Spectral neural approximations for models of transcriptional dynamics

Gennady Gorin[1,*], Maria Carilli[1,2,*], Tara Chari[2], and Lior Pachter[2,3,**]

[1]Division of Chemistry and Chemical Engineering, California Institute of Technology, Pasadena, CA, 91125
[2]Division of Biology and Biological Engineering, California Institute of Technology, Pasadena, CA, 91125
[3]Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, CA, 91125
[*]These authors contributed equally to this work.
[**]Address correspondence to Lior Pachter (lpachter@caltech.edu)

November 30, 2023

Abstract

The advent of high-throughput transcriptomics provides an opportunity to advance mechanistic understanding of transcriptional processes and their connections to cellular function at an unprecedented, genome-wide scale. These transcriptional systems, which involve discrete, stochastic events, are naturally modeled using Chemical Master Equations (CMEs), which can be solved for probability distributions to fit biophysical rates that govern system dynamics. While CME models have been used as standards in fluorescence transcriptomics for decades to analyze single species RNA distributions, there are often no closed-form solutions to CMEs that model multiple species, such as nascent and mature RNA transcript counts. This has prevented the application of standard likelihood-based statistical methods for analyzing high-throughput, multi-species transcriptomic datasets using biophysical models. Inspired by recent work in machine learning to learn solutions to complex dynamical systems, we leverage neural networks and statistical understanding of system distributions to produce accurate approximations to a steady-state bivariate distribution for a model of the RNA life-cycle that includes nascent and mature molecules. The steady-state distribution to this simple model has no closed-form solution and requires intensive numerical solving techniques: our approach reduces likelihood evaluation time by several orders of magnitude. We demonstrate two approaches, where solutions are approximated by (1) learning the weights of kernel distributions with constrained parameters, or (2) learning both weights and scaling factors for parameters of kernel distributions. We show that our strategies, denoted by kernel weight regression (KWR) and parameter scaled kernel weight regression (psKWR), respectively, enable broad exploration of parameter space and can be used in existing likelihood frameworks to infer transcriptional burst sizes, RNA splicing rates, and mRNA degradation rates from experimental transcriptomic data.

Statement of significance

The life-cycles of RNA molecules are governed by a set of stochastic events that result in heterogeneous gene expression patterns in genetically identical cells, resulting in the vast diversity of cellular types, responses, and functions. While stochastic models have been used in the field of fluorescence transcriptomics to understand how cells exploit and regulate this inherent randomness, biophysical models have not been widely applied to high-throughput transcriptomic data, as solutions are often intractable and computationally impractical to scale. Our neural approximations of solutions to a two-species transcriptional system enable efficient inference of rates that drive the dynamics of gene expression, thus providing a scalable route to extracting mechanistic information from increasingly available multi-species single-cell transcriptomics data.

1 Introduction

The production, processing, and degradation of RNA molecules are governed by a set of stochastic events that lead to variable levels of RNA transcripts and proteins between genetically identical cells in identical conditions [1]. The inherent stochasticity and resulting variability in RNA transcript levels play an essential role in cellular function, development, and response, but must also be tightly regulated to produce deterministic functional outcomes at precise times. Extracting the rates that underlie distributions of RNA molecules is essential for understanding the intricate balance between stochastic gene expression and precise cellular responses required for life.

To explore the dynamics of transcriptional systems, specific biophysical models can be proposed that are explicitly parameterized by relevant biological rates, such as rates of transcriptional initiation, RNA splicing, or RNA degradation. These stochastic systems are naturally modeled by discrete-valued, continuous-time Markov chains (CTMCs). The evolution of a system's microstate probabilities under a CTMC model of biology is given by a chemical master equation (CME) [2]. For example, a CME tracks the changes in probability of there being a certain number of nascent (n) and mature (m) RNA molecules at time t given certain biophysical parameters θ, or $P(n, m, t; \theta)$. This can be used to obtain insights into transcriptional biophysics [3] by fitting θ most consistent with experimental distributions of molecule numbers [4].

Until recently, mechanistic modeling and insight into the underpinnings of transcriptional processes have been tied to intensive, temporal, fluorescence datasets which generally measure the production of one species of RNA molecules in one cell line [1,5,6]. Recent high-throughput transcriptomics methods, such as single-cell RNA sequencing (scRNA-seq), offer discrete molecular readouts at genome-wide scale across diverse cell populations [7], thus making possible the exploration of transcriptional parameters across distinct cell-types, tissues, and organisms. In addition to increased cell-type coverage, the diversity of molecular species captured, such as the simultaneous quantification of unspliced and spliced RNA molecules [8–10], proteins [11], and chromatin accessibility [12], has the potential to deepen our understanding of transcriptional system dynamics [13]. While single-cell RNA-seq data has been modeled using CME solutions to extract biophysical rates of transcriptional initation and burst sizes, mechanistic modeling has not been adopted in recent analyses of multi-species transcriptomic experiments [14]. One major challenge is that closed-form solutions to CMEs that model multi-species systems with several interacting species are rarely available [3,15], and current methods to solve them numerically for probabilities are prohibitive to the recursive calculations necessary for maximum likelihood inference of biophysical parameters.

The methods currently used to circumvent the lack of closed-form solutions to CMEs are computationally expensive and compromise accuracy. For example, CTMCs are often simulated using Gillespie's stochastic simulation algorithm [16], which generates statistically realistic molecular trajectories over possible state-spaces, but the number of simulations required for accurate parameter approximation is inefficient for parameter space exploration. Alternatively, application of matrix methods to a truncated state space, an approach called finite state projection (FSP) [17,18], has been applied for inference [4,6,13,19] but is extremely inefficient when the state space is much larger than the number of data points. In certain cases, CMEs can be solved using generating functions that transform infinite-dimensional CMEs to complex partial differential equations (PDEs). However, this approach requires performing an inverse Fast Fourier Transform over a finite, truncated array [20]. Both FSP and generating function methods require state-space truncation, which leads to error in calculated probabilities, with a trade-off between higher accuracy and increased evaluation time with increased grid size. Further, these methods are not amenable for efficient calcu-

lation of likelihood gradients necessary for widely used machine-learning based numerical inference procedures [21,22]. In addition, the challenge of approximation suffers from the so-called "curse of dimensionality": the size of the state-space grows exponentially with the number of modeled species.

In the fields of machine learning and data-driven modeling, recent work has demonstrated the ability of sparse function approximations to distributions [23] and learned mappings between function spaces rather than between finite-dimensional Euclidean spaces [24] to learn generalizable solutions that govern the dynamics of complex systems. For example, [24] has introduced a neural operator formulation that models fluid dynamics in turbulent flows governed by such PDEs as the Navier-Stokes equations several orders of magnitude faster than traditional solvers.

Inspired by these approaches, we present a computationally efficient approach that uses neural networks coupled to statistical analysis of system distributions to approximate multidimensional solutions to CME models of transcriptional dynamics. In particular, we investigate a CME system involving transcriptional bursting, splicing and degradation (Figure 1a). This system of bursty transcription, although experimentally validated [25], does not yield a closed-form steady-state solution and requires costly numerical evaluation procedures [26]. We alleviate this problem by using features of the system's univariate conditional distributions to define a statistical approximation for conditional nascent RNA distributions in terms of a mixture of negative binomial kernel functions. The weights for these kernel functions are learned by a neural network (Figure 1b). We then multiply by marginal mature RNA probability to reconstruct full joint distributions (Figure 1c). We denote this procedure kernel weight regression (KWR).

To increase flexibility of the neural approximation while further decreasing evaluation time, we also implement an approximation approach by learning scaling factors for pre-calculated mean and variances of fewer negative binomial distribution kernels in addition to weights. The joint probability is again approximated by applying conditional probability. We call this approach parameter scaled kernel weight regression (psKWR). Both procedures reconstruct joint distributions by approximating conditional distributions for mature RNA molecules with neural-network scaled kernel distributions, then reconstructing full joint distributions using an analytical solution for marginal nascent RNA distribution. KWR and psKWR also provide approximations on an unbounded support and are compatible with gradient calculations required for machine-learning based likelihood optimization strategies, thereby facilitating inference of system parameters at increased resolution [27].

Our work complements other recent reports that leverage neural networks to approximate CME solutions for single-cell biophysics models. [28] use recurrent neural networks to directly learn CME solutions over a grid but cannot extrapolate beyond a predefined state space. [29] have proposed neural approximations for continuous-valued statistics of CME systems but cannot be used to perform inference. Most pertinently, [30] have independently proposed a method for approximating CME solutions with a set of negative binomial mixture functions, implemented using the Nessie framework. However, Nessie learns mixture function locations and weights, while our approach simplifies the learning process by learning weights and location scaling factors, initially placing kernel functions by exploiting the unimodality and moments of the system's CME solution. Further, Nessie is designed to fit univariate distributions; we present a strategy for learning multivariate distributions, an essential feature for the analysis of increasingly available multimodal data.

2 Methods

2.1 Transcriptional system

We consider the system illustrated in Figure 1a, which encodes a standard model of the mRNA life-cycle [26]. Transcriptional events arrive according to a Poisson process with rate α, and generate B nascent mRNAs (N), where B is a geometric distribution with mean b. After an exponentially-distributed delay with rate β, nascent mRNAs are spliced into mature mRNAs (M). Finally, after an exponentially-distributed delay with rate γ, mature mRNAs are degraded. This system induces a stationary distribution with the probability mass function (PMF) $P(n, m; b, \beta, \gamma)$, where β and γ are considered in units of α with no loss of generality at steady state. Although the system does not afford a closed-form solution, its joint PMF can be approximated to high precision through generating functions evaluated over a state-space grid (the larger the state-space, the greater the accuracy) [26]. Further, its nascent marginal $P(n)$ is negative binomial [26].

2.2 KWR approximation overview

To approximate full joint distributions, we train a neural network to approximate the distributions of mature RNA molecules conditional on a nascent copy number n (Figure 1b), i.e., the univariate PMFs $P(m|n; b, \beta, \gamma)$. We first generate rate vectors $\theta := \{b, \beta, \gamma\}$ and compute high-quality generating function solutions over state-space grids to obtain joint PMFs $P(m, n; \theta)$, following Equation 18. These solutions are denoted QV20 (as they are evaluated on grids with limits 20 standard deviations above nascent and mature marginal means) and treated as ground truth. We then divide by nascent probabilities $P(n; \theta)$ to extract conditional mature probabilities $P(m|n; \theta)$.

To construct a basis for the conditional distribution at the nascent copy number n, we use a deterministic function $\psi(\theta, n)$ to calculate kernel means μ_k and standard deviations σ_k of negative binomial basis kernels P_{k}, where k indexes the negative binomial kernels (Section S1.3). The function takes advantage of the marginal means and variances of the system, constructing an initial basis by moment-matching to an analytically tractable lognormal distribution (Section S1.3). Next, we use a neural network to map $\tilde{\theta}_{\mathrm{KWR}}$, a transformation of θ and n, to a set of weights $\mathbf{w}$ and a kernel width scaling factor h. We train the network to minimize truncated Kullback-Leibler divergence (KL divergence, Section S1.4) between the conditional true distributions $P(m|n; \theta)$ and the conditional kernel approximations $Q_{\mathrm{KWR}}(m|n, \theta)$:

$$Q_{\mathrm{KWR}}(m|n; \theta) = \sum_k w_k P_{\mathrm{k}}(m; \mu_k, h\sigma_k). \tag{1}$$

Finally, to reconstruct the joint distributions, we multiply each $Q_{\mathrm{KWR}}(m|n; \theta)$ by the analytical nascent marginal probability $P(n; \theta)$, as in Figure 1c:

$$Q_{\mathrm{KWR}}(m, n; \theta) = Q_{\mathrm{KWR}}(m|n; \theta) P(n; \theta). \tag{2}$$

We call this method kernel weight regression (KWR), with a full description in Section S1.7. Model architecture choice is motivated in Figure S3. Figure S1 depicts final model architecture and Table S3 lists final model parameters.

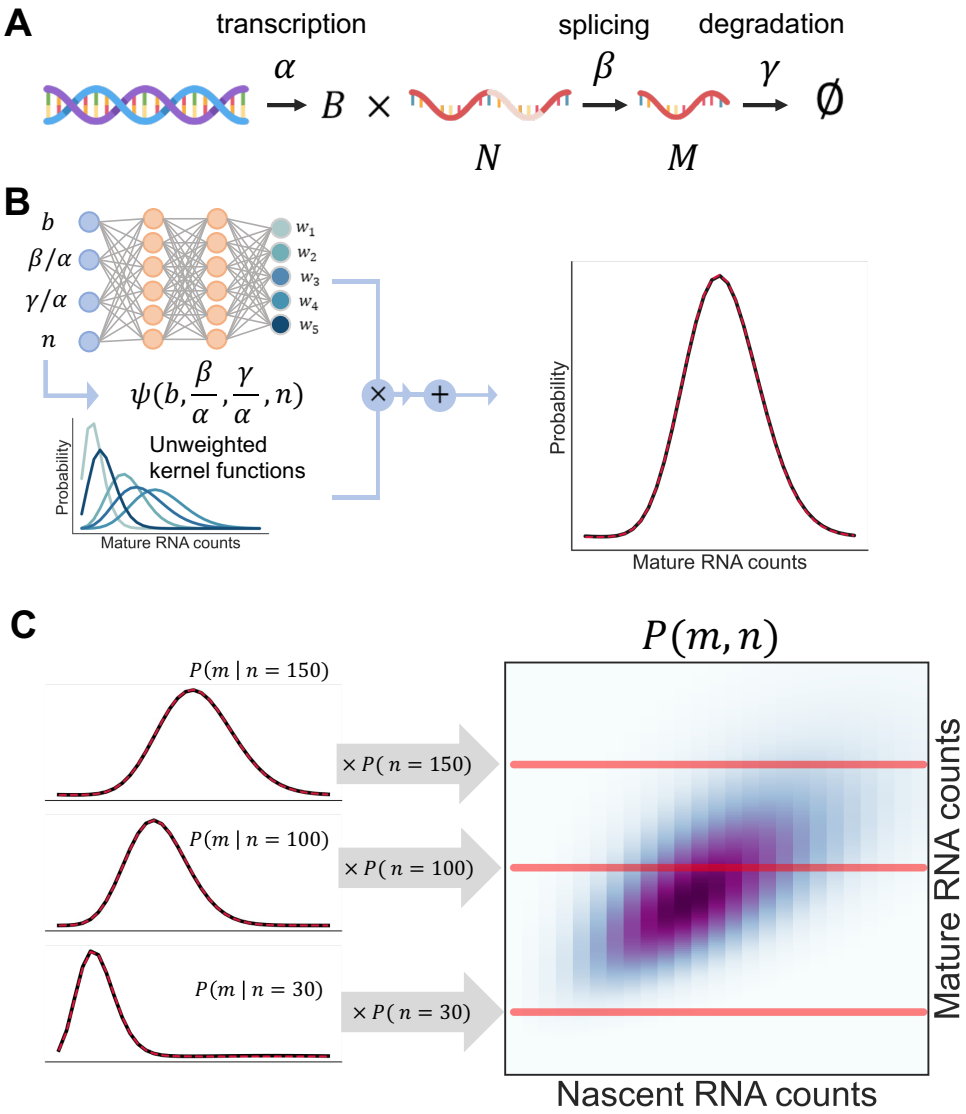

Figure 1: A schematic representation of the KWR approximation procedure. **A.** The biological process under consideration: bursty transcription of nascent RNA, followed by splicing to produce mature RNA and mRNA degradation. **B.** Univariate conditional distributions are approximated by summing a set of kernel functions with neural network learned weights. **C.** Joint distributions are reconstructed by multiplying conditional mature RNA probabilities by marginal nascent RNA probabilities.

2.3 Parameter scaled kernel weight approximation overview

To further allow for flexibility of distribution approximation, we present an alternative strategy, which scales the pre-calculated kernel means and variances. Here, we use the same function $\psi(\theta, n)$ for a given n to estimate kernel distribution means μ_k and standard deviations σ_k. We then use a neural network to map $\tilde{\theta}_{\text{psKWR}}$, a slightly modified transformation of θ and n, to weights $\mathbf{w}$ and scaling factors $\mathbf{c}_\mu$ and $\mathbf{c}_\sigma$ to construct the approximate conditional distribution $Q_{\text{psKWR}}(m|n;\theta)$:

$$Q_{\text{psKWR}}(m|n;\theta) = \sum_k w_k P_k(m; c_{\mu_k}\mu_k, c_{\sigma_k}\sigma_k), \tag{3}$$

where k again indexes the kernel. The joint probabilities are then reconstructed by multiplying each $Q_{\text{psKWR}}(m|n;\theta)$ by marginal $P(n;\theta)$:

$$Q_{\text{psKWR}}(m, n;\theta) = Q_{\text{psKWR}}(m|n;\theta)P(n;\theta). \tag{4}$$

We train the network to minimize the Hellinger distance between the joint true distributions $P(m, n;\theta)$ and the joint approximations $Q_{psKWR}(m, n;\theta)$.

We call this approach parameter scaled kernel weight regression (psKWR). It is fully described in Section S1.8, with model architecture choice motivated in Figure S4, final model architecture illustrated in Figure S2, and final model parameters listed in Table S4.

2.4 Data generation and generating function solutions

To generate training, validation, and testing distributions, we sampled vectors of biophysical rate vectors θ within the following biologically motivated bounds:

$$\begin{aligned} b &\in [1, 300], \\ \beta &\in [0.05, 50], \\ \gamma &\in [0.05, 50], \end{aligned} \tag{5}$$

where the sampling measure was log-uniform. As described previously, β and γ are unitless quantities, scaled to the value of α. We generated a vector of three random numbers within the bounds. Next, we computed μ_N and μ_M, the average nascent and mature copy numbers (Table S1). To ensure that training distributions matched the low-copy number regime of experimental scRNA-seq data, we restricted to vectors with μ_N and μ_M both less than 1,000. For each θ, we evaluated the previously derived generating function solution to the system, given in Equation 18, up to the bounds $\mu_z + 20\sigma_z$ (where z is either nascent n or mature m value) using adaptive quadrature (implemented in the `scipy` function `integrate.quad_vec` [31]) and the inverse real fast Fourier transform (implemented in the `scipy` function `fft.irfft2` [32]). We denote this high-precision procedure by QV20 and treat it as ground truth.

We further generated lower-order generating function solutions for benchmarking. We calculated generating function solutions using adaptive quadrature on grids of 10, 4, and 1 standard deviations above nascent and mature means, or within the bounds $\mu_z + 10\sigma_z$, $\mu_z + 4\sigma_z$, and $\mu_z + \sigma_z$ in each dimension $z \in \{N, M\}$. We refer to these solutions as QV10, QV4, and QV1 in the text and Figure 2.

We trained KWR models on normalized conditional values and psKWR models on joint microstate probabilities. To generate normalized conditional training data for KWR, we extracted

the entries corresponding to each n and divided them by the analytical solution for the marginal $P(n;\theta)$ (Equation 7, implemented in the `scipy` function `stats.nbinom.pmf` [31] with $r = \beta^{-1}$ and $p = (1+b)^{-1}$ [33]). The resulting conditional distributions were stored along with their values of θ and n.

2.5 Model training and architectures

KWR models were trained to minimize the truncated KL divergence (see Section S1.4) between approximated conditional distributions $Q_{\text{KWR}}(m|n;\theta)$ and normalized QV20 conditional distributions $P(m|n;\theta)$ for mature RNA transcripts.

Architectures of varying number of layers, nodes per layer, and number of kernel functions were trained and compared, with testing results for different models shown in Figure S3. The architecture chosen for final comparisons is a MLP comprising 2 hidden layers of 256 nodes the output of which is passed to one layer of 256 nodes that produces weights for 10 basis functions, and a different layer of 256 nodes that produces the scaling factor h (described in 2.2). This final model was trained for 35 training epochs with a learning rate of 0.001 with 1,536 training rate vectors θ (corresponding to 211,515 vectors $\tilde{\theta}_{\text{KWR}}$ conditional vectors) and 512 validation rate vectors θ (corresponding to 72,906 vectors $\tilde{\theta}_{\text{KWR}}$ conditional vectors). All models were implemented in PyTorch with the Adam optimizer [34]. Full descriptions of training and architectures are included in Section S1.7, Figure S1, and Table S3.

psKWR models were trained to minimize the Hellinger distance between joint approximations $Q_{psKWR}(m,n;\theta)$ and QV20 joint microstate probabilities $P(m,n;\theta)$. We again compared model architectures of different numbers of layers, nodes per layers, and basis functions to obtain a final model (Section S1.7 and Figure S3). The final model chosen for psKWR approximation is a MLP with 3 hidden layers (L1, L2, and L3) of 512 nodes per layer that produce weights $\mathbf{w}$ for five kernel functions functions. It further includes one layer of 512 nodes that takes outputs from L2 to produce scaling factors $\mathbf{c}_\mu$ for kernel means described in Section layer of 512 nodes that takes outputs from L2 to produce scaling factors $\mathbf{c}_\sigma$ for kernel standard deviations. It was trained for 100 epochs on 1,280 rate vectors θ (corresponding to 211,515 conditional vectors $\tilde{\theta}_{\text{psKWR}}$) and 512 validation rate vectors θ (corresponding to 72,906 conditional vectors $\tilde{\theta}_{\text{psKWR}}$). The model was implemented in PyTorch and optimized using the Adam optimizer with a learning rate of 0.001 [34] . A full description of training is included in S1.8 with final model architecture diagrammed in S2 and final model parameters listed in Table S4.

2.6 Inference of biophysical parameters in mouse brain cells

We implemented psKWR and KWR neural approximation strategies into an existing maximum likelihood inference framework *Monod* [35] to infer burst sizes and rates of splicing and degradation for 200 genes in Gabaergic cells in the mouse brain [36]. We obtained data from mouse sample C01 (donor ID 427378) from the Allen Institute Mouse Primary Motor Cortex (MOp) dataset [36]. Raw FASTQ files for the data were obtained and processed using `kallisto` — `bustools` as described in [35]. Briefly, raw reads were aligned to a reference of unspliced and spliced transcripts (generated using the flag `--lamanno`) to obtain counts for unspliced (intron containing, treated as nascent transcripts) and spliced transcripts (non-intron containing, treated as mature transcripts). Cells were filtered to remove low-quality cells (as in [35] by removing those that did not pass `kallisto` — `bustools` filtering and those with fewer that 10^4 unique molecular identifier counts.

We then restricted our analysis to cells of a single cell sub-type, Gabaergic neurons, according to cell type annotations from the original study [36]. For these 1,704 subset Gabaergic cells, gene selection was performed using *Monod* to obtain genes with unspliced and spliced means (across all cells) of at least 0.01, maximum of unspliced and spliced counts (across all cells) of at least 3, and maximum spliced counts (across all cells) no greater than 400. We then selected the top 200 of remaining genes by expression (as in [35]).

For all 200 genes, inference was performed to obtain burst size b, relative splicing rate β/α, and relative degradation rate γ/α, according to the bursty model of transcription (Section 2.1), by maximizing data likelihood under the model's steady-state distribution. For all genes, three different methods of evaluating data likelihoods were used: 1) the default generating function method with adaptive quadrature (QV, Section S1.2 , 2) KWR approximation, and 3) psKWR approximation. Inference bounds of $[-1.0, 4.2]$, $[-1.8, 2.5]$,$[-1.8, 3.5]$ were set for $\log_{10}b$, $\log_{10}\beta$, and $\log_{10}\gamma$, respectively. Inference was performed with a maximum of 15 iterations, a single search (no restarts), and parameters initiated at their method of moments estimates.

3 Results and discussion

To benchmark the algorithm's performance, we investigated the accuracy and runtime for PMF grid evaluation for a set of testing rate vectors (Figure 2a). To quantify accuracy, we calculated the Hellinger distance, which measures the discrepancy between probability distributions P and Q and takes the form $\frac{1}{\sqrt{2}}\|\sqrt{P}-\sqrt{Q}\|_2$, using the high-quality QV20 solutions as ground truth.

3.1 Comparison to generating function grid evaluations

We first compared KWR to three lower-order generating function-based solutions, denoted QV1, QV4, and QV10, which evaluate the numerical steady-state solution (Section S1.6) to varying accuracies based on grid size. We found that the KWR approximation achieves accuracy comparable to lower-order generating function solutions while reducing evaluation time (Figure 2a). For example, the average Hellinger distances for testing examples (normalized by grid size) were 3.085×10^{-5} and 3.370×10^{-6} for QV4 and KWR, respectively, but KWR approximation was more than 6 times faster (even over a grid) with an average time of 9.6307×10^{-6} seconds for QV4 evaluation and 1.624×10^{-6} seconds for KWR approximation (Figure 2**A**). KWR was both faster and almost an order of magnitude more accurate than generating function solution QV1, which had the larger average evaluation time (normalized by grid size) of 4.978×10^{-5} seconds and the greater Hellinger distance of 2.637×10^{-4} (both normalized by grid size, Figure 2**A**). While psKWR was slightly less accurate than KWR, its accuracy was still comparable to the generating function solutions of QV4 and more accurate than QV1 with a 3.976×10^{-5} average Hellinger distance. psKWR even further reduced evaluation time with an average normalized grid evaluation runtime of 4.414×10^{-6} seconds, two orders of magnitudes faster than QV1 (Figure 2**A**). This benchmark used 256 testing examples, with Hellinger distances and runtimes normalized by the grid size.

Comparison to statistical controls and baseline models

As a control to assess how learning kernel function weights contributes to reconstruction accuracy, we investigated approximations obtained using random weights (RW) $\mathbf{w}$ in Equation 1. RW employs the same function $\psi_{KWR}(\theta, n)$ to generate kernel means and variances as KWR, and identically

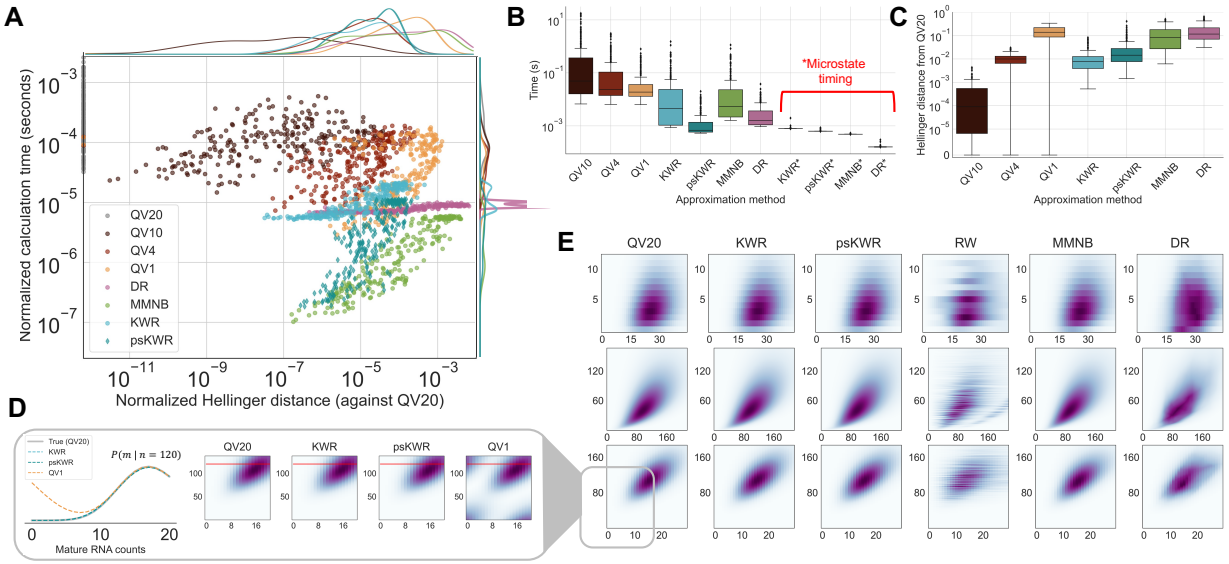

Figure 2: KWR and psKWR approximations accurately reconstruct joint probability distributions, with low runtime. **A.** Time to generate joint distributions over grid versus Hellinger distance, both normalized by grid size, for 256 testing rate vectors, comparing three generating function based methods (QV10, QV4, and QV1), direct regression (DR), a moment-matched negative binomial (MMNB) approximation, kernel weight regression (KWR), and parameter scaled kernel weight regression (psKWR) to ground truth (QV20). **B.** Time (unnormalized) to calculate probability for 768 testing rate vectors, where the first seven bars are grid-evaluation and an asterisk indicates method runtime for a single microstate probability. **C.** Hellinger distance (unnormalized) over grid for 768 testing rate vectors. **D.** Conditional probabilities and joint probabilities calculated using KWR, psKWR, and QV1 on a grid one standard deviation above nascent and mature means for a single testing rate vector show that significant error (as compared to QV20) is introduced on a small grid by truncation for quadrature methods but not neural approximation methods. **E.** Three example distributions approximated by KWR, psKWR, and several controls (RW, MMNB, and DR) demonstrate qualitative agreement between neural approximation methods and high-quality numerical solutions (QV20) but not all controls.

scales variances by learned factor h, but weights kernels with random weights that sum to unity. As shown quantitatively in Figure 2**B**, RW does not accurately reconstruct joint probabilities, motivating the use of learned weights.

To explore how much accuracy improves when using a neural network to scale means and variances rather than using mere statistical approximations, we also benchmarked against a non-neural, statistical approximation (moment-matched negative binomial or MMNB). The MMNB approach (Section S3) generates statistically plausible means and variances for mature RNA conditional distributions (using the same function $\psi_{KWR}(\theta, n)$). These means and variances parameterize a single negative binomial distribution, which are then multiplied by the analytical nascent marginal probability to produce join probability approximations. MMNB underperforms KWR and psKWR in Hellinger distance accuracy over a grid, evidence that the neural procedure is necessary to achieve highly accurate approximations.

To assess the utility of kernel functions, we also compared KWR and psKWR to a method of direct approximation that uses a fully connected multilayer perceptron to map from an input $\{\theta, n, m\}$ to a single-valued joint probability $P(m, n; \theta)$. The network was trained to reduce the mean squared error between numerically calculated and predicted joint probabilities. We refer to this method as direct regression (DR), described in Section S2 with testing results in Figure S5 and network parameters in Table S5. DR was considerably less accurate than the KWR and psKWR approximations (Figure 2**A,B**). This highlights that basis function learning methods like KWR and psKWR, which output parametric forms rather than single probabilities, are desirable and efficient for this application.

Evaluation time for microstate probabilities

In the context of statistical inference from data, optimizing likelihoods involves the evaluation of a single microstate $P(n, m; \theta)$. Here, KWR and psKWR outperform generating function-based methods by design by bypassing grid evaluation. Generating function evaluation over grids can take tens of seconds (QV10, Figure 2**B**), extremely prohibitive for inference processes in high-throughput genomics in which likelihoods must be iteratively calculated for thousands of genes. Evaluating single microstate probabilities using the neural approximations, by contrast, took 4.205×10^{-2} and 3.284×10^{-3} seconds on average, decreasing evaluation time by three and four orders of magnitude (Figure 2**B**) with similar accuracies (Figure 2**A,C**). While DR and MMNB slightly outperform KWR on microstate timing (Figure 2**B**), their accuracy is substantially lower (Figure 2**A,C**). Further, psKWR has evaluation time comparable to DR and MMNB and is more accurate (Figure 2**A,B,C**).

Joint distribution reconstruction

Some examples of probability distributions obtained by the approximation procedures are shown in Figure 2**D**, where we again treat QV20 as ground truth. Both KWR and psKWR approximations are qualitatively indistinguishable from ground truth, suggesting that they are appropriate for moderate-precision inference. While MMNB results look qualitatively similar, Hellinger distances from ground truth are substantially greater (Figure 2**A,C**), suggesting that while it is appropriate as a coarse approximation, it may not be not sufficient for precise inference. The random weight control recapitulates the general shape of the distribution, but introduces considerable error into microstate probabilities, confirming that the learned weights are essential for an accurate approx-

imation (Figure 2**D**, fourth column). Finally, the DR results noticeably distort the distribution shapes, further evidencing the utility of the kernel function approach (Figure 2**D**, sixth column).

KWR and psKWR use in biophysical parameter inference

KWR and psKWR can be used in methods in an existing maximum likelihood framework [35] to infer biophysical parameters from experimental data. We used both neural approximation strategies and the default generating function solver to infer burst sizes (denoted QV in Figure 3), relative splicing rates, and relative degradation rates for 200 highly expressed genes in Gabaergic cells from the mouse primary motor cortex (see Section 2.6) [36].

For the 200 fit genes, we found that inference results using KWR and psKWR approaches for data likelihood evaluation highly correlate with results using the standard generating function approach (Figure 3**A**), with better defined likelihood landscapes (Figure 3**B**). Pearson correlations of greater than 0.9 were obtained for $\log_{10}$ of inferred burst sizes, relative splicing rates, and relative degradation rates for KWR against QV and psKWR against QV (Figure 3**A**). KL divergence between empirical and model distributions at optimal inferred parameters also agree between QV and both neural approximation strategies (Figure 3**C**). In addition, distributions produced by all methods capture empirical RNA distributions, with two example genes shown in Figure 3**D** and **E**. These results suggest that KWR and psKWR are reliable alternatives to numerical solutions for parameter inference.

Conclusion

After training on a relatively small set of example rate vectors, the KWR and psKWR neural approximation approaches generalize to produce highly accurate and unbounded solutions to a mechanistic model of the lifecycle of RNA in single cells (Figure 2). They are as accurate as lower-order generating function methods, but bypass quadrature and Fourier inversion, achieving orders of magnitude faster runtimes. Thus, these approaches can greatly reduce the amount of time necessary to explore steady-state distributions across a wide range of transcriptional parameters for inference, parameter sensitivity investigations, and experimental design [37]. In particular, we have recently employed KWR application in a popular variational autoencoder framework to infer biophysical parameters for thousands of cells and genes, made possible because KWR, unlike generating function solutions, is compatible with gradient descent algorithms [21,27].

Recent advances in simultaneous quantification [38] and modeling [39] of mRNA and proteins present natural further applications as trivariate unspliced mRNA – spliced mRNA – protein data are available, but the corresponding stochastic system does not afford an analytical solution. Single cell measurements of accessible regions of chromatin, obtained using ATAC-seq [40], may also be modeled using CMEs to produce multivariate steady-state distributions, an example of yet another system for which our neural approximation strategies can facilitate mechanistic analysis. As single-cell assays with multiple molecular species become increasingly more common, so increases the need for computationally efficient methods for modeling and inference that go beyond statistical summaries to yield mechanistic insight into system dynamics.

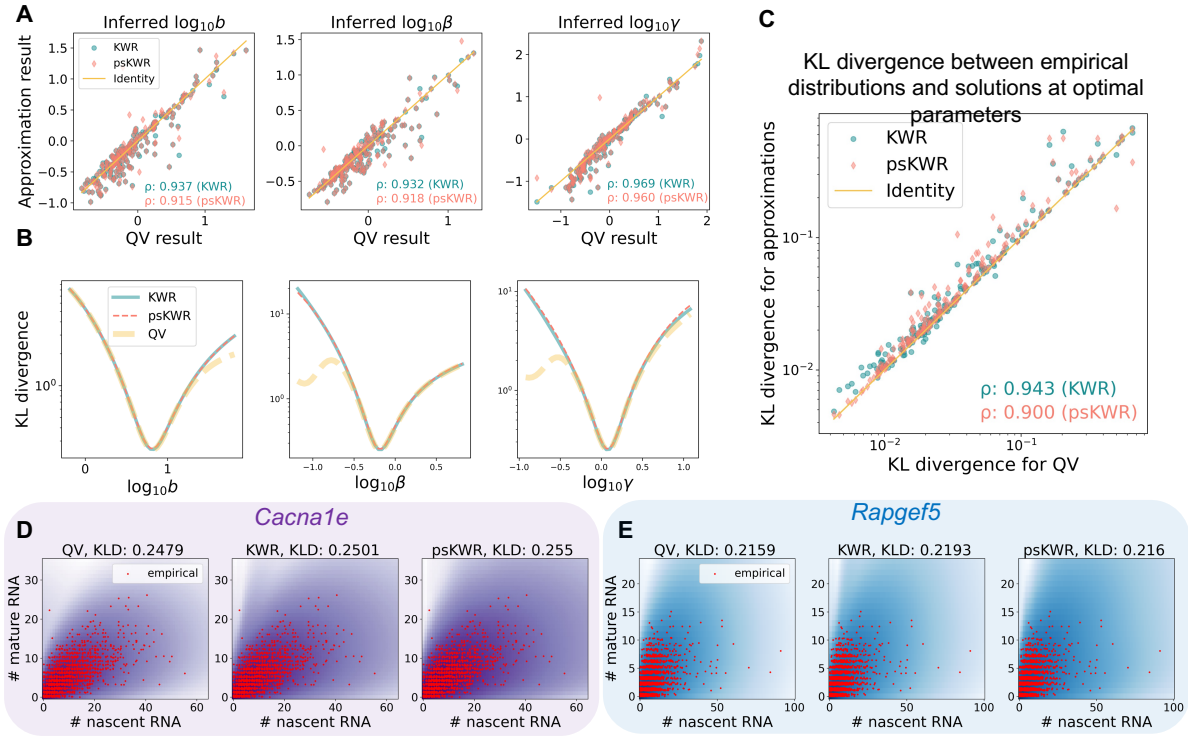

Figure 3: Use of KWR and psKWR in inference frameworks results in likelihoods and parameters consistent with those obtained using standard numerical solutions, with better behaved likelihood landscapes. **A.** Parameters for the bursty model of transcription (burst size b, splicing rate β, and degradation rateγ, with splicing and degradation rate considered in units of transcription rate α) highly correlate with results using adaptive quadrature. **B.** For the gene *Cacna1e*, Kullback-Leibler divergence between data and biophysical model likelihoods calculated at two of the optimal parameters with the indicated third varied. Likelihood landscapes for biophysical parameters evaluated using KWR and psKWR show one clear local and global minimum, while truncation necessary for numerical solution QV introduces off-optimum decreases in KL divergence. **C.** The $\log_{10}$ of lowest KL divergence obtained between data distributions and neural approximation strategies and data distributions and QV solutions at their respective optimal parameters highly correlate. **D.** Joint count distributions for *Cacna1e* and **E.** *Rapgef5*, both empirical (red scatter points) and evaluated using QV, KWR, and psKWR (colored density) at the optimal parameters they respectively produced.

4 Data and code availability

All training and validation datasets, trained models, and scripts to generate manuscript figures are available at https://github.com/pachterlab/GCCP_2022.

5 Author contributions

G.G. conceptualized the neural approximation strategies and was involved in implementation, statistical analyses, and basis function design. M.C. implemented and adapted the strategies, trained models, and performed analyses. T.C. and L.P. provided feedback, design suggestions, and troubleshooting advice. All authors wrote the article.

6 Declaration of interests

The authors declare no competing interests.

7 Acknowledgments

We thank Dr. Yisong Yue, Yongin Choi, Dr. John J. Vastola, and Dr. Zachary Fox for valuable discussions. G.G. and L.P. were partially funded by NIH U19MH114830. M.C. is funded by the National Science Foundation Graduate Research Fellowship Program and Amazon AI4Science Fellowship. Figures 1, 2, and S3 use color palettes from MetBrewer, developed by Blake R. Mills and available at https://github.com/BlakeRMills/MetBrewer.

References

[1] Ido Golding, Johan Paulsson, Scott M. Zawilski, and Edward C. Cox. Real-Time Kinetics of Gene Activity in Individual Bacteria. *Cell*, 123(6):1025–1036, December 2005.

[2] Crispin Gardiner. *Handbook of Stochastic Methods for Physics, Chemistry, and the Natural Sciences*. Springer, third edition, 2004.

[3] Gennady Gorin, John J. Vastola, Meichen Fang, and Lior Pachter. Interpretable and tractable models of transcriptional noise for the rational design of single-molecule quantification experiments. *Nature Communications*, 13(1):7620, December 2022.

[4] Brian Munsky, Guoliang Li, Zachary R. Fox, Douglas P. Shepherd, and Gregor Neuert. Distribution shapes govern the discovery of predictive models for gene regulation. *Proceedings of the National Academy of Sciences*, 115(29):7533–7538, 2018.

[5] Xiaoming Fu, Heta P Patel, Stefano Coppola, Libin Xu, Zhixing Cao, Tineke L Lenstra, and Ramon Grima. Quantifying how post-transcriptional noise and gene copy number variation bias transcriptional parameter inference from mRNA distributions. *eLife*, 11:e82493, October 2022.

[6] Brian Munsky, Zachary Fox, and Gregor Neuert. Integrating single-molecule experiments and discrete stochastic models to understand heterogeneous gene transcription dynamics. *Methods*, 85:12–21, 2015.

[7] Grace X. Y. Zheng, Jessica M. Terry, Phillip Belgrader, Paul Ryvkin, Zachary W. Bent, Ryan Wilson, Solongo B. Ziraldo, Tobias D. Wheeler, Geoff P. McDermott, Junjie Zhu, Mark T. Gregory, Joe Shuga, Luz Montesclaros, Jason G. Underwood, Donald A. Masquelier, Stefanie Y. Nishimura, Michael Schnall-Levin, Paul W. Wyatt, Christopher M. Hindson, Rajiv Bharadwaj, Alexander Wong, Kevin D. Ness, Lan W. Beppu, H. Joachim Deeg, Christopher McFarland, Keith R. Loeb, William J. Valente, Nolan G. Ericson, Emily A. Stevens, Jerald P. Radich, Tarjei S. Mikkelsen, Benjamin J. Hindson, and Jason H. Bielas. Massively parallel digital transcriptional profiling of single cells. *Nature Communications*, 8(1):14049, April 2017.

[8] Kristján Eldjárn Hjörleifsson, Delaney K. Sullivan, Guillaume Holley, Páll Melsted, and Lior Pachter. Accurate quantification of single-nucleus and single-cell RNA-seq transcripts. Preprint, bioRxiv: 2022.12.02.518832, December 2022.

[9] Gioele La Manno, Ruslan Soldatov, Amit Zeisel, Emelie Braun, Hannah Hochgerner, Viktor Petukhov, Katja Lidschreiber, Maria E. Kastriti, Peter Lönnerberg, Alessandro Furlan, Jean Fan, Lars E. Borm, Zehua Liu, David van Bruggen, Jimin Guo, Xiaoling He, Roger Barker, Erik Sundström, Gonçalo Castelo-Branco, Patrick Cramer, Igor Adameyko, Sten Linnarsson, and Peter V. Kharchenko. RNA velocity of single cells. *Nature*, 560(7719):494–498, August 2018.

[10] Páll Melsted, A. Sina Booeshaghi, Lauren Liu, Fan Gao, Lambda Lu, Kyung Hoi Min, Eduardo da Veiga Beltrame, Kristján Eldjárn Hjörleifsson, Jase Gehring, and Lior Pachter. Modular, efficient and constant-memory single-cell RNA-seq preprocessing. *Nature Biotechnology*, 39(7):813–818, July 2021.

[11] Eleni P. Mimitou, Anthony Cheng, Antonino Montalbano, Stephanie Hao, Marlon Stoeckius, Mateusz Legut, Timothy Roush, Alberto Herrera, Efthymia Papalexi, Zhengqing Ouyang, Rahul Satija, Neville E. Sanjana, Sergei B. Koralov, and Peter Smibert. Multiplexed detection of proteins, transcriptomes, clonotypes and CRISPR perturbations in single cells. *Nature Methods*, 16(5):409–412, May 2019.

[12] Junyue Cao, Darren A. Cusanovich, Vijay Ramani, Delasa Aghamirzaie, Hannah A. Pliner, Andrew J. Hill, Riza M. Daza, Jose L. McFaline-Figueroa, Jonathan S. Packer, Lena Christiansen, Frank J. Steemers, Andrew C. Adey, Cole Trapnell, and Jay Shendure. Joint profiling of chromatin accessibility and gene expression in thousands of single cells. *Science*, 361(6409):1380–1385, September 2018.

[13] Heng Xu, Samuel O. Skinner, Anna Marie Sokac, and Ido Golding. Stochastic Kinetics of Nascent RNA. *Physical Review Letters*, 117(12):128101, 2016.

[14] Yuhan Hao, Stephanie Hao, Erica Andersen-Nissen, William M. Mauck, Shiwei Zheng, Andrew Butler, Maddie J. Lee, Aaron J. Wilk, Charlotte Darby, Michael Zager, Paul Hoffman, Marlon Stoeckius, Efthymia Papalexi, Eleni P. Mimitou, Jaison Jain, Avi Srivastava, Tim Stuart, Lamar M. Fleming, Bertrand Yeung, Angela J. Rogers, Juliana M. McElrath, Catherine A.

Blish, Raphael Gottardo, Peter Smibert, and Rahul Satija. Integrated analysis of multimodal single-cell data. *Cell*, 184(13):3573–3587.e29, June 2021.

[15] John J Vastola. *In search of a coherent theoretical framework for stochastic gene regulation*. PhD thesis, Vanderbilt, March 2021.

[16] Daniel T Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22(4):403–434, December 1976.

[17] Brian Munsky and Mustafa Khammash. The finite state projection algorithm for the solution of the chemical master equation. *The Journal of Chemical Physics*, 124(4):044104, 2006.

[18] Ankit Gupta, Jan Mikelson, and Mustafa Khammash. A finite state projection algorithm for the stationary solution of the chemical master equation. *The Journal of Chemical Physics*, 147(15):154101, October 2017.

[19] Zachary Fox, Gregor Neuert, and Brian Munsky. Finite state projection based bounds to compare chemical master equation models using single-cell data. *The Journal of Chemical Physics*, 145(7):074101, August 2016.

[20] Pavol Bokes, John R. King, Andrew T. A. Wood, and Matthew Loose. Exact and approximate distributions of protein and mRNA levels in the low-copy regime of gene expression. *Journal of Mathematical Biology*, 64(5):829–854, April 2012.

[21] Romain Lopez, Jeffrey Regier, Michael B. Cole, Michael I. Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15(12):1053–1058, December 2018.

[22] Adam Gayoso, Zoë Steier, Romain Lopez, Jeffrey Regier, Kristopher L. Nazor, Aaron Streets, and Nir Yosef. Joint probabilistic modeling of single-cell multi-omic data with totalVI. *Nature Methods*, 18(3):272–282, March 2021.

[23] Kathleen Champion, Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116(45):22445–22451, November 2019.

[24] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier Neural Operator for Parametric Partial Differential Equations. Preprint, arXiv: 2010.08895, May 2021.

[25] Long Cai, Nir Friedman, and X. Sunney Xie. Stochastic protein expression in individual cells at the single molecule level. *Nature*, 440(7082):358–362, March 2006.

[26] Abhyudai Singh and Pavol Bokes. Consequences of mRNA Transport on Stochastic Variability in Protein Levels. *Biophysical Journal*, 103(5):1087–1096, September 2012.

[27] Maria T. Carilli, Gennady Gorin, Yongin Choi, Tara Chari, and Lior Pachter. Biophysical modeling with variational autoencoders for bimodal, single-cell RNA sequencing data. Preprint, bioRxiv: 2023.01.13.523995, May 2023.

[28] Shangying Wang and Simone Bianco. AI-assisted Biology: Predict the Conditional Probability Distributions from Noisy Measurements. Preprint, bioRxiv: 2021.10.07.463577, October 2021.

[29] Ankit Gupta, Christoph Schwab, and Mustafa Khammash. DeepCME: A deep learning framework for computing solution statistics of the chemical master equation. *PLOS Computational Biology*, 17(12):e1009623, December 2021.

[30] Augustinas Sukys, Kaan Öcal, and Ramon Grima. Approximating solutions of the Chemical Master equation using neural networks. *iScience*, 25(9):105010, August 2022.

[31] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, Ilhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, SciPy 1.0 Contributors, Aditya Vijaykumar, Alessandro Pietro Bardelli, Alex Rothberg, Andreas Hilboll, Andreas Kloeckner, Anthony Scopatz, Antony Lee, Ariel Rokem, C. Nathan Woods, Chad Fulton, Charles Masson, Christian Häggström, Clark Fitzgerald, David A. Nicholson, David R. Hagen, Dmitrii V. Pasechnik, Emanuele Olivetti, Eric Martin, Eric Wieser, Fabrice Silva, Felix Lenders, Florian Wilhelm, G. Young, Gavin A. Price, Gert-Ludwig Ingold, Gregory E. Allen, Gregory R. Lee, Hervé Audren, Irvin Probst, Jörg P. Dietrich, Jacob Silterra, James T Webber, Janko Slavič, Joel Nothman, Johannes Buchner, Johannes Kulick, Johannes L. Schönberger, José Vinícius de Miranda Cardoso, Joscha Reimer, Joseph Harrington, Juan Luis Cano Rodríguez, Juan Nunez-Iglesias, Justin Kuczynski, Kevin Tritz, Martin Thoma, Matthew Newville, Matthias Kümmerer, Maximilian Bolingbroke, Michael Tartre, Mikhail Pak, Nathaniel J. Smith, Nikolai Nowaczyk, Nikolay Shebanov, Oleksandr Pavlyk, Per A. Brodtkorb, Perry Lee, Robert T. McGibbon, Roman Feldbauer, Sam Lewis, Sam Tygier, Scott Sievert, Sebastiano Vigna, Stefan Peterson, Surhud More, Tadeusz Pudlik, Takuya Oshima, Thomas J. Pingel, Thomas P. Robitaille, Thomas Spura, Thouis R. Jones, Tim Cera, Tim Leslie, Tiziano Zito, Tom Krauss, Utkarsh Upadhyay, Yaroslav O. Halchenko, and Yoshiki Vázquez-Baeza. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3):261–272, March 2020.

[32] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

[33] Lisa Amrhein, Kumar Harsha, and Christiane Fuchs. A mechanistic model for the negative binomial distribution of single-cell mRNA counts. Preprint, bioRxiv: 657619, June 2019.

[34] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy,

Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[35] Gennady Gorin and Lior Pachter. Distinguishing biophysical stochasticity from technical noise in single-cell RNA sequencing using *Monod*. Preprint, bioRxiv: 2022.06.11.495771, April 2023.

[36] Zizhen Yao, Hanqing Liu, Fangming Xie, Stephan Fischer, Ricky S. Adkins, Andrew I. Aldridge, Seth A. Ament, Anna Bartlett, M. Margarita Behrens, Koen Van den Berge, Darren Bertagnolli, Hector Roux de Bézieux, Tommaso Biancalani, A. Sina Booeshaghi, Héctor Corrada Bravo, Tamara Casper, Carlo Colantuoni, Jonathan Crabtree, Heather Creasy, Kirsten Crichton, Megan Crow, Nick Dee, Elizabeth L. Dougherty, Wayne I. Doyle, Sandrine Dudoit, Rongxin Fang, Victor Felix, Olivia Fong, Michelle Giglio, Jeff Goldy, Mike Hawrylycz, Brian R. Herb, Ronna Hertzano, Xiaomeng Hou, Qiwen Hu, Jayaram Kancherla, Matthew Kroll, Kanan Lathia, Yang Eric Li, Jacinta D. Lucero, Chongyuan Luo, Anup Mahurkar, Delissa McMillen, Naeem M. Nadaf, Joseph R. Nery, Thuc Nghi Nguyen, Sheng-Yong Niu, Vasilis Ntranos, Joshua Orvis, Julia K. Osteen, Thanh Pham, Antonio Pinto-Duarte, Olivier Poirion, Sebastian Preissl, Elizabeth Purdom, Christine Rimorin, Davide Risso, Angeline C. Rivkin, Kimberly Smith, Kelly Street, Josef Sulc, Valentine Svensson, Michael Tieu, Amy Torkelson, Herman Tung, Eeshit Dhaval Vaishnav, Charles R. Vanderburg, Cindy van Velthoven, Xinxin Wang, Owen R. White, Z. Josh Huang, Peter V. Kharchenko, Lior Pachter, John Ngai, Aviv Regev, Bosiljka Tasic, Joshua D. Welch, Jesse Gillis, Evan Z. Macosko, Bing Ren, Joseph R. Ecker, Hongkui Zeng, and Eran A. Mukamel. A transcriptomic and epigenomic cell atlas of the mouse primary motor cortex. *Nature*, 598(7879):103–110, October 2021.

[37] Zachary R Fox and Brian Munsky. The finite state projection based Fisher information matrix approach to estimate information and optimize single-cell experiments. *PLOS Computational Biology*, 15(1):e1006365, January 2019.

[38] Marlon Stoeckius, Christoph Hafemeister, William Stephenson, Brian Houck-Loomis, Pratip K Chattopadhyay, Harold Swerdlow, Rahul Satija, and Peter Smibert. Simultaneous epitope and transcriptome measurement in single cells. *Nature Methods*, 14(9):865–868, September 2017.

[39] Gennady Gorin, Valentine Svensson, and Lior Pachter. Protein velocity and acceleration from single-cell multiomics experiments. *Genome Biology*, 21:39, February 2020.

[40] Florian De Rop, Joy N Ismail, Carmen Bravo González-Blas, Gert J Hulselmans, Christopher Campbell Flerin, Jasper Janssens, Koen Theunis, Valerie M Christiaens, Jasper Wouters, Gabriele Marcassa, Joris de Wit, Suresh Poovathingal, and Stein Aerts. HyDrop enables droplet based single-cell ATAC-seq and single-cell RNA-seq using dissolvable hydrogel beads. *eLife*, 11:e73971, February 2022.

[41] Yung-Sung Cheng. Bivariate Lognormal Distribution for Characterizing Asbestos Fiber Aerosols. *Aerosol Science and Technology*, 5(3):359–368, January 1986.

[42] Gennady Gorin and Lior Pachter. Modeling bursty transcription and splicing with the chemical master equation. *Biophysical Journal*, 121(6):1056–1069, February 2022.

[43] C. W. Gardiner and S. Chaturvedi. The poisson representation. I. A new technique for chemical master equations. *Journal of Statistical Physics*, 17(6):429–468, December 1977.

[44] Martin Bengtsson, Anders Ståhlberg, Patrik Rorsman, and Mikael Kubista. Gene expression profiling in single cells from the pancreatic islets of Langerhans reveals lognormal distribution of mRNA levels. *Genome Research*, 15(10):1388–1392, October 2005.