\begin{aligned}\frac{\partial}{\partial\beta_l}\widehat{\delta}_j^{\Delta t}(k,\mathbf{x};\boldsymbol{\beta}) &= a_j(\mathbf{x})\frac{1}{2}\sqrt{\frac{\widehat{u}(\frac{(k+1)\Delta t}{T},\mathbf{x};\boldsymbol{\beta})}{\widehat{u}(\frac{(k+1)\Delta t}{T},\max(\mathbf{x}+\nu_j,0)}}\\ &\cdot\left(\frac{\frac{\partial}{\partial\beta_l}\widehat{u}(\frac{(k+1)\Delta t}{T},\max(\mathbf{x}+\nu_j,0);\boldsymbol{\beta})}{\widehat{u}(\frac{(k+1)\Delta t}{T},\mathbf{x};\boldsymbol{\beta})}\right.\\ &\left.-\frac{\widehat{u}(\frac{(k+1)\Delta t}{T},\max(\mathbf{x}+\nu_j,0);\boldsymbol{\beta})\frac{\partial}{\partial\beta_l}\widehat{u}(\frac{(k+1)\Delta t}{T},\mathbf{x};\boldsymbol{\beta})}{\widehat{u}(\frac{(k+1)\Delta t}{T},\mathbf{x};\boldsymbol{\beta})^2}\right)\\ &=\frac{a_j(\mathbf{x})^2}{2\widehat{\delta}_j^{\Delta t}(k,\mathbf{x};\boldsymbol{\beta})}\cdot\left(\frac{\frac{\partial}{\partial\beta_l}\widehat{u}(\frac{(k+1)\Delta t}{T},\max(\mathbf{x}+\nu_j,0);\boldsymbol{\beta})}{\widehat{u}(\frac{(k+1)\Delta t}{T},\mathbf{x};\boldsymbol{\beta})}\right.\\ &\left.-\frac{\widehat{u}(\frac{(k+1)\Delta t}{T},\max(\mathbf{x}+\nu_j,0);\boldsymbol{\beta})\frac{\partial}{\partial\beta_l}\widehat{u}(\frac{(k+1)\Delta t}{T},\mathbf{x};\boldsymbol{\beta})}{\widehat{u}(\frac{(k+1)\Delta t}{T},\mathbf{x};\boldsymbol{\beta})^2}\right)\end{aligned}\tag{B.5}$$

where $\frac{\partial}{\partial\beta_l}\widehat{u}_{\Delta t}(t,\mathbf{x};\boldsymbol{\beta})$ depends on the chosen ansatz.

Combining the previous steps, the gradient can be expressed as

$$\begin{aligned}&\frac{\partial}{\partial\beta_l}\left(\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)\\ &=\underbrace{\left(\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)}_{:=L(\widehat{\mathbf{X}}^{\Delta t};\boldsymbol{\beta})}\\ &\underbrace{\left(\sum_{k=1}^{N-1}\sum_{j=1}^{J}\left(\Delta t-\frac{P_{k,j}}{\widehat{\delta}_j^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})}\right)\cdot\frac{\partial}{\partial\beta_l}\widehat{\delta}_j^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)}_{:=S(\widehat{\mathbf{X}}^{\Delta t};\boldsymbol{\beta})},\end{aligned}\tag{B.6}$$

where the gradient of $\widehat{\delta}_j^{\Delta t}$ is dependent on the ansatz used and given by (B.5).

Since the MC estimator (B.1) may have a large variance, we again apply IS,

$$\begin{aligned}&\frac{\partial}{\partial\beta_l}\mathbb{E}\left[g^2\left(\overline{\mathbf{X}}_N^{\Delta t,\boldsymbol{\beta}}\right)\prod_{k=0}^{N-1}L_k^2\left(\overline{\mathbf{P}}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\overline{\mathbf{X}}_k^{\Delta t,\boldsymbol{\beta}};\boldsymbol{\beta})\right)\right]\\ &=\mathbb{E}\left[g^2\left(\widehat{\mathbf{X}}_N^{\Delta t}\right)\frac{\partial}{\partial\beta_l}\left(\prod_{k=0}^{N-1}L_k\left(\mathbf{P}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\widehat{\mathbf{X}}_k^{\Delta t};\boldsymbol{\beta})\right)\right)\right]\\ &\overset{IS}{=}\mathbb{E}\left[\left(\prod_{k=0}^{N-1}L_k\left(\overline{\mathbf{P}}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\overline{\mathbf{X}}_k^{\Delta t,\boldsymbol{\beta}};\boldsymbol{\beta})\right)\right)g^2\left(\overline{\mathbf{X}}_N^{\Delta t,\boldsymbol{\beta}}\right)\right.\\ &\left.\frac{\partial}{\partial\beta_l}\left(\prod_{k=0}^{N-1}L_k\left(\overline{\mathbf{P}}_k,\widehat{\boldsymbol{\delta}}^{\Delta t}(k,\overline{\mathbf{X}}_k^{\Delta t,\boldsymbol{\beta}};\boldsymbol{\beta})\right)\right)\right]\\ &=\mathbb{E}\left[L(\overline{\mathbf{X}}^{\Delta t,\boldsymbol{\beta}};\boldsymbol{\beta})^2\cdot g^2\left(\overline{\mathbf{X}}_N^{\Delta t,\boldsymbol{\beta}}\right)\cdot S(\overline{\mathbf{X}}^{\Delta t,\boldsymbol{\beta}};\boldsymbol{\beta})\right].\end{aligned}\tag{B.7}$$

References

1. Abdulle, A., Yucheng, H., Li, T.: Chebyshev methods with discrete noise: the τ-ROCK methods. J. Comput. Math. **28**, 195–217 (2010)
2. Ahn, T.-H., Sandu, A., Han, X. Implicit simulation methods for stochastic chemical kinetics (2013). arXiv:1303.3614
3. Anderson, D., Higham, D.: Multilevel Monte Carlo for continuous Markov chains, with applications in biochemical kinetics. SIAM Multiscale Model. Simul. **10**(1), 146–179 (2012)
4. Anderson, D.F.: A modified next reaction method for simulating chemical systems with time dependent propensities and delays. J. Chem. Phys. **127**(21), 214107 (2007)
5. Anderson, D.F., Kurtz, T.G.: Stochastic Analysis of Biochemical Systems, vol. 1. Springer, Berlin (2015)
6. Aparicio, J.P., Solari, H.G.: Population dynamics: Poisson approximation and its relation to the Langevin process. Phys. Rev. Lett. **86**(18), 4183 (2001)
7. Banisch, R., Hartmann, C. Meshless discretization of LQ-type stochastic control problems (2013). arXiv:1309.7497
8. Bayer, C., Moraes, A., Tempone, R., Vilanova, P.: An efficient forward-reverse expectation–maximization algorithm for statistical inference in stochastic reaction networks. Stoch. Anal. Appl. **34**(2), 193–231 (2016)
9. Ben Hammouda, C.: Hierarchical approximation methods for option pricing and stochastic reaction networks. Ph.D. Thesis (2020)
10. Ben Hammouda, C., Rached, N.B., Tempone, R.: Importance sampling for a robust and efficient multilevel Monte Carlo estimator for stochastic reaction networks. Stat. Comput. **30**(6), 1665–1689 (2020)
11. Ben Hammouda, C., Moraes, A., Tempone, R.: Multilevel hybrid split-step implicit tau-leap. Numer. Algorithms **74**(2), 527–560 (2017)
12. Ben Rached, N., Haji-Ali, A.-L., Rubino, G., Tempone, R.: Efficient importance sampling for large sums of independent and identically distributed random variables. Stat. Comput. **31**(6), 1–13 (2021)
13. Brauer, F., Castillo-Chavez, C.: Mathematical Models in Population Biology and Epidemiology, vol. 40. Springer, Berlin (2012)
14. Cao, Y., Petzold, L.: Trapezoidal tau-leaping formula for the stochastic simulation of biochemical systems. In: Proceedings of Foundations of Systems Biology in Engineering (FOSBE 2005), pp. 149–152 (2005)
15. Cao, Y., Liang, J.: Adaptively biased sequential importance sampling for rare events in reaction networks with comparison to exact solutions from finite buffer dCME method. J. Chem. Phys. **139**(2), 07B605_1 (2013)
16. Daigle Jr, B.J., Roh, M.K., Gillespie, D.T., Petzold, L.R.: Automated estimation of rare event probabilities in biochemical systems. J. Chem. Phys. **134**(4), 01B628 (2011)
17. Dupuis, P., Spiliopoulos, K., Wang, H.: Importance sampling for multiscale diffusions. Multiscale Model. Simul. **10**(1), 1–27 (2012)
18. Engblom, S.: On the stability of stochastic jump kinetics (2012). arXiv:1202.3892
19. Ethier, S.N., Kurtz, T.G.: Markov Processes: Characterization and Convergence. Wiley Series in Probability and Mathematical Statistics, Wiley, New York (1986)
20. Fleming, W.H., Soner, H.M.: Controlled Markov Processes and Viscosity Solutions, vol. 25. Springer, Berlin (2006)
21. Giles, M.B.: Multilevel Monte Carlo path simulation. Oper. Res. **56**(3), 607–617 (2008)
22. Giles, M.B.: Multilevel Monte Carlo methods. Acta Numer. **24**, 259–328 (2015)

23. Gillespie, C.S., Golightly, A.: Guided proposals for efficient weighted stochastic simulation. J. Chem. Phys. **150**(22), 224103 (2019)
24. Gillespie, D.T., Roh, M., Petzold, L.R.: Refining the weighted stochastic simulation algorithm. J. Chem. Phys. **130**(17), 174103 (2009)
25. Gillespie, D.T.: A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J. Comput. Phys. **22**(4), 403–434 (1976)
26. Gillespie, D.T.: Approximate accelerated stochastic simulation of chemically reacting systems. J. Chem. Phys. **115**(4), 1716–1733 (2001)
27. Gupta, A., Briat, C., Khammash, M.: A scalable computational framework for establishing long-term behavior of stochastic reaction networks. PLoS Comput. Biol. **10**(6), e1003669 (2014)
28. Hartmann, C., Banisch, R., Sarich, M., Badowski, T., Schütte, C.: Characterization of rare events in molecular dynamics. Entropy **16**(1), 350–376 (2014)
29. Hartmann, C., Kebiri, O., Neureither, L., Richter, L.: Variational approach to rare event simulation using least-squares regression. Chaos Interdisc. J. Nonlinear Sci. **29**(6), 063107 (2019)
30. Hartmann, C., Richter, L., Schütte, C., Zhang, W.: Variational characterization of free energy: theory and algorithms. Entropy **19**(11), 626 (2017)
31. Hartmann, C., Schütte, C., Weber, M., Zhang, W.: Importance sampling in path space for diffusion processes with slow-fast variables. Probab. Theory Relat. Fields **170**(1), 177–228 (2018)
32. Hensel, S.C., Rawlings, J.B., Yin, J.: Stochastic kinetic modeling of vesicular stomatitis virus intracellular growth. Bull. Math. Biol. **71**(7), 1671–1692 (2009)
33. Kebiri, O., Neureither, L., Hartmann, C. Adaptive importance sampling with forward-backward stochastic differential equations. In: International workshop on Stochastic Dynamics out of Equilibrium, pp. 265–281. Springer, Berlin (2017)
34. Kingma, D.P., Ba, J. Adam: A Method for Stochastic Optimization (2014). arXiv:1412.6980
35. Kroese, D.P., Taimre, T., Botev, Z.I.: Handbook of Monte Carlo Methods. Wiley, New York (2013)
36. Kuwahara, H., Mura, I.: An efficient and exact stochastic simulation method to analyze rare events in biochemical systems. J. Chem. Phys. **129**(16), 10B619 (2008)
37. L'Ecuyer, P.: Note: On the interchange of derivative and expectation for likelihood ratio derivative estimators. Manag. Sci. **41**(4), 738–747 (1995)
38. Lester, C., Yates, C.A., Giles, M.B., Baker, R.E.: An adaptive multi-level simulation algorithm for stochastic biological systems. J. Chem. Phys. **142**(2), 01B612_1 (2015)
39. Li, T.: Analysis of explicit tau-leaping schemes for simulating chemically reacting systems. Multiscale Model. Simul. **6**(2), 417–436 (2007)
40. Moraes, A., Tempone, R., Vilanova, P.: A multilevel adaptive reaction-splitting simulation method for stochastic reaction networks. SIAM J. Sci. Comput. **38**(4), A2091–A2117 (2016)
41. Nüsken, N., Richter, L.: Solving high-dimensional Hamilton–Jacobi–Bellman PDEs using neural networks: perspectives from the theory of controlled diffusions and measures on path space. Partial Differ. Equ. Appl. **2**(4), 1–48 (2021)
42. Rached, N.B., Haji-Ali, A.L., Mohan, S., Tempone, R.: Double Loop Monte Carlo Estimator with Importance Sampling for Mckean–Vlasov Stochastic Differential Equation (2022). arXiv:2207.06926
43. Rao, C.V., Arkin, A.P.: Stochastic chemical kinetics and the quasi-steady-state assumption: application to the Gillespie algorithm. J. Chem. Phys. **118**(11), 4999–5010 (2003)
44. Rathinam, M.: Moment Growth Bounds on Continuous Time Markov Processes on Non-negative Integer Lattices (2013). arXiv:1304.5169
45. Rathinam, M., El Samad, H.: Reversible-equivalent-monomolecular tau: a leaping method for small number and stiff stochastic chemical systems. J. Comput. Phys. **224**(2), 897–923 (2007)
46. Roh, M.K.: Data-driven method for efficient characterization of rare event probabilities in biochemical systems. Bull. Math. Biol. **81**(8), 3097–3120 (2019)
47. Roh, M.K., Gillespie, D.T., Petzold, L.R.: State-dependent biasing method for importance sampling in the weighted stochastic simulation algorithm. J. Chem. Phys. **133**(17), 174106 (2010)
48. Ranjan, S., You, L., Summers, J., Stochastic vs, J.Y.: deterministic modeling of intracellular viral kinetics. J. Theor. Biol. **218**(3), 309–321 (2002)
49. Zhang, W., Wang, H., Hartmann, C., Weber, M., Schütte, C.: Applications of the cross-entropy method to importance sampling and optimal control of diffusions. SIAM J. Sci. Comput. **36**(6), A2654–A2672 (2014)