Weak error analysis of tau-leaping methods for multi-scale stochastic chemical kinetic systems

David F. Anderson[1] and Masanori Koyama[2]

June 23, 2011

Abstract

A chemical reaction network is a chemical system involving multiple reactions and chemical species. The simplest stochastic models of such networks treat the system as a continuous time Markov chain with the state being the number of molecules of each species and with reactions modeled as possible transitions of the chain. For such models, there is typically a wide variation in temporal and other quantitative scales. In this multi-scale setting it is typically an extremely difficult task to perform approximations, such as Langevin approximations or law of large number type arguments, to simplify a system. Therefore, numerical methods oftentimes are the only reasonable means by which such models can be understood in real time.

In this paper we provide a general framework for understanding the weak error of numerical approximation techniques in the multi-scale setting. We quantify how the error of three different methods depends upon both the natural scalings within a given system, and with the step-size of the numerical method. Further, we introduce a new algorithm in this setting, the weak trapezoidal algorithm, which was developed originally as an approximate method for diffusion processes, and prove that the leading order of the error process scales with the square of the time discretization, making it the first second order method in this setting.

1 Introduction

A chemical reaction network is a chemical system involving multiple reactions and chemical species. The simplest stochastic models of such networks treat the system as a continuous time Markov chain with the state being the number of molecules of each species and with reactions modeled as possible transitions of the chain. For such models, there is typically a wide variation in scales in that the different species and reaction rates vary over several orders of magnitude. In this multi-scale setting it is typically an extremely difficult task to analytically perform approximations, such as Langevin approximations or law of large number type arguments, to simplify or reduce a system. In fact, this is an active research area [6, 18]. Therefore, numerical methods oftentimes are the only reasonable means by which such models can be understood in real time. This paper provides a general framework for understanding the weak error of numerical approximation techniques for stochastically modeled chemical reaction networks, and introduces the first second order method in this setting.

[1] Department of Mathematics, University of Wisconsin, Madison, Wi. 53706, anderson@math.wisc.edu, grant support from NSF-DMS-1009275.

[2] Department of Mathematics, University of Wisconsin, Madison, Wi. 53706, koyama@math.wisc.edu, grant support from NSF-DMS-1009275 and NSF-DMS-0805793.

[0] AMS 2000 subject classifications: Primary 60H35, 65C99; Secondary 92C40

More specifically, suppose we have a process X, and a function of the state of the system, f, giving us some quantity of interest, perhaps the abundance of a particular species. Supposing that Z is an approximation of X, we wish to quantify the behavior of

$$\mathbb{E}f(X(t)) - \mathbb{E}f(Z(t)). \tag{1.1}$$

To answer this question, we will demonstrate the need to take the natural scales of the problem into account.

We will show that Euler's method, also known as explicit τ-leaping in this setting, is an order one method, in that the leading order of the error (1.1) decreases linearly with the step-size. Also, we will show that the leading order of the error of the approximate midpoint method, introduced in [15], involves some terms that scale linearly and others that scale quadratically with the step-size. That is,

$$\mathbb{E}f(X(t)) - \mathbb{E}f(Z_M(t)) = c_1^N h + c_2^N h^2 + o(c_1^N h + c_2^N h^2),$$

where c_1^N, c_2^N depend upon the natural scalings of the system, quantified here by $N > 0$, and Z_M denotes the path generated by the approximate midpoint method. The term that dominates this error then depends upon the specific scalings of a system, encapsulated in the constants c_1^N and c_2^N, and the size of the time discretization h. We will extend the weak trapezoidal method of Anderson and Mattingly, which was originally formulated in the diffusive setting [5], to the discrete, stochastic chemical kinetic setting. We will prove that it is second order accurate in the step-size regardless of the scaling of the system.

This paper is best viewed as a merging and extension of [3], which provided the first error analysis in the stochastic chemical kinetic setting that took the natural scales of the system into account, and [5], which introduced and analyzed the weak trapezoidal algorithm in the diffusive setting.

1.1 Paper outline

The paper is organized as follows. In Section 2 we introduce the basic models to be considered, and discuss the relevant numerical methods. In Section 3, we discuss the multi-scale nature of stochastic models of chemical reaction networks, and incorporate this into our model. In Section 4, we show how bounds on the global weak errors of the different approximation methods can be expressed using respective bounds on the local, one-step errors. In Section 5, we provide the local, one-step analyses for the three methods considered in this paper. In Section 6, we provide results on how the behavior of the exact process affects the global error of the numerical method. In Section 7, we provide a brief discussion of the stability of the different explicit methods we have analyzed. We then finish with examples demonstrating our results in Section 8.

2 The basic model and numerical methods

2.1 The basic model

An example of a chemical reaction is

$$2S_1 + S_2 \ \to \ S_3,$$

where we would interpret the above as saying two molecules of type S_1 combine with a molecule of type S_2 to produce a molecule of type S_3. The S_i are called chemical *species*. Letting

$$\nu_1 = \begin{pmatrix} 2 \\ 1 \\ 0 \end{pmatrix}, \quad \nu_1' = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad \text{and} \quad \zeta_1 = \nu_1' - \nu_1 = \begin{pmatrix} -2 \\ -1 \\ 1 \end{pmatrix},$$

we see that every instance of the reaction changes the state of the system by addition of ζ_1. Here the subscript "1" is used to denote the first (and in this case only) reaction of the system.

In the general setting we denote the number of species by d, and for $i \in \{1, \dots, d\}$ we denote the ith species as S_i. We then consider a finite set of R reactions, where the model for the kth reaction is determined by

(*i*) a vector of inputs ν_k specifying the number of molecules of each chemical species that are consumed in the reaction,

(*ii*) a vector of outputs ν_k' specifying the number of molecules of each species that are created in the reaction, and

(*iii*) a function of the state λ_k that gives the *transition intensity* or rate at which the reaction occurs. (Note that in the chemical literature, transition intensities are referred to as *propensities.*)

Specifically, if we denote the state of the system at time t by $X(t) \in \mathbb{Z}^d_{\geq 0}$, and if the kth reaction occurs at time t, we update the state by addition of the *reaction vector*

$$\xi_k \stackrel{\text{def}}{=} \nu_k' - \nu_k$$

and the new state becomes $X(t) = X(t-) + \xi_k$. For the standard Markov chain model, the number of times that the kth reaction occurs by time t can be represented by the counting process $R_k(t) = Y_k\left(\int_0^t \lambda_k(X(s))ds\right)$, where the Y_k are independent unit-rate Poisson processes (see, for example, [20], [10] Chapter 6, or the recent survey [4]). The state of the system then satisfies the equation

$$X(t) = X(0) + \sum_k Y_k\left(\int_0^t \lambda_k(X(s))ds\right)\xi_k. \tag{2.1}$$

The above formulation is termed a "random time change representation" and is equivalent to the "chemical master equation representation" found in much of the biology and chemistry literature, where the master equation is Kolmogorov's forward equation in the terminology of probability.

A common choice of intensity function for chemical reaction systems, and the one we adopt throughout, is that of mass action kinetics. Under mass action kinetics, the intensity function for the kth reaction is

$$\lambda_k(x) = \kappa_k \prod_{i=1}^{d} \frac{x_i!}{(x_i - \nu_{ki})!}, \tag{2.2}$$

where, ν_{ki} is the ith component of ν_k. Implicit in the assumption of mass action kinetics is that the vessel under consideration is "well-stirred." Note that λ_k defined in (2.2) is a polynomial function, and is therefore defined, and infinitely differentiable. While the natural state space is the non-negative orthant, we extend λ_k to all of $\mathbb{R}^d$ by setting it to zero outside $\mathbb{R}^d_{\geq 0}$.

This model is a continuous time Markov chain in $\mathbb{Z}^d$ with generator

$$(\mathcal{A}f)(x) = \sum_k \lambda_k(x)(f(x+\zeta_k) - f(x)),$$

where $f : \mathbb{Z}^d \to \mathbb{R}$. Throughout the paper, in order to gain control of the terms in (1.1), we will make extensive use of Dynkin's formula,

$$\mathbb{E}_x f(X(t)) = f(x) + \int_0^t \mathbb{E}_x(\mathcal{A}f)X(s)ds,$$

for suitably chosen f [10], and where the above equality holds so long as the expectations exist.

Kolmogorov's forward equation for this model is

$$\frac{d}{dt}P(x,t|\pi) = \sum_k \lambda_k(x-\zeta_k)1_{\{x-\zeta_k\in\mathbb{Z}^d_{\geq 0}\}}P(x-\zeta_k,t|\pi) - \sum_k \lambda_k(x)P(x,t|\pi),$$

where for $x \in \mathbb{Z}^d_{\geq 0}$, $P(x,t|\pi)$ represents the probability that $X(t) = x$, conditioned upon the initial distribution π.

Example 1. *To solidify notation, we consider the network*

$$S_1 \underset{\kappa_2}{\overset{\kappa_1}{\rightleftarrows}} S_2, \qquad 2S_2 \overset{\kappa_k}{\to} S_3,$$

where we have placed the rate constants κ_k above or below their respective reactions. For this example, equation (2.1) *is*

$$\begin{aligned} X(t) = X(0) &+ Y_1\left(\int_0^t \kappa_1 X_1(s)ds\right)\begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix} + Y_2\left(\int_0^t \kappa_2 X_2(s)ds\right)\begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \\ &+ Y_3\left(\int_0^t \kappa_3 X_2(s)(X_2(s)-1)ds\right)\begin{bmatrix} 0 \\ -2 \\ 1 \end{bmatrix}. \end{aligned}$$

Defining $\zeta_1 = [-1,1,0]^T$, $\zeta_2 = [1,-1,0]^T$, and $\zeta_3 = [0,-2,1]^T$, the generator $\mathcal{A}$ satisfies

$$(\mathcal{A}f)(x) = \kappa_1 x_1(f(x+\zeta_1) - f(x)) + \kappa_2 x_2(f(x+\zeta_2) - f(x)) + \kappa_3 x_2(x_2-1)(f(x+\zeta_3) - f(x))$$

2.2 Numerical methods

Exact methods. There are a number of numerical methods that produce statistically exact sample paths for the model described above. These include the stochastic simulation algorithm, best known as Gillespie's algorithm [13, 14], the first reaction method [13], and the next reaction method [1, 12]. All such algorithms perform the same two basic steps multiple times until a sample path is produced over a desired time interval. Conditioned on the current state of the system, both (i) the amount of time that passes until the next reaction takes place, Δt, is computed and (ii) the specific reaction that has taken place is found. Note that Δt is an exponential random variable with a parameter of $\sum_k \lambda_k(X(t))$. Therefore, if

$$\sum_k \lambda_k(X(t)) \gg 1 \qquad \text{so that} \qquad \mathbb{E}\Delta t = \frac{1}{\sum_k \lambda_k(X(t))} \ll 1, \tag{2.3}$$

then the runtime needed to produce a single exact sample path may be prohibitive. As such methods are usually coupled with Monte Carlo techniques, this observation motivates the need to consider approximate simulation methods, which we now introduce.

Approximate methods. Throughout the paper, we let X denote the solution to (2.1). For approximate methods, we will sometimes consider an arbitrary method and will in such cases denote the solution as Z. Other times we will distinguish the Euler, midpoint, and Trapezoidal methods, by Z_E, Z_M, Z_{trap}, respectively. We will always begin the methods from the same initial condition, namely $X(0) = Z(0) = x_0$.

Remark 2.1. Historically the time discretization parameter for the methods described in this paper has been τ, thus giving these methods the general name "τ-leaping methods." We choose to break from this tradition and denote our time-step by h so as not to confuse τ with a stopping time.

The approximate algorithm termed *explicit tau-leaping* was developed by Dan Gillespie in [15] in an effort to overcome the problem described above that Δt, the amount of time that must pass before the next reaction event, may be prohibitively small. The basic idea of tau-leaping is to hold the intensity functions fixed over the time interval $[t_n, t_n + h]$ at the values $\lambda_k(X(t_n))$, where $X(t_n)$ is the current state of the system, and, under this assumption, compute the number of times each reaction takes place over this period. Note that this method will potentially yield lower runtimes only if $h \gg 1/\sum_k \lambda_k(X(t_n)) \approx \Delta t$. As the waiting times for the reactions are exponentially distributed this leads to the following algorithm, which simulates up to a time of $T > 0$. Below and in the sequel, for $x \geq 0$ we will write Poisson(x) for a Poisson random variable with a parameter of x.

Algorithm 1 (Euler tau-leaping). Fix $h > 0$. Set $Z_E(0) = x_0$, $t_0 = 0$, $n = 0$ and repeat the following until $t_{n+1} = T$:

- (i) Set $t_{n+1} = t_n + h$. If $t_{n+1} \geq T$, set $t_{n+1} = T$ and $h = T - t_n$.
- (ii) For $k \in \{1, \ldots, R\}$, let $\Lambda_k = \text{Poisson}(\lambda_k(Z_E(t_n))h)$ be independent of each other and all previous random variables.
- (iii) Set $Z_E(t_{n+1}) = Z_E(t_n) + \sum_k \Lambda_k \zeta_k$.
- (iv) Set $n \leftarrow n + 1$.

Several improvements and modifications have been made to the basic algorithm described above over the years. However, they are mainly concerned with how to choose the step-size adaptively [8, 16] and/or how to ensure that population values do not go negative during the course of a simulation [2, 7, 9], and are not explicitly relevant to the current discussion of convergence analysis.

Similar to (2.1), a path-wise representation of Euler tau-leaping defined for all $t \geq 0$ can be given through a random time change of Poisson processes:

$$Z_E(t) = Z_E(0) + \sum_k Y_k \left(\int_0^t \lambda_k(Z_E \circ \eta(s)) ds \right) \zeta_k, \tag{2.4}$$

where the Y_k are as before, and $\eta(s) \overset{\text{def}}{=} \left\lfloor \frac{s}{h} \right\rfloor h$. Thus, $Z_E(\eta(s)) = Z_E(t_n)$ if $t_n \leq s < t_{n+1}$. Noting that $\int_0^{t_{n+1}} \lambda_k(Z_E \circ \eta(s)) ds = \sum_{i=0}^n \lambda_k(Z_E(t_i))(t_{i+1} - t_i)$ explains why this method is called "Euler tau-leaping." Defining the operator

$$(\mathcal{B}_z f)(x) \overset{\text{def}}{=} \sum_k \lambda_k(z)(f(x+\zeta_k) - f(x)), \tag{2.5}$$

we see that for $t > 0$

$$\mathbb{E}f(Z_E(t)) = \mathbb{E}f(Z_E \circ \eta(t)) + \mathbb{E}\int_{\eta(t)}^t (\mathcal{B}_{Z_E\circ\eta(t)}f)(Z_E(s))ds, \tag{2.6}$$

so long as the expectations exist. Equation (2.6) points out why we care about the associated operators for each of our approximate methods: they will be used to gain the necessary control over the difference $\mathbb{E}f(X(t)) - \mathbb{E}f(Z(t))$, called the *weak error* of the approximation, which is the focus of our paper.

A midpoint type method was first described in [15][1] and analyzed in [3]. Define the function

$$\rho(z) \overset{\text{def}}{=} z + \frac{1}{2}h\sum_k \lambda_k(z)\zeta_k,$$

which computes an approximate midpoint for the system assuming the state of the system is z and the time-step is h. The following algorithm simulates up to a time of $T > 0$. Note that only step (ii) changes from Euler tau-leaping.

Algorithm 2 (Midpoint tau-leaping). Fix $h > 0$. Set $Z_M(0) = x_0$, $t_0 = 0$, $n = 0$ and repeat the following until $t_{n+1} = T$:

(i) Set $t_{n+1} = t_n + h$. If $t_{n+1} \geq T$, set $t_{n+1} = T$ and $h = T - t_n$.

(ii) For $k \in \{1, \dots, R\}$, let $\Lambda_k = \text{Poisson}(\lambda_k \circ \rho(Z_M(t_n))h)$ be independent of each other and all previous random variables.

(iii) Set $Z_M(t_{n+1}) = Z_M(t_n) + \sum_k \Lambda_k\zeta_k$.

(iv) Set $n \leftarrow n + 1$.

Similar to (2.1) and (2.4), $Z_M(t)$ constructed via Algorithm 2 can be represented for all $t \geq 0$ via a random time change of Poisson processes:

$$Z_M(t) = Z_M(0) + \sum_k Y_k\left(\int_0^t \lambda_k \circ \rho(Z_M(\eta(s)))ds\right)\zeta_k, \tag{2.7}$$

where $\eta(\cdot)$ is as before. For $\mathcal{B}_z$ defined via (2.5), any $t > 0$, and Z_M generated via (2.7) we have

$$\mathbb{E}f(Z_M(t)) = \mathbb{E}f(Z_M \circ \eta(t)) + \mathbb{E}\int_{\eta(t)}^t (\mathcal{B}_{\rho\circ Z_M\circ\eta(t)}f)(Z_M(s))ds,$$

so long as the expectations exist.

The weak trapezoidal method. We will now propose a trapezoidal type algorithm to approximate the solutions of (2.1). The method was originally introduced in the work of Anderson and

[1] The midpoint method detailed in [15] is actually a slight variant of the method described here. In [15] the approximate midpoint, called $\rho(z)$ above, is rounded to the nearest integer value.

Mattingly in the diffusive setting where it is best understood by using a path-wise representation that incorporates space-time white noise processes, see [5]. It can similarly be understood in the current setting of jump processes by using a representation that utilizes Poisson random measures. See Appendix A.

In the algorithm below, which simulates a path up to a time $T > 0$, it is notationally convenient to define $[x]^+ = x \vee 0 = \max\{x, 0\}$.

Algorithm 3 (Weak trapezoidal method). Fix $h > 0$. Set $Z(0) = x_0$, $t_0 = 0$, and $n = 0$. Fixing a $\theta \in (0, 1)$, we define

$$\xi_1 \overset{\text{def}}{=} \frac{1}{2}\frac{1}{\theta(1-\theta)} \qquad \text{and} \quad \xi_2 \overset{\text{def}}{=} \frac{1}{2}\frac{(1-\theta)^2+\theta^2}{\theta(1-\theta)}\,. \tag{2.8}$$

We repeat the following steps until $t_{n+1} = T$, in which we first compute a θ-midpoint y^*, and then the new value $Z_{trap}(t_{n+1})$:

(*i*) Set $t_{n+1} = t_n + h$. If $t_{n+1} \geq T$, set $t_{n+1} = T$ and $h = T - t_n$.

(*ii*) For $k \in \{1, \ldots, R\}$, let $\Lambda_{k,1} = \text{Poisson}(\lambda_k(Z_{trap}(t_n))\theta h)$ be independent of each other and all previous random variables.

(*iii*) Set $y^* = Z_{trap}(t_n) + \sum_k \Lambda_{k,1}\zeta_k$.

(*iv*) For $k \in \{1, \ldots, R\}$, let $\Lambda_{k,2} = \text{Poisson}([\xi_1\lambda_k(y^*) - \xi_2\lambda_k(t_n)]^+(1-\theta)h)$ be independent of each other and all previous random variables.

(*v*) Set $Z_{trap}(t_{n+1}) = y^* + \sum_k \Lambda_{k,2}\zeta_k$.

(*vi*) Set $n \leftarrow n + 1$.

Remark 2.2. Notice that on the $(n+1)$st-step, y^* is the Euler approximation to $X(nh + \theta h)$ starting from $Z_{trap}(t_n)$ at time nh.

Remark 2.3. Notice that for all $\theta \in (0, 1)$ one has $\xi_1 > \xi_2$ and $\xi_1 - \xi_2 = 1$.

We define the operator $\mathcal{B}_{z_1,z_2}$ by

$$(\mathcal{B}_{z_1,z_2}f)(x) \overset{\text{def}}{=} \sum_k [\xi_1\lambda_k(z_1) - \xi_2\lambda_k(z_2)]^+(f(x+\zeta_k) - f(x)).$$

Then, for $\eta(t) \leq t \leq \eta(t) + \theta h$, the process Z_{trap} satisfies

$$\mathbb{E}f(Z_{trap}(t)) = \mathbb{E}f(Z_{trap}(\eta(t))) + \mathbb{E}\int_{\eta(t)}^t (\mathcal{B}_{Z_{trap}(\eta(t))}f)(Z_{trap}(s))ds,$$

where we recall that $\mathcal{B}_z$ is defined via (2.5), and for $\eta(t) + \theta h \leq t \leq \eta(t) + h$, the process Z_{trap} satisfies

$$\mathbb{E}f(Z_{trap}(t)) = \mathbb{E}f(Z_{trap}(\eta(t)+\theta h)) + \mathbb{E}\int_{\eta(t)+\theta h}^t (\mathcal{B}_{Z_{trap}(\eta(t)+\theta h),Z_{trap}(\eta(t))}f)(Z_{trap}(s))ds.$$

2.3 Previous error analyses

Under the scaling $h \to 0$, Rathinam et al. performed a consistency check of Euler tau-leaping and found that the local truncation error was $O(h^2)$ for all moments [22]. They also showed that under this same scaling Euler tau-leaping is first order accurate in a weak sense in the case that the intensity functions λ_k are linear [22]. Li extended these results by showing that as $h \to 0$, Euler tau-leaping has a strong error (in the L^2 norm) of order 1/2 and a weak error of order one [21], which agree with classical results pertaining to numerical analysis of SDEs driven by Brownian motions (see, for example, [19]).

Under the scaling $h \to 0$ it is readily seen that midpoint tau-leaping is no more accurate than Euler tau-leaping. This follows since midpoint tau-leaping consists of making an $O(h^2)$ correction to the intensity functions used in Euler tau-leaping. As $h \to 0$, this correction becomes negligible as Poisson processes "ignore" $O(h^2)$ corrections, and the accuracy of the two methods will be the same.

However, in many examples the midpoint method is readily seen to be more accurate than Euler's method in the discrete stochastic setting. In [3], Anderson, Ganguly, and Kurtz provided an error analysis of Euler's method and the approximate midpoint method under the assumptions that (*i*) the system of interest satisfies the classical scaling described in Section 3 below and (*ii*) the time discretization satisfies the requirement

$$h \gg \frac{1}{\sum_k \lambda_k(Z(t))}, \tag{2.9}$$

where $Z(t)$ is the state of the system at time t. The requirement (2.9) is reasonable as such approximation methods would only be used in a regime where $h \gg \Delta t$, where Δt is the expected amount of time between reactions, for otherwise an exact method would be performed. They proved that, in this specific setting, Euler's method is an order one method in both a weak and a strong (in the L^1 norm) sense. They proved that the strong error of the midpoint method falls between order one and two (see [3] for precise statements), and that the weak error of the midpoint method scales quadratically with the step-size *when condition* (2.9) *is satisfied*. The importance of the analysis in [3] is that it pointed out the need to incorporate the natural scales of the system into the analysis.

3 Scaled models

As discussed in and around (2.3), the approximate algorithms being considered are only useful on the class of models which satisfy $\sum_k \lambda_k(X(\cdot)) \gg 1$. There are at least two different ways this behavior can be achieved. The first is that there could be a large number of reactions, $R \gg 1$, in which case the approximate algorithms currently being discussed will not provide an appreciable improvement in terms of runtime over the exact simulation methods. The other common way for $\sum_k \lambda_k(X(\cdot)) \gg 1$ to hold is to have either large abundances of certain species, or to have large rate constants, or both. We will study the behavior of the different algorithms under this latter assumption. To do so, we will introduce a scaling parameter, N, used to quantify the variations in the sizes of the abundances and parameters. We emphasize that the scaling detailed below is an analytical tool used to understand the behavior of the different processes, and that the actual simulations using the different methods make no use of, nor have need for, an understanding of N.

The specifics of the scaling used here have previously been used in [4, 6, 18]. Let $N \gg 1$. Assume that we are given a model of the form

$$X(t) = X(0) + \sum_k Y_k \left(\int_0^t \lambda_k'(X(s))ds \right) \zeta_k,$$

where the λ_k' are of the form

$$\lambda_k'(x) = \kappa_k' \prod_{i=1}^d \frac{x_i!}{(x_i - \nu_{ki})!},$$

and where we recall that $\zeta_k \overset{\text{def}}{=} \nu_k' - \nu_k$. For each species i, define the *normalized abundance* (or simply, the abundance) by

$$X_i^N(t) = N^{-\alpha_i} X_i(t),$$

where $\alpha_i \geq 0$ should be selected so that $X_i^N = O(1)$. Here X_i^N may be the species number ($\alpha_i = 0$) or the species concentration or something else.

Since the rate constants may also vary over several orders of magnitude, we write $\kappa_k' = \kappa_k N^{\beta_k}$ where the β_k are selected so that $\kappa_k = O(1)$. Note that for a binary reaction

$$\kappa_k' X_i X_j = N^{\beta_k + \alpha_i + \alpha_j} \kappa_k X_i^N X_j^N,$$

and we can write

$$\beta_k + \alpha_i + \alpha_j = \beta_k + \nu_k \cdot \alpha.$$

We also have,

$$\kappa_k' X_i = N^{\beta_k + \nu_k \cdot \alpha} \kappa_k X_i^N, \quad \kappa_k' X_i^N (X_i^N - 1) = N^{\beta_k + \nu_k \cdot \alpha} \kappa_k X_i^N (X_i^N - N^{-\alpha_i}),$$

where the source vectors are $\nu_k = e_i$ in the first example and $\nu_k = 2e_i$ in the second, with similar expressions for intensities involving higher order reactions. That is, under the mass-action kinetics assumption, we always have that $\lambda_k'(X(s)) = N^{\beta_k + \nu_k \cdot \alpha} \lambda_k(X^N(s))$, where λ_k is deterministic mass-action kinetics with rate constants κ_k. Note that for reactions of the form $2S_i \to *$, where $*$ represents an arbitrary linear combination of the species, the rate is $N^{\beta_k + 2\alpha_i} \kappa_k X_i^N(t)(X_i^N(t) - N^{-\alpha_i})$, so if $\alpha_i > 0$, we should write λ_k^N instead of λ_k, but to simplify notation, we will simply write λ_k. Our model has become

$$X_i^N(t) = X_i^N(0) + \sum_k N^{-\alpha_i} Y_k \left(\int_0^t N^{\beta_k + \nu_k \cdot \alpha} \lambda_k(X^N(s)) ds \right) \zeta_{ki}, \quad i \in \{1, \dots, d\}. \tag{3.1}$$

Remark 3.1. We emphasize that the models (3.1) and (2.1) are equivalent in that X^N is a scaled version of X. This scaling will allow us to quantify the behavior of the different algorithms, though plays *no role* in the simulation of the processes.

Remark 3.2. If $\beta_k + \nu_k \cdot \alpha = \alpha_i = 1$ for all i, k in (3.1), then we have what is typically called the *classical scaling.* It was specifically this scaling that was used in the analyses of Euler and midpoint τ-leaping found in [3]. In this case it is natural to consider X^N as a vector whose ith component gives the concentration, in moles per unit volume, of the ith species.

The focus of the paper will now shift from (2.1) to the equivalent (3.1). To analyze how the different algorithms approximate X^N, we will need some terminology which we collect below.

3.1 Terminology and definitions

For any vector $w \in \mathbb{R}^d$, define w^N to be the vector with ith component

$$w_i^N \overset{\text{def}}{=} \frac{w_i}{N^{\alpha_i}},$$

and define

$$\mathbb{L}_N = \left\{ x^N \mid x \in \mathbb{Z}^d \right\}.$$

By construction, the process (3.1) lives in $\mathbb{L}_N$, and its generator is

$$\mathcal{A}^N f(x) = \sum_k N^{\beta_k + \nu_k \cdot \alpha} \lambda_k(x)(f(x + \zeta_k^N) - f(x)). \tag{3.2}$$

For $f : \mathbb{L}^N \to \mathbb{R}$ and any $t \geq 0$, Dynkin's formula is now

$$\mathbb{E}_{x_0} f(X^N(t)) = f(x_0) + \mathbb{E}_{x_0} \int_0^t \mathcal{A}^N f(X(s))ds, \tag{3.3}$$

which holds so long as the expectations exist.

To quantify the natural time-scale of the system, define $\gamma \in \mathbb{R}$ via

$$\gamma \overset{\text{def}}{=} \max_{\{i,k \,:\, \zeta_{ki}^N \neq 0\}} \{\beta_k + \nu_k \cdot \alpha - \alpha_i\},$$

where we recall that ν_k is the source vector for the kth reaction. It is worth noting that $\gamma = 0$ if one assumes the system satisfies the classical scaling discussed in Remark 3.2. However, $\gamma = 0$ in many other settings as well. We will see that our main results are most useful when $\gamma \leq 0$.

Example 2. *As an instructive example, consider the system*

$$S_1 \underset{100}{\overset{100}{\rightleftarrows}} S_2$$

with $X_1(0) = X_2(0) =$ *10,000. In this case, it is natural to take* $N =$ *10,000 and* $\alpha_1 = \alpha_2 = 1$*. As the rate constants are* $100 = \sqrt{10{,}000}$*, we take* $\beta_1 = \beta_2 = 1/2$ *and find that* $\gamma = 1/2$*. The equation governing the normalized process* X_1^N *is*

$$X_1^N(t) = X_1^N(0) - Y_1\left(N^{1/2} N \int_0^t X_1^N(s)ds\right)\frac{1}{N} + Y_2\left(N^{1/2} N \int_0^t (2 - X_1^N(s))ds\right)\frac{1}{N}$$

where we have used that $X_1^N + X_2^N \equiv 2$.

We define the operator ∇_k^N for the kth reaction via

$$\nabla_k^N f(x) \overset{\text{def}}{=} N^{\beta_k + \nu_k \cdot \alpha - \gamma}(f(x + \zeta_k^N) - f(x)). \tag{3.4}$$

In Example 2 above, we have

$$\begin{aligned} \nabla_1^N f(x) &= N(f(x + (e_2 - e_1)/N) - f(x)) \\ \nabla_2^N f(x) &= N(f(x + (e_1 - e_2)/N) - f(x)), \end{aligned}$$

where $S_1 \to S_2$ is arbitrarily labeled as the first reaction, and $e_i \in \mathbb{Z}^2$ is the vector of all zeros except with a one in the ith location.

Note that if f is globally Lipschitz, then by the definition of γ, $\nabla_k^N f(x)$ is uniformly bounded over k and x. We may now write (3.2) as

$$\mathcal{A}^N f(x) = \sum_k N^\gamma \lambda_k(x) \nabla_k^N f(x).$$

Defining the vector valued operators

$$\lambda \overset{\text{def}}{=} [\lambda_1, \dots, \lambda_R], \quad \nabla^N \overset{\text{def}}{=} [\nabla_1^N, \dots, \nabla_R^N], \tag{3.5}$$

where we recall that R is the number of reactions, we obtain

$$\mathcal{A}^N f(x) = (N^\gamma \lambda \cdot \nabla^N) f(x).$$

For $k \in \{1, \dots, R\}$ we define

$$c_k \overset{\text{def}}{=} \beta_k + \nu_k \cdot \alpha - \gamma, \tag{3.6}$$

so that (3.4) becomes

$$\nabla_k^N f(x) = N^{c_k}(f(x + \zeta_k^N) - f(x)).$$

For $i \in \{1, \dots, d\}$ and $k \in \{1, \dots, R\}$, we define

$$m_k \overset{\text{def}}{=} \min\{\alpha_i \ : \ \zeta_{ki}^N \neq 0\},$$

so that $O(|\zeta_k^N|) = N^{-m_k}$. Note that $m_k \geq 0$, and by the choice of γ we have $c_k - m_k \leq 0$ for all k. Further, we point out that γ is chosen so that $c_k = 0$ for at least one k. Finally, we note that if $\|\nabla f\|_\infty$ is bounded, then $\nabla^N f$ is in $O(N^{c_k - m_k})$.

To obtain the analog of (3.3) for the approximate methods we first define the operator $\mathcal{B}_z^N$ by

$$\mathcal{B}_z^N f(x) \overset{\text{def}}{=} (N^\gamma \lambda(z) \cdot \nabla^N) f(x).$$

If Z_E^N represents the approximation to (3.1) via Euler's method, then for all $t > 0$

$$\mathbb{E} f(Z_E^N(t)) = \mathbb{E} f(Z_E^N(\eta(t))) + \mathbb{E} \int_{\eta(t)}^t (\mathcal{B}^N_{Z_E^N(\eta(t))} f)(Z_E^N(s)) ds,$$

so long as the expectations exist. If Z_M^N represents the approximation to (3.1) via the midpoint method, then for $t > 0$

$$\mathbb{E} f(Z_M^N(t)) = \mathbb{E} f(Z_M^N(\eta(t))) + \mathbb{E} \int_{\eta(t)}^t (\mathcal{B}^N_{\rho(Z_M^N \circ \eta(t))} f)(Z_M^N(s)) ds,$$

so long as the expectations exist, where now

$$\rho(z) = z + \frac{1}{2} h \sum_k N^{\beta_k + \nu_k \cdot \alpha} \lambda_k(z) \zeta_k^N.$$

While we should write ρ^N in the above, we repress the "N" in this case for ease of notation. Finally, define the operator $\mathcal{B}^N_{z_1, z_2}$ by

$$(\mathcal{B}^N_{z_1, z_2} f)(x) \overset{\text{def}}{=} (N^\gamma [\xi_1 \lambda(z_1) - \xi_2 \lambda(z_2)]^+ \cdot \nabla^N) f(x),$$

where for some $\theta \in (0, 1)$, ξ_1 and ξ_2 satisfy (2.8), and for $v \in \mathbb{R}^d$ the ith component of v^+ is $[v_i]^+ = \max\{v_i, 0\}$. Then, if Z_{trap}^N represents the approximation to (3.1) via the weak trapezoidal method, then for $\eta(t) \leq t < \eta(t) + \theta h$

$$\mathbb{E} f(Z_{trap}^N(t)) = \mathbb{E} f(Z_{trap}^N(\eta(t))) + \mathbb{E} \int_{\eta(t)}^t (\mathcal{B}^N_{Z_{trap}^N(\eta(t))} f)(Z_{trap}^N(s)) ds,$$

whereas for $\eta(t) + \theta h \leq t < \eta(t) + h$

$$\mathbb{E}f(Z^N_{trap}(t)) = \mathbb{E}f(Z^N_{trap}(\eta(t) + \theta h)) + \mathbb{E}\int_{\eta(t)+\theta h}^{t} (\mathcal{B}^N_{Z^N_{trap}(\eta(t)+\theta h), Z^N_{trap}(\eta(t))} f)(Z^N_{trap}(s))ds.$$

As in [3], we modify the kinetics λ via multiplication by a C^∞ cutoff function, which sets the intensities to zero outside of our scaling region of interest. This has the effect of confining the dynamics to a compact subset of $\mathbb{L}_N$, which we denote by $\overline{\mathbb{L}}_N \subset \mathbb{R}^d_{\geq 0}$. Supposing the cutoff function were denoted $g \in C^\infty_c(\mathbb{R}^d, \mathbb{R}^R)$, we should technically henceforth write $g\lambda$, with the multiplication defined component-wise, as our intensity function. The function g is chosen so that $g\lambda(x) = \lambda(x)$ for all x in a region of interest in the interior of $\overline{\mathbb{L}}_N$. However, for ease of notation, we refrain from adding the cutoff function in the notation, and continue to solely write λ. Note that $\|\lambda\|_n$ is now bounded for all $n \geq 0$ where $\|\cdot\|_n$ is defined in (5.1). See Section 2.2 of [3] for a further discussion of the need for such a cutoff function.

For any function $f : \mathbb{R}^d \to \mathbb{R}$, we denote

$$\|f\|_\infty \overset{\text{def}}{=} \sup_{x \in \overline{\mathbb{L}}_N} \{|f(x)|\}.$$

We abuse notation this way because the processes X^N and Z^N have domain $\overline{\mathbb{L}}_N$ throughout our analysis.

4 Global error from local error

Throughout the section, we will denote the vector valued process whose ith component satisfies (3.1) by X^N, and denote an arbitrary approximate process via Z^N. Also, we define the following semigroup operators acting on $f \in C_0(\mathbb{L}_N, \mathbb{R})$ as follows

$$\mathcal{P}_t f(x) \overset{\text{def}}{=} \mathbb{E}_x f(X^N(t))$$
$$P_t f(x) \overset{\text{def}}{=} \mathbb{E}_x f(Z^N(t)),$$

where for ease of notation we choose not to incorporate the notation N into either $\mathcal{P}_t$ or P_t.

We will interpret the difference between the above two operators, for $t \in [0, T]$, as the *weak error* of the approximate process Z^N on the interval $[0, T]$. For our purposes, there will be a time discretization associated with Z^N, and we will then interpret $\mathcal{P}_h - P_h$ as the one step local error, as is common in the literature. These concepts are defined formally below.

Definition 1. Let n be an arbitrary non-negative integer, and $\mathcal{M}$ be a m dimensional vector of $C(\mathbb{R}^d, \mathbb{R})$ valued operators on $C(\mathbb{R}^d, \mathbb{R})$, with its ℓth coordinate denoted by $\mathcal{M}_\ell$. Then we define

$$\|f\|^{\mathcal{M}}_n = \sup\left\{ \left\| \left(\prod_{i=1}^{p} \mathcal{M}_{\ell_i} \right) f \right\|_\infty, 1 \leq \ell_i \leq m,\ p \leq n \right\}.$$

For example, if $j, k, \ell \in \{1, ..., R\}$ then

$$|(\nabla^N_j \nabla^N_k \nabla^N_\ell f)(x)| \leq \|f\|^{\nabla^N}_3,$$

where we recall that ∇^N is defined in (3.5). Note that, for any $\mathcal{M}$,

$$\|f\|_0^{\mathcal{M}} = \|f\|_0 = \|f\|_\infty. \tag{4.1}$$

Also note that, by definition, for $n \geq 0$

$$\|f\|_n^{\mathcal{M}} \leq \|f\|_{n+1}^{\mathcal{M}}.$$

Definition 2. Suppose $\mathcal{M} : C(\mathbb{R}^d, \mathbb{R}) \to C(\mathbb{R}^d, \mathbb{R}^R)$ and $Q : C(\mathbb{R}^d, \mathbb{R}) \to C(\mathbb{R}^d, \mathbb{R})$ are operators. Then define

$$\|Q\|_{j\to\ell}^{\mathcal{M}} \stackrel{\text{def}}{=} \sup_{f\in C^j, f\neq 0} \frac{\|Qf\|_\ell^{\mathcal{M}}}{\|f\|_j^{\mathcal{M}}}.$$

The purpose of this paper can now be stated succinctly. We will derive bounds for the global weak error of the different approximate processes, which, due to (4.1), consists of deriving bounds for $\|(P_h^n - \mathcal{P}_{nh})\|_{m\to 0}^{\mathcal{M}}$, for an appropriately defined $\mathcal{M}$ and a reasonable choice of $m \geq 0$. Theorem 4.1 below quantifies how the global error $\|(P_h^n - \mathcal{P}_{nh})\|_{m\to 0}^{\mathcal{M}}$ can be bounded using the one-step local error $\|P_h - \mathcal{P}_h\|_{m\to 0}^{\mathcal{M}}$. As is common, we will denote by $O(h)$ a set of values bounded by a fixed constant multiple of h. Later, in Section 5, we will derive the requisite bounds for the local weak error.

Theorem 4.1. *Let $\mathcal{M}$ be a $C(\mathbb{R}^d, \mathbb{R}^R)$ valued operator on $C(\mathbb{R}^d, \mathbb{R})$. Then for any $n, m \geq 0$, and $h > 0$*

$$\|(P_h^n - \mathcal{P}_{nh})\|_{m\to 0}^{\mathcal{M}} = O(n\ \|P_h - \mathcal{P}_h\|_{m\to 0}^{\mathcal{M}} \max_{\ell\in\{1,\dots,n\}} \{\|\mathcal{P}_{\ell h}\|_{m\to m}^{\mathcal{M}}\})$$

Proof. Let $f \in C_0(\mathbb{R}^d, \mathbb{R})$. Note that, since $\|g\|_0 = \|g\|_0^{\mathcal{M}}$ for any g,

$$\|P_h^{j-1}\|_{0\to 0}^{\mathcal{M}} \|P_h - \mathcal{P}_h\|_{m\to 0}^{\mathcal{M}} = \|P_h^{j-1}\|_{0\to 0} \|P_h - \mathcal{P}_h\|_{m\to 0}^{\mathcal{M}}.$$

With this in mind

$$\begin{aligned}
\|(P_h^n - \mathcal{P}_{nh})f\|_0 &= \Big\| \sum_{j=1}^n (P_h^j \mathcal{P}_{h(n-j)} - P_h^{j-1} \mathcal{P}_{h(n-j+1)}) f \Big\|_0 \\
&\leq \sum_{j=1}^n \|P_h^{j-1} (P_h - \mathcal{P}_h) \mathcal{P}_{h(n-j)} f\|_0 \\
&\leq \sum_{j=1}^n \|P_h^{j-1}\|_{0\to 0} \|P_h - \mathcal{P}_h\|_{m\to 0}^{\mathcal{M}} \|\mathcal{P}_{h(n-j)}\|_{m\to m}^{\mathcal{M}} \|f\|_m^{\mathcal{M}}.
\end{aligned}$$

Since P_h is a contraction, i.e. $\|P_h\|_{0\to 0} \leq 1$, the result is shown. □

From the proof of the above theorem, the following result is immediate with ∇^N in place of $\mathcal{M}$.

Corollary 4.2. *Under the same assumptions of Theorem 4.1 and with $f \in C_0^m(\mathbb{R}^d, \mathbb{R})$,*

$$\|(P_h^n - \mathcal{P}_{nh})f\|_0^{\nabla^N} = O(n\|P_h - \mathcal{P}_h\|_{m\to 0}^{\nabla^N} \max_{\ell\in\{1,\dots,n\}} \{\|\mathcal{P}_{\ell h} f\|_m^{\nabla^N}\}).$$

The following generalization, which allows for variable step sizes, is straightforward.

Corollary 4.3. *For $f \in C_0^m(\mathbb{R}^d, \mathbb{R})$*

$$\|\mathbb{E}_x[f(Z_{t_n})] - \mathbb{E}_x[f(X_{t_n})]\|_\infty = O(n \max_{i=1,\dots,n}\{\|P_{h_i} - \mathcal{P}_{h_i}\|_{m\to 0}^{\nabla^N}\} \max_{\ell\in\{1,\dots,n\}}\{\|\mathcal{P}_{t_\ell} f\|_m^{\nabla^N}\}).$$

Thus, once we compute the local one step error $\|P_h - \mathcal{P}_h\|_{m\to 0}^{\nabla^N}$, we have a bound on the weak error of the algorithm that depends only on the semigroup $\mathcal{P}_t$ of the original process. We will delay discussion of $\|\mathcal{P}_t f\|_m^{\nabla^N}$ for now, as this term is independent of the numerical approximation method. Instead, in the next section we provide a bound of $\|P_h - \mathcal{P}_h\|_{m\to 0}^{\nabla^N}$ for each of the different algorithms.

5 Local errors

Section 5.1 will present some necessary analytic tools. Sections 5.2, 5.3, and 5.4 will present the local analysis of the Euler, midpoint, and weak trapezoidal tau-leaping, respectively.

5.1 Analytical tools

Definition 3. Denote the jth directional derivative of f into the direction $[v_1, v_2, ...v_j]$ by $f'[v_1, ..., v_j]$ and

$$\|f\|_j \overset{\text{def}}{=} \sup_x\{f'[v_1, ..., v_j](x), \|v\| = 1\} \tag{5.1}$$

Proposition 5.1. *For $d, R \geq 0$, let $f \in C_0^1(\mathbb{R}^d, \mathbb{R}^R)$. Then, for any $k \in \{1, \dots, R\}$*

$$\nabla_k^N f \in O(N^{c_k - m_k}\|f\|_1) \subset O(1).$$

In particular, $N^{-c_k}\nabla_k^N f$ is bounded.

Proof. The result follows from the fact that for any $w \in \mathbb{R}^d$

$$|f(x+w) - f(x)| \leq |w|\|f\|_1.$$

□

Define, for any multi-subset I of $\{1, ..., R\}$,

$$\nabla_I^N f \overset{\text{def}}{=} \left\{ (\prod_{i=1}^{|I|} \nabla_{\ell_i}^N) f \right\},$$

so that,

$$\|f\|_n^{\nabla^N} = \sup_{|I|\leq n} \|\nabla_I^N f\|_\infty.$$

Proposition 5.2. *For $d, R \geq 0$, let $f \in C_0^j(\mathbb{R}^d, \mathbb{R}^R)$. Then,*

$$\|f\|_j^{\nabla^N} = O(\|f\|_j).$$

Proof. The case $j = 1$ follows from Proposition 5.1. Now consider $\nabla_I^N f(x)$ for a multi-set I of $\{1, \dots, R\}$, with $|I| = j \geq 2$. If $m_k > 0$ for all $k \in I$, the statement is clear. If on the other hand, $m_k = 0$ for some $k \in I$, then for this specific k, we have $c_k \leq 0$ and

$$\|\nabla_I^N f\|_\infty \leq 2N^{c_k} \|\nabla_{I\backslash k}^N f\|_\infty = O(\|f\|_{j-1}) = O(\|f\|_j),$$

where the second to last equality follows by an inductive hypothesis. □

We make some definitions associated with ∇^N. Let $g : \mathbb{R}^d \to \mathbb{R}^R$. For $i, j \in \{1, \dots, R\}$

$$\begin{aligned} [D^N g(x)]_{ij} &\overset{\text{def}}{=} \nabla_j^N g_i(x) \\ [(\nabla^N)^2]_{ij} &\overset{\text{def}}{=} \nabla_i^N \nabla_j^N \\ diag(N^c) &\overset{\text{def}}{=} diag(N^{c_1}, ..., N^{c_R}). \end{aligned} \tag{5.2}$$

Also, we define $\mathbf{1}_R$ to be the R dimensional vector whose entries are all 1.

Lemma 5.3. *(Product Rule) Let $g, q : \mathbb{R}^d \to \mathbb{R}^R$ be vector valued functions. Then*

$$\nabla_k^N (g \cdot q)(x) = (\nabla_k^N g \cdot q)(x) + (g \cdot \nabla_k^N q)(x) + N^{-c_k} (\nabla_k^N g \cdot \nabla_k^N q)(x).$$

Also,

$$\nabla^N (g \cdot q)(x) = [D^N g]^T q(x) + [D^N q]^T g(x) + diag(N^c)^{-1} ([D^N g]^T \times [D^N q]^T)(x) \mathbf{1}_R f.$$

Proof. Note that, for any k,

$$\begin{aligned} \nabla_k^N (g \cdot q)(x) =& N^{c_k} (g(x + \zeta_k^N) q(x + \zeta_k^N) - g(x) q(x)) \\ &= N^{c_k} (g(x + \zeta_k^N) - g(x)) q(x) + N^{c_k} (q(x + \zeta_k^N) - q(x)) g(x) \\ &\quad + N^{-c_k} N^{c_k} (q(x + \zeta_k^N) - q(x)) N^{c_k} (g(x + \zeta_k^N) - g(x)) \\ &= (\nabla_k^N g) \cdot q)(x) + (\nabla_k^N q \cdot g)(x) + N^{-c_k} (\nabla_k^N g \cdot \nabla_k^N q)(x), \end{aligned}$$

verifying the first statement. To verify the second, one simply notes that the above calculation holds for every coordinate, and the result follows after simple bookkeeping. □

Corollary 5.4. *Let $\lambda : \mathbb{R}^d \to \mathbb{R}^R$ be a vector valued function, and $f : \mathbb{R}^d \to \mathbb{R}$. Then*

$$\nabla_k^N (\lambda \cdot \nabla^N f)(x) = (\nabla_k^N \lambda \cdot \nabla^N) f + \lambda \cdot \nabla^N \nabla_k^N f + N^{-c_k} \nabla_k^N \lambda \cdot \nabla^N \nabla_k^N f.$$

Also,

$$\nabla^N (\lambda \cdot \nabla^N f) = [D^N \lambda]^T \nabla^N f + [(\nabla^N)^2 f] \lambda + diag(N^c)^{-1} ([D^N \lambda \times (\nabla^N)^2] \mathbf{1}_R f. \tag{5.3}$$

Proof. Simply put $g = \lambda$ and $q = \nabla^N f$, and recall that ∇^2 is symmetric. □

5.2 Euler's method

Throughout subsection 5.2, we let Z_E^N be the Euler approximation to X^N computed via Algorithm 1, and for $x \in \mathbb{L}_N$ let

$$P_{E,h} f(x) \overset{\text{def}}{=} \mathbb{E}_x f(Z_E^N(h)),$$

where h is the step-size taken in the algorithm. Below, we will assume $h < N^{-\gamma}$, which is a natural stability condition, and is discussed further in Section 7.

Theorem 5.5. *Suppose that the step size h satisfies $h < N^{-\gamma}$. Then*

$$\|P_{E,h} - \mathcal{P}_h\|_{2\to 0}^{\nabla^N} = O(N^{2\gamma}h^2).$$

Proof. For Euler's method with initial condition x_0,

$$P_{E,h}f(x_0) = f(x_0) + h\mathcal{B}_{x_0}^N f(x_0) + \frac{h^2}{2}(\mathcal{B}_{x_0}^N)^2 f(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3), \tag{5.4}$$

where, noting $\nabla^N \lambda(x_0) = 0$ and using the product rule in Lemma 5.3, we have

$$\begin{aligned} \mathcal{B}_{x_0}^N f &= N^\gamma \lambda(x_0) \cdot \nabla^N f \\ (\mathcal{B}_{x_0}^N)^2 f &= N^\gamma \lambda(x_0) \cdot \nabla^N (N^\gamma \lambda(x_0) \cdot \nabla^N f) \\ &= N^{2\gamma} \lambda(x_0)^T [(\nabla^N)^2 f] \lambda(x_0). \end{aligned} \tag{5.5}$$

On the other hand, for the exact process (3.1),

$$\mathcal{P}_h f(x_0) = f(x_0) + h\mathcal{A}^N f(x_0) + \frac{h^2}{2}(\mathcal{A}^N)^2 f(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3), \tag{5.6}$$

where, again,

$$\mathcal{A}^N f = N^\gamma \lambda \cdot \nabla^N f.$$

Noting that,

$$\begin{aligned} (\mathcal{A}^N)^2 f(x) &= N^{2\gamma}(\lambda \cdot \nabla^N(\lambda \cdot \nabla^N f(x))) \\ &= N^{2\gamma}\lambda^T([D^N\lambda]^T \nabla^N f(x) + [(\nabla^N)^2 f]\lambda(x) + N^{2\gamma}\lambda^T(diag(N^{-c})[D^N\lambda \times (\nabla)^2]1_R f) \end{aligned} \tag{5.7}$$

and defining

$$\begin{aligned} a(x) &\stackrel{\text{def}}{=} N^{2\gamma}\lambda^T[D^N\lambda]^T \nabla^N f(x) \\ b(x) &\stackrel{\text{def}}{=} N^{2\gamma}\lambda^T[(\nabla^N)^2 f]\lambda(x) \\ c(x) &\stackrel{\text{def}}{=} N^{2\gamma}\lambda^T[diag(N^{-c})[D^N\lambda \times (\nabla^N)^2]1_R f(x)], \end{aligned}$$

we can write

$$\mathcal{P}_h f(x_0) = f(x_0) + h\mathcal{A}^N f(x_0) + \frac{h^2}{2}(a(x_0) + b(x_0) + c(x_0)) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3).$$

Note that $\mathcal{B}_{x_0}^N f(x_0) = \mathcal{A}^N f(x_0)$ and $b(x_0) = (\mathcal{B}_{x_0}^N)^2 f(x_0)$. We may then compare (5.4) and (5.6)

$$\begin{aligned} (P_{E,h} - \mathcal{P}_h)f(x_0) &= \frac{h^2}{2}((\mathcal{B}_{x_0}^N)^2 f(x_0) - (a(x_0) + b(x_0) + c(x_0))) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3) \\ &= \frac{h^2}{2}(-a(x_0) - c(x_0)) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3). \end{aligned}$$

The term $a(x) + c(x) = O(N^{2\gamma}\|f\|_2^{\nabla^N})$ is clearly non-zero in general, giving the desired result. □

5.3 Approximate midpoint method

Throughout subsection 5.3, we let Z_M^N be the midpoint method approximation to X^N computed via Algorithm 2, and for $x \in \mathbb{L}_N$ let

$$P_{M,h}f(x) \overset{\text{def}}{=} \mathbb{E}_x f(Z_M^N(h)),$$

where h is the step-size taken in the algorithm. As before, we will assume $h < N^{-\gamma}$, which is a natural stability condition, and is discussed further in Section 7.

Theorem 5.6. *Suppose that the step size h satisfies $h < N^{-\gamma}$. Then*

$$\|(P_{M,h} - \mathcal{P}_h)\|_{3\to 0}^{\nabla^N} = O(N^{3\gamma}h^3 + N^{2\gamma - \min\{m_k\}}h^2).$$

Remark 5.7. Theorem 5.6 predicts that the midpoint method behaves locally like a third order method and globally like a second order method if h is in a regime satisfying $N^\gamma h \gg N^{-\min\{m_k\}}$, or equivalently if $h \gg N^{-\gamma-\min\{m_k\}}$. This agrees with the result found in [3] pertaining to the midpoint method, which had $\gamma = 0$, $m_k \equiv 1$, and the running assumption that $h \gg 1/N$. This behavior is demonstrated via numerical example in Section 8.

Proof. (of Theorem 5.6) Let ζ^N denote the matrix with kth column ζ_k^N, i.e.

$$[\zeta^N] = [\zeta_1^N, \zeta_2^N, ..., \zeta_R^N].$$

Recall that ρ is defined via

$$\rho(z) = z + \frac{h}{2}N^\gamma \sum_k \lambda_k(z) N^{c_k}\zeta_k^N.$$

After some algebra, we have

$$\begin{aligned}\mathcal{B}^N_{\rho(x_0)}f(x) &= N^\gamma(\lambda(x_0 + \frac{h}{2}N^\gamma \sum_k \lambda_k(x_0)N^{c_k}\zeta_k^N)) \cdot \nabla^N f(x)\\ &= N^\gamma \lambda(x_0)\cdot \nabla^N f(x) + w(x_0) + O(N^{2\gamma}\|f\|_1^{\nabla^N}h^2).\end{aligned}$$

where

$$w(x) \overset{\text{def}}{=} N^{2\gamma}\frac{h}{2}[D\lambda(x_0)][\zeta^N]diag(N^c)\lambda(x_0)\cdot \nabla^N f(x).$$

Next, using the product rule (5.3), we see

$$\begin{aligned}(\mathcal{B}^N_{\rho(x_0)})^2 f(x) &= N^\gamma\lambda(x_0 + \frac{h}{2}[\zeta^N]diag(N^c)\lambda(x_0))\cdot\nabla^N(N^\gamma\lambda(x_0 + \frac{h}{2}[\zeta^N]diag(N^c)\lambda(x_0))\cdot\nabla^N f)(x)\\ &= N^{2\gamma}\lambda(x_0 + \frac{h}{2}[\zeta^N]diag(N^c)\lambda(x_0))^T[(\nabla^N)^2 f]\lambda(x_0 + \frac{h}{2}[\zeta^N]diag(N^c)\lambda(x_0))\cdot\nabla^N f)(x)\\ &= g(x_0) + O(N^{2\gamma}\|f\|_2^{\nabla^N}h),\end{aligned}$$

where

$$g(x_0) \overset{\text{def}}{=} N^{2\gamma}\lambda(x_0)^T[(\nabla^N)^2 f(x)]\lambda(x_0).$$

Therefore, since $N^\gamma\lambda(x_0)\cdot\nabla^N f(x_0) = \mathcal{A}^N f(x_0)$,

$$\begin{aligned}P_{M,h}f(x_0) &= f(x_0) + h\mathcal{B}^N_{\rho(x_0)}f(x_0) + \frac{h^2}{2}(\mathcal{B}^N_{\rho(x_0)})^2 f(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N}h^3)\\ &= f(x_0) + h\left(\mathcal{A}^N f(x_0) + w(x_0) + O(N^{2\gamma}\|f\|_2^{\nabla^N}h^2)\right)\\ &\quad + \frac{h^2}{2}\left(g(x_0) + O(N^{2\gamma}\|f\|_2^{\nabla^N}h)\right) + O(N^{3\gamma}\|f\|_3^{\nabla^N}h^3).\end{aligned}$$

Recall that

$$(\mathcal{A}^N)^2 f(x) = a(x) + b(x) + c(x),$$

where

$$\begin{aligned} a(x) &= N^{2\gamma}\lambda^T[D^N\lambda]^T\nabla^N f(x), \\ b(x) &= N^{2\gamma}\lambda^T[(\nabla^N)^2 f]\lambda(x), \\ c(x) &= N^{2\gamma}\lambda^T[diag(N^{-c})[D^N\lambda \times (\nabla^N)^2]1_R f(x)], \end{aligned} \tag{5.8}$$

and

$$\mathcal{P}_h f(x_0) = f(x_0) + h\mathcal{A}^N f(x_0) + \frac{h^2}{2}(a(x_0) + b(x_0) + c(x_0)) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3).$$

Noting that $b(x_0) = g(x_0)$, we see

$$\begin{aligned} (P_{M,h} - \mathcal{P}_h)f(x_0) &= hw(x_0) + \frac{h^2}{2}\left(g(x_0) - (a(x_0) + b(x_0) + c(x_0))\right) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3) \\ &= (hw(x_0) - \frac{h^2}{2}a(x_0)) - \frac{h^2}{2}c(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3). \end{aligned} \tag{5.9}$$

We will now gain control over the terms $(hw(x_0) - \frac{h^2}{2}a(x_0))$ and $\frac{h^2}{2}c(x_0)$, separately.

Handling $\frac{h^2}{2}c(x_0)$ first, we simply note that by the discussion surrounding (3.6), we have that $\nabla^N\lambda_k \in O(N^{c_k - m_k})$, and so

$$c(x_0) = O(N^{2\gamma - \min\{m_k\}}\|f\|_2^{\nabla^N}).$$

Next, we will show that

$$hw(x_0) - \frac{h^2}{2}a(x_0) = O(N^{2\gamma - \min\{m_k\}}\|f\|_1^{\nabla^N} h^2).$$

We have

$$\begin{aligned} hw(x_0) - \frac{h^2}{2}a(x_0) &= \frac{h^2}{2}N^{2\gamma}[D\lambda(x_0)][\zeta^N]diag(N^c)\lambda(x_0)\cdot\nabla^N f(x_0) - \frac{h^2}{2}N^{2\gamma}\lambda^T[D^N\lambda]^T\nabla^N f(x) \\ &= \frac{h^2}{2}N^{2\gamma}\Big([D\lambda(x_0)][\zeta^N]diag(N^c) - [D^N\lambda(x_0)]\Big)\lambda(x_0)\cdot\nabla^N f(x_0). \end{aligned} \tag{5.10}$$

By Proposition 5.2, $\nabla^N f(x)$ is bounded by $\|f\|_1^{\nabla^N}$. Therefore, we just need to show that the difference between the two square matrices

$$[D^N\lambda(x_0)] \qquad \textit{and} \qquad [D\lambda(x_0)][\zeta^N]diag(N^c) \tag{5.11}$$

is $O(N^{-\min\{m_k\}})$. Recalling the definitions in (5.2), the (i,j)th entry of the left side of (5.11) is

$$N^{c_j}(\lambda_i(x_0 + \zeta_j^N) - \lambda_i(x_0))$$

whereas that of the right side of (5.11) is

$$N^{c_j}\nabla\lambda_i\cdot\zeta_j^N.$$

Also, note that, for $\lambda \in C_c^2(\mathbb{R}^d, \mathbb{R})$,

$$((\lambda(x+v) - \lambda(x)) - \nabla\lambda(x) \cdot v) \in O(|v|^2 \|\lambda\|_2).$$

where

$$\|\lambda\|_2 = \sup\{\|\lambda\|_\infty, \|\partial_{x_i}\lambda\|_\infty, \|\partial_{x_j}\partial_{x_\ell}\lambda\|_\infty, i, j, k \leq d.\}$$

Since $\|\lambda_k\|_2$ is bounded for any k, the difference between the (i,j)th entries of the two expressions in (5.11) is hence

$$O(N^{c_j} N^{-2m_j})$$

Also, recall that $c_j - m_j \leq 0$ by our choice of γ, with equality at at least one j. Thus the above is also

$$O(N^{-\min\{m_k\}}).$$

Therefore (5.10) is of order

$$O(N^{2\gamma - \min\{m_k\}} h^2 \|f\|_1^{\nabla^N}),$$

as desired. Combining the above with (5.9) gives us

$$\begin{aligned} \|(\mathcal{P}_h - P_{M,h})f\|_0 &= O(N^{2\gamma-\min\{m_k\}}\|f\|_1^{\nabla^N} h^2 + N^{2\gamma-\min\{m_k\}}\|f\|_2^{\nabla^N} h^2 + N^{3\gamma}\|f\|_3^{\nabla^N} h^3) \\ &= O(\|f\|_3^{\nabla^N}[N^{3\gamma}h^3 + N^{2\gamma-\min\{m_k\}}h^2]), \end{aligned} \tag{5.12}$$

implying

$$\|P_{M,h} - \mathcal{P}_h\|_{3\to 0}^{\nabla^N} = O(N^{3\gamma}h^3 + N^{2\gamma-\min\{m_k\}}h^2),$$

as desired. □

We can strengthen Theorem 5.6 slightly. Suppose that f explicitly depends only on a subset of species $S_0 \subset S$. More precisely, suppose that

$$S_0(f) = \{i \in \{1,..,d\} \ : \ f(x + \epsilon e_i) \neq f(x) \quad \text{for some } x \text{ and } e_i \in \mathbb{R}^d,\ \epsilon \in \mathbb{R}\ \}.$$

Next define R_0 to be the subset of all reactions that affect any element in S_0; that is,

$$R_0(f) = \{k \in \{1,..,R\} \ : \ \zeta_{ki} \neq 0 \quad \text{for some } i \in S_0\}.$$

Therefore,

$$\nabla_k^N f = 0 \qquad \forall k \notin R_0. \tag{5.13}$$

Finally, define

$$R_{00}(f) = \{k \in \{1,..,R\} \ : \ \lambda_\ell(x + \zeta_k) \neq \lambda_\ell(x), \text{ for some } \ell \in R_0 \text{ and some } x \in \mathbb{L}_N\}$$

Thus, by construction, the set R_{00} corresponds to those reactions that affect those of R_0, which in turn affect S_0.

Corollary 5.8. *For a given f, define $S_0(f)$, $R_0(f)$,and $R_{00}(f)$ as above. Then if $h < N^{-\gamma}$,*

$$\|(P_{M,h} - \mathcal{P}_h)f\|_\infty = O\left((N^{3\gamma}h^3 + N^{2\gamma - \min_{k\in R_{00}(f)}\{m_k\}}h^2)\|f\|_3^{\nabla^N}\right).$$

Proof. Returning to the proof of Theorem 5.6, we consider again (5.9). First we handle

$$c(x_0) = N^{2\gamma}\lambda^T[diag(N^{-c})[D^N\lambda \times (\nabla^N)^2]1_R f(x_0)].$$

By (5.13), $\nabla_i^N\nabla_j^N f(x_0) = 0$ if either $i \notin R_0(f)$ or $j \notin R_0(f)$. Thus, in the matrix $D^N\lambda \times (\nabla^N)^2$, the only non-zero entries are those with indices $i, j \in R_0(f)$. Further, by construction

$$[D^N\lambda]_{i,j} = \nabla_j\lambda_i \neq 0 \qquad \text{only if} \qquad i \in R_0(f), j \in R_{00}(f).$$

Therefore, after recalling that $\nabla_j\lambda_i(x_0) \in N^{c_j - m_j}$ we have that,

$$c(x_0) \in N^{2\gamma - \min_{k\in R_{00}(f)}\{m_k\}}.$$

By similar arguments,

$$hw(x_0) - \frac{h^2}{2}a(x_0) \in O(N^{2\gamma - \min_{k\in R_{00}(f)}\{m_k\}}\|f\|_1^{\nabla^N}h^2),$$

and the remainder of the argument proceeds in exactly the same manner as the proof of Theorem 5.6. □

Definition 4. The system (2.1) is called a *first order system* if each λ_k is linear.

Noting that it is always possible to solve for the means of first order systems, see [11], the following is pointed out for completeness.

Corollary 5.9. *If X^N is the solution of a first order system, and if f is linear, then*

$$\|(P_{M,h} - \mathcal{P}_h)f\|_\infty = O(N^{3\gamma}h^3).$$

Proof. It is sufficient to show that $c(x_0)$ in (5.8) and the right hand side of (5.10) are both zero. The fact that $c(x_0) = 0$ follows immediately from the linearity of f, as $(\nabla^N)^2 f = 0$. Moreover, by the linearity of the intensity functions,

$$D^N\lambda(\cdot) = [D\lambda(\cdot)][\zeta^N]diag(N^c),$$

showing the right hand side of (5.10) is zero. □

5.4 Weak trapezoidal method

Throughout subsection 5.4, we let Z_{trap}^N be the approximation to X^N computed via Algorithm 3, and for $x \in \mathbb{L}_N$ let

$$P_{trap,h}f(x) \overset{\text{def}}{=} \mathbb{E}_x f(Z_{trap}^N(h)),$$

where h is the size of the time discretization. We will again only consider the case $h < N^{-\gamma}$, which is a natural stability condition and is discussed further in Section 7.

We make the standing assumption that for all $x \in \overline{\mathbb{L}}_N$ and $k, j \in \{1, \dots, R\}$, we have

$$\xi_1\lambda_k(x + \zeta_j) - \xi_2\lambda_k(x) \geq 0, \tag{5.14}$$

where $\xi_1 > \xi_2$ are defined in (2.8) for some $\theta \in (0, 1)$.

Theorem 5.10. *Suppose that the step size h satisfies $h < N^{-\gamma}$. Then*

$$\|(P_{trap,h} - \mathcal{P}_h)\|_{3\to 0}^{\nabla^N} = O(N^{3\gamma}h^3).$$

Proof. Consider one step of the method with a step-size of size h and with initial value x_0. Note that the first step of the algorithm produces a value y^* that is distributionally equivalent to one produced by a Markov process with generator B_1^N given by

$$B_1^N f(x) = N^\gamma \lambda(x_0) \cdot \nabla^N f(x).$$

Next, given both x_0 and y^*, step 2 produces a value which is distributionally equivalent to one produced by a Markov process with generator

$$B_2^N f(x) = N^\gamma [\xi_1 \lambda(y^*) - \xi_2 \lambda(x_0)]^+ \cdot \nabla^N f(x). \tag{5.15}$$

Recall that for the exact process,

$$\mathcal{P}_h f(x_0) = f(x_0) + h\mathcal{A}^N f(x_0) + \frac{h^2}{2}(\mathcal{A}^N)^2 f(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3).$$

For the approximate process we have,

$$\begin{aligned} P_{trap,h} f(x_0) &= \mathbb{E}_{x_0}[\mathbb{E}_{x_0}[f(Z^N_{trap}(h))|y^*]] \\ &= \mathbb{E}_{x_0} f(y^*) + (1-\theta)h\mathbb{E}_{x_0}[B_2^N f(y^*)] + \frac{(1-\theta)^2 h^2}{2}\mathbb{E}_{x_0}[(B_2^N)^2 f(y^*)] + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3). \end{aligned} \tag{5.16}$$

We will expand each piece of (5.16) in turn. Noting that $B_1^N f(x_0) = \mathcal{A}^N f(x_0)$, the first term is

$$\begin{aligned} \mathbb{E}_{x_0} f(y^*) &= f(x_0) + \mathbb{E}_{x_0}\left[\int_0^{\theta h} B_1^N f(Z_s) ds\right] \\ &= f(x_0) + \theta h \mathcal{A}^N f(x_0) + \frac{\theta^2 h^2}{2}(B_1^N)^2 f(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3). \end{aligned}$$

We turn attention to the second term, $(1-\theta)h\mathbb{E}_{x_0}[B_2^N f(y^*)]$, and begin by making the following definition:

$$g(y^*) \overset{\text{def}}{=} B_2^N f(y^*) = N^\gamma[\xi_1\lambda(y^*) - \xi_2\lambda(x_0)]^+ \cdot \nabla^N f(y^*),$$

so that $g(x) = N^\gamma([\xi_1\lambda(x) - \xi_2\lambda(x_0)]^+ \cdot \nabla^N) f(x)$. Because $\xi_1 - \xi_2 = 1$, we have

$$g(x_0) = N^\gamma \lambda(x_0) \cdot \nabla^N f(x_0) = \mathcal{A}^N f(x_0).$$

By our standing assumption (5.14)

$$g(x_0 + \zeta_k) - g(x_0) = N^\gamma(\xi_1\lambda(x_0+\zeta_k) - \xi_2\lambda(x_0)) \cdot \nabla^N f(x_0 + \zeta_k) - N^\gamma \lambda(x_0) \cdot \nabla^N f(x_0).$$

After some algebra

$$\begin{aligned} B_1^N g(x_0) &= N^\gamma(\lambda(x_0) \cdot \nabla^N g)(x_0) = N^\gamma \sum_k N^{c_k}\lambda_k(x_0)[g(x_0+\zeta_k) - g(x_0)] \\ &= \xi_1 N^\gamma \lambda(x_0) \cdot \nabla^N(N^\gamma \lambda \cdot f)(x_0) - \xi_2 N^\gamma \lambda(x_0) \cdot \nabla^N(\lambda(x_0) \cdot f)(x_0) \\ &= \xi_1(B_1^N \mathcal{A}^N f(x_0)) - \xi_2((B_1^N)^2 f)(x_0). \end{aligned}$$

Thus,

$$\begin{aligned}
\mathbb{E}_{x_0}[B_2^N f(y^*)] &= \mathbb{E}_{x_0}[g(y^*)] = g(x_0) + \theta h B_1^N g(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^2) \\
&= \mathcal{A}^N f(x_0) + \theta h \left[\xi_1 (B_1^N \mathcal{A}^N f)(x_0) - \xi_2 (B_1^N)^2 f(x_0)\right] + O(N^{3\gamma}\|f\|_2^{\nabla^N} h^2) \\
&= \mathcal{A}^N f(x_0) + \theta h \left[\xi_1 (\mathcal{A}^N)^2 f(x_0) - \xi_2 (B_1^N)^2 f(x_0)\right] + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^2),
\end{aligned}$$

where the last line follows since $B_1^N f(x_0) = \mathcal{A}^N f(x_0)$ for any f.

Finally, we turn the the last term in (5.16). Define

$$\begin{aligned}
q(y^*) &\overset{\text{def}}{=} (B_2^N)^2 f(y^*) \\
&= [\xi_1 \lambda(y^*) - \xi_2 \lambda(x_0)]^+ \cdot \nabla^N([\xi_1 \lambda(y^*) - \xi_2\lambda(x_0)]^+ \nabla^N f)(y^*),
\end{aligned}$$

so that

$$q(x) = [\xi_1 \lambda - \xi_2 \lambda(x_0)]^+ \cdot \nabla^N([\xi_1\lambda - \xi_2\lambda(x_0)]^+\nabla^N f)(x).$$

By our standing assumption (5.14) we have

$$\begin{aligned}
\mathbb{E}_{x_0}[(B_2^N)^2 f(y^*)] &= \mathbb{E}_{x_0}[q(y^*)] \\
&= q(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h) \\
&= (B_1^N)^2 f(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h).
\end{aligned} \tag{5.17}$$

Noting that

$$(1-\theta)\theta\xi_1 = \frac{1}{2} \qquad \text{and} \qquad (1-\theta)\theta\xi_2 = \frac{(1-\theta)^2+\theta^2}{2},$$

we may conclude the following from the above calculations

$$\begin{aligned}
\mathbb{E}_{x_0}[f(Z^N_{trap,h})] &= \mathbb{E}_{x_0} f(y^*) + (1-\theta)h\mathbb{E}_{x_0}[B_2^N f(y^*)] + \frac{(1-\theta)^2h^2}{2}\mathbb{E}_{x_0}[(B_2^N)^2 f(y^*)] \\
&\quad + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3) \\
&= f(x_0) + \theta h \mathcal{A}^N f(x_0) + \frac{\theta^2 h^2}{2}(B_1^N)^2 f(x_0) \\
&\quad + (1-\theta)h\mathcal{A}^N f(x_0) + \frac{h^2}{2}(\mathcal{A}^N)^2 f(x_0) - \frac{h^2}{2}[(1-\theta)^2+\theta^2](B_1^N)^2 f(x_0) \\
&\quad + \frac{(1-\theta)^2h^2}{2}(B_1^N)^2 f(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3) \\
&= f(x_0) + \mathcal{A}^N f(x_0) + \frac{h^2}{2}(\mathcal{A}^N)^2 f(x_0) + O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3).
\end{aligned}$$

Thus

$$\|(P_{trap,h} - \mathcal{P}_h)f\|_0 \in O(N^{3\gamma}\|f\|_3^{\nabla^N} h^3),$$

and the proof is complete. □

6 Bound on $\|\mathcal{P}_t f\|_n^{\nabla^N}$

In this section we will provide a bound on $\|\mathcal{P}_t f\|_n^{\nabla^N}$ for any nonnegative n. We point out, however, that for any process X^N for which $\mathcal{P}_t$ is well behaved, in that $\|\mathcal{P}_t\|_{n\to 0}^{\nabla^N}$ is bounded without any N dependence, the following results are not needed, and, in fact, would most likely be a *least optimal* bound, as the bound grows exponentially in $N^\gamma t$. Note that any system satisfying the classical scaling has $\gamma = 0$.

For $t \geq 0$ and $x \in \mathbb{L}_N$, We define

$$v(t,x) \overset{\text{def}}{=} \mathcal{P}_t f(x) = \mathbb{E}_x[f(X_t^N)].$$

Theorem 6.1. *If $\|f\|_n^{\nabla^N} < \infty$, then*

$$\|v(t,\cdot)\|_n^{\nabla^N} = \|\mathcal{P}_t f\|_n^{\nabla^N} \leq \|f\|_n^{\nabla^N} e^{N^\gamma C_n t}$$

where

$$C_n = 2\left(\|\lambda\|_1^{\nabla^N} n\ R + R(n-1)\|\lambda\|_n^{\nabla^N}\right). \tag{6.1}$$

We delay the proof of Theorem 6.1 until the following Lemma is shown, the proof of which is similar to that found in [17], which itself was an extension of the proof of Lemma 4.3 in [3].

Lemma 6.2. *Given a multiset I of $\{1, \cdots, R\}$, there exists a function $q_I(x)$ that is a linear function of terms of the form $\nabla_J^N v(t,x)$ with $|J| < |I|$, so that*

$$\partial_t \nabla_I^N v(t,x) = N^\gamma(\lambda \cdot \nabla^N)\nabla_I^N v(t,x) + N^\gamma \sum_{i=1}^{|I|} (\beta_i \cdot \nabla^N)\nabla_{I\setminus \ell_i} v(t, x+\zeta_{\ell_i}) + N^\gamma q_I(x),$$

where $\beta_i = \nabla_{\ell_i}^N \lambda$. Further, q_I consists of at most $R(|I|-1)$ terms of the form $\nabla_J^N v(t,x)$, each of whose coefficients are bounded above by $\|\lambda\|_{|I|}^{\nabla^N}$.

Proof. This goes by induction. For $|I| = 0$, the statement follows because

$$\partial_t v(t,x) = N^\gamma(\lambda \cdot \nabla^N) v(t,x). \tag{6.2}$$

Note that in this case, there are no β_i or q terms. It is instructive to perform the $|I| = 1$ case. We have

$$\begin{aligned}
\partial_t \nabla_k^N v(t,x) &= \nabla_k^N \partial_t v(t,x) \\
&= \nabla_k^N (N^\gamma \lambda \cdot \nabla^N v(t,x)) \\
&= N^\gamma(\nabla_k^N \lambda \cdot \nabla^N) v(t,x) + N^\gamma \lambda \cdot \nabla_k^N \nabla^N v(t,x) + N^\gamma(N^{-c_k} \nabla_k^N \lambda \cdot \nabla_k^N \nabla^N v(t,x)).
\end{aligned}$$

Note that for any $g : \mathbb{R}^d \to \mathbb{R}$

$$(\nabla_k^N \lambda \cdot \nabla^N) g(x) + (N^{-c_k} \nabla_k^N \lambda \cdot \nabla^N)\nabla_k^N g(x) = (\nabla_k^N \lambda \cdot \nabla^N) g(x+\zeta_k). \tag{6.3}$$

Therefore, with $g(x) = v(t,x)$ in the above, we have

$$\partial_t \nabla_k^N v(t,x) = N^\gamma(\lambda(x) \cdot \nabla^N)\nabla_k^N v(t,x) + N^\gamma(\nabla_k^N \lambda(x) \cdot \nabla^N) v(t, x+\zeta_k).$$

Now assume that it holds for a set of size $\leq |I|$. Then, using the inductive hypothesis, Lemma 5.3, and equation (6.3) yields

$$
\begin{aligned}
&\partial_t \nabla_k^N \nabla_I^N v(t,x) \\
&\quad = \nabla_k^N \partial_t \nabla_I^N v(t,x) \\
&\quad = N^\gamma \nabla_k^N \left[(\lambda \cdot \nabla^N) \nabla_I^N v(t,x) + \sum_{i=1}^{|I|} (\beta_i \cdot \nabla^N) \nabla_{I \backslash \ell_i} v(t, x+\zeta_{\ell_i}) + q_I(x) \right] \\
&\quad = N^\gamma \left[(\lambda \cdot \nabla^N) \nabla_{I \cup k}^N v(t,x) + (\nabla_k^N \lambda \cdot \nabla^N) \nabla_I^N v(t, x+\zeta_k) \right] \\
&\qquad + N^\gamma \sum_{i=1}^{|I|} \left[(\beta_i \cdot \nabla^N) \nabla_k^N \nabla_{I \backslash \ell_i}^N v(t, x+\zeta_{\ell_i}) + (\nabla_k^N \beta_i \cdot \nabla^N) \nabla_{I \backslash \ell_i}^N v(t, x+\zeta_{\ell_i}+\zeta_k) \right] \\
&\qquad + N^\gamma \nabla_k^N q_I(x) \\
&\quad = N^\gamma (\lambda \cdot \nabla^N) \nabla_{I \cup k}^N v(t,x) + N^\gamma \left[(\nabla_k^N \lambda \cdot \nabla^N) \nabla_{I \cup k \backslash k}^N v(t, x+\zeta_k) + \sum_{i=1}^{|I|} (\beta_i \cdot \nabla^N) \nabla_{I \cup k \backslash \ell_i}^N v(t, x+\zeta_{\ell_i}) \right] \\
&\qquad + N^\gamma \left[\nabla_k^N q_I(x) + (\nabla_k^N \beta_i \cdot \nabla^N) \nabla_{I \backslash \ell_i}^N v(t, x+\zeta_{\ell_i}+\zeta_k) \right],
\end{aligned}
$$

showing the result. □

Proof. (*of Theorem 6.1*)

Let $n \geq 0$. Define

$$
U_n(t) \overset{\text{def}}{=} \max_{x \in \mathbb{L}_N, |I| \leq n} |\nabla_I^N v(t,x)| = \|v\|_n^{\nabla^N}.
$$

Each $\nabla_I^N v(t,x)$ is a continuously differentiable function with respect to t. Therefore, the maximum above is achieved at some (I^*, x^*) for all $t \in [0, t_1]$ where $t_1 > 0$. Fixing this choice of (I^*, x^*), we have

$$
U_n(t) = \nabla_{I^*}^N v(t, x^*)
$$

for all $t < t_1$.

Note that

$$
\begin{aligned}
[(\lambda \cdot \nabla^N) \nabla_{I^*}^N v(t,x^*)] \nabla_{I^*}^N v(t,x^*) &= \sum_k \lambda_k(x) (\nabla_k^N \nabla_{I^*}^N v(t,x^*)) \nabla_{I^*}^N v(t,x^*) \\
&= \sum_k N^{c_k} \lambda_k(x) (\nabla_{I^*}^N v(t, x^* + \zeta_k) - \nabla_{I^*}^N v(t,x^*)) \nabla_{I^*}^N v(t,x^*) \qquad (6.4) \\
&\leq 0,
\end{aligned}
$$

where the final inequality holds by the specific choice of I^* and x^*. Also note that for any $\ell_i \in I^*$ and any choice of $x \in \overline{\mathbb{L}}_N$

$$
|\nabla^N \nabla_{I^* \backslash \ell_i}^N v(t,x)| \leq \sum_{k=1}^R |\nabla_k \nabla_{I^* \backslash \ell_i}^N v(t,x)| \leq R |\nabla_{I^*}^N v(t,x^*)|. \qquad (6.5)
$$

From Lemma 6.2 and equations (6.4) and (6.5), we have

$$\begin{aligned}\frac{1}{2}\partial_t(\nabla^N_{I^*}v(t,x^*))^2 &= (\partial_t\nabla^N_{I^*}v(t,x^*))\nabla^N_{I^*}v(t,x^*)\\ &= N^\gamma\left[(\lambda\cdot\nabla^N)\nabla^N_{I^*}v(t,x^*)+\sum_{i=1}^{|I^*|}(\beta_i\cdot\nabla^N)\nabla_{I^*\setminus\ell_i}v(t,x^*+\zeta_{\ell_i})+q_{I^*}(x^*)\right]\nabla^N_{I^*}v(t,x^*) \qquad (6.6)\\ &\le N^\gamma\left[\|\lambda\|_1^{\nabla^N}|I^*|\ R\ |\nabla^N_{I^*}v(t,x^*)|^2+R(|I^*|-1)\|\lambda\|^{\nabla^N}_{|I^*|}|\nabla^N_{I^*}v(t,x^*)|^2\right],\end{aligned}$$

where we have used the fact that each $\beta_i=\nabla_{\ell_i}\lambda$ for $\ell_i\in I^*$. Setting

$$C_n = 2\left(\|\lambda\|_1^{\nabla^N}n\ R+R(n-1)\|\lambda\|_n^{\nabla^N}\right), \qquad (6.7)$$

we see by an application of Gronwall's inequality that the conclusion of the theorem holds for all $t<t_1$. That is, for $t<t_1$

$$U_n(t)\le\|f\|_n^{\nabla^N}e^{N^\gamma C_n t}.$$

To continue, repeat the above argument on the interval $[t_1,t_2)$, with I^*,x^* again chosen to maximize U_n on that interval, and note that

$$U_n(t_1)\le\|f\|_n^{\nabla^N}e^{N^\gamma C_n t_1},$$

so that we may conclude that for $t_1\le t<t_2$,

$$U_n(t)\le\|f\|_n^{\nabla^N}e^{N^\gamma C_n t_1}e^{N^\gamma C_n(t-t_1)}=\|f\|_n^{\nabla^N}e^{N^\gamma C_n t}.$$

Continuing on, we see that $t_i\to\infty$ as $i\to\infty$ by the boundedness of the time derivatives of $v(t,x)$, thereby concluding the proof. □

Remark 6.3. In the theorem above, $C_n\in\|\lambda\|_n^{\nabla^N}$.

Combining all of the above results, we have the following theorems.

Theorem 6.4. ***(Global bound for the Euler method)***
For Algorithm 1, suppose that the step size h satisfies $h<N^{-\gamma}$, and $T=nh$. Then

$$\|(P^n_{E,h}-\mathcal{P}_{nh})\|^{\nabla^N}_{2\to0}=O(N^{2\gamma}he^{C_2N^\gamma T})$$

where $C_2\in O(\|\lambda\|_2^{\nabla^N})$ is defined in (6.1).

Theorem 6.5. ***(Global bound for the midpoint tau-leap method)***
For Algorithm 2, suppose that the step size h satisfies $h<N^{-\gamma}$, and $T=nh$. Then

$$\|(P^n_{M,h}-\mathcal{P}_{nh})\|^{\nabla^N}_{3\to0}=O([N^{3\gamma}h^2+N^{2\gamma-\min\{m_k\}}h]e^{C_3N^\gamma T})$$

where $C_3\in O(\|\lambda\|_3^{\nabla^N})$ is defined in (6.1).

The following immediate corollary to the theorem above recovers the result in [3].

Corollary 6.6. *Under the additional condition $h>N^{-\gamma-\min\{m_k\}}$ in Theorem* (6.5)*, the leading order of the error of midpoint tau-leaping is $O(h^2)$.*

Theorem 6.7. ***(Global bound for the weak trapezoidal method)***
For Algorithm 3, suppose that the step size h satisfies $h < N^{-\gamma}$, and $T = nh$. Then

$$\|P_h^n - \mathcal{P}_{nh}\|_{3\to 0}^{\nabla^N} = O(h^2 N^{3\gamma} e^{N^\gamma C_3 T})$$

where $C_3 \in O(\|\lambda\|_3^{\nabla^N})$ is defined in (6.1).

Thus, we see that the weak trapezoidal method detailed in Algorithm 3 is the only method that boasts a global error of second order in the stepsize h in an honest sense. That is, it is a second order method for the multiple scalings regardless of the relation of h with respect to N. This is in contrast to the midpoint tau leaping method, Algorithm 2, which has second order accuracy only when the step-size h is larger than $N^{-\gamma-\min\{m_k\}}$.

7 Stability

The main results and proofs of our paper have incorporated stability concerns into the analysis. This is seen in the statements of the theorems by the running condition that $h < N^{-\gamma}$, where we recall that N^γ should be interpreted as the time-scale of the system. Without this condition, the methods are unstable, and in fact the convergence results do not hold. It is an interesting question, and the subject of future work, to determine the stability properties of other methods in this setting.

As an instructive example, again consider the system

$$S_1 \underset{100}{\overset{100}{\rightleftarrows}} S_2$$

with $X_1(0) = X_2(0) = 10{,}000$. In this case, it is natural to take $N = 10{,}000$. As the rate constants are $100 = \sqrt{10{,}000}$, we take $\beta_1 = \beta_2 = 1/2$ and find that $\gamma = 1/2$. The equation governing the normalized process X_1^N is

$$X_1^N(t) = X_1^N(0) - Y_1\left(N^{1/2}N\int_0^t X_1^N(s)ds\right)\frac{1}{N} + Y_2\left(N^{1/2}N\int_0^t (2 - X_1^N(s))ds\right)\frac{1}{N}$$

where we have used that $X_1^N + X_2^N \equiv 2$. It is now clear that if the condition $h < N^{-\gamma}$ is violated a path generated by any of the explicit methods discussed in this paper will behave quite poorly.

8 Example

Consider the following first order reaction network

$$A \underset{\kappa_2}{\overset{\kappa_1}{\rightleftarrows}} B \underset{\kappa_4}{\overset{\kappa_3}{\rightleftarrows}} C,$$

with $\kappa_1 = 0.03, \kappa_2 = 1, \kappa_3 = 0.1$, and $\kappa_4 = 1$. Starting from the initial state

$$X(0) = (X_A(0), X_B(0), X_C(0)) = (13000, 100, 20),$$

where we make the obvious associations $X_1 = X_A, X_2 = X_B$, and $X_3 = X_C$. We approximate $X(2)$ using the three methods considered in this paper: Euler, midpoint, and weak trapezoidal. For first order systems, we may find the first moments and the covariances of $X(t)$ as solutions of linear ODEs using a Moment Generating function approach [11].

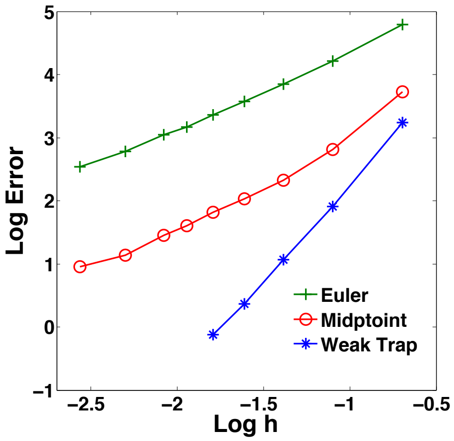


Figure 1: The log-log plot of $|\mathbb{E}[X_3^2(2)] - \mathbb{E}[Z_3^2(2)]|$ against h for the three approximation methods. The slope for Euler's method is 1.21, whereas the slope for the weak trapezoidal solution is 3.06, which is better than expected. The curve governing the solution from the midpoint method appears to not be linear; a behavior predicted by Theorem 5.6.

In Figure 1, we show a log-log plot of $|\mathbb{E}[X_3^2(2)] - \mathbb{E}[Z_3^2(2)]|$ against h for the three approximation methods. Each data point was found from either $10^6, 2.9 \times 10^6, 3.9 \times 10^6, 4.9 \times 10^6, 8 \times 10^6$, or 10^7 independent simulations, with the number of simulations depending upon the size of h and the method being used. The slope for Euler's method is 1.21, whereas the slope for the weak trapezoidal solution is 3.06, which is better than expected. The curve governing the solution from the midpoint method appears to not be linear; a behavior predicted by Theorem 5.6.

In Figure 2 we again consider the log-log plots of $|\mathbb{E}[X_3^2(2)] - \mathbb{E}[Z_3^2(2)]|$ against h, but now only for Euler's method and the midpoint method so that we may see the change in behavior in the midpoint method predicted in Theorem 5.6. In (a), we see that for larger h the slope generated via the midpoint method is 2.03, whereas in (b) the slope is 1.12 when h is smaller. For reference, in (a) the slope generated by Euler's method is 1.366, whereas in (b) it is 1.09.

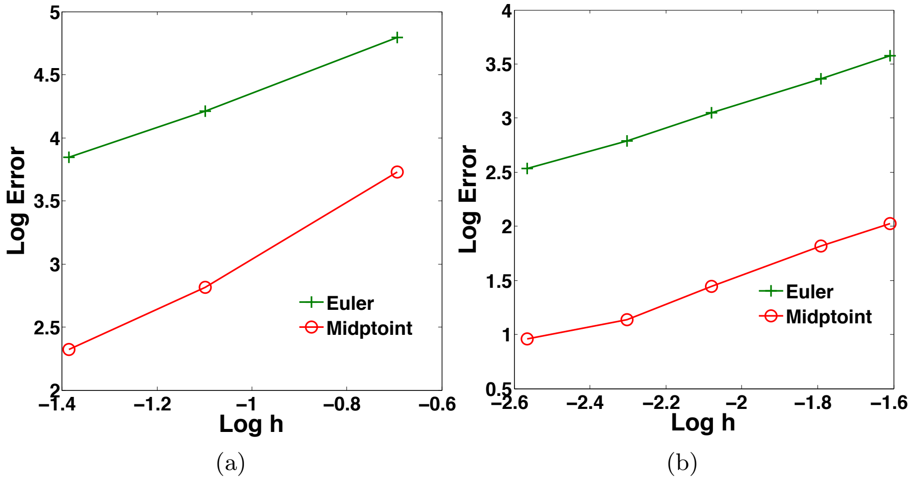


Figure 2: The log-log plot of $|\mathbb{E}[X_3^2(2)] - \mathbb{E}[Z_3^2(2)]|$ against h. The slope for generated via midpoint tauleaping shifts from 2.03 in (a) to 1.12 in (b).

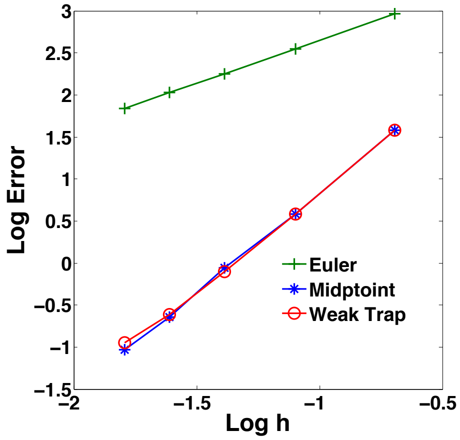


Figure 3: Log-log plot of $|\mathbb{E}[X_3(2)] - \mathbb{E}[Z_3(2)]|$ against h for the three approximation methods. The slopes are: 1.02 for Euler's methods, 2.372 for midpoint method, and 2.3 for the trapezoidal method.

While the simulations make no use of the scalings inherent in the system, it is instructive for us to quantify them in this example so that we are able to understand the behavior of the midpoint method. We have $N \approx 10^4$, $\alpha_1 = 1, \alpha_2 = 1/2$, $\alpha_3 = 1/4$, and $m_k = 1/4$. Also, $\gamma \approx 0$. Therefore, Theorem 5.6 predicts the midpoint method will behave as an order two method if $h \gg N^{-1/4} \approx 1/10$, or if $\log(h) \gg -2.3$, which roughly agrees with what is shown in Figures 1 and 2. Note that Theorem 5.6 will never provide a sharp estimate as to when the behavior will change as it is a local result and the scalings in the system will change during the course of a simulation.

The fact that the trapezoidal method gave an order three convergence rate above does not hold in general. This was demonstrated in the proof of Theorem 5.10, but it is helpful to also show this via example. In Figure 3 we present a log-log plot of $|\mathbb{E}X_2(2) - \mathbb{E}Z_2(2)|$ for the different algorithms on this same example. The slopes are: 1.02 for Euler's methods, 2.372 for midpoint method, and 2.3 for the trapezoidal method. Note that Corollary 5.9 explains why the midpoint method is as accurate as the weak trapezoidal algorithm in this case. We point out that all of the plots above represent results pertaining to the *non-normalized processe* as the simulation methods themselves make no use of the scalings.

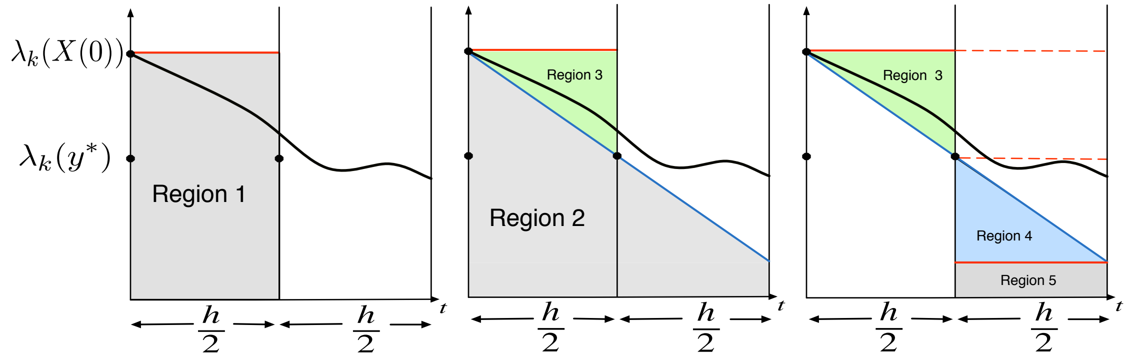
A Why the weak trapezoidal algorithm works

This discussion follows that of Section 3.1 in [5]. Equation (2.1) is distributionally equivalent[2] to

$$X(t) = X(0) + \sum_k \zeta_k \int_0^\infty \int_0^t 1_{[0,\lambda_k(X(s)))}(u)\mu_k(du \times ds), \tag{A.1}$$

where the μ_k are independent, unit-rate Poisson random measures on $[0,\infty)^2$ with Lebesgue mean measure. That is, if $A, B \subset [0,\infty)^2$ with $A \cap B = \emptyset$, then $\mu_k(A)$ and $\mu_k(B)$ are independent Poisson random variables with parameters Area(A) and Area(B), respectively. All other notation is as before. In order to approximate the integral in (A.1) over the time interval $[0,h)$, we must approximate $\mu_k(A_{[0,h)}(\lambda_k))$ where $A_{[0,h)}(\lambda_k)$ is the region under the curve $\lambda_k(X(t))$ for $0 \le t < h$.

[2] in that solutions to (A.1) are Markov processes that solve the same martingale problem as solutions to (2.1); that is, they have the same generator [10].



(a) First step (b) Desired second step (c) Used second step

Figure 4: A graphical depiction of the Weak trapezoidal algorithm with $\theta = 1/2$. In (a) the region of space-time used in the first step of the Weak trapezoidal algorithm is depicted by the grey shaded Region 1. In (b) the desired region to use, in order to perform a trapezoidal approximation, would be Region 2. However, we have used Region 3 in our previous calculation and this is analytically problematic to undo. In (c) we see that Region 5 gives the correct amount of new area wanted as subtracting off the area of Region 4 "offsets" the used area of Region 3. The case $\theta \neq 1/2$ is similar.

We consider a natural way to approximate $X(h)$ and focus on the double integral in (A.1) for a single k. We also take $\theta = 1/2$ for simplicity and simply note that the case $\theta \neq 1/2$ follows similarly. We begin by approximating the value $X(h/2)$ by y^* obtained via an Euler approximation of the system on the interval $[0, h/2)$. To do so, we hold $X(t)$ fixed at $X(0)$ and see that we need to calculate μ_k(Region 1), where Region 1 is the grey shaded region in Figure 4(a). Because

$$\mu_k(\text{Region 1}) \overset{\mathcal{D}}{=} \text{Poisson}(\lambda_k(X(0))h/2),$$

we see that this step is equivalent in distribution to the first step of Algorithm 3, where "$\overset{\mathcal{D}}{=}$" denotes "equal in distribution."

If we were trying to determine the area under the curve $\lambda_k(X(t))$ using an estimated midpoint y^* for a deterministic $X(t)$, one natural (and common) way would be to use the area of Region 2, where Region 2 is the grey shaded region in Figure 4(b). Such a method would be equivalent to a trapezoidal rule. However, in our setting we would have to ignore, or subtract off, the area already accounted for in Region 3, which is depicted as the shaded green section of Figure 4(b). In doing so, the random variable needed in order to perform this step would necessarily be dependent upon the past (via Region 3), and our current analysis would break down. However, noting that Region 3 has the same area as Region 4, as depicted by the blue shaded region in Figure 4(c), we see that it would be reasonable to expect that if one only uses Region 5, as depicted as the grey shaded region in Figure 4(c), then the accuracy of the method should be improved as we have performed a trapezoidal type approximation. Because, after some algebra,

$$\mu_k(\text{Region 5}) \overset{\mathcal{D}}{=} \text{Poisson}(2\lambda_k(y^*) - \lambda_k(X(0))),$$

we see that this is precisely what is carried out by Step 2 of the Weak Trapezoidal Algorithm.

References

[1] David F. Anderson, *A modified next reaction method for simulating chemical systems with time dependent propensities and delays*, J. Chem. Phys. **127** (2007), no. 21, 214107.

[2] ______, *Incorporating postleap checks in tau-leaping*, J. Chem. Phys. **128** (2008), no. 5, 054103.

[3] David F. Anderson, Arnab Ganguly, and Thomas G. Kurtz, *Error analysis of tau-leap simulation methods*, to appear in Annals of Applied Probability.

[4] David F. Anderson and Thomas G. Kurtz, *Continuous time markov chain models for chemical reaction networks*, Design and Analysis of Biomolecular Circuits: Engineering Approaches to Systems and Synthetic Biology (H. Koeppl et al., ed.), Springer, 2011, pp. 3–42.

[5] David F. Anderson and Jonathan C. Mattingly, *A weak trapezoidal method for a class of stochastic differential equations*, Comm. Math. Sci. **9** (2011), no. 1, 301 – 318.

[6] Karen Ball, Thomas G. Kurtz, Lea Popovic, and Greg Rempala, *Asymptotic analysis of multiscale approximations to reaction networks*, Ann. Appl. Prob. **16** (2006), no. 4, 1925–1961.

[7] Yang Cao, Daniel T. Gillespie, and Linda R. Petzold, *Avoiding negative populations in explicit poisson tau-leaping*, J. Chem. Phys. **123** (2005), 054104.

[8] ______, *Efficient step size selection for the tau-leaping simulation method*, J. Chem. Phys. **124** (2006), 044109.

[9] Abhijit Chatterjee and Dionisios G. Vlachos, *Binomial distribution based τ-leap accelerated stochastic simulation*, J. Chem. Phys. **122** (2005), 024112.

[10] Stewart N. Ethier and Thomas G. Kurtz, *Markov processes: Characterization and convergence*, John Wiley & Sons, New York, 1986.

[11] Chetan Gadgil, Chang Hyeong Lee, and Hans G. Othmer, *A stochastic analysis of first-order reaction networks*, Bull. Math. Bio. **67** (2005), 901–946.

[12] M.A. Gibson and J. Bruck, *Efficient exact stochastic simulation of chemical systems with many species and many channels*, J. Phys. Chem. A **105** (2000), 1876–1889.

[13] D. T. Gillespie, *A general method for numerically simulating the stochastic time evolution of coupled chemical reactions*, J. Comput. Phys. **22** (1976), 403–434.

[14] ______, *Exact stochastic simulation of coupled chemical reactions*, J. Phys. Chem. **81** (1977), no. 25, 2340–2361.

[15] ______, *Approximate accelerated simulation of chemically reaction systems*, J. Chem. Phys. **115** (2001), no. 4, 1716–1733.

[16] D. T. Gillespie and Linda R. Petzold, *Improved leap-size selection for accelerated stochastic simulation*, J. Chem. Phys. **119** (2003), no. 16, 8229–8234.

[17] Yucheng Hu, Tiejun Li, and Bin Min, *The weak convergence analysis of tau-leaping methods: revisited*, accepted to Comm. Math. Sci.

[18] Hye-Won Kang and Thomas G. Kurtz, *Separation of time-scales and model reduction for stochastic reaction networks*, submitted.

[19] Peter E. Kloeden and Eckhard Platen, *Numerical solution of stochastic differential equations*, Applications of Mathematics (New York), vol. 23, Springer-Verlag, Berlin, 1992. MR MR1214374 (94b:60069)

[20] Thomas G. Kurtz, *Approximation of population processes*, CBMS-NSF Reg. Conf. Series in Appl. Math.: 36, SIAM, 1981.

[21] Tiejun Li, *Analysis of explicit tau-leaping schemes for simulating chemically reacting systems*, SIAM Multiscale Model. Simul. **6** (2007), no. 2, 417–436.

[22] Muruhan Rathinam, Linda R. Petzold, Yang Cao, , and Daniel T. Gillespie, *Consistency and stability of tau-leaping schemes for chemical reaction systems*, SIAM Multiscale Model. Simul. **3** (2005), 867–895.