Advanced Mathematical Biology IV (MATH4411)

Dr Denis Patterson, Durham University

2025-06-01

Table of contents

Michaelmas Term Overview

In Michaelmas term of this course, we will study a range of stochastic models that allow us to analyse populations of biological agents moving and interacting in both space and time. We will consider general individual-based models in the form of stochastic simulation algorithms accounting for reactions among chemical species, or interactions between individuals in a population. We show how these models can, in various limits, give rise to the same continuum-level descriptions obtained via conservation laws at the macroscale. However, we will also demonstrate novel features of stochastic individual-based models, such as stochastic resonance and noise-induced attractor switching, that are not present in the deterministic macroscale models.

A key focus will be on designing and understanding stochastic simulation algorithms to simulate various stochastic processes on a computer. Simultaneously, we will develop a toolkit of analytic approaches to studying these processes that will validate and complement direct numerical simulations. We begin by studying how to simulate spatially homogeneous reaction systems, before moving on to more complex spatially extended systems later in the course. Applications will include models from ecology, neuroscience, chemistry, genetics and cell biology.

Content

The content of Term 1 is divided into 4 Chapters, some of which will be longer than others as we devote more time earlier in the course to foundational concepts that will be needed throughout.

- **Chapter 1: Stochastic Simulation of Chemical Reactions**
  - Stochastic models of 1st, 2nd, and higher order chemical reaction systems
  - "Naive" versus Gillespie stochastic simulation algorithms
  - The chemical master equation
  - Stationary distributions and probability generating functions
- **Chapter 2: Deterministic vs Stochastic Models**
  - Disagreement between ODEs and stochastic systems
  - Stochastic resonance
  - Finite-size effects and stochastic forcing

- **Chapter 3: Stochastic Differential Equations (SDEs)**
  - Computational definition of SDEs
  - Examples with drift, diffusion and bistability
  - Introduction to the Fokker-Planck equation (and Chemical F-P equation)
  - Mean switching times and bistability
- **Chapter 4: Stochastic Reaction-diffusion Models**
  - Modeling diffusion with SDEs
  - Discrete approach to diffusion
  - Spatially discrete approach to reaction-diffusion
  - SDE approach to reaction-diffusion
  - Pattern formation in stochastic models

Lectures, Problem Classes & Homeworks

Lectures will primarily be used to present new material, but will also feature computer demonstrations of the algorithms and models presented. As such, students are encouraged to bring along their own laptops to both lectures and problem classes so that they can also run the examples themselves. The MATLAB code for all of the examples will be available on the course Github page:

https://github.com/patterd2/MATH4411_Adv_Math_Bio

You don't need any prior coding experience in MATLAB to run this code; mostly you will just want to tweak parameter values, analyse the output of the code, and compare the algorithms to the pseudocode in the lecture notes.

Problem classes will not contain any new material, but will focus on the presentation of additional examples, discussion of lecture material, and solving problems from the problem sheets. There will be 4 short homework assignments that will be graded for the purposes of formative assessment (i.e. letter grade of A-D with additional comments and feedback). Homeworks will be due every two weeks with the first homework due in week 4 (precise submission instructions are available on Ultra and on the homework question sheets themselves).

	Activities	Content
Week 1	Introductory lecture, 1 lecture	Chapter 1
Week 2	2 lectures, 1 problem class	Chapter 1
Week 3	2 lectures, HW1 due	Chapter 1
Week 4	2 lectures, 1 problem class	Chapter 2
Week 5	2 lectures, HW2 due	Chapter 2
Week 6	2 lectures, 1 problem class	Chapter 3
Week 7	2 lectures, HW3 due	Chapter 3

	Activities	Content
Week 8	2 lectures, 1 problem class	Ch. 3/Ch. 4
Week 9	2 lectures, HW4 due	Chapter 4
Week 10	1 lecture, 1 rev. class + extra prob. sheet	Chapter 4

Additional Reading

The lecture notes are designed to be sufficient for the course and hence students do not need to purchase a textbook to successfully complete the course. References for additional reading will also be given at the end of each chapter. The main reference for the course content in Michaelmas term is:

- **Erban, R. & Chapman, S.J.** *Stochastic Modelling of Reaction-Diffusion Processes*, Vol. 60, Cambridge University Press, 2020.

Supplementary material can also be found in:

- **Murray, J.D.** *Mathematical Biology: II: Spatial Models and Biomedical Applications*, Vol. 3, Springer, 2003.
- **Keener, J.P.** *Biology in Time and Space: A Partial Differential Equation Modeling Approach*, Vol. 50, American Mathematical Society, 2021.

Contact Information

For questions or clarifications on any of the above, please come speak to me in lectures, office hours or drop me an email at denis.d.patterson@durham.ac.uk.

1 Stochastic Simulation of Chemical Reactions

The goal of this chapter is to introduce stochastic chemical reaction processes and some algorithms to simulate them numerically. We will also develop some analytic tools to study these processes and begin to discuss similarities and differences between deterministic and stochastic models of the same phenomena.

1.1 Stochastic Simulation of Degradation

1.1.1 Introducing Stochastic Simulation Algorithms (SSAs)

We begin with the simplest possible chemical reaction process. Consider an experiment starting with n_0 molecules of a chemical species denoted by A; A reacts at rate k in the absence of any stimulus and upon reacting, a molecule of A transitions into some other species that is not of interest to us.

We can represent this reaction process more succinctly as:

$$A \xrightarrow{k} \emptyset.$$

The symbol $\emptyset$ here simply denotes a species not of interest in the experiment, rather than meaning that a molecule of A disappears after reacting. Note also that we are not considering the spatial extent of this experiment; we are assuming that the system is spatially homogeneous and that each molecule of A reacts exactly as described above regardless of its spatial position (an assumption we will relax later in the course). We also choose to model this system as evolving in continuous time.

Let $A(t)$ denote the number of molecules of species A in the experiment at time t. Suppose $dt > 0$ is a small quantity (think $\epsilon > 0$ that will later be sent to zero) and consider the dynamics of the process in the time interval $[t, t + dt)$. There are 3 possibilities for what can happen in $[t, t + dt)$ and we assign them the following probabilities:

- **No reactions occur:**
  $\mathbb{P}[A(t + dt) = A(t)] \approx 1 - A(t)kdt + \mathcal{O}(dt^2)$
- **Exactly one reaction occurs:**
  $\mathbb{P}[A(t + dt) = A(t) - 1] \approx A(t)kdt + \mathcal{O}(dt^2)$

- **More than one reaction occurs:**
  $\mathbb{P}[A(t+dt) < A(t) - 1] \approx \mathcal{O}(dt^2)$

If dt is sufficiently small, we can neglect the $\mathcal{O}(dt^2)$ terms and thus ignore the possibility of multiple reactions occurring "simultaneously".

> **Note**
>
> From its definition, the degradation process has the **Markov property**, i.e.
>
> $$\mathbb{P}[A(t+dt) = n \mid A(s), s \leq t] = \mathbb{P}[A(t+dt) = n \mid A(t)].$$
>
> Can you give an intuitive explanation of why this is the case? Since it is posed in continuous time, A is formally called a **Markov jump process**.

Deferring the mathematical analysis of this process for now, how could we simulate a sample path of the process to find the number of molecules of A remaining at time t given $A(0) = n_0$? We could try to find a chemical with reaction rate k and conduct the actual experiment, or we can simulate the process on a computer! To this end, we may introduce the following "naive" stochastic simulation algorithm (SSA) for the process:

At time $t = 0$, set $A(0) = n_0$, then:

1. Generate a random number $r \sim U([0, 1])$.
2. If $r < A(t)k\Delta t$, then
   - set $A(t+\Delta t) = A(t)-1$, set $t = t+\Delta t$, and go back to step 1, else set $A(t+\Delta t) = A(t)$, set $t = t + \Delta t$, and go back to step 1.

We can stop this procedure whenever there are no molecules of A left to react or when t reaches some maximal stopping time T that we choose in advance. The quantity $\Delta t > 0$ is some fixed discretisation parameter that we have chosen a priori.

But does this algorithm really simulate the process described above?

To understand the motivation for the steps, recall that a uniform random variable $r \sim U([0, 1])$ has CDF given by

$$F(x) = \mathbb{P}[r < x] = \begin{cases} 0, & x < 0, \\ x, & x \in [0, 1], \\ 1, & x > 1. \end{cases}$$

Hence the probability of one reaction occurring in the interval $[t, t+\Delta t)$ is $\mathbb{P}[r < A(t)k\Delta t] = A(t)k\Delta t$, assuming that $A(t)k\Delta t < 1$. This emphasises the need to choose Δt sufficiently small for the algorithm to have the appropriate reaction probabilities. In fact, we want to choose $\Delta t \ll 1/A(t)k$ because this ensures that the probability of multiple reactions taking place in $[t, t+\Delta t)$ is very small, which is what we want according to the definition of the process above. However, there is a trade-off between accuracy and speed here: a smaller Δt means lower probability of multiple reactions and numerical error, but a smaller Δt also means more compute time (and more intervals in which no reactions take place).

Virtually all modern programming languages have algorithms for generating (pseudo-) random numbers and we can use such routines to generate the uniformly distributed random numbers called for in step 1 to execute the **naive SSA**. The output of some simulations of the degradation process using this algorithm are shown in Figure 1.1. The left panel shows two sample paths compared to what we called the **deterministic mean**. The motivation for this comparison is that a principle called **the law of mass action** can be invoked to yield a simple deterministic model of the reaction process.

> **Note**
>
> The law of mass action states that "the rate of a chemical reaction is directly proportional to the product of the activities or concentrations of the reactants."

If we assume our chemical is at unit volume (for simplicity), then A reacts at a constant rate and so the deterministic model is given by the linear ODE

$$\frac{d}{dt}A(t) = -kA(t), \quad A(0) = n_0.$$

This equation approximately describes the evolution of the mean of the stochastic model and is readily solved to show that

$$A(t) = n_0 e^{-kt}, \quad t \geq 0.$$

The deterministic model of the mean behaviour is strictly decreasing in time and thus does not capture the qualitative behaviour of individual trajectories very well. However, the right panel of Figure 1.1 shows that it captures the behaviour of the average over a large number of simulations quite well; the stochastic mean at time t is calculated as the average value at that time over 20 simulations of the process.

It is worth noting that the naive SSA is very inefficient in this example as we chose our parameters to ensure a very small probability of multiple reactions occurring in $[t, t+\Delta t)$. In fact, we had $\mathbb{P}[\text{one reaction in } [t, t+\Delta t)] \approx 0.01$. But this means that step 1 is mostly wasted as we generate lots of random numbers that we don't use! We can be more efficient than that...

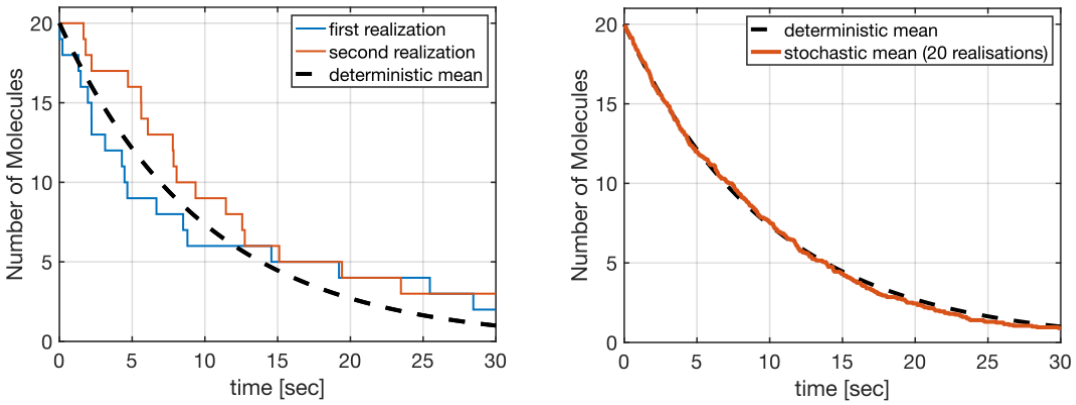

Figure 1.1: Left: Some paths of the degradation process compared to the corresponding deterministic model of the same process. Right: Stochastic mean compared to the mean predicted by the deterministic model. Parameters: $n_0 = 20$, $k = 0.1$, $\Delta t = 0.005$.

Exercise 1.1

Open the course Github page and try running the MATLAB script:
`CH1_naive_SSA_degradation.m`

1. How does the difference between the deterministic and stochastic means change as you vary the number of realisations?

2. What happens if you set $\Delta t = 1$ and how can we explain this behaviour?

From an efficiency standpoint, we can do much better than the naive SSA, but this advancement requires a change of perspective. In developing the naive SSA we focused on matching the transition probabilities of the underlying process on short time intervals, stepping forward a short time step and then repeating the probability matching step. We might instead ask: **When is the next reaction going to take place?** If we know when the next reaction will take place, then we can just skip ahead to that time, let the reaction occur, and repeat this process.

The problem we have now is that the time of the next reaction is random. But since it is just some (continuous) random variable related to the reaction rate, we can compute its distribution. To this end, let τ denote the next reaction time and take $t, s > 0$. Define the function $f(A(t), s)ds$ to be the probability that no reaction occurs in the interval $[t, t+s]$ and that exactly one reaction occurs in $[t+s, t+s+ds]$. $f(A(t), s)$ is the PDF of τ, the random variable giving the next reaction time, and so by computing f, we can identify the distribution of τ.

Suppose there are n molecules of A at time t, then

$$\begin{aligned} f(A(t),s)ds &= \mathbb{P}\left[A(t+s+ds) = n-1,\, A(t+s) = n \mid A(t) = n\right] \\ &= \underbrace{\mathbb{P}\left[A(t+s+ds) = n-1 \mid A(t+s) = n\right]}_{\text{one reaction in } [t+s,t+s+ds]} \\ &\quad \times \underbrace{\mathbb{P}\left[A(t+s) = n \mid A(t) = n\right]}_{\text{no reactions in } [t+s]}. \end{aligned}$$

Letting $g(A(t),s)$ denote the probability that no reaction occurs in the interval $[t,t+s]$, we can write the formula above more succinctly as

$$\begin{aligned} f(A(t),s)ds &= g(A(t),s)A(t+s)kds \\ &= g(A(t),s)A(t)kds, \end{aligned}$$

where the second equality holds because we are conditioning on no reactions occurring in $[t,t+s]$.

For any $\sigma > 0$, using the definition of $g(A(t),s)$ yields that

$$\begin{aligned} g(A(t),\sigma+d\sigma) &= g(A(t),\sigma)\left[1 - A(t+\sigma)kd\sigma\right] \\ &= g(A(t),\sigma)\left[1 - A(t)kd\sigma\right] \end{aligned}$$

where the last equality follows because $A(t) = A(t+\sigma)$ if there is no reaction in $[t,t+\sigma]$. Rearranging the equality above and letting $d\sigma \downarrow 0$ then gives that g obeys the ODE

$$\frac{d}{d\sigma}g(A(t),\sigma) = -A(t)kg(A(t),\sigma).$$

Note that the derivative is with respect to σ here, not t! Solving this ODE gives

$$g(A(t),\sigma) = e^{-A(t)k\sigma}$$

since $g(A(t),0) = \mathbb{P}[\text{no reaction at time } t] = 1$. Plugging this back into the previous formula and cancelling ds on both sides shows that

$$f(A(t),s) = A(t)ke^{-A(t)ks}.$$

Recall that the PDF of an exponential random variable with rate parameter λ is given by

$$f(x) = \lambda e^{-\lambda x}, \quad x \geq 0,$$

and hence we can conclude that $\tau \sim \text{Exp}[kA(t)]$.

> **Note**
>
> In general, the waiting times (time intervals between reactions/jumps) of a Markov jump process are exponentially distributed.

We want to simulate the degradation process and so we now need to simulate exponentially distributed random numbers in order to find the next reaction time at each step. Previously, we assumed that our computer could generate random numbers that are uniformly distributed on $[0, 1]$, but fortunately, there is a simple way to generate an exponential random variable from a uniform one.

Suppose $U \sim U([0, 1])$ and F denotes the CDF of a continuous random variable. If F is invertible, then $V = F^{-1}(U)$ is distributed according to F. To see this, just calculate the CDF of V as follows:

$$\mathbb{P}[V \leq x] = \mathbb{P}[F^{-1}(U) \leq x] = \mathbb{P}[U \leq F(x)] = F(x).$$

Can you justify all of the equalities in the calculation above? This procedure is called the **inverse transform method** and allows us to sample from any continuous distribution with an invertible CDF given uniform random numbers.

The CDF of the exponential distribution with rate $\lambda > 0$ is given by

$$F(x) = \begin{cases} 0, & x < 0, \\ 1 - e^{-\lambda x}, & x \geq 0, \end{cases}$$

and hence

$$F^{-1}(x) = \begin{cases} -\frac{1}{\lambda}\log(1-x) = \frac{1}{\lambda}\log\left(\frac{1}{1-x}\right), & x \in (0, 1), \\ 0, & \text{else.} \end{cases}$$

Thus, if $U \sim U([0, 1])$, then $\log(1/(1 - U))/\lambda \sim \text{Exp}(\lambda)$. Furthermore, $1 - U \sim U([0, 1])$ and so $\log(1/U)/\lambda \sim \text{Exp}(\lambda)$ as well; this is the formula that we will use to generate exponentially distributed random numbers in the algorithms that follow.

Exercise 1.2

1. Calculate the inverse CDF for the exponential distribution and verify the formula above!

2. Apply the inverse transform method to the Pareto distribution, which has PDF given by

$$f(x) = \begin{cases} \frac{\alpha\,\beta^{\alpha}}{x^{\alpha+1}}, & x \geq \beta 0, \\ 0, & \text{else.} \end{cases}$$

We're now ready to introduce our **more efficient SSA**:

At time $t = 0$, set $A(0) = n_0$, then:

1. Generate a random number $r \sim U([0, 1])$.
2. Compute the next reaction time by calculating

$$\tau = \frac{1}{A(t)k} \log(1/r).$$

3. Set $t = t + \tau$, set $A(t + \tau) = A(t) - 1$ and go back to step 1.

Figure 1.2 below shows the results of some simulations of the degradation process using the more efficient SSA. There is a notable speed-up compared to the naive SSA when plotting large numbers of realisations. We take advantage of this to illustrate further differences between the deterministic model and stochastic one in the right panel of Figure 1.2; this panel shows a comparison between the estimated PMF of the process at $t = 10$ and the deterministic mean at the same time. We used 200 paths of the process for this plot but more are probably required to fully capture the PMF of even this simple process, further highlighting the need for efficient SSAs!

Exercise 1.3

Open the course Github page and try running the MATLAB script:
`CH1_more_efficient_SSA_degradation.m`

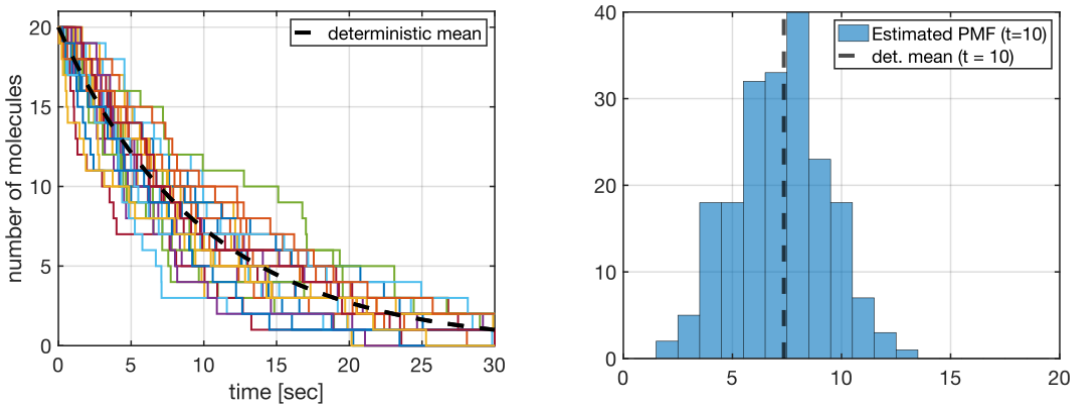

Figure 1.2: Left: Some paths of the degradation process simulated using the more efficient SSA. Right: Estimated PMF of the degradation process based on 200 realisations at time $t = 10$. Parameters: $n_0 = 20$, $k = 0.1$.

1.1.2 The Chemical Master Equation

We now have a couple of ways to simulate the degradation process numerically, but we also need some mathematical tools to analyse the process analytically. In modelling, there is very often an interplay between theory and numerics; we typically want to prove as much as possible about a model rigorously and use numerics to generate conjectures and address questions that are not possible to answer with theoretical tools.

To this end, we introduce the **chemical master equation** for the degradation process by letting

$$P_n(t) = \mathbb{P}[A(t) = n], \quad t \geq 0, \quad n \in \mathbb{N}.$$

For a fixed value of t, $P_n(t)$ is simply the probability mass function of the random variable $A(t)$, so we have a collection of PMFs indexed by (continuous) time. Our goal is to deduce an evolution equation for how $P_n(t)$ changes over time. As always, we assume there is an infinitesimally small quantity dt, so that $\mathcal{O}(dt^2)$ terms are negligible, and begin by considering the event $\{A(t + dt) = n\}$. There are two ways we can arrive at this event, either

- $A(t) = n$ and there were no reactions in $[t, t + dt)$, **or**
- $A(t) = n + 1$ and there was one reaction in $[t, t + dt)$.

Hence

$$P_n(t + dt) = \underbrace{P_n(t)(1 - kndt)}_{\text{no reactions in } [t,t+dt)} + \underbrace{P_{n+1}(t)(k(n + 1)dt)}_{\text{one reaction in } [t,t+dt)}.$$

Rearrangement yields

$$\frac{P_n(t+dt) - P_n(t)}{dt} = k(n+1)P_{n+1}(t) - knP_n(t),$$

and letting $dt \downarrow 0$ thus gives the *system* of ODEs

$$\frac{d}{dt}P_n(t) = k(n+1)P_{n+1}(t) - knP_n(t), \quad t \geq 0, \quad n \geq 0.$$

> **Note**
>
> In the theory of Markov jump processes, this equation is called the **Kolmogorov Forward Equation** (sometimes just the Kolmogorov equation) for the process. You might see this terminology used in other texts or papers.

The function $P_n(t)$ contains all of the information we could ever want to know about the stochastic process, but to obtain this information we need to solve the system of equations above! In general, this is far from a trivial task for complex processes with many chemical species and reactions (and that is before we even consider adding spatial extent to the system!). However, we can solve this system directly for the degradation process.

We can simplify the system immediately by thinking about the initial conditions of the process. Since $A(0) = n_0$, we have

$$P_{n_0}(0) = 1, \quad P_n(0) = 0 \quad \forall n \neq n_0.$$

Moreover, the process $A(t)$ is non-increasing because we can never produce any new molecules of A. Hence

$$P_n(t) = 0, \quad \forall n > n_0.$$

These two facts already provide a significant simplification because we now have a finite system of ODEs (after starting with a countable number of ODEs to solve) and we have the initial condition at $t = 0$ for each value of n from above. Our strategy from here is to start with the equation for P_{n_0} and try to solve the system iteratively (until we hopefully see a pattern). The equation for P_{n_0} reads:

$$\frac{d}{dt}P_{n_0}(t) = k(n_0+1)P_{n_0+1}(t) - kn_0P_{n_0}(t), \quad P_{n_0}(0) = 1.$$

But, as we just noted, $P_{n_0+1}(t) = 0$ so we can immediately solve to find that

$$P_{n_0}(t) = P_{n_0}(0)e^{-kn_0t} = e^{-kn_0t}.$$

Next tackle the equation for $P_{n_0-1}(t)$, which now reads:

$$\frac{d}{dt}P_{n_0-1}(t) = kn_0e^{-kn_0t} - k(n_0-1)P_{n_0-1}(t).$$

Thankfully, this equation is still linear in P_{n_0-1}, but it is inhomogeneous so we need to use the variation of constants formula to solve it. We thus obtain

$$\begin{aligned} P_{n_0-1}(t) &= P_{n_0-1}(0)e^{-k(n_0-1)t} + e^{-k(n_0-1)t}\int_0^t e^{k(n_0-1)s}kn_0e^{-kn_0s}\,ds \\ &= n_0e^{-k(n_0-1)t}\left(1-e^{-kt}\right). \end{aligned}$$

Now we begin to see the pattern: We can take the solution for P_{n_0-1}, plug it into the equation for P_{n_0-2} and apply the variation of constants formula to then solve that linear inhomogeneous ODE, and carry on this procedure until we reach P_0. In fact, we can show by induction that the general formula for P_n is given by

$$P_n(t) = \begin{cases} 0, & n > n_0, \quad t \geq 0, \\ \binom{n_0}{n}e^{-knt}(1-e^{-kt})^{n_0-n}, & 0 \leq n \leq n_0, \quad t \geq 0. \end{cases}$$

We could have guessed the case $n > n_0$: there is no chance that we can have more than n_0 molecules of A! The second part of the formula is less obvious but should look familiar: at each fixed time t, this is the PMF of a Binomial random variable with parameters n_0 (number of trials) and e^{-kt} (probability of success per trial). Succinctly, we may write

$$P_n(t) \sim \text{Binomial}(n_0,\, e^{-kt}).$$

Given the solution to the **chemical master equation**, we can answer virtually any question regarding the process. In practice, we may want to know about its average behaviour, its fluctuations (which are often characterised by the variance), or the probability of it hitting a certain value by a given time. We can compute the mean of the process, $M(t)$, directly from the formula above as follows:

$$
\begin{aligned}
M(t) := \mathbb{E}[A(t)] &= \sum_{n=-\infty}^{\infty} nP_n(t) = \sum_{n=0}^{n_0} nP_n(t) \\
&= \sum_{n=0}^{n_0} n\binom{n_0}{n} e^{-knt}(1-e^{-kt})^{n_0-n} \\
&= n_0 e^{-kt} \sum_{m=0}^{n_0-1} \binom{n_0-1}{m} \left(e^{-kt}\right)^m (1-e^{-kt})^{(n_0-1)-m} \\
&= n_0 e^{-kt}.
\end{aligned}
$$

This exactly matches the deterministic model for the mean behaviour that we obtained by solving the ODE obtained via the law of mass action. However, this is not true in general and it is also worth noting that the deterministic ODE can't tell us more detailed information about the paths, like their fluctuations or hitting times. For more complicated examples (coming in later chapters), we will see more dramatic disagreement between the mean behaviour predicted by the law of mass action and the dynamics of the underlying stochastic process.

Exercise 1.4

1. Complete the inductive step from $n_0 - k$ to $n_0 - (k+1)$ to obtain the formula above.

2. Use the more efficient SSA code to show that with enough realisations we obtain an approximate Binomial distribution, in agreement with the formula above.

1.2 Stochastic Simulation of Production & Degradation

1.2.1 The production/degradation process

We next consider a variation on the degradation process where we now add production of species A to the set of possible reactions. We are still assuming that there is some container of volume ν that holds the molecules of chemical A and that we can neglect the spatial extent of the experiment. The new reaction process can be written as:

$$A \xrightarrow{k_1} \emptyset, \quad \emptyset \xrightarrow{k_2} A.$$

Once more, the $\emptyset$ simply denotes some chemical species not of interest, rather than denoting that A is being produced out of thin air! Given that the volume of the container is ν, we will assume that production of A is proportional to $k_2\nu$ so that the input scales appropriately with the system size. It is worth remarking that the units of k_1 will be sec^{-1} and those of k_2 are $\text{sec}^{-1}m^{-3}$ because production was chosen to scale with volume. This all leads to the following characterization of the transition rates in a (small) interval $[t, t+dt)$:

- No reactions occur:
  $\mathbb{P}[A(t+dt) = A(t)] \approx 1 - A(t)k_1 dt - k_2 \nu dt + \mathcal{O}(dt^2)$
- One molecule of A is produced:
  $\mathbb{P}[A(t+dt) = A(t) + 1] \approx k_2 \nu dt + \mathcal{O}(dt^2)$
- One molecule of A is degraded:
  $\mathbb{P}[A(t+dt) = A(t) - 1] \approx A(t)k_1 dt + \mathcal{O}(dt^2)$
- More than one reaction occurs:
  $\mathbb{P}[\text{multiple reactions in } [t, t+dt)] \approx \mathcal{O}(dt^2)$.

Based on the transition rates outlined above, we can define the **propensity function** of the process

$$\alpha(t) := A(t)k_1 + k_2\nu.$$

This definition is motivated by the fact that the probability of a reaction occurring in $[t, t+dt)$ is given by $\alpha(t)dt$ ($+\mathcal{O}(dt^2)$ terms). We might also call the propensity function the **total reaction rate** of the process.

Next we outline a version of our **efficient SSA** that we claim simulates the production/degradation process described above:

At time $t = 0$, set $A(0) = n_0$, then:

1. Generate random numbers r_1, $r_2 \sim U([0, 1])$.
2. Compute the propensity function for the process, i.e.
$$\alpha(t) = k_1 A(t) + k_2 \nu.$$
3. Compute the next reaction time by calculating
$$\tau = \frac{1}{\alpha(t)} \log(1/r_1).$$
4. Compute the number of A molecules at time $t = t + \tau$:
$$A(t+\tau) = \begin{cases} A(t) + 1, & \text{if } r_2 < k_2\nu/\alpha(t), \\ A(t) - 1, & \text{if } r_2 \geq k_2\nu/\alpha(t), \end{cases}$$
and set $t = t + \tau$.
Go back to step 1.

Does this scheme correctly simulate the production/degradation process?

Step 3 in the scheme generates an exponentially distributed random number with rate parameter $\alpha(t)$ as the next reaction time. We could consider a single reaction, as in the last section, by letting **production or degradation** be one reaction, the reaction rate of this simplified process would be:

$$\begin{aligned}\mathbb{P}[\text{one reaction in } [t,t+dt)] &= \mathbb{P}[\text{production in } [t,t+dt) \text{ or degradation in } [t,t+dt)] \\ &= \mathbb{P}[\text{production, no degradation}] + \mathbb{P}[\text{no production, degradation}] \\ &= (k_1 A(t)dt)(1-k_2\nu dt) + (k_2\nu dt)(1-k_1 A(t)dt) \\ &= (k_1 A(t) + k_2\nu)dt = \alpha(t)\,dt.\end{aligned}$$

In other words, the single reaction or reaction/no-reaction process has reaction rate $\alpha(t)$ and we showed in the previous section that the waiting time before its next reaction time is exponential distributed with rate parameter $\alpha(t)$, thus justifying step 3.

Step 4 tells us, conditional on a reaction having taken place in $[t,t+dt)$, whether a production or degradation reaction took place. Since $r_2 \sim U([0,1])$, we can calculate the (conditional) probabilities of production or degradation:

$$\begin{aligned}\mathbb{P}[\text{production}] &= \mathbb{P}\left[r_2 < \frac{k_2\nu}{k_1 A(t) + k_2\nu}\right] = \frac{k_2\nu}{k_1 A(t) + k_2\nu}, \\ \mathbb{P}[\text{degradation}] &= \mathbb{P}\left[r_2 \geq \frac{k_2\nu}{k_1 A(t) + k_2\nu}\right] \\ &= 1 - \frac{k_2\nu}{k_1 A(t) + k_2\nu} = \frac{k_1 A(t)}{k_1 A(t) + k_2\nu}.\end{aligned}$$

These calculations were facilitated by the fact that, given positive rate constants, volume and a non-zero number of molecules, we have

$$\frac{k_2\nu}{k_1 A(t) + k_2\nu} \in (0,1), \quad \frac{k_1 A(t)}{k_1 A(t) + k_2\nu} \in (0,1).$$

We conclude from these computations that the relative probabilities (and they are true probabilities being in $(0,1)$) of production and degradation are proportional to their rates, which at least seems a sensible state of affairs!

A more rigorous justification of steps 3 and 4 is to note that **conditional on the current value of the process**, $A(t)$, the production and degradation reactions are independent single-reaction processes (until the next reaction occurs). Therefore the waiting time until the next

degradation reaction is exponentially distributed with rate parameter $k_1 A(t)$ and the waiting time until the next production reaction is exponentially distributed with rate parameter $k_2\nu$. Hence the waiting time until the next reaction time for the production/degradation system is simply the minimum of these two waiting times. Once the next reaction takes place, $A(t)$ changes, these two (independent) clocks are reset and we wait again to see which takes place first.

Suppose $E_D \sim \text{Exp}(k_1 A(t))$ and $E_P \sim \text{Exp}(k_2\nu)$ so that $\tau = \min(E_D, E_P)$ is the waiting time until the next reaction in the production/degradation process. Crucially, E_D and E_P are independent here because $A(t)$ is fixed. What is the distribution of τ? Compute the CDF of τ as follows:

$$\begin{aligned}\mathbb{P}[\tau > x] &= \mathbb{P}[\min(E_D, E_P) > x] = \mathbb{P}[E_P > x,\, E_D > x] \\ &= \mathbb{P}[E_D > x]\,\mathbb{P}[E_P > x] \\ &= e^{-k_1 A(t) x} e^{-k_2 \nu x} = e^{-(k_1 A(t) + k_2\nu)x} \\ &= e^{-\alpha(t) x}, \quad x > 0.\end{aligned}$$

Thus $\tau \sim \text{Exp}(\alpha(t))$, as claimed before. Moreover, we can directly compute the probability that a production reaction occurs next by computing $\mathbb{P}[\text{production}] = \mathbb{P}[E_P < E_D]$ since this is just the probability that one exponential random variable is less than another (and they are both independent!). Carrying out this calculation shows that

$$\mathbb{P}[\text{production}] = \mathbb{P}[E_P < E_D] = \frac{k_2\nu}{k_1 A(t) + k_2\nu},$$

in agreement with the formulae above.

Exercise 1.5

Directly compute $\mathbb{P}[E_P < E_D]$ to show that the formula above holds.

Figure 1.3 shows some sample paths of the production/degradation process generated using the more efficient SSA. All sample paths begin with $A(0) = 0$ but, for this parameter choice, production outpaces degradation initially before the process levels off somewhat around 10 molecules (on average). The deterministic mean in this plot is generated by using the law of mass action to derive a deterministic model for the average behaviour of the system.

Exercise 1.6

1. Open the course Github page and try running the MATLAB script: `CH1_production_degradation.m`
2. Use the law of mass action to derive a deterministic model for the produc-

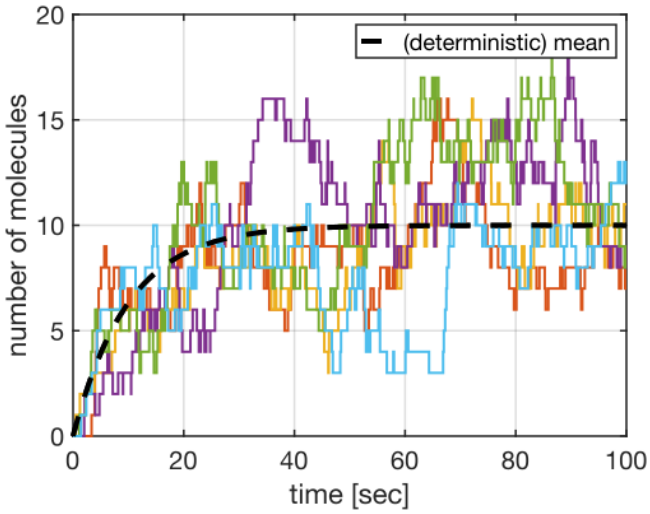

Figure 1.3: Sample paths of the production/degradation process generating using the more efficient SSA and compared to the deterministic mean; the deterministic mean was obtained by solving the deterministic model generated by the law of mass action. Parameters: $k_1 = 0.1$, $k_2 = 1$, $\nu = 1$, $A(0) = n_0 = 0$.

tion/degradation process and verify the formula for the deterministic mean in the code above.

1.2.2 Deriving the chemical master equation

To derive the chemical master equation for the production/degradation process we again consider the question: How can the number of molecules change between time t and time $t+dt$? Suppose we have n molecules at time $t+dt$, i.e. $A(t) = n$, then we either had $n+1$ molecules at time t and a degradation reaction took place in the interval $[t, t+dt)$, **or** we had $n-1$ molecules and a production took place in $[t, t+dt)$, **or** we had n molecules at time t and nothing happened in $[t, t+dt)$! Letting $P_n(t) = \mathbb{P}[A(t) = n]$, we can express this probabilistically as follows:

$$P_n(t+dt) = P_n(t)\,[1 - k_1 n dt - k_2 \nu dt] + P_{n+1}(t)\,[k_1(n+1)dt] + P_{n-1}(t)\,[k_2 \nu dt]\,.$$

Rearranging and letting $dt \downarrow 0$ thus yields

$$\frac{d}{dt}P_n(t) = k_1(n+1)P_{n+1}(t) - k_1 n P_n(t) + k_2 \nu P_{n-1}(t) - k_2 \nu P_n(t), \quad \text{for all } n \geq 0,$$

with the convention that $P_n(t) \equiv 0$ for all $n < 0$.

Unlike the pure degradation process, there is no upper bound on the number of molecules in the production/degradation process and hence we can't reduce this countably-infinite system of ODEs to a finite system like we could previously. In practice, we would hope to show (or somehow know by other means!) that $P_n(t)$ tends to zero as $n \to \infty$; this would justify truncating the system at some large but finite value of n in order to solve it numerically.

Even for this relatively simple process, the chemical master equations are challenging to solve analytically. Instead, we will compute the mean and variance processes, i.e.

$$M(t) = \sum_{n=0}^{\infty} nP_n(t), \quad V(t) = \sum_{n=0}^{\infty} (n - M(t))^2 P_n(t),$$

as these are typically more tractable and still offer considerable insight into the dynamics of the process.

To derive an evolution equation for the mean, multiply the master equation by n and sum over n:

$$\begin{aligned}
\frac{d}{dt}\sum_{n=0}^{\infty} nP_n(t) &= k_1 \sum_{n=0}^{\infty} n(n+1)P_{n+1}(t) - k_1 \sum_{n=0}^{\infty} n^2 P_n(t) \\
&\quad + k_2\nu \sum_{n=0}^{\infty} nP_{n-1}(t) - k_2\nu \sum_{n=0}^{\infty} nP_n(t) \\
&= k_1 \sum_{m=0}^{\infty} m(m-1)P_m(t) - k_1 \sum_{n=0}^{\infty} n^2 P_n(t) \\
&\quad + k_2\nu \sum_{m=0}^{\infty} (m+1)P_m(t) - k_2\nu \sum_{n=0}^{\infty} nP_n(t) \\
&= -k_1 \sum_{n=0}^{\infty} nP_n(t) + k_2\nu \sum_{n=0}^{\infty} P_n(t) \\
\iff \frac{d}{dt}M(t) &= -k_1 M(t) + k_2\nu,
\end{aligned}$$

where the final equality used the fact that $\sum_{n=0}^{\infty} P_n(t) = 1$.

> **Note**
>
> In the preceding calculation we freely exchanged the limiting operations of differentiation

and infinite summation, i.e. we implicitly claimed that

$$\frac{d}{dt}\sum_{n=0}^{\infty} nP_n(t) = \sum_{n=0}^{\infty} n\frac{d}{dt}P_n(t).$$

Changing the order of limiting operations can, and often does, change the result of a calculation. For example, consider the function $f(n,m) = n/(n+m)$ and let both n and m tend to infinity, order matters here! However, we can interchange limits without worrying about changing the result when the conditions of **uniform convergence** are satisfied. You can assume we have uniform convergence throughout this course, unless explicitly noted otherwise.

It is straightforward to then solve this linear inhomogeneous ODE for $M(t)$ to show that

$$M(t) = M(0)e^{-k_1 t} + \frac{k_2\nu}{k_1}\left(1 - e^{-k_1 t}\right), \quad t \geq 0.$$

Similarly, but with slightly more pain along the way, we can deduce an evolution equation for $V(t)$. This begins by simplifying the definition of $V(t)$ first:

$$\begin{aligned}
V(t) &= \sum_{n=0}^{\infty} (n - M(t))^2 P_n(t) = \sum_{n=0}^{\infty} (n^2 - 2nM(t) + M(t)^2)P_n(t) \\
&= \sum_{n=0}^{\infty} n^2 P_n(t) - 2M(t)\sum_{n=0}^{\infty} nP_n(t) + M(t)^2 \sum_{n=0}^{\infty} P_n(t) \\
&= \sum_{n=0}^{\infty} n^2 P_n(t) - 2M(t)^2 + M(t)^2 \\
&= -M(t)^2 + \sum_{n=0}^{\infty} n^2 P_n(t).
\end{aligned}$$

We need to write the sum involving n^2 in terms of $M(t)$ and $V(t)$ to obtain a closed system so go back to the master equation, multiply across by n^2 and sum over n:

$$\begin{aligned}
\frac{d}{dt}\sum_{n=0}^{\infty} n^2 P_n(t) &= k_1 \sum_{n=0}^{\infty} n^2(n+1)P_{n+1}(t) - k_1 \sum_{n=0}^{\infty} n^3 P_n(t) \\
&\quad + k_2\nu \sum_{n=0}^{\infty} n^2 P_{n-1}(t) - k_2\nu \sum_{n=0}^{\infty} n^2 P_n(t) \\
&= k_1 \sum_{n=0}^{\infty} (-2n^2 + n)P_n(t) + k_2\nu \sum_{n=0}^{\infty} (2n+1)P_n(t)
\end{aligned}$$

Thus,

$$\frac{d}{dt}V(t) + 2M(t)\frac{d}{dt}M(t) = -2k_1(V(t) + M(t)^2) + k_1M(t) + 2k_2\nu M(t) + k_2\nu,$$

where the last implication requires us to insert the simplified $V(t)$ to replace the sums with n^2 terms. Tidying this up leaves us with the following evolution equation for $V(t)$:

$$\frac{d}{dt}V(t) = -2k_1V(t) + k_1M(t) + k_2\nu, \quad t \geq 0.$$

Equation above is another linear inhomogeneous ODE and can be solved given that we know $M(t)$. Instead, we focus on an approach that tends to be tractable for more complex systems, namely, to focus on the large time or asymptotic behaviour of the process. To this end, define the quantities M_s and V_s by

$$M_s := \lim_{t\to\infty} M(t), \quad V_s := \lim_{t\to\infty} V(t).$$

Assuming that these limiting quantities are well-defined for this system (which is far from a given in general!), we can compute them by solving the steady-state versions of the evolution equations (setting the time derivatives to zero):

$$0 = -k_1M_s + k_2\nu, \quad 0 = -2k_1V_s + k_1M_s + k_2\nu.$$

Thus

$$M_s = V_s = \frac{k_2\nu}{k_1}.$$

This already gives us good information on the average behaviour of the system, and fluctuations around that average, for large times but we can actually take our asymptotic analysis a step further. To do so, define the **stationary distribution** ϕ of the production/degradation process by:

$$\phi(n) := \lim_{t\to\infty} P_n(t), \quad n \geq 0.$$

If the limit above is well-defined for each $n \geq 0$, then we can compute the stationary distribution by solving the steady-state version of the chemical master equation, i.e.

$$0 = k_1(n+1)\phi(n+1) - k_1n\phi(n) + k_2\nu\phi(n-1) - k_2\nu\phi(n), \quad n \geq 0,$$

where $\phi(n) \equiv 0$ for all $n < 0$. We can solve this system recursively by starting at $n = 0$ and trying to then guess the general form of the solution. For $n = 0$, we have

$$0 = k_1\phi(1) - k_2\nu\phi(0) \implies \phi(1) = \frac{k_2\nu}{k_1}\phi(0).$$

Since ϕ is a PMF, we have the additional normalisation constraint that $\sum_{n=0}^{\infty}\phi(n) = 1$ and this will allow us to determine $\phi(0)$ later if we can find the general formula for ϕ up to a multiplicative constant. It can be shown by induction that

$$\phi(n) = \frac{1}{n!}\left(\frac{k_2\nu}{k_1}\right)^n e^{-k_2\nu/k_1}, \quad n \geq 0,$$

and hence, for large times, the production/degradation process is approximately Poisson distributed with parameter $k_2\nu/k_1$.

Figure 1.4 shows a comparison of the stationary distribution with the estimated PMF of the process at different times points. In all simulations, $A(0) = 0$ and we can see how the bias of the initial condition begins to disappear gradually as we take larger and larger time intervals. By $t = 100$, the estimated PMF is already in very close agreement with the asymptotic behaviour predicted by the stationary distribution.

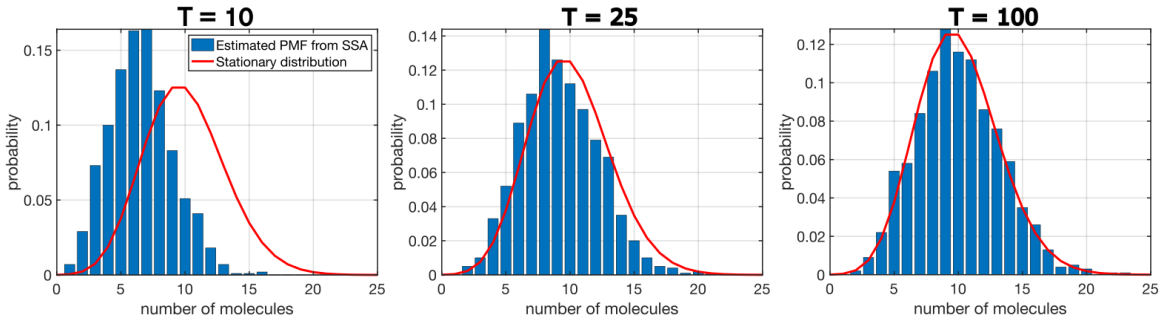

Figure 1.4: Simulations of the production/degradation process starting with $A(0) = 0$ with the PMF estimated at different times and compared to the stationary distribution. Parameters: $k_1 = 0.1$, $k_2 = 1$, $\nu = 1$, $A(0) = n_0 = 0$.

Exercise 1.7

Open the course Github page and try playing with the MATLAB script:
`CH1_production_degradation_CME.m`
Try varying the initial conditions and the time interval parameter T to observe how the mean and skewness of the estimated PMF vary.

1.3 Higher Order Chemical Reactions

Thus far we have only dealt with chemical reaction processes in which reactions depend linearly on the number of molecules of a given species. In reality, most chemical reactions of interest involve two or more molecules interacting. For example, a simple **second order** chemical reaction would be the reaction process:

$$A + B \xrightarrow{k} C,$$

where a molecule of A and a molecule of B react at rate k to produce a molecule of C. We once more neglect spatial extent here but clearly the A and B molecules would need to come together in space to cause such a reaction. We are assuming the molecules are in some container with volume ν and hence the reaction rates should naturally scale with volume. For example, if the volume is increased, then A and B molecules will take longer on average to bump into each other (assuming there is no attraction/repulsion between them and that both species are distributed uniformly randomly in the container). Thus we will assume

$$\mathbb{P}\left[\text{one reaction between an } A \text{ and a } B \text{ molecule in } [t, t+dt)\right] \approx \frac{k}{\nu} dt + \mathcal{O}(dt^2),$$

so that the reaction rates scale appropriately with volume. The total number of possible A-B pairs that could react together at time t is $A(t)B(t)$ and hence

$$\mathbb{P}\left[\text{one reaction in } [t, t+dt)\right] \approx \frac{k}{\nu} A(t)B(t)dt + \mathcal{O}(dt^2).$$

A slightly different type of second order chemical reaction is the process

$$A + A \xrightarrow{k} C,$$

where two molecules of A join together by ions or bonds to form a new chemical species C, a process called **dimerisation**. In this case, the reaction rate of the system will depend on the number of possible A-A pairs, which is given by

$$\binom{A(t)}{2} = \frac{A(t)(A(t)-1)}{2}.$$

Thus the propensity function for the dimerisation process above is given by

$$\alpha(t) = \begin{cases} \frac{kA(t)(A(t)-1)}{2\nu}, & A(t) \geq 2, \\ 0, & A(t) < 2. \end{cases}$$

Table 1.1 below lists some simple reaction processes of different orders, along with their propensity functions ($\alpha(t)$) and the units of their reaction rates (k).

Table 1.1: Examples of some simple low order chemical reactions and their corresponding propensity functions.

Reaction	order	$\alpha(t)$	units of k
$\emptyset \xrightarrow{k} A$	zero	$k\nu$	m^{-3} sec^{-1}
$A \xrightarrow{k} \emptyset$	first	$kA(t)$	sec^{-1}
$A + B \xrightarrow{k} \emptyset$	second	$kA(t)B(t)/\nu$	m^{-3} sec^{-1}
$A + B + C \xrightarrow{k} \emptyset$	third	$kA(t)B(t)C(t)/\nu^2$	m^{-6} sec^{-1}

Implicit in all of these processes is the assumption that the molecules are **well-mixed** spatially, meaning that a molecule of any species is equally likely to encounter a molecule of any other species. Later in the course we will consider biological processes where this well-mixedness assumption is very strongly violated, such as in chemotaxis in cell biology (in which cells are attracted to/repulsed by other cells) or prey that move to avoid predators in ecological models.

Exercise 1.8

Write down the propensity functions for the following reaction processes:
1. $2A + B \xrightarrow{k} \emptyset$
2. $A + 2B + C \xrightarrow{k} \emptyset$
3. $2A + 3B \xrightarrow{k} \emptyset$

1.4 Stochastic Simulation of Dimerisation

The chemical master equations for complex higher-order reaction processes are typically very hard to solve analytically. In this section we will use an example of a dimerisation process to introduce the probability generating function approach to solving the chemical master equations. Dimerisation in chemistry is the process where two identical or similar molecules, known as monomers, join together to form a single, larger molecule called a dimer – we will use it as the simplest example of a second order chemical reaction process.

1.4.1 Probability Generating Functions

Before introducing the probability generating function (PGF) of a stochastic process, we may define the PGF of a random variable as follows:

For a discrete random variable X taking values in the set $\{x_0, x_1, \dots\}$, its **probability generating function** G is the function

$$G : [-1, 1] \mapsto \mathbb{R}, \quad G(x) = \sum_{n=0}^{\infty} x^n \mathbb{P}[X = x_n].$$

The PGF of a random variable contains all of the information about the distribution of the random variable. In particular, it is straightforward to show from the formula above that

$$\mathbb{P}[X = x_n] = \frac{1}{n!} G^{(n)}(x)\Big|_{x=0},$$

where $G^{(n)}(x)$ denotes the nth derivative of G with respect to x. This gives us a very direct way to recover the PMF of a random variable once we know its PGF. Similarly, the PGF and the moments of a random variable are related by the formula:

$$\mathbb{E}\left[\frac{X!}{(X-k)!}\right] = G^{(k)}(x)\Big|_{x=1}, \quad k \geq 0.$$

The formula above refers to the kth factorial moment of the random variable but from this we can deduce formulae for the mean and variance, which are given by

$$\mathbb{E}[X] = \frac{d}{dx} G(x)\Big|_{x=1}, \quad \mathrm{Var}[X] = \left(\frac{d^2}{dx^2} G(x) + \frac{d}{dx} G(x) - \left(\frac{d}{dx} G(x)\right)^2\right)\Big|_{x=1}.$$

For example, if we consider a Poisson distributed random variable Y with parameter $\lambda > 0$, its PGF is given by

$$G_Y(x) = \sum_{n=0}^{\infty} x^n \frac{\lambda^n e^{-\lambda}}{n!} = e^{-\lambda} \sum_{n=0}^{\infty} \frac{(x\lambda)^n}{n!} = e^{\lambda(x-1)}.$$

We can thereby calculate the mean of Y as suggested above:

$$\mathbb{E}[Y] = \frac{d}{dx} G_Y(x)\Big|_{x=1} = \lambda e^{\lambda(x-1)}\Big|_{x=1} = \lambda.$$

1.4.2 Dimerisation Process Analysis

Consider the second-order stochastic reaction process given by

$$A + A \xrightarrow{k_1} \emptyset, \quad \emptyset \xrightarrow{k_2} A.$$

The propensity function of the process is thus given by

$$\alpha(t) = \frac{k_1}{\nu} A(t)(A(t) - 1) + k_2 \nu,$$

where ν denotes the system volume and we have absorbed the constant $1/2$ owing to the number of possible A pairs into the rate constant k_1. The more efficient SSA applied to the dimerisation process takes the following form:

At time $t = 0$, set $A(0) = n_0$, then:

1. Generate random numbers r_1, $r_2 \sim U([0, 1])$.
2. Compute the propensity function $\alpha(t)$.
3. Compute the next reaction time by calculating

   $$\tau = \frac{1}{\alpha(t)} \log(1/r_1),$$

   and set $t = t + \tau$.
4. Compute the number of A molecules at time $t = t + \tau$:

   $$A(t + \tau) = \begin{cases} A(t) + 1, & \text{if } r_2 < k_2 \nu / \alpha(t), \\ A(t) - 2, & \text{if } r_2 \geq k_2 \nu / \alpha(t), \end{cases}$$

   and go back to step 1.

We defer simulating the process for now and instead proceed directly to its analysis via the chemical master equations. Adopting our usual approach, we see that

$$\begin{aligned} P_n(t + dt) = P_n(t) \left[1 - \frac{k_1}{\nu} n(n-1)dt - k_2 \nu dt \right] \\ + \frac{k_1}{\nu}(n+1)(n+2) P_{n+2}(t)dt + k_2 \nu P_{n-1}(t)dt. \end{aligned}$$

Rearranging and letting $dt \downarrow 0$, we obtain the system of ODEs:

$$\frac{d}{dt}P_n(t) = \frac{k_1}{\nu}(n+1)(n+2)P_{n+2}(t) - \frac{k_1}{\nu}n(n-1)P_n(t) + k_2\nu P_{n-1}(t) - k_2\nu P_n(t),$$

with the convention that $P_n \equiv 0$ for $n < 0$.

Clearly, it will not be particularly fun or easy to solve this system! Worse still, it is not possible to write a closed system of equations for the mean and variance of this process via manipulating the chemical master equations (as we did before). Instead we proceed via the probability generating function of the process:

$$G : [-1, 1] \times (0, \infty) \mapsto \mathbb{R} : \quad G(x, t) = \sum_{n=0}^{\infty} x^n P_n(t).$$

As we can see, the PGF of a stochastic process is defined analogously to the PGF of a random variable, except now we have an extra argument to account for the time at which we are evaluating the process $A(t)$. Formulae for the PMF, mean and variance of $A(t)$ are exactly the same as those shown above. Therefore, if we can obtain the PGF of the process, it contains all of the same information as the PMF and is thus functionally equivalent to solving the chemical master equation!

To derive an evolution equation for the PGF of the process, multiply the master equation across by x^n and sum over n to show that:

$$\begin{aligned}\frac{\partial}{\partial t}\sum_{n=0}^{\infty} x^n P_n(t) = \frac{k_1}{\nu}\sum_{n=0}^{\infty} x^n(n+1)(n+2)P_{n+2}(t) - \frac{k_1}{\nu}\sum_{n=2}^{\infty} x^n n(n-1)P_n(t) \\ + k_2\nu\sum_{n=0}^{\infty} x^n P_{n-1}(t) - k_2\nu\sum_{n=0}^{\infty} x^n P_n(t).\end{aligned}$$

Next note the identity

$$\frac{\partial^2}{\partial x^2}G(x, t) = \sum_{n=2}^{\infty} n(n-1)x^{n-2}P_n(t),$$

which we will use to simplify the sums above. Changing the indices in the 1st and 3rd sums yields

$$\frac{\partial}{\partial t}G(x,t) = \frac{k_1}{\nu}\sum_{n=2}^{\infty} n(n-1)x^{n-2}P_n(t) - \frac{k_1}{\nu}x^2\sum_{n=2}^{\infty} n(n-1)x^{n-2}P_n(t)$$
$$+ k_2\nu x\sum_{n=0}^{\infty} x^n P_n(t) - k_2\nu\sum_{n=0}^{\infty} x^n P_n(t)$$
$$\implies \frac{\partial}{\partial t}G(x,t) = \frac{k_1}{\nu}(1-x^2)\frac{\partial^2}{\partial x^2}G(x,t) + k_2\nu(x-1)G(x,t).$$

Thus this is a second-order PDE for the PGF of the dimerisation process and if we can solve it, we will recover all of the information contained in the PMF of the process. To solve the PDE, we need to supply some initial and boundary conditions. Firstly,

$$G(x,0) = \sum_{n=0}^{\infty} x^n P_n(0), \quad x \in [-1,1]$$

so we can compute the initial condition for each x given the initial conditions of the process. We also need boundary conditions at $x = 1$ and $x = -1$. At $x = 1$, we have

$$G(1,t) = \sum_{n=0}^{\infty} P_n(t) = 1$$

and evaluating the PDE at $x = -1$ gives us

$$\frac{\partial}{\partial t}G(-1,t) = -2k_2\nu G(-1,t)$$

so that $G(-1,t) = G(-1,0)e^{-2k_2\nu t}$. These conditions plus the PDE thus constitute a well-posed initial-value problem that could be solved analytically or numerically, although this is not straightforward either!

Instead of trying to solve the PDE directly, we will analyse the asymptotic behaviour of the process via the **stationary probability generating function** G_s, which is given by:

$$G_s : [-1,1] \mapsto \mathbb{R}, \quad G_s(x) = \lim_{t\to\infty} G(x,t) = \sum_{n=0}^{\infty} x^n \phi(n),$$

where ϕ is the stationary distribution of the process. We can then obtain the asymptotic mean, $M_s := \lim_{t\to\infty} M(t)$, the asymptotic variance, $V_s := \lim_{t\to\infty} V(t)$, and the stationary distribution, ϕ, via the stationary PGF. Since the stationary PGF, G_s, does not depend on t, the PDE becomes

$$0 = \frac{k_1}{\nu}(1-x^2)\frac{d^2}{dx^2}G_s(x) + k_2\nu(x-1)G_s(x),$$

which simplifies to the second order ODE

$$G_s''(x) = \frac{k_2\nu^2}{k_1}\frac{1}{1+x}G_s(x), \quad x \in (-1,1).$$

The general solution of this ODE can be written as

$$G_s(x) = C_1\sqrt{1+x}I_1\left(2\sqrt{\frac{k_2\nu^2(1+x)}{k_1}}\right) + C_2\sqrt{1+x}K_1\left(2\sqrt{\frac{k_2\nu^2(1+x)}{k_1}}\right),$$

where C_1, C_2 are constants and the modified Bessel functions I_1 and K_1 are two independent solutions to the equation

$$z^2I_n''(z) + zI_n'(z) - (z^2+n^2)I_n(z) = 0.$$

Since we are now studying the stationary PGF, we use our old boundary conditions as $t \to \infty$, i.e.

$$\lim_{t\to\infty} G(1,t) = 1, \quad \lim_{t\to\infty} G(-1,t) = \lim_{t\to\infty} G(-1,0)e^{-2k_2\nu t} = 0.$$

The functions I_1 and K_1 obey

$$I_1(z) \sim \frac{z}{2} \text{ as } z \downarrow 0, \quad K_1(z) \sim \frac{1}{z} \text{ as } z \downarrow 0$$

and combining this with the boundary conditions allows us to deduce that $C_2 = 0$ and

$$C_1 = \left[\sqrt{2}I_1\left(2\sqrt{\frac{2k_2\nu^2}{k_1}}\right)\right]^{-1}.$$

We thus have an explicit formula for $G_s(x)$ in terms of the modified Bessel function I_1:

$$G_s(x) = \sqrt{1+x}\, I_1\left(2\sqrt{\frac{k_2\nu^2(1+x)}{k_1}}\right)\left[\sqrt{2}I_1\left(2\sqrt{\frac{2k_2\nu^2}{k_1}}\right)\right]^{-1}, \quad x \in [-1,1].$$

This formula can be evaluated numerically as the function I_1 is implemented in most mathematical software (such as in MATLAB or appropriate Python packages).

1.4.3 Simulation of the Dimerisation Process

In this section, we summarise and conclude our analysis of the dimerisation process. We have a number of tools and approaches we can use to understand the dynamics of this process, including:

- **direct simulation** (using the more efficient SSA),
- **analytic approaches**: chemical master equation, PGF, stationary PGF, etc.
- **deterministic modelling** via the law of mass action.

In order to compare all of these tools in a unified analysis, we need to apply the law of mass action to develop a deterministic model of the dimerisation process. This yields the **nonlinear** ODE:

$$\frac{d}{dt}a(t) = -2k_1 a(t)^2 + k_2,$$

where $a(t) = A(t)/\nu$ is the concentration of A molecules. We can solve this ODE numerically or solve the steady-state version to see that it predicts the long-run mean of $A(t)$ will be $\nu\sqrt{k_2/2k_1}$.

Figure 1.5 presents a synthesis of the three forms of analysis of the dimerisation process outlined above. In the left panel, we show sample paths of the process, together with the long-run mean obtained via analytic calculations (red dashed line) and the solution to the ODE model given by law of mass action (dashed black line). The law of mass action predicts a slightly lower mean value than the long-run mean, M_s, and these quantities differ in general for this model (which we will show in problem classes). We also plot $M_s \pm 2\sqrt{V_s}$ in the left-hand panel to show that these bounds give a good idea of the fluctuations of the process around its long-run mean. This works because Chebyshev's inequality tells us that any distribution with a finite mean and variance has approximately 75% of its mass within two standard deviations of either side of the mean (where the standard deviation here is $\sqrt{V_s}$).

The right-hand panel of Figure 1.5 shows the estimated PMF of the dimerisation process at $t = 100$ using $1,000$ sample paths of the process obtained via the more efficient SSA. The solid red line denoted by "analysis" in the legend is the stationary distribution of the process calculated from an analytic formula based on the calculations of the preceding section (see problem classes for more details!).

Exercise 1.9

Open the course Github page and try playing with the MATLAB script:
`CH1_dimerisation_process.m`
Try varying the parameters (time interval, initial conditions, reaction rates) to observe how the analysis presented above changes.

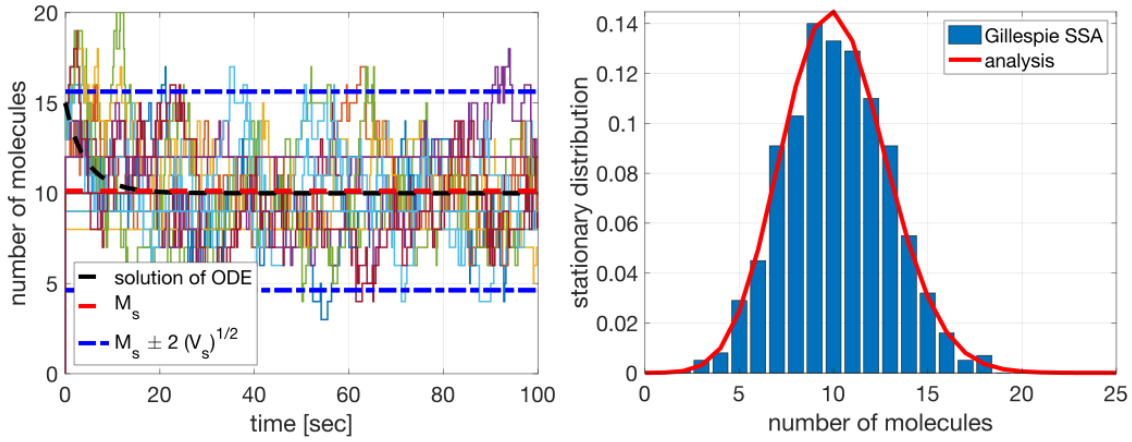

Figure 1.5: Left: Sample paths compared to the solution of the mass action ODE model and to the long run mean and variance. Right: Comparison between the stationary distributed obtained analytically and the estimated PMF from simulations. Parameters: $A(0) = 15$, $k_1 = 0.005$, $k_2 = 1$.

1.5 The Gillespie Stochastic Simulation Algorithm

1.5.1 Gillespie SSA Formulation

The "more efficient SSA" that we introduced in this chapter is a special case of the so-called **Gillespie algorithm** for simulating Markov jump processes. The Gillespie algorithm is a classic example of Stigler's law of eponymy, i.e. the "rule" that scientific discoveries are not named after the person who discovered them. Physicist Dan Gillespie popularised the algorithm that now bears his name in a 1977 paper concerning the simulation of chemical systems [@gillespie1977], but the algorithm was known some 35 years prior. The Gillespie algorithm was first implemented in 1950 by English mathematician and statistician David George Kendall on the Manchester Mark 1 computer.

In this section, we formulate the Gillespie algorithm for a general stochastic reaction process with $q \geq 1$ chemical reactions. Let $\alpha_i(t)$ denote the propensity function of the ith reaction. Note that we don't need to specify the number of chemical species involved; what matters is the number of distinct possible reactions.

At time $t = 0$, set the initial number of molecules in each species, then:

1. Generate random numbers r_1, $r_2 \sim U([0, 1])$.

2. Compute the propensity function of the system:

$$\alpha_0(t) = \sum_{i=1}^{q} \alpha_i(t).$$

3. Compute the next reaction time by calculating

$$\tau = \frac{1}{\alpha_0(t)} \log(1/r_1),$$

and set $t = t + \tau$.

4. Figure out which of the q reactions took place by finding the integer j such that:

$$r_2 \geq \frac{1}{\alpha_0(t)} \sum_{i=1}^{j-1} \alpha_i(t), \quad r_2 < \frac{1}{\alpha_0(t)} \sum_{i=1}^{j} \alpha_i(t).$$

Carry out reaction j, i.e. adjust the number of molecules in each species to account for reaction j occurring at time $t = t + \tau$. Go back to step 1.

The justification for this algorithm's correctness is a natural generalisation of the rationale we provided for the two species case. Step 3 is asserting that the next reaction time is exponentially distributed with parameter $\alpha_0(t)$, where $\alpha_0(t)$ is the sum over all of the q individual reaction rates. To see that this is correct, we can once again note that conditional on the current state of the system, the waiting time until the ith reaction occurs is $\tau_i \sim \text{Exp}(\alpha_i(t))$ and each pair of waiting times τ_i, τ_j are (conditionally) independent for $i \neq j$. Hence,

$$\tau = \min\left(\tau_1, \tau_2, \ldots, \tau_q\right),$$

where τ is the waiting time for the entire system. Using the mutual independence of the individual waiting times, we have

$$\mathbb{P}[\tau > x] = \mathbb{P}[\tau_1 > x, \ldots, \tau_q > x] = \prod_{i=1}^{q} \mathbb{P}[\tau_i > x] = e^{-x \sum_{i=1}^{q} \alpha_i(t)} = e^{-\alpha_0(t)x}.$$

Therefore, $\tau \sim \text{Exp}(\alpha_0(t))$, as claimed.

Step 4 ensures that we have

$$\mathbb{P}[\text{reaction } i \text{ occurs} \mid \text{some reaction occurs}] = \frac{\alpha_i(t)}{\alpha_0(t)},$$

which can be verified as the correct probability by computing $\mathbb{P}[\min(\tau_1, \ldots, \tau_q) = \tau_i]$. Intuitively, we can think of the condition in step 4 as breaking up the interval $[0, 1]$ into q subintervals

where the length of subinterval j is proportional to $\alpha_j(t)$. We then draw $r_2 \sim U([0,1])$ and see which subinterval r_2 falls in; if r_2 falls in subinterval j, then reaction j takes place.

There are many ways to improve upon the algorithm outlined above and this can be especially important for efficiently simulating processes that have a large number of reactions per unit time. Some improvements exploit special structure in the processes we are simulating, other improvements, such as the tau-leaping method [@cao2006efficient], are more broadly applicable. For example, we only need to update the propensity functions $\alpha_i(t)$ which changed the last time a reaction took place. This often means that most propensity functions do not need to be changed after every reaction and offers significant speed up for systems with large numbers of reactions.

1.5.2 Gillespie SSA Example

Consider the following stochastic reaction process involving dimerisation and production:

$$A + A \xrightarrow{k_1} \emptyset, \quad A + B \xrightarrow{k_2} \emptyset, \quad \emptyset \xrightarrow{k_3} A, \quad \emptyset \xrightarrow{k_4} B.$$

We will use this process to illustrate some of the difficulties that we typically encounter when considering more complex reactions with multiple species and multiple higher-order reactions; both of these features are naturally present in most practical problems of interest.

The propensity functions for the reaction process above are:

$$\begin{aligned} \alpha_1(t) &= \frac{k_1}{\nu} A(t)(A(t)-1), & \alpha_2(t) &= \frac{k_2}{\nu} A(t)B(t), \\ \alpha_3(t) &= k_3\nu, & \alpha_4(t) &= k_4\nu. \end{aligned}$$

To simulate the process, we need to compute the total propensity function

$$\alpha_0(t) = \alpha_1(t) + \alpha_2(t) + \alpha_3(t) + \alpha_4(t),$$

in order to compute the next reaction time. If τ is the next reaction time, then the step to update the number of A and B molecules according to the Gillespie algorithm will be:

$$A(t+\tau) = \begin{cases} A(t) - 2, & 0 \le r_2 < \alpha_1/\alpha_0, \\ A(t) - 1, & \alpha_1/\alpha_0 \le r_2 < (\alpha_1 + \alpha_2)/\alpha_0, \\ A(t) + 1, & (\alpha_1 + \alpha_2)/\alpha_0 \le r_2 < (\alpha_1 + \alpha_2 + \alpha_3)/\alpha_0, \\ A(t), & \text{else}, \end{cases}$$

and

$$B(t+\tau) = \begin{cases} B(t), & 0 \leq r_2 < \alpha_1/\alpha_0, \\ B(t)-1, & \alpha_1/\alpha_0 \leq r_2 < (\alpha_1+\alpha_2)/\alpha_0, \\ B(t), & (\alpha_1+\alpha_2)/\alpha_0 \leq r_2 < (\alpha_1+\alpha_2+\alpha_3)/\alpha_0, \\ B(t)+1, & \text{else.} \end{cases}$$

At this point we could simulate the process directly via our SSA but first we will consider our other typical lines of attack: the chemical master equations and the law of mass action.

As the number of reactions and species increases, so does the complexity of the chemical master equations. In particular, we now need to consider the joint density of the species. To this end, define

$$P_{n,m}(t) = \mathbb{P}[A(t) = n,\, B(t) = m].$$

Proceeding in the usual way, we may write

$$\begin{aligned} P_{n,m}(t+dt) = &\left[1 - \frac{k_1}{\nu}n(n-1)dt - \frac{k_2}{\nu}nmdt - k_3\nu dt - k_4\nu dt\right] P_{n,m} \\ &+ \frac{k_1}{\nu}(n+2)(n+1)P_{n+2,m}dt + \frac{k_2}{\nu}(n+1)(m+1)P_{n+1,m+1}dt \\ &+ k_3\nu P_{n-1,m}dt + k_4\nu P_{n,m-1}dt, \end{aligned}$$

where we have suppressed the t arguments on the right-hand side. Letting $dt \downarrow 0$, the chemical master equations are thus given by

$$\begin{aligned} \frac{d}{dt}P_{n,m}(t) = &\frac{k_1}{\nu}(n+2)(n+1)P_{n+2,m} + \frac{k_2}{\nu}(n+1)(m+1)P_{n+1,m+1} \\ &+ k_3\nu P_{n-1,m} + k_4\nu P_{n,m-1} - \frac{k_1}{\nu}n(n-1)P_{n,m} \\ &- \frac{k_2}{\nu}nmP_{n,m} - k_3\nu P_{n,m} - k_4\nu P_{n,m}, \end{aligned}$$

for $n, m \geq 0$, with the standard convention that $P_{n,m} \equiv 0$ if either $n < 0$ or $m < 0$.

Due to the presence of the second-order reactions, we cannot even write a closed system of evolution equations for the mean and variance via the chemical master equations! We could solve these equations, or their steady-state analogue, numerically if we wanted detailed information on the distribution or stationary distribution.

Finally, we can employ the law of mass action to write an approximate deterministic model for the mean behaviour of the process. Letting $a(t) = A(t)/\nu$ and $b(t) = B(t)/\nu$, we obtain the following pair of nonlinear ODEs:

$$\frac{d}{dt}a = -2k_1a^2 - k_2ab + k_3,$$
$$\frac{d}{dt}b = -k_2ab + k_4.$$

Figure 1.6 below shows some sample paths of the process with the corresponding solutions to the deterministic model overlaid for comparison (dashed black lines).

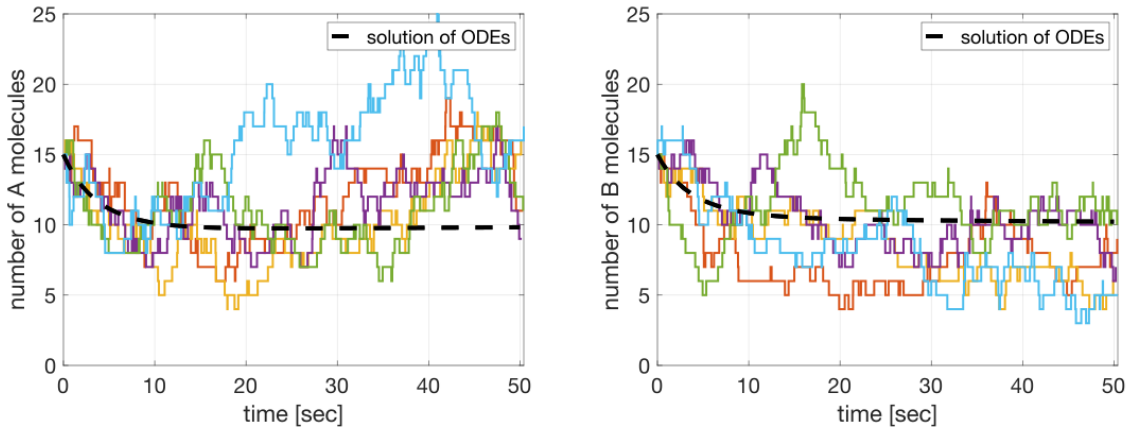

Figure 1.6: Sample paths and the deterministic mean predicted by the law of mass action. Parameters: $k_1 = 0.001$, $k_2 = 0.01$, $k_3 = 1.2$, $k_4 = 1$.

We can gain more information about the dynamics of the process by running more and longer simulations. This allows us to estimate the stationary distribution which characterizes the asymptotic behaviour of the process. Figure 1.7 shows the estimated stationary distribution in the left-hand panel, i.e. the joint PMF of $A(t)$ and $B(t)$ as $t \to \infty$. The right-hand panel of Figure 1.7 shows the marginal stationary distribution of $A(t)$, which can be obtained from the joint stationary distribution via the formula

$$\phi(n) = \sum_{m=0}^{\infty} \phi(n, m),$$

where $\phi(n) := \lim_{t\to\infty} \mathbb{P}[A(t) = n]$ and $\phi(n, m) := \lim_{t\to\infty} \mathbb{P}[A(t) = n,\ B(t) = m]$.

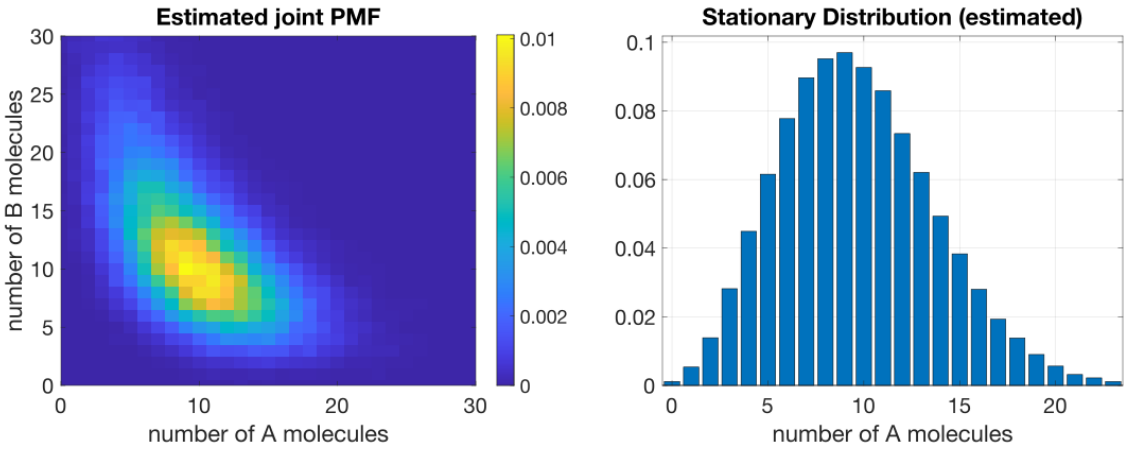

Figure 1.7: Estimated stationary distributions. Parameters: $k_1 = 0.001$, $k_2 = 0.01$, $k_3 = 1.2$, $k_4 = 1$.

Knowledge checklist

Key topics:

1. Chemical reaction processes (continuous time jump processes)
2. The law of mass action (approximation)
3. "Naive" & Gillespie SSAs; the inverse transform method
4. Chemical master equations (CMEs)
5. Probability generating functions
6. Stationary distributions

Key skills:

- Define and interpret chemical reaction processes using either:
  - The reaction notation (e.g. $A + B \to C$), or
  - the leading order probabilistic formulations (e.g. $\mathbb{P}[A(t + dt) = n] \approx f(A(t))dt + O(dt^2)$).
- Write and analyse mass action approximations of chemical reaction processes
- Write and analyse pseudocode for SSAs (Gillespie and the "naive" SSA)
  - Given a chemical reaction process, write the propensity functions for each reaction (especially for higher order reactions)
  - Comment on the efficiency and justify the correctness of SSAs
- Write and analyse chemical master equations, i.e.
  - computing stationary distributions from CMEs
  - calculating and solving moment equations
  - compute/analyse probability generating functions
- Describe the differences between and advantages/disadvantages of different approaches to studying a given chemical reaction process (i.e. mass action versus SSAs versus analytic approaches).

References