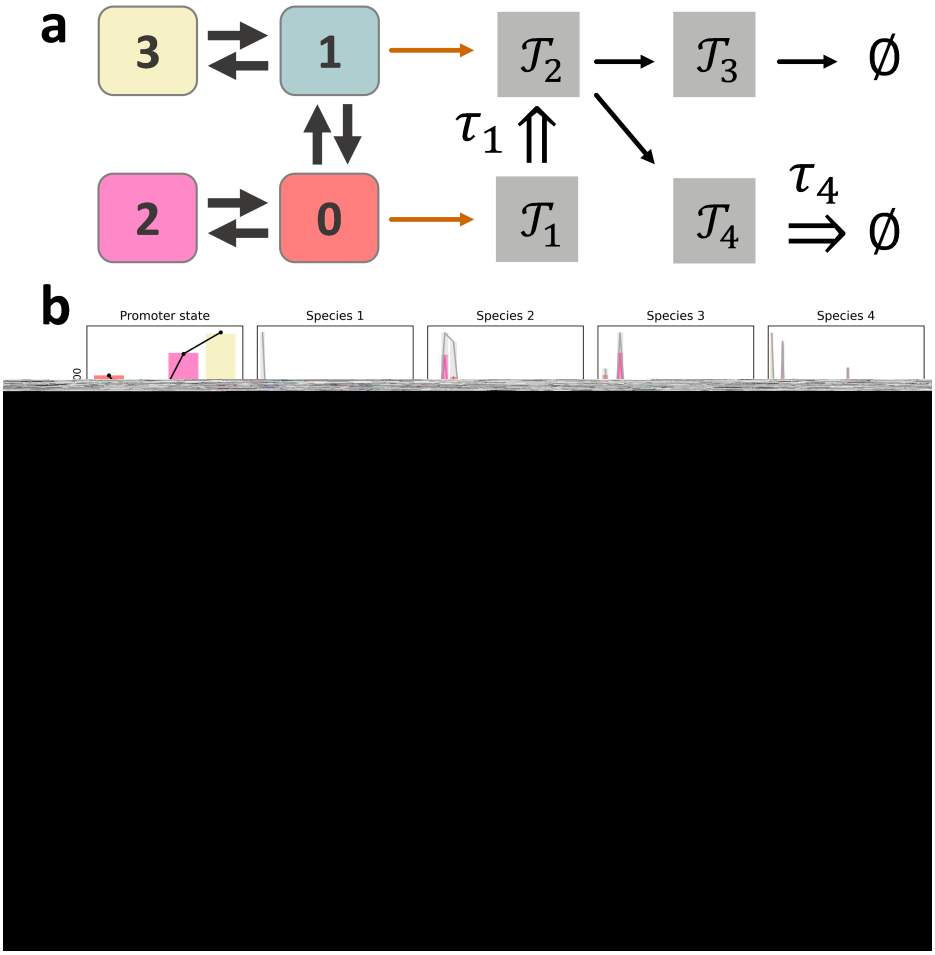

Figure 2: Validating a complicated non-Markovian transcriptional system. **a**. The structure of a system with four promoter states and four RNA species, two of which have deterministic lifetimes (rounded squares: promoter states; thick arrows: promoter state transitions; gray squares: RNA species; thin orange arrows: transcription reactions; thin black arrows: Markovian splicing and degradation reactions; wide arrows: deterministically delayed reactions). **b**. The analytical solutions for marginal distributions of promoter states and molecular species match simulations (Rows: time points. Leftmost column: promoter state distributions; histograms: simulation; lines: matrix exponential solution. Columns 2–5: the marginal distributions of molecular species; colored histograms: non-normalized marginals at each promoter state; gray histograms: molecular marginals aggregated over promoter states; lines: ODE solutions).

The characteristics were computed using the MATLAB Symbolic Math Toolbox (55). As shown in Figure 2b, the procedure generates results that match simulations, and affords full conditional distributions at arbitrary times and for arbitrary initial conditions.

3.2 Single-nucleus data do not show identifiable mechanistic differences from single-cell data

As discussed in Section 1, we do not *a priori* know that splicing and nuclear export are Markovian. Using the suite of mathematical tools described in Section 2.2 and relatively homogeneous data, we revisit (10) the question of RNA life-cycle modeling and consider the differences between nuclear and whole-cell systems in further detail.

We are primarily interested in comparing the stationary distributions induced by the DCME reaction system with delayed efflux

$$\varnothing \xrightarrow{\alpha} B \times \mathcal{T}_1 \xrightarrow{\beta} \mathcal{T}_2 \overset{\tau}{\Longrightarrow} \varnothing \tag{7}$$

to those induced by the CME system

$$\varnothing \xrightarrow{\alpha} B \times \mathcal{T}_1 \xrightarrow{\beta} \mathcal{T}_2 \xrightarrow{\gamma} \varnothing, \tag{8}$$

where B is distributed according to a geometric distribution on $\mathbb{N}_0$ with mean b. We solve the DCME system in Section S2.2. In addition, we solve the closely related system with delayed splicing, or deterministic retention of the molecule at the gene locus, in Section S2.3:

$$\varnothing \xrightarrow{\alpha} B \times \mathcal{T}_1 \overset{\tau}{\Longrightarrow} \mathcal{T}_2 \xrightarrow{\gamma} \varnothing. \tag{9}$$

By using a computer algebra system, such as the MATLAB Symbolic Math Toolbox (55), it is straightforward to obtain the lower moments. For comparison, Table 1 displays the moments for the Markovian and delayed systems.

Table 1: Model moments

Moment	Markovian	Delayed efflux	Delayed splicing
μ_1	$\frac{\alpha b}{\beta}$	$\frac{\alpha b}{\beta}$	$\alpha b \tau$
σ_1^2	$\mu_1(1+b)$	$\mu_1(1+b)$	$\mu_1(1+2b)$
μ_2	$\frac{\alpha b}{\gamma}$	$\alpha b \tau$	$\frac{\alpha b}{\gamma}$
σ_2^2	$\mu_2\left(1+\frac{b\beta}{\beta+\gamma}\right)$	$\mu_2 + 2\alpha b^2\left[\tau + \frac{1}{\beta}\left(e^{-\beta\tau}-1\right)\right]$	$\mu_2(1+b)$
$\mathrm{Cov}(X_1, X_2)$	$\frac{\alpha b^2}{\beta+\gamma}$	$\frac{\alpha b^2}{\beta}\left[1-e^{-\beta\tau}\right]$	0

We fit the three models introduced above, as well as a qualitatively similar overdispersed model with a gamma-distributed transcription rate (56–58) (Figure 3a), interpreting unspliced RNA counts as $\mathcal{T}_1$ and spliced counts as $\mathcal{T}_2$. The analysis was performed using *Monod* (11), using gradient descent to obtain maximum likelihood estimates (MLEs) under the four models (Section S3). As misspecification was evident with or without length bias in the original report (cf. `brain_5k_nuc` in Figs. S5 and S6 of (10)), we omitted the modeling of technical noise. The models were separately fit to GABAergic and glutamatergic cell types from two mouse brain samples (59), one generated using single-cell sequencing and one generated using single-nucleus sequencing, as well as single-cell and single-nucleus data from pericentral, periportal, and interzonal hepatocytes from a single human liver sample (60).

Upon fitting the MLEs, we computed the models' likelihood ratios relative to the standard bursty Markovian model and plotted them (Figure 3b). We did not observe a systematic bias toward the delayed-efflux model in the nuclear data (left column); the log-likelihood ratios were near zero in all considered cases, suggesting the data were insufficient to strongly favor either model in any of the datasets, nuclear or whole-cell. In contrast, the other two models were less favored relative to the standard model, with log-likelihood ratios tending to be negative. The strength of evidence against the models was lower in the single-nucleus data. This loss of statistical identifiability concords with intuition: the bursty Markovian, delayed-efflux, and extrinsic models afford identical negative binomial unspliced RNA marginals, requiring a large amount of spliced RNA to distinguish the models, which the nuclear sequencing protocol lacks. The delayed-splicing model has a different nascent marginal (described by the geometric-Poisson or Pólya–Aeppli distribution), and appeared to be more distinguishable from the negative binomial.

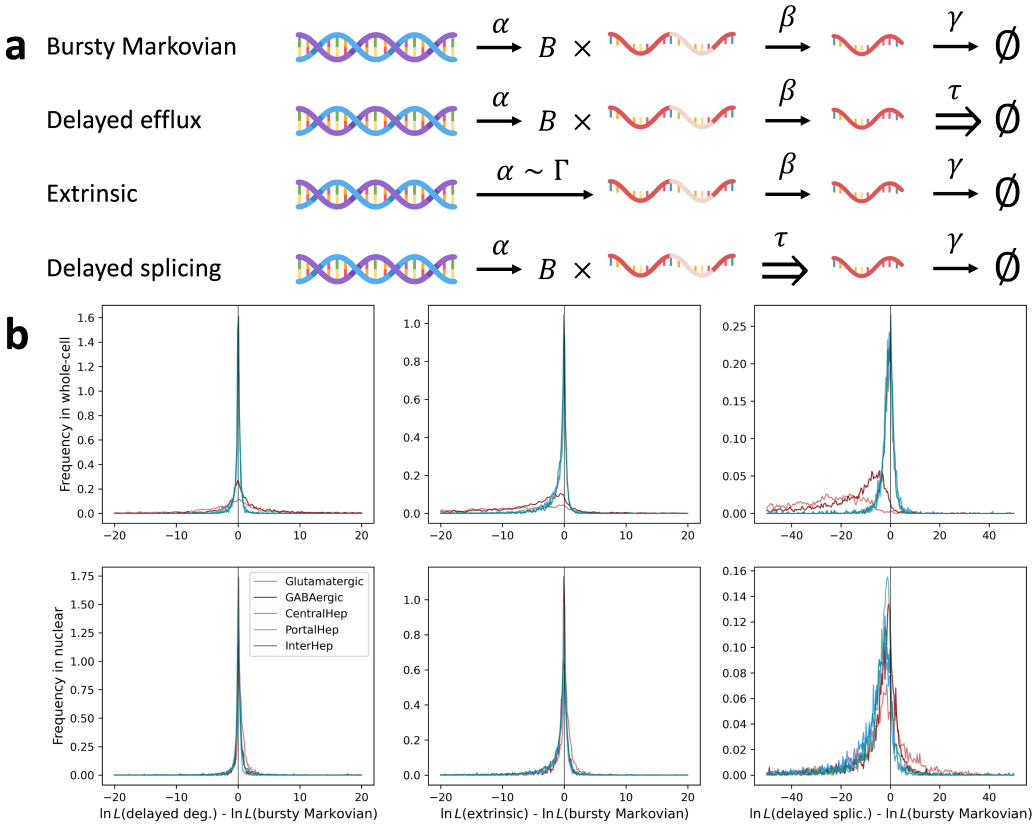

Figure 3: Certain models of RNA transcription and processing may be distinguished based on single-cell data, but single-nucleus data do not suggest that nuclear export biophysics are identifiably non-Markovian. **a.** The reaction schema of the considered models: DNA generates unspliced RNA with transcriptional burst frequency α, the unspliced RNA are converted to spliced RNA; the spliced RNA are removed from the system. In the "extrinsic" model, α is gamma-distributed; in the others, it is identical for all cells, and the transcriptional events produce bursts of size $B \sim$ Geom with mean b. **b.** Single-cell data (top row) support the bursty Markovian and delayed-efflux models to approximately the same degree, but offer consistently less support to the other two models, as quantified by the likelihood ratio. Single-nucleus data (bottom row) appear to show similar broad trends, but at lower magnitudes, limiting model identifiability (colors: cell types; red: Allen data; blue-green: Andrews data; lines: kernel density estimates).

4 DISCUSSION

This report extends the solutions reported in (24) to a broad class of delayed systems, and unifies them with the theory of switching genes outlined in (61). The approach augments the toolbox of stochastic biophysical analysis: it affords generating function-based analytical solutions for systems which would otherwise be outside the purview of the standard analysis of Markov chains. The derivation of these solutions does not require the usual manipulations of master equations at several time points, merely fairly standard integrals. These solutions may be fit (e.g., through the *Monod* implementation) and used as alternative hypotheses to investigate whether the standard assumptions of memorylessness hold, whether they are violated, or whether the available data are equivocal.

In Section 3.2, we used these analytical solutions to fit several superficially similar bivariate overdispersed models to single-cell and single-nucleus data, and observed that the typical Markovian model is a strong candidate, but cannot be easily distinguished from the competing delayed-efflux model, especially in the low-spliced expression single-nucleus data. On the other hand, two alternative models are consistently judged less plausible. This suggests that the assumption of Markovian, one-step splicing is useful despite its simplicity. This investigation is far from comprehensive. We forgo modeling technical noise or identifying cell types; using cell type definitions based on the spliced RNA counts introduces some degree of circularity. However, even with these limitations, the results illustrate fundamental challenges associated with single-nucleus data. To sum up, the data analysis is consistent with the following answers to the questions posed in Section 1:

- Single-nucleus data do not appear to require qualitatively different models. Both single-cell and single-nucleus data are consistent with the delayed efflux model.
- The Markovian splicing hypothesis is considerably more consistent with data than the delayed-splicing hypothesis.
- As may be expected, the low amounts of spliced RNA in nuclear data mean that only models with substantial differences in the unspliced distributions can be distinguished.

Furthermore, even when generating functions cannot be obtained, which is the typical case for systems with multiple RNA species, our method provides a simple numerical recipe for evaluating their distributions. The numerical approach is guaranteed to run in $O(N \ln N)$, can use off-the-shelf integration routines (e.g., quadrature for bursty systems and Runge–Kutta methods for switching systems), and enables the evaluation of arbitrary marginal distributions, which is not feasible for matrix- or simulation-based methods.

The approach is largely modular with respect to the specific details of the downstream processes, i.e., the causal relationships between molecules and the Markovian or deterministic waiting times for the reactions. In Section S5, we discuss a set of extensions to even more generic systems, with waiting times described by combinations of exponential and degenerate distributions. The same generating function-based approach can be used to implement models of technical noise and molecular non-identifiability under the assumption of independent sampling (10). Although the solutions are fairly generic, they do not yet provide a route to treating more complex systems with protein translation or molecular feedback. We speculate that a sufficiently general treatment of such systems may uncover similar relationships between CMEs and DCMEs, and provide a simulation routine that can be extended to model such phenomena.

5 ACKNOWLEDGMENTS

G.G. thanks Dr. Joseph J. Vastola for valuable discussions. G.G. and L.P. were partially funded by NIH U19MH114830. A part of the reported results were obtained during a Data Sciences Co-op with Celsius Therapeutics, Inc.

6 CODE AVAILABILITY

All scripts used to process data and generate the figures are located at https://github.com/pachterlab/GYP_2022. The raw count matrices, as well as the outputs of the *Monod* pipeline, are available on Zenodo (62).

REFERENCES

1. Lammers, N. C., Y. J. Kim, J. Zhao, and H. G. Garcia, 2020. A matter of time: Using dynamics and theory to uncover mechanisms of transcriptional bursting. *Current Opinion in Cell Biology* 67:147–157. https://linkinghub.elsevier.com/retrieve/pii/S0955067420300971.

2. Fukaya, T., B. Lim, and M. Levine, 2016. Enhancer Control of Transcriptional Bursting. *Cell* 166:358–368. https://linkinghub.elsevier.com/retrieve/pii/S0092867416305736.

3. Dar, R. D., B. S. Razooky, A. Singh, T. V. Trimeloni, J. M. McCollum, C. D. Cox, M. L. Simpson, and L. S. Weinberger, 2012. Transcriptional burst frequency and burst size are equally modulated across the human genome. *Proceedings of the National Academy of Sciences* 109:17454–17459. http://www.pnas.org/cgi/doi/10.1073/pnas.1213530109.

4. Rodriguez, J., and D. R. Larson, 2020. Transcription in Living Cells: Molecular Mechanisms of Bursting. *Annual Review of Biochemistry* 89:189–212. https://www.annualreviews.org/doi/10.1146/annurev-biochem-011520-105250.

5. Sanchez, A., and I. Golding, 2013. Genetic Determinants and Cellular Constraints in Noisy Gene Expression. *Science* 342:1188–1193. http://www.sciencemag.org/cgi/doi/10.1126/science.1242975.

6. Peccoud, J., and B. Ycard, 1995. Markovian Modeling of Gene Product Synthesis. *Theoretical Population Biology* 48:222–234.

7. Zhou, T., and J. Zhang, 2012. Analytical Results for a Multistate Gene Model. *SIAM Journal on Applied Mathematics* 72:789–818. http://epubs.siam.org/doi/10.1137/110852887.

8. Ham, L., D. Schnoerr, R. D. Brackston, and M. P. H. Stumpf, 2020. Exactly solvable models of stochastic gene expression. *The Journal of Chemical Physics* 152:144106. http://aip.scitation.org/doi/10.1063/1.5143540.

9. Sharova, L. V., A. A. Sharov, T. Nedorezov, Y. Piao, N. Shaik, and M. S. Ko, 2009. Database for mRNA Half-Life of 19 977 Genes Obtained by DNA Microarray Analysis of Pluripotent and Differentiating Mouse Embryonic Stem Cells. *DNA Research* 16:45–58. https://academic.oup.com/dnaresearch/article-lookup/doi/10.1093/dnares/dsn030.

10. Gorin, G., and L. Pachter, 2021. Length Biases in Single-Cell RNA Sequencing of pre-mRNA. Preprint, bioRxiv: 2021.07.30.454514. https://www.biorxiv.org/content/10.1101/2021.07.30.454514v1.

11. Gorin, G., and L. Pachter, 2022. *Monod*: mechanistic analysis of single-cell RNA sequencing count data. Preprint, bioRxiv: 2022.06.11.495771. http://biorxiv.org/lookup/doi/10.1101/2022.06.11.495771.

12. Zheng, G. X. Y., J. M. Terry, P. Belgrader, P. Ryvkin, Z. W. Bent, R. Wilson, S. B. Ziraldo, T. D. Wheeler, G. P. McDermott, J. Zhu, M. T. Gregory, J. Shuga, L. Montesclaros, J. G. Underwood, D. A. Masquelier, S. Y. Nishimura, M. Schnall-Levin, P. W. Wyatt, C. M. Hindson, R. Bharadwaj, A. Wong, K. D. Ness, L. W. Beppu, H. J. Deeg, C. McFarland, K. R. Loeb, W. J. Valente, N. G. Ericson, E. A. Stevens, J. P. Radich, T. S. Mikkelsen, B. J. Hindson, and J. H. Bielas, 2017. Massively parallel digital transcriptional profiling of single cells. *Nature Communications* 8:14049. http://www.nature.com/articles/ncomms14049.

13. La Manno, G., R. Soldatov, A. Zeisel, E. Braun, H. Hochgerner, V. Petukhov, K. Lidschreiber, M. E. Kastriti, P. Lönnerberg, A. Furlan, J. Fan, L. E. Borm, Z. Liu, D. van Bruggen, J. Guo, X. He, R. Barker, E. Sundström, G. Castelo-Branco, P. Cramer, I. Adameyko, S. Linnarsson, and P. V. Kharchenko, 2018. RNA velocity of single cells. *Nature* 560:494–498. http://www.nature.com/articles/s41586-018-0414-6.

14. Melsted, P., A. S. Booeshaghi, L. Liu, F. Gao, L. Lu, K. H. Min, E. da Veiga Beltrame, K. E. Hjörleifsson, J. Gehring, and L. Pachter, 2021. Modular, efficient and constant-memory single-cell RNA-seq preprocessing. *Nature Biotechnology* 39:813–818. http://www.nature.com/articles/s41587-021-00870-2.

15. Ding, J., X. Adiconis, S. K. Simmons, M. S. Kowalczyk, C. C. Hession, N. D. Marjanovic, T. K. Hughes, M. H. Wadsworth, T. Burks, L. T. Nguyen, J. Y. H. Kwon, B. Barak, W. Ge, A. J. Kedaigle, S. Carroll, S. Li, N. Hacohen, O. Rozenblatt-Rosen, A. K. Shalek, A.-C. Villani, A. Regev, and J. Z. Levin, 2020. Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nature Biotechnology* 38:737–746. https://www.nature.com/articles/s41587-020-0465-8.

16. Habib, N., I. Avraham-Davidi, A. Basu, T. Burks, K. Shekhar, M. Hofree, S. R. Choudhury, F. Aguet, E. Gelfand, K. Ardlie, D. A. Weitz, O. Rozenblatt-Rosen, F. Zhang, and A. Regev, 2017. Massively parallel single-nucleus RNA-seq with DroNc-seq. *Nature Methods* 14:955–958. http://www.nature.com/articles/nmeth.4407.

17. Thrupp, N., C. Sala Frigerio, L. Wolfs, N. G. Skene, N. Fattorelli, S. Poovathingal, Y. Fourne, P. M. Matthews, T. Theys, R. Mancuso, B. de Strooper, and M. Fiers, 2020. Single-Nucleus RNA-Seq Is Not Suitable for Detection of Microglial Activation Genes in Humans. *Cell Reports* 32:108189. https://linkinghub.elsevier.com/retrieve/pii/S2211124720311785.

18. Selewa, A., R. Dohn, H. Eckart, S. Lozano, B. Xie, E. Gauchat, R. Elorbany, K. Rhodes, J. Burnett, Y. Gilad, S. Pott, and A. Basu, 2020. Systematic Comparison of High-throughput Single-Cell and Single-Nucleus Transcriptomes during Cardiomyocyte Differentiation. *Scientific Reports* 10:1535. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6992778/.

19. Litviňuková, M., C. Talavera-López, H. Maatz, D. Reichart, C. L. Worth, E. L. Lindberg, M. Kanda, K. Polanski, M. Heinig, M. Lee, E. R. Nadelmann, K. Roberts, L. Tuck, E. S. Fasouli, D. M. DeLaughter, B. McDonough, H. Wakimoto, J. M. Gorham, S. Samari, K. T. Mahbubani, K. Saeb-Parsy, G. Patone, J. J. Boyle, H. Zhang, H. Zhang, A. Viveiros, G. Y. Oudit, O. A. Bayraktar, J. G. Seidman, C. E. Seidman, M. Noseda, N. Hubner, and S. A. Teichmann, 2020. Cells of the adult human heart. *Nature* 588:466–472. https://www.nature.com/articles/s41586-020-2797-4.

20. Filatova, T., N. Popović, and R. Grima, 2022. Modulation of nuclear and cytoplasmic mRNA fluctuations by time-dependent stimuli: Analytical distributions. *Mathematical Biosciences* 347:108828. https://linkinghub.elsevier.com/retrieve/pii/S0025556422000372.

21. Singh, A., and P. Bokes, 2012. Consequences of mRNA Transport on Stochastic Variability in Protein Levels. *Biophysical Journal* 103:1087–1096. https://linkinghub.elsevier.com/retrieve/pii/S0006349512007904.

22. Munsky, B., G. Li, Z. R. Fox, D. P. Shepherd, and G. Neuert, 2018. Distribution shapes govern the discovery of predictive models for gene regulation. *Proceedings of the National Academy of Sciences* 115:7533–7538.

23. Hansen, M. M., R. V. Desai, M. L. Simpson, and L. S. Weinberger, 2018. Cytoplasmic Amplification of Transcriptional Noise Generates Substantial Cell-to-Cell Variability. *Cell Systems* 7:384–397.e6. https://linkinghub.elsevier.com/retrieve/pii/S240547121830317X.

24. Gorin, G., and L. Pachter, 2022. Modeling bursty transcription and splicing with the chemical master equation. *Biophysical Journal* 121:1056–1069. https://www.cell.com/biophysj/fulltext/S0006-3495(22)00104-7.

25. Corrigan, A. M., E. Tunnacliffe, D. Cannon, and J. R. Chubb, 2016. A continuum model of transcriptional bursting. *eLife* 5:e13051. https://elifesciences.org/articles/13051.

26. Coté, A., C. Coté, S. Bayatpour, H. L. Drexler, K. A. Alexander, F. Chen, A. T. Wassie, E. S. Boyden, S. Berger, L. S. Churchman, and A. Raj, 2021. pre-mRNA spatial distributions suggest that splicing can occur post-transcriptionally. Preprint, bioRxiv: 2020.04.06.028092. https://doi.org/10.1101/2020.04.06.028092.

27. Drexler, H. L., K. Choquet, and L. S. Churchman, 2020. Splicing Kinetics and Coordination Revealed by Direct Nascent RNA Sequencing through Nanopores. *Molecular Cell* 77:985–998.e8. https://linkinghub.elsevier.com/retrieve/pii/S1097276519308652.

28. Haque, A., J. Engel, S. A. Teichmann, and T. Lönnberg, 2017. A practical guide to single-cell RNA-sequencing for biomedical research and clinical applications. *Genome Medicine* 9:75. http://genomemedicine.biomedcentral.com/articles/10.1186/s13073-017-0467-4.

29. Gorin, G., and L. Pachter, 2020. Special function methods for bursty models of transcription. *Physical Review E* 102:022409. https://link.aps.org/doi/10.1103/PhysRevE.102.022409.

30. Choubey, S., J. Kondev, and A. Sanchez, 2015. Deciphering Transcriptional Dynamics In Vivo by Counting Nascent RNA Molecules. *PLOS Computational Biology* 11:e1004345.

31. Choubey, S., 2018. Nascent RNA kinetics: Transient and steady state behavior of models of transcription. *Physical Review E* 97:022402.

32. Ali, M. Z., and S. Choubey, 2019. Decoding the grammar of transcriptional regulation from RNA polymerase measurements: models and their applications. *Physical Biology* 16:061001. https://iopscience.iop.org/article/10.1088/1478-3975/ab45bf.

33. Xu, H., S. O. Skinner, A. M. Sokac, and I. Golding, 2016. Stochastic Kinetics of Nascent RNA. *Physical Review Letters* 117:128101. https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.117.128101.

34. Zhang, X., H. Jin, Z. Yang, and J. Lei, 2014. Effects of elongation delay in transcription dynamics. *Mathematical Biosciences and Engineering* 11:1431–1448. http://www.aimspress.com/article/10.3934/mbe.2014.11.1431.

35. Fu, X., H. P. Patel, S. Coppola, L. Xu, Z. Cao, T. L. Lenstra, and R. Grima, 2021. Accurate inference of stochastic gene expression from nascent transcript heterogeneity. Preprint, bioRxiv: 2021.11.09.467882. http://biorxiv.org/lookup/doi/10.1101/2021.11.09.467882.

36. Gorin, G., M. Wang, I. Golding, and H. Xu, 2020. Stochastic simulation and statistical inference platform for visualization and estimation of transcriptional kinetics. *PLOS ONE* 15:e0230736. https://dx.plos.org/10.1371/journal.pone.0230736.

37. Liu, B., S. Yan, Q. Wang, and S. Liu, 2011. Oscillatory expression and variability in p53 regulatory network. *Physica D: Nonlinear Phenomena* 240:259–264. https://linkinghub.elsevier.com/retrieve/pii/S016727891000254X.

38. Barrio, M., K. Burrage, A. Leier, and T. Tian, 2006. Oscillatory Regulation of Hes1: Discrete Stochastic Delay Modelling and Simulation. *PLOS Computational Biology* 2:e117. https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.0020117.

39. Zhang, J., and T. Zhou, 2019. Markovian approaches to modeling intracellular reaction processes with molecular memory. *Proceedings of the National Academy of Sciences* 116:23542–23550. https://pnas.org/doi/full/10.1073/pnas.1913926116.

40. Gedeon, T., and P. Bokes, 2012. Delayed Protein Synthesis Reduces the Correlation between mRNA and Protein Fluctuations. *Biophysical Journal* 103:377–385. https://linkinghub.elsevier.com/retrieve/pii/S0006349512006820.

41. Bhadana, J., A. L. Chanu, M. Z. Malik, and R. K. B. Singh, 2021. Noise and delay can shape distribution functions in stochastic reaction dynamics. *Nonlinear Dynamics* 105:797–811. https://link.springer.com/10.1007/s11071-021-06643-5.

42. Singh, S. N., A. L. Chanu, M. Z. Malik, and R. B. Singh, 2021. Interplay of cellular states: Role of delay as control mechanism. *Physica A: Statistical Mechanics and its Applications* 572:125869. https://linkinghub.elsevier.com/retrieve/pii/S0378437121001412.

43. Bratsun, D., D. Volfson, L. S. Tsimring, and J. Hasty, 2015. Delay-induced stochastic oscillations in gene regulation. *Proceedings of the National Academy of Sciences* 102:14593–14598. https://www.pnas.org/doi/10.1073/pnas.0503858102.

44. Miekisz, J., J. Poleszczuk, M. Bodnar, and U. Foryś, 2011. Stochastic Models of Gene Expression with Delayed Degradation. *Bulletin of Mathematical Biology* 73:2231–2247. http://link.springer.com/10.1007/s11538-010-9622-4.

45. MacDonald, N., 1978. Time Lags in Biological Models, volume 27 of *Lecture Notes in Biomathematics*. Springer Berlin Heidelberg, Berlin, Heidelberg. http://link.springer.com/10.1007/978-3-642-93107-9.

46. Lafuerza, L. F., and R. Toral, 2011. Role of delay in the stochastic creation process. *Physical Review E* 84:021128. https://journals.aps.org/pre/abstract/10.1103/PhysRevE.84.021128.

47. Leier, A., and T. T. Marquez-Lago, 2015. Delay chemical master equation: direct and closed-form solutions. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 471:20150049. https://royalsocietypublishing.org/doi/10.1098/rspa.2015.0049.

48. Jia, T., and R. V. Kulkarni, 2011. Intrinsic Noise in Stochastic Models of Gene Expression with Molecular Memory and Bursting. *Physical Review Letters* 106:058102. https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.106.058102.

49. Cai, X., 2007. Exact stochastic simulation of coupled chemical reactions with delays. *The Journal of Chemical Physics* 126:124108. http://aip.scitation.org/doi/10.1063/1.2710253.

50. Tian, T., K. Burrage, P. M. Burrage, and M. Carletti, 2007. Stochastic delay differential equations for genetic regulatory networks. *Journal of Computational and Applied Mathematics* 205:696–707. https://linkinghub.elsevier.com/retrieve/pii/S0377042706003943.

51. Gupta, C., J. M. López, R. Azencott, M. R. Bennett, K. Josić, and W. Ott, 2014. Modeling delay in genetic networks: From delay birth-death processes to delay stochastic differential equations. *The Journal of Chemical Physics* 140:204108. http://aip.scitation.org/doi/10.1063/1.4878662.

52. Fatehi, F., Y. N. Kyrychko, and K. B. Blyuss, 2020. A new approach to simulating stochastic delayed systems. *Mathematical Biosciences* 322:108327. https://linkinghub.elsevier.com/retrieve/pii/S0025556420300225.

53. Jiang, Q., X. Fu, S. Yan, R. Li, W. Du, Z. Cao, F. Qian, and R. Grima, 2021. Neural network aided approximation and parameter inference of non-Markovian models of gene expression. *Nature Communications* 12:2618. http://www.nature.com/articles/s41467-021-22919-1.

54. Lafuerza, L. F., and R. Toral, 2011. Exact solution of a stochastic protein dynamics model with delayed degradation. *Physical Review E* 84:051121. https://journals.aps.org/pre/abstract/10.1103/PhysRevE.84.051121.

55. MathWorks, T., 2022. MATLAB R2022a Symbolic Math Toolbox. https://www.mathworks.com/products/symbolic.html.

56. Gorin, G., and L. Pachter, 2020. Intrinsic and extrinsic noise are distinguishable in a synthesis – export – degradation model of mRNA production. Preprint, bioRxiv: 2020.09.25.312868. http://biorxiv.org/lookup/doi/10.1101/2020.09.25.312868.

57. Ham, L., R. D. Brackston, and M. P. H. Stumpf, 2020. Extrinsic Noise and Heavy-Tailed Laws in Gene Expression. *Physical Review Letters* 124:108101. https://link.aps.org/doi/10.1103/PhysRevLett.124.108101.

58. Gorin, G., J. J. Vastola, M. Fang, and L. Pachter, 2021. Interpretable and tractable models of transcriptional noise for the rational design of single-molecule quantification experiments. Preprint, bioRxiv: 2021.09.06.459173. https://www.biorxiv.org/content/10.1101/2021.09.06.459173v1.

59. Yao, Z., H. Liu, F. Xie, S. Fischer, R. S. Adkins, A. I. Aldridge, S. A. Ament, A. Bartlett, M. M. Behrens, K. Van den Berge, D. Bertagnolli, H. R. de Bézieux, T. Biancalani, A. S. Booeshaghi, H. C. Bravo, T. Casper, C. Colantuoni, J. Crabtree, H. Creasy, K. Crichton, M. Crow, N. Dee, E. L. Dougherty, W. I. Doyle, S. Dudoit, R. Fang, V. Felix, O. Fong, M. Giglio, J. Goldy, M. Hawrylycz, B. R. Herb, R. Hertzano, X. Hou, Q. Hu, J. Kancherla, M. Kroll, K. Lathia, Y. E. Li, J. D. Lucero, C. Luo, A. Mahurkar, D. McMillen, N. M. Nadaf, J. R. Nery, T. N. Nguyen, S.-Y. Niu, V. Ntranos, J. Orvis, J. K. Osteen, T. Pham, A. Pinto-Duarte, O. Poirion, S. Preissl, E. Purdom, C. Rimorin, D. Risso, A. C. Rivkin, K. Smith, K. Street, J. Sulc, V. Svensson, M. Tieu, A. Torkelson, H. Tung, E. D. Vaishnav, C. R. Vanderburg, C. van Velthoven, X. Wang, O. R. White, Z. J. Huang, P. V. Kharchenko, L. Pachter, J. Ngai, A. Regev, B. Tasic, J. D. Welch, J. Gillis, E. Z. Macosko, B. Ren, J. R. Ecker, H. Zeng, and E. A. Mukamel, 2021. A transcriptomic and epigenomic cell atlas of the mouse primary motor cortex. *Nature* 598:103–110. https://www.nature.com/articles/s41586-021-03500-8.

60. Andrews, T. S., J. Atif, J. C. Liu, C. T. Perciani, X.-Z. Ma, C. Thoeni, M. Slyper, G. Eraslan, A. Segerstolpe, J. Manuel, S. Chung, E. Winter, I. Cirlan, N. Khuu, S. Fischer, O. Rozenblatt-Rosen, A. Regev, I. D. McGilvray, G. D. Bader, and S. A. MacParland, 2022. Single-Cell, Single-Nucleus, and Spatial RNA Sequencing of the Human Liver Identifies Cholangiocyte and Mesenchymal Heterogeneity. *Hepatology Communications* 6:821–840. https://aasldpubs.onlinelibrary.wiley.com/doi/abs/10.1002/hep4.1854.

61. Vastola, J. J., 2021. In search of a coherent theoretical framework for stochastic gene regulation. Ph.D. thesis, Vanderbilt. https://ir.vanderbilt.edu/handle/1803/16646.

62. Gorin, G., 2022. Supporting data for GYP_2022 (10.5281/zenodo.7217195). https://zenodo.org/record/7217195.

63. Iserles, A., and S. MacNamara, 2017. Magnus expansions and pseudospectra of Master Equations. Preprint, arXiv: 1701.02522. http://arxiv.org/abs/1701.02522, arXiv: 1701.02522.

64. Iyer-Biswas, S., F. Hayot, and C. Jayaprakash, 2009. Stochasticity of gene products from transcriptional pulsing. *Physical Review E* 79:031911. https://link.aps.org/doi/10.1103/PhysRevE.79.031911.

65. Virtanen, P., R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, I. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, SciPy 1.0 Contributors, A. Vijaykumar, A. P. Bardelli, A. Rothberg, A. Hilboll, A. Kloeckner, A. Scopatz, A. Lee, A. Rokem, C. N. Woods, C. Fulton, C. Masson, C. Häggström, C. Fitzgerald, D. A. Nicholson, D. R. Hagen, D. V. Pasechnik, E. Olivetti, E. Martin, E. Wieser, F. Silva, F. Lenders, F. Wilhelm, G. Young, G. A. Price, G.-L. Ingold, G. E. Allen, G. R. Lee, H. Audren, I. Probst, J. P. Dietrich,

J. Silterra, J. T. Webber, J. Slavič, J. Nothman, J. Buchner, J. Kulick, J. L. Schönberger, J. V. de Miranda Cardoso, J. Reimer, J. Harrington, J. L. C. Rodríguez, J. Nunez-Iglesias, J. Kuczynski, K. Tritz, M. Thoma, M. Newville, M. Kümmerer, M. Bolingbroke, M. Tartre, M. Pak, N. J. Smith, N. Nowaczyk, N. Shebanov, O. Pavlyk, P. A. Brodtkorb, P. Lee, R. T. McGibbon, R. Feldbauer, S. Lewis, S. Tygier, S. Sievert, S. Vigna, S. Peterson, S. More, T. Pudlik, T. Oshima, T. J. Pingel, T. P. Robitaille, T. Spura, T. R. Jones, T. Cera, T. Leslie, T. Zito, T. Krauss, U. Upadhyay, Y. O. Halchenko, and Y. Vázquez-Baeza, 2020. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods* 17:261–272. http://www.nature.com/articles/s41592-019-0686-2.

66. Gillespie, D. T., 1976. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics* 22:403–434. https://linkinghub.elsevier.com/retrieve/pii/0021999176900413.

67. Harper, C. V., B. Finkenstädt, D. J. Woodcock, S. Friedrichsen, S. Semprini, L. Ashall, D. G. Spiller, J. J. Mullins, D. A. Rand, J. R. E. Davis, and M. R. H. White, 2011. Dynamic Analysis of Stochastic Transcription Cycles. *PLoS Biology* 9:e1000607. https://dx.plos.org/10.1371/journal.pbio.1000607.

68. Herbach, U., 2019. Stochastic Gene Expression with a Multistate Promoter: Breaking Down Exact Distributions. *SIAM Journal on Applied Mathematics* 79:1007–1029. https://epubs.siam.org/doi/10.1137/18M1181006.

69. Stinchcombe, A. R., C. S. Peskin, and D. Tranchina, 2012. Population density approach for discrete mRNA distributions in generalized switching models for stochastic gene expression. *Physical Review E* 85:061919. https://link.aps.org/doi/10.1103/PhysRevE.85.061919.

70. Pressé, S., K. Ghosh, J. Lee, and K. A. Dill, 2013. Principles of maximum entropy and maximum caliber in statistical physics. *Reviews of Modern Physics* 85:1115–1141. https://link.aps.org/doi/10.1103/RevModPhys.85.1115.

S1 THEORETICAL FOUNDATIONS

We extend the foundations in (24) to include interconversion between promoter states. We assume there are n species, with microstates $\mathbf{m} \in \mathbb{N}_0^n$, and N promoter states, with microstates $j \in \{0, ..., N-1\}$. The species are one-indexed; a reaction from a species to "0" corresponds to degradation. The promoter states are zero-indexed, to reflect the common notation wherein a "0" state is inactive whereas the "1" state is active. A microstate is defined by the instantaneous variables $j, \mathbf{m}$ and t: a particular combination of promoter state, molecular counts, and a time. To define the process, we must also specify $\mathbf{w}$ and $\mathbf{P}^0$, the promoter state probabilities and the RNA distributions at $t = 0$.

We assume the system contains the following Markovian reactions:

- Interconversion between promoter states.
- Synthesis of new molecules.
- Interconversion of molecular species.
- Degradation of molecular species.

The coupled generating function ODEs take the following form in terms of $\mathbf{Y} := [Y_0, ..., Y_{N-1}]^T$:

$$\frac{d\mathbf{Y}}{dt} = H^T\mathbf{Y} + (A\mathbf{u}) \odot \mathbf{Y} + (C\mathbf{u})^T \left(\nabla \sum_j Y_j \right) \tag{10}$$

$H \in \mathbb{R}^N$ is the matrix encoding the continuous-time Markov chain (CTMC) governing the promoter states, such that $\sum_k H_{jk} = 0$ for all j. $\mathbf{u} \in \mathbb{C}^n$ is the vector of PGF generating function arguments corresponding to the molecular degrees of freedom, shifted down by 1, i.e., $u_i = x_i - 1$. $A \in (\mathbb{R}_{\geq 0})^{N \times n}$ is the matrix describing the production of each species at each state, i.e., $A_{ji} = \alpha_{j,i}$ is the rate of production of transcript i while the gene is in state j. $C \in \mathbb{R}^{n \times n}$ is the matrix encoding the CTMC governing the transitions between molecular species, such that $C_{ik} = c_{ik}$ is the rate of transitioning from species i to species k, whereas the total efflux rate $C_{ii} = -\sum_{k \neq i} c_{ik} - c_{i0}$, where c_{i0} is the degradation rate of species i. The third term on the right-hand side is a complex scalar. The symbol $\odot$ denotes the Hadamard, or entrywise, product of two matrices of identical dimensions. By applying the method of characteristics, the PDE reduces to an ODE:

$$\frac{d\mathbf{Y}}{dt} = H^T\mathbf{Y} + (A\mathbf{U}) \odot \mathbf{Y}, \tag{11}$$

where $\mathbf{U}$ are the system's characteristics $[U_1(\mathbf{u}, s), ..., U_n(\mathbf{u}, s)]^T = \mathbf{U}(\mathbf{u}, s)$.

It remains to specify the initial conditions. The generating function, assuming a homogeneous initial condition with $m_i = 0$ at $t = 0$ for all i, is obtained by exploiting the stationary distribution of the gene state degrees of freedom, $\mathbf{h} = \ker H^T$.

$$Y(\mathbf{u}, t) = \sum_j Y_j(\mathbf{u}, t) \frac{w_j}{h_j} := (\mathbf{w} \oslash \mathbf{h})^T \mathbf{Y}(\mathbf{u}, t), \tag{12}$$

where the vector $\mathbf{w}$ encodes the categorical distribution over promoter state abundances at $t = 0$, whereas $\oslash$ denotes the (entrywise) Hadamard division operator. This result generalizes and formalizes the dependence on f_1 and f_2 reported on p. 232 of (61): these non-negative coefficients are not arbitrary, but reflect the initial condition. It also generalizes Eq. B.4 of (20) for a non-bidiagonal C, but imposes the restriction of a time-independent H. We do not consider the case where H is time-dependent, but we speculate that the governing ODE system (the non-homogeneous form of Equation 11) is identical for arbitrary time dependence in H and A, and any time-inhomogeneity would affect only the $(\mathbf{w} \oslash \mathbf{h})$ term.

The system of ODEs in Equation 11 likewise requires an initial condition. This initial condition is simply $\mathbf{h}$ if we are interested in obtaining $G(\mathbf{u}, t)$ or $\mathbf{h} \odot \boldsymbol{\delta}_j$ if we are interested in obtaining $G_j(\mathbf{u}, t)$. The vector $\boldsymbol{\delta}_j \in \{0, 1\}^N$ denotes the Kronecker delta, such that $\delta_j = 1$ and $\delta_{k \neq j} = 1$.

Finally, it remains to treat any potential initial conditions due to molecules present in the system at $t = 0$. The molecular initial conditions are fully specified by a set of N n-variate distributions of molecules *conditional* on gene state, and represented by $\mathbf{G}^0 := \left[G_0^0(\mathbf{u}), ..., G_{N-1}^0(\mathbf{u})\right]^T = \mathbf{G}^0(\mathbf{u})$; we use the shifted coordinates $\mathbf{u}$ for convenience as elsewhere. If we denote the *h*omogeneous solution of Equation 11 by $\mathbf{G}^h$, correcting for the initial conditions merely requires evaluating the Hadamard product

$$\mathbf{Y}^h \odot \mathbf{G}^0(\mathbf{U}(\mathbf{u}, t)). \tag{13}$$

Putting it all together, the following schema holds:

$$\begin{aligned} G(\mathbf{u},t|\mathbf{w},\mathbf{G}^0,0) &= (\mathbf{w}\oslash\mathbf{h})^T\left[\mathbf{Y}^h(\mathbf{u},t)\odot\mathbf{G}^0(\mathbf{U}(\mathbf{u},t))\right], \text{ where} \\ \frac{d\mathbf{Y}^h}{dt} &= H^T\mathbf{Y}^h + (A\mathbf{U})\odot\mathbf{Y}^h, \text{ such that } \mathbf{Y}^h(\mathbf{u},0)=\mathbf{h}. \end{aligned} \tag{14}$$

Further, if we wish to obtain $G_j(\mathbf{u},t|\mathbf{w},\mathbf{G}^0,0)$, which can be inverted to yield $P_j(\mathbf{m},t|\mathbf{w},\mathbf{G}^0,0)$, we can use the formula in Equation 14 with $\mathbf{Y}^h(\mathbf{u},0)=\mathbf{h}\odot\boldsymbol{\delta}_j$. This non-normalized distribution can be converted to a conditional distribution through dividing by $P(j)=\left(e^{tH^T}\mathbf{h}\right)_j$.

S1.1 Solution through the lens of matrix ODEs

In practical terms, Equation 14 is the solution: the ODE can be easily plugged into a standard Runge–Kutta-type solver to obtain the generating function. However, we can, in principle, write down the ODE in more compact form:

$$\begin{aligned} \mathcal{D}(t) &= H^T + \operatorname{diag} A\mathbf{U} = H^T + (A\mathbf{U}\mathbf{1}^T)\odot I \\ \frac{d\mathbf{Y}^h}{dt} &= \mathcal{D}(t)\mathbf{Y}^h, \end{aligned} \tag{15}$$

where I is the dimension-N identity matrix and $\mathcal{D}$ is the matrix that encodes system *d*ynamics. We observe:

$$\mathcal{D}(t_1)\mathcal{D}(t_2) = \left[H^T + \operatorname{diag} A\mathbf{U}(t_1)\right]\left[H^T + \operatorname{diag} A\mathbf{U}(t_2)\right] \neq \mathcal{D}(t_2)\mathcal{D}(t_1) \tag{16}$$

in general; the product of diagonal matrices commutes, but the product of H^T and a diagonal matrix does not (except in the trivial case $N=1$). Therefore, $\mathbf{Y}^h$ cannot be represented by a finite matrix exponential, but can be approximated by a Magnus series (63).

S1.2 Solution through the lens of special functions

Certain steady-state solutions can be obtained by appealing to the theory of special functions. The stationary PGF of the solution to the standard telegraph model (with $N=2$ and $n=1$) is given by the confluent hypergeometric function ${}_1F_1$ (64); transient solutions have an analogous form. The combinatorial extension of the telegraph model (with $N=2^p$, $p\in\mathbb{N}$), whose state transitions are given by the Hamming graph, has a stationary PGF given by the p-fold product of confluent hypergeometric functions (8). Certain systems with $n=1$ and an irreversible refractory state dynamics have solutions given by the generalized hypergeometric function ${}_{N-1}F_{N-1}$ (7). Thus, in principle, solutions may be obtained by casting the series of N ODEs in Equation 14 into a single order-N ODE, then solving it using the properties of special functions. However, this tends to be considerably more involved than the original problem, both due to the complexity of manipulating hypergeometric functions and the challenges of actually evaluating them. Further, although hypergeometric representations are available for $n=1$, the case of $n>1$ does not even afford a formal solution. We have found the ODE approach yields a satisfactory combination of numerical stability and robustness for most purposes.

S2 SOLUTIONS TO TWO-STAGE BURSTY SYSTEMS

S2.1 Bursty Markovian

The steady state of the system given in Equation 8 has been previously reported (21) and has the log-PGF

$$\begin{aligned} U_1^M(s;\beta,\gamma) &= u_2\frac{\beta}{\beta-\gamma}e^{-\gamma s} + \left(u_1 - u_2\frac{\beta}{\beta-\gamma}\right)e^{-\beta s} \\ \phi^M &= \alpha\int_0^\infty\left[\frac{1}{1-bU_1^M(s;\beta,\gamma)} - 1\right]ds, \end{aligned} \tag{17}$$

where U_1^M is the characteristic of the *M*arkovian system.

S2.2 Bursty with delayed efflux

The system given in Equation 7 is governed by the following equations:

$$\begin{aligned} U_2^D &= u_2 \mathbb{I}(s < \tau) \\ \frac{dU_1^D}{ds} &= \beta \left(U_2^D - U_1^D \right) \\ U_1^D(s;\beta,\tau) &= u_1 e^{-\beta s} + u_2 \left(1 - e^{-\beta s}\right) - u_2 \left(1 - e^{-\beta(s-\tau)}\right) \mathbb{I}(s > \tau) \end{aligned} \tag{18}$$

where $\mathbb{I}$ is the indicator function. This can be converted to a piecewise constant representation:

$$\begin{aligned} U_1^D(s;\beta,\tau) &= u_2 + (u_1 - u_2)e^{-\beta s} \text{ whenever } s \in (0,\tau) \text{ and} \\ &= (u_1 - u_2)e^{-\beta s} + u_2 e^{-\beta(s-\tau)} \\ &= \left(u_2 + (u_1 - u_2)e^{-\beta\tau}\right) e^{\beta\tau} e^{-\beta s} \text{ whenever } s \in (\tau,\infty). \end{aligned} \tag{19}$$

Defining $U(s;\beta) := u_2 + (u_1 - u_2)e^{-\beta s}$ for convenience and integrating:

$$\begin{aligned} \phi^D &= \alpha \int_0^\tau \left[\frac{1}{1 - bU_1^D(s;\beta,\tau)} - 1 \right] ds + \alpha \int_\tau^\infty \left[\frac{1}{1 - bU_1^D(s;\beta,\tau)} - 1 \right] ds \\ &= \frac{\alpha/\beta}{1 - bu_2} \ln\left(\frac{bU(\tau;\beta) - 1}{bu_1 - 1} \right) + \alpha\tau \frac{bu_2}{1 - bu_2} - \frac{\alpha}{\beta} \ln\left(1 - bU(\tau;\beta)\right) \end{aligned} \tag{20}$$

S2.3 Bursty with delayed splicing

The system given in Equation 9 supposes that each molecule of nascent RNA is retained in the system for duration τ. This is model is typical for studies concerned with modeling submolecular details of transcriptional elongation (33). This system is governed by the following characteristics:

$$\begin{aligned} U_2^D &= u_2 e^{-\gamma s} \\ U_1^D(s;\tau,\gamma) &= u_1 \mathbb{I}(s < \tau) + u_2 e^{-\gamma(s-\tau)} \mathbb{I}(s > \tau) \end{aligned} \tag{21}$$

Upon plugging the characteristics into the burst size MGF and integrating, we find:

$$\begin{aligned} \phi^D &= \alpha \int_0^\tau \left[\frac{1}{1 - bU_1^D(s;\gamma,\tau)} - 1 \right] ds + \alpha \int_\tau^\infty \left[\frac{1}{1 - bU_1^D(s;\gamma,\tau)} - 1 \right] ds \\ &= \frac{\alpha\tau b u_1}{1 - bu_1} - \frac{\alpha}{\gamma} \ln(1 - bu_2), \end{aligned} \tag{22}$$

i.e., the joint distribution is merely the product of two independent marginal distributions. This result can be derived by appealing to the bursty limit of the argument in Section 14.4 in the supplementary material of (33).

S3 DATA ANALYSIS

To compare the biophysical mechanisms consistent with single-cell and single-nucleus datasets, we used mouse cortex data generated by Allen Institute for Brain Science ("Allen data") (59) and human liver data generated by Toronto's University Health Network ("Andrews data") (60). The datasets analyzed for this study are reported in Table S1.

S3.1 Allen data pre-processing

The Allen data did not contain technical replicates, i.e., a single mouse tissue processed using two single-cell and single-nucleus technologies. In lieu of this, we chose two libraries from female mice: B08, obtained from donor 457911, and A02, obtained from donor 427311. The raw single-cell FASTQs were obtained from https://data.nemoarchive.org/biccn/grant/u19_zeng/zeng/transcriptome/scell/10x_v3/mouse/raw/M0p/; the annotations were obtained from https://data.

Table S1: Dataset cell counts

Source	Reference	Donor	Dataset	Cell type	Cells
Allen	(59)	457911	B08	Glutamatergic	5352
Allen	(59)	457911	B08	GABAergic	789
Allen	(59)	427311	A02	Glutamatergic	6731
Allen	(59)	427311	A02	GABAergic	1253
Andrews	(60)	4	C72 sc	CentralHep	1567
Andrews	(60)	4	C72 sc	PortalHep	4421
Andrews	(60)	4	C72 sc	InterHep	991
Andrews	(60)	4	C72 sn	CentralHep	2885
Andrews	(60)	4	C72 sn	PortalHep	2862
Andrews	(60)	4	C72 sn	InterHep	1879

nemoarchive.org/biccn/grant/u19_zeng/zeng/transcriptome/scell/10x_v3/mouse/processed/analysis/10X_cells_v3_AIBS/. The raw single-nucleus FASTQs were obtained from https://data.nemoarchive.org/biccn/grant/u19_zeng/zeng/transcriptome/sncell/10x_v3/mouse/raw/MOp/; the annotations were obtained from https://data.nemoarchive.org/biccn/grant/u19_zeng/zeng/transcriptome/sncell/10x_v3/mouse/processed/analysis/10X_nuclei_v3_AIBS/.

For quantification, we used a pre-built mm10 reference genome released by 10x Genomics, obtained from https://support.10xgenomics.com/single-cell-gene-expression/software/downloads/latest, to generate cDNA and intron references using *kallisto|bustools* 0.26.0 (`kb ref` with option `--lamanno`). Both libraries were produced using the 10x Genomics v3 chemistry. We pseudoaligned the B08 and B02 libraries using the 10x v3 cell barcode whitelist and the default *kallisto|bustools* filter (`kb count` with options `--lamanno`, `-x 10xv3`, and `--filter bustools`) to generate spliced and unspliced count matrices. The data were further filtered, excluding all cells with too few total (spliced + unspliced) UMIs, with a threshold of 10^4 for B08 and 8×10^3 for A02. These thresholds were chosen manually based on knee plots.

To choose cells for analysis by *Monod*, we identified cell barcodes which were (1) assigned to either the GABAergic or glutamatergic cell type according to annotations, (2) passed the *kallisto|bustools* filter and knee plot filters, and (3) did not belong to one of the excluded rare cell subtypes (Scng or L6 IT Car3). This procedure yielded the cell numbers reported in the final column of Table S1.

S3.2 Andrews data pre-processing

The Andrews data contained technical replicates, generated using different sequencing chemistries. We analyzed data for donor 4 (`C72_reseq` for single-cell, here referred to as "C72 sc"; `C72_TST` for single-nucleus, here referred to as "C72 sn"). The single-cell data were obtained in the Sequence Read Archive format (experiment SRX12509405, runs SRR16227578–83) and converted to paired-end FASTQ files with *fasterq-dump* 2.11.2 (with options `--include-technical` and `-S`). The single-nucleus data were obtained in the originally submitted BAM format (experiment SRX12509406, run SRR16227584) and converted to paired-end FASTQ files with the 10x Genomics software *bamtofastq* 1.3.1. The metadata were obtained from the Gene Expression Omnibus (series GSE185477, file `GSE185477_Final_Metadata.txt.gz`).

For quantification, we used a pre-built GRCh38 reference genome released by 10x Genomics, obtained from https://support.10xgenomics.com/single-cell-gene-expression/software/downloads/latest, to generate cDNA and intron references using *kallisto|bustools* 0.26.0 (`kb ref` with option `--lamanno`). The single-cell library was produced using the 10x v2 chemistry, whereas the single-nucleus library was produced using the v3 chemistry. To simplify the procedure of extracting and storing the count data, we used the lists of barcodes present in each sample's cell type annotations as the whitelists for pseudoalignment. The *kallisto|bustools* was run using the `--lamanno` option and the appropriate `-x` arguments for each dataset. Due to the whitelist definition, we did not use the default filter. Instead, we excluded all cells with too few total UMIs, with a threshold of 8×10^2 for single-cell and 2×10^3 for single-nucleus, and too many total UMIs, with corresponding thresholds of 10^4 and 5×10^4. These thresholds were chosen manually based on knee plots.

To choose cells for analysis by *Monod*, we identified cell barcodes which were (1) assigned to pericentral (CentralHep), periportal (PortalHep), or interzonal (InterHep) hepatocyte cell types according to annotations and (2) passed the knee plot filters. This procedure yielded the cell numbers reported in the final column of Table S1.

S3.3 *Monod* analysis

The analysis was performed using *Monod* 0.2.4.0. For each species, we selected 3,000 genes with moderate to high expression in all datasets. This procedure imposed a set of constraints (each species' mean copy number > 0.01, each species' maximum copy number > 3 and < 400) and computed the number of datasets in which the criterion was met. The top 3,000 genes were identified by choosing the top quantiles of this metric, using random non-replacement sampling to break ties.

Next, we fit all of the models to the data. The procedure used gradient descent to minimize the Kullback-Leibler divergence (KLD) between model and data, and output the maximum likelihood estimates (MLEs) for the parameters. To compute the proposed probability distributions for the KLD, we evaluated the generating functions on a two-dimensional grid of microstates, with the lower bound set to zero and the upper bound set at 10 molecules more than the observed maximum of each species. The generating function of the bursty Markovian model, which requires numerical integration, was evaluated by order-60 Gaussian quadrature, implemented through `integrate.fixed_quad` in *SciPy* (65), evaluated on the interval $[0, T]$, where $T = 10\left(1 + \beta^{-1} + \gamma^{-1}\right)$ to ensure the system reached equilibrium. For the bursty models, α was set to unity with no loss of generality at steady state. For the extrinsic noise model, θ, the scale parameter of the gamma distribution, was set to unity.

Gradient descent was implemented using L-BFGS-B, a constrained optimization algorithm, implemented through the *SciPy* function `optimize.minimize` (65). The optimization was performed in a 3-dimensional space of $\log_{10}$ parameter values, with domain of $(-2, 3.1)$ for the burst sizes and gamma shape parameters and $(-1.8, 3.5)$ for β, γ, and τ^{-1}. These parameters were chosen based on the parameter distributions seen in Figure S5 of (10). Optimization ran for a maximum of 15 steps, with 5 independent trials. The first trial was initialized at the method of moments (MoM) estimate. The method of moments estimates for the bursty Markovian models and the extrinsic models are reported in the supplement of (11). The MoM estimate for the delayed efflux model was identical to that of the bursty Markovian model, with τ^{-1} taking the place of γ. The MoM estimate for the delayed splicing model was identical to that of the bursty Markovian model, with b corrected by a factor of $\frac{1}{2}$ and τ^{-1} taking the place of β. These estimates follow directly from the results in Table 1.

The log-likelihood ratios visualized in Figure 3 were evaluated by computing the proposal PMFs as above, then computing the difference between the log-likelihoods of the data under the alternative model and under the bursty model:

$$\ln \mathrm{LR}_{j,\mu} = \ln \mathcal{L}(\mathcal{D}_j|\hat{\Theta}_{j,\mu}) - \ln \mathcal{L}(\mathcal{D}_j|\hat{\Theta}_{j,\mathrm{bursty}}) \tag{23}$$

where μ is the alternative model type (delayed-efflux, extrinsic, delayed-splicing), j is the gene index, $\mathcal{D}_j$ is the dataset (spliced and unspliced copy numbers in a particular cell type) for gene j, $\hat{\Theta}_{j,\mu}$ is the vector of parameter MLEs under the model, and $\mathcal{L}$ is the data likelihood.

Log-likelihood ratios with magnitude > 100 for single-cell and > 50 for single-nucleus data were omitted as potential signatures of failure to converge to a satisfactory MLE. Kernel density estimates in Figure 3b were generated from the remaining $\ln \mathrm{LR}_j$ values using the SciPy class `stats.gaussian_kde`, with bandwidth set to 0.01 to avoid oversmoothing.

S4 MULTI-STATE SYSTEM PARAMETERS

The parameters for the system in Figure 2a are given in Table S2. The global parameters are $N = 4$ and $n = 4$. k_{ij} denotes the rate of transition from promoter state i to promoter state j. $\alpha_{j,i}$ denotes the rate of production of transcription $\mathcal{T}_i$ at promoter state j. c_{ik} denotes the rate of conversion of $\mathcal{T}_i$ to $\mathcal{T}_k$. c_{i0} denotes the rate of degradation of $\mathcal{T}_i$. r_i denotes the total efflux rate $\sum_k c_{ik}$. τ_i denotes the residence time for deterministically delayed conversion. We generated realizations of trajectories for 10,000 cells, simulated until $t = 10$.

The initial conditions are outlined in Table S3. The first row reports the probability of the system starting in each state. The following rows encode the distributions of each species at $t = 0$, conditional on starting in each state. D-x corresponds to a deterministic initial condition with x molecules, whereas P-μ corresponds to a Poisson initial condition with an average of μ molecules.

Table S2: Multi-state model parameters

Parameter	Value
k_{01}	0.75
k_{10}	0.5
k_{02}	1
k_{20}	0.1
k_{13}	0.15
k_{31}	2
$\alpha_{0,1}$	5.2
$\alpha_{1,2}$	9.2
c_{23}	1.8
c_{24}	2.5
c_{30}	0.8
τ_1	1.1
τ_4	2.1

Table S3: Multi-state model initial conditions

	State 0	State 1	State 2	State 3
Probability	0.24	0.1	0.3	0.36
Species 1	D-0	P-10	D-0	P-2.3
Species 2	D-2	P-3.5	D-1	P-3
Species 3	D-0	P-2	D-2	P-5
Species 4	D-21	P-1.1	D-3	P-0.2

The system is described by the following characteristics:

$$
\begin{aligned}
U_4 &= u_4 \mathbb{I}(s < \tau_4) \\
U_3 &= u_3 e^{-c_{30} s} \\
U_2 &= u_2 e^{-r_2 s} \\
&\quad + u_3 \frac{c_{23}}{r_2 - c_{30}} \left[e^{-c_{30} s} - e^{-r_2 s} \right] \\
&\quad + u_4 \frac{c_{24}}{r_2} \left[\mathbb{I}(s < \tau_4) + \mathbb{I}(s > \tau_4) e^{-r_2 (s - \tau_4)} - e^{-r_2 s} \right] \\
U_1 &= u_1 \mathbb{I}(s < \tau_1) \\
&\quad + u_2 \mathbb{I}(s > \tau_1) e^{-r_2 (s - \tau_1)} \\
&\quad + u_3 \frac{c_{23}}{r_2 - c_{30}} \mathbb{I}(s > \tau_1) \left[e^{-s (c_{30} - \tau_1)} - e^{-r_2 (s - \tau_1)} \right] \\
&\quad + u_4 \frac{c_{24}}{r_2} \left[\mathbb{I}(s \in (\tau_1, \tau_4)) - \mathbb{I}(s > \tau_1) e^{-r_2 (s - \tau_1)} + \mathbb{I}(s > \tau_1 + \tau_4) e^{-r_2 (s - \tau_1 - \tau_4)} \right]
\end{aligned}
\tag{24}
$$

S4.1 Numerical methods

To compute the analytical solutions, we applied a standard Runge-Kutta integration method, implemented using `integrate.rk45` in the *SciPy* package (65).

To simulate the delayed system, we implemented a custom algorithm that introduces three modifications to Gillespie's stochastic simulation algorithm (66). The Markovian form of the algorithm takes the following form:

1. Initialize system at time $t = t_0$ and state $x = x_0$.
2. Compute instantaneous reaction rates of the μth reaction, $\text{flux}_\mu(x)$ and the net state efflux rate, $\text{flux}(x) = \sum_\mu \text{flux}_\mu(x)$.
3. Compute sojourn time $t_s = \frac{1}{\text{flux}(x)} \log \frac{1}{u_1}$ and reaction index, μ, such that $\sum_{\nu=1}^{\mu-1} < u_2 \text{flux}(x) \leq \sum_{\nu=1}^{\mu}$, where u_1 and u_2 are random variables uniformly distributed over $[0, 1]$.
4. Advance system to time $t \leftarrow t + t_s$ and state $x \leftarrow x + \nu_\mu$.
5. Return to step 2 or terminate simulation.

The first modification treats removal events of delayed species, i.e, transcripts that undergo reactions with deterministic waiting times. Two empty queues are initialized: one for times and one for reaction indices. Then, if the reaction index determined in step 3 is the creation of a delayed species, the queues are populated with the time and reaction index of the deterministic removal of that species in the correct order. If the system is initialized with delayed species, the queues of times and reaction indices must be pre-defined accordingly. For simplicity, we always assume that existing delayed species were created at $t = 0$.

The second modification alters the calculation of flux, specifically accounting for the contributions of species that don't yet exist, but will after some delay. The total flux, $\text{flux}(x)$, is computed at each queued reaction event, and sojourn time corresponding to the random flux generated by $\log \frac{1}{u_1}$ is found analytically. The computation of the sojourn time is essentially equivalent to the direct method outlined in (49).

The third modification ensures that all reactions happen in the correct order. After step 3, the reaction time and event are stored, and before advancing the system in step 4, all queued reactions that are to happen before the stored reaction event are sequentially applied and stored. The resulting ordered list is then converted into system times and states.

The algorithm is implemented in Python, and is designed to simulate the class of systems outlined in Section S1. With minor modifications, it can be used to simulate multi-molecular reactions, catalysis, feedback schema, or more general waiting time distributions, analogously to (49).

S5 METHODOLOGICAL EXTENSIONS

In the current section, we draw useful connections to theory we previously reported in (10), contextualize the work with respect to standard tools of DCME analysis, and discuss potential extensions.

S5.1 Generating function identifiability

Consider a system with n species, described by a generating function $G(\mathbf{u})$. Some of these species may not be mutually identifiable. Suppose there are $\mathcal{n}$ identifiable species, where each species $i \in \{1, ..., n\}$ may be assigned to each category $\mathcal{i}$ with

probability $p_{i,\imath}$ for each molecule. If this assignment process is independent, the process induces a collection of n generating functions $\mathbf{G}^s(\boldsymbol{u})$, such that each component is the PGF of a categorical distribution (i.e., a multinomial distribution with a single trial):

$$G_i^s(\boldsymbol{u}) = \sum_{\imath=0}^{n} (u_\imath + 1) p_{i,\imath}, \tag{25}$$

where $p_{i,0}$ is defined as $1 - \sum_\imath p_{i,\imath}$, the probability of assigning species i to none of the observable species, and losing it. Evidently,

$$G_i^s(\boldsymbol{u}) - 1 = \sum_{\imath=0}^{n} u_\imath p_{i,\imath}. \tag{26}$$

From standard properties of generating functions, the distribution resulting from "filtering" G through $\mathbf{G}^s$ is a function composition:

$$G(\boldsymbol{u}) = G(\mathbf{G}^s(\boldsymbol{u}) - \mathbf{1}), \tag{27}$$

where $\mathbf{1}$ denotes the length-n vector of ones.

S5.2 Erlang-distributed delays

Consider the case of a system with bursty transcription, Markovian degradation, and Erlang-distributed waiting time for splicing, with shape q and mean waiting time τ. First, set up a system with $q + 1$ species:

$$\varnothing \xrightarrow{\alpha} B \times \mathcal{T}_1 \xrightarrow{q/\tau} ... \mathcal{T}_q \xrightarrow{q/\tau} \mathcal{T}_{q+1} \xrightarrow{\gamma} \varnothing \tag{28}$$

The characteristic corresponding to $\mathcal{T}_{q+1}$ is simply $u_{q+1}e^{-\gamma s}$. By somewhat tedious computation, amounting to repeatedly solving ordinary differential equations of the form $\frac{dU_i}{dt} = \frac{q}{\tau}(U_{i+1} - U_i)$, we find that the characteristic corresponding to $\mathcal{T}_1$ takes the following form:

$$U_1 = u_{q+1}e^{-\gamma s} \times \left(\frac{\lambda}{\lambda - \gamma}\right)^q + e^{-\lambda s} \sum_{i=0}^{q-1} \frac{(u_{i+1}\lambda s)^i}{i!} - u_{q+1}e^{-\lambda s} \frac{(\lambda)^q}{(\lambda - \gamma)^q} \sum_{i=0}^{q-1} \frac{s^i(\lambda - \gamma)^i}{i!}, \tag{29}$$

where λ is defined as q/τ.

To aggregate $\mathcal{T}_1, ..., \mathcal{T}_q$ into a single species $\mathcal{X}_1$, and represent $\mathcal{T}_{q+1}$ as the downstream species $\mathcal{X}_2$, we evaluate the arguments at $u_1 = ... = u_q = u_1$ and $u_{q+1} = u_2$. This yields:

$$U_1 = u_2 e^{-\gamma s} \times \left(\frac{\lambda}{\lambda - \gamma}\right)^q + u_1 \frac{\Gamma(q, \lambda s)}{\Gamma(q)} - u_2 e^{-\gamma s} \frac{\lambda^q}{(\lambda - \gamma)^q} \frac{\Gamma(q, (\lambda - \gamma)s)}{\Gamma(q)}. \tag{30}$$

To evaluate the PGF, we plug U_1 into the burst distribution MGF and integrate.

S5.3 Linear chain trick

The same approach can be used to arrive at the solution for deterministically delayed systems, commonly known as the "linear chain trick." Taking the limit as $q \to \infty$ and exploiting the properties of special functions:

$$\begin{aligned} \lim_{q\to\infty} U_1 &= u_2 e^{-\gamma s} e^{\gamma \tau} + u_1 \mathbb{I}(s < \tau) - \lim_{q\to\infty} u_2 e^{-\gamma s} e^{\gamma \tau} \frac{\Gamma(q, (q/\tau - \gamma)s)}{\Gamma(q)} \\ &= u_2 e^{-\gamma(s-\tau)} + u_1 \mathbb{I}(s < \tau) - u_2 e^{-\gamma(s-\tau)} \mathbb{I}(s < \tau) \\ &= u_1 \mathbb{I}(s < \tau) + u_2 e^{-\gamma(s-\tau)} \mathbb{I}(s > \tau). \end{aligned} \tag{31}$$

To obtain the PGF, we plug $\lim_{q\to\infty} U_1$ into the burst distribution MGF and integrate, eliding the functional analysis justification for interchanging the limit and the integral. This reproduces the results in Section S2.3, albeit with considerably more effort.

S5.4 General waiting time distributions

The same approach may be used to evaluate systems with more generic waiting time distributions. For example, the characteristic appropriate for a molecule that remains in the system for time τ, then exhibits Markovian efflux is immediately implied by the results of the previous section:

$$U = u\left[\mathbb{I}(s < \tau) + e^{-\gamma(s-\tau)}\mathbb{I}(s > \tau)\right], \tag{32}$$

obtained by evaluating the previous characteristic U_1 at $u_1 = u_2 = u$. This holds for any combination of Markovian and deterministic delays.

Throughout the current section, we have considered delays only in downstream molecular processing. However, studies have indicated that non-Markovian refractory periods can occur in promoter state transitions (67). If the refractory period can be effectively described by a series of Markovian processes (e.g., if the waiting time is Erlang), such systems can be solved by defining "internal" inactive states and adding their P_j, obtained through Equation 14, in the spirit of (68). Stinchcombe et al. treated the case of arbitrary switching time distributions (69); however, this approach requires a degree of ingenuity and is fairly challenging to formalize in the language of generating functions.

The case of more general waiting times was explored in considerable detail by Zhang and Zhou (39), albeit with somewhat more emphasis on theoretical foundations and implications for noise buffering than here. Essentially, arbitrary waiting time distributions may be encoded by properly exploiting their statistical structure; for example, it is immediately evident that the characteristic in Equation 32 is simply the survival function (complementary cumulative distribution function) of each molecule, evaluated at its birth. In the same vein, the U_2^D terms in Equations 18 and 21 are the survival functions of the degenerate and exponential distributions, whereas the u_2 term of the characteristic in Equation 17 is the survival function of the two-parameter hypoexponential distribution.

It appears that more general waiting time distributions can be incorporated in the characteristic framework by appropriately manipulating their survival functions. We forgo detailed discussion of this extension, as it is less suitable to automation and tends to lead to intractable integrals. Further, Markovian and deterministic delays are relatively easy to motivate from first principles by appealing to memorylessness or perfect molecular memory, respectively (either maximum or zero entropy in the language of statistical physics (70)). On the other hand, "intermediate" non-Markovian cases require a specific hypothesis to instantiate a functional form for the waiting time distribution.

The simulation algorithm outlined in Section S4.1 generalizes to arbitrary waiting time distributions, and may be used to investigate the impact of different assumptions even when analytical solutions are unavailable. Further, the tools outlined here can be used to explore some pathological cases, such as the loss of stationarity due to particularly ill-behaved waiting time distributions. For example, if the RNA production process is constitutive with unity transcription rate and the waiting time distribution is standard half-Cauchy, the characteristic and log-PGF take the following form:

$$\begin{aligned} U &= u\left[1 - \frac{2}{\pi}\tan^{-1} s\right] \\ \phi &= \frac{u}{\pi}\left[\pi t - 2t\tan^{-1}(t) + \ln(1+t^2)\right], \end{aligned} \tag{33}$$

where ϕ is immediately recognizable as the log-PGF of a Poisson distribution, albeit one that does not possess a limit as $t \to \infty$. This implies that the $M/G/\infty$ queue with half-Cauchy service times has no stationary distribution. In contrast, if the waiting time distribution is standard Pareto, with support on $[1, \infty)$, we find:

$$\begin{aligned} U &= us^{-\alpha} \\ \phi &= \frac{u}{\alpha - 1}\left(1 - t^{1-\alpha}\right) \\ \text{or } \phi &= u\ln t \text{ when } \alpha = 1. \end{aligned} \tag{34}$$

This distribution possesses a Poisson steady state with mean $(\alpha - 1)^{-1}$ whenever $\alpha > 1$. This implies that the $M/G/\infty$ queue with Pareto service times has a stationary distribution, but only if the service time distribution is endowed with a mean. Approaches such as this may be of use in identifying classes of non-Markovian processes admissible under particular axioms about biology, e.g., the existence of steady states, without assuming the existence of all Laplace transforms for the waiting time distributions (39).