nature

SUPPLEMENTARY INFORMATION

https://doi.org/10.1038/s41586-018-0414-6

In the format provided by the authors and unedited.

RNA velocity of single cells

Gioele La Manno[1,2], Ruslan Soldatov[3], Amit Zeisel[1,2], Emelie Braun[1,2], Hannah Hochgerner[1,2], Viktor Petukhov[3,4], Katja Lidschreiber[5], Maria E. Kastriti[6], Peter Lönnerberg[1,2], Alessandro Furlan[1], Jean Fan[3], Lars E. Borm[1,2], Zehua Liu[3], David van Bruggen[1], Jimin Guo[3], Xiaoling He[7], Roger Barker[7], Erik Sundström[8], Gonçalo Castelo-Branco[1], Patrick Cramer[5,9], Igor Adameyko[6], Sten Linnarsson[1,2]* & Peter V. Kharchenko[3,10]*

[1]Laboratory of Molecular Neurobiology, Department of Medical Biochemistry and Biophysics, Karolinska Institutet, Stockholm, Sweden. [2]Science for Life Laboratory, Solna, Sweden. [3]Department of Biomedical Informatics, Harvard Medical School, Boston, MA, USA. [4]Department of Applied Mathematics, Peter The Great St. Petersburg Polytechnic University, St, Petersburg, Russia. [5]Department of Biosciences and Nutrition, Karolinska Institutet, Stockholm, Sweden. [6]Department of Physiology and Pharmacology, Karolinska Institutet, Stockholm, Sweden. [7]John van Geest Centre for Brain Repair, Department of Clinical Neurosciences, University of Cambridge, Cambridge, UK. [8]Division of Neurodegeneration, Department of Neurobiology, Care Sciences and Society, Karolinska Institutet, Stockholm, Sweden. [9]Max Planck Institute for Biophysical Chemistry, Department of Molecular Biology, Göttingen, Germany. [10]Harvard Stem Cell Institute, Cambridge, MA, USA. *e-mail: sten.linnarsson@ki.se; peter_kharchenko@hms.harvard.edu

RNA velocity of single cells

Authors: Gioele La Manno[1,2], Ruslan Soldatov[3], Amit Zeisel[1,2], Emelie Braun[1,2], Hannah Hochgerner[1,2], Viktor Petukhov[3,4], Katja Lidschreiber[5], Maria E. Kastriti[6], Peter Lönnerberg[1,2], Alessandro Furlan[1], Jean Fan[3], Lars E. Borm[1,2], Zehua Liu[3], David van Bruggen[1], Jimin Guo[3], Xiaoling He[7], Roger Barker[7], Erik Sundström[8], Gonçalo Castelo-Branco[1], Patrick Cramer[5,9], Igor Adameyko[6], Sten Linnarsson[1,2]† and Peter V. Kharchenko[3,10]†

Supplementary Note 1: Theory and Computational Methods

Theoretical description of RNA velocity

Based on the model of transcription shown in Fig. 1, we can write down the rate equations for a single gene, which describes how the expected number of unspliced mRNA molecules u, and spliced molecules s, evolve over time:

$$\frac{du}{dt} = \alpha(t) - \beta(t)\, u(t) \quad (1)$$

$$\frac{ds}{dt} = \beta(t)\, u(t) - \gamma(t) s(t) \quad (2)$$

Here, $\alpha(t)$ is the time-dependent rate of transcription, $\beta(t)$ is the rate of splicing, $\gamma(t)$ is the rate of degradation. Under an assumption of constant (time-independent) rates $\alpha(t) = \alpha$, $\gamma(t) = \gamma$, and setting $\beta(t) = 1$ (*i.e.* measuring all rates in units of the splicing rate), the rate equations simplify to:

$$\frac{du}{dt} = \alpha - u(t) \quad (3)$$

$$\frac{ds}{dt} = u(t) - \gamma s(t) \quad (4)$$

The complete solution to the rate equations is given by:

$$u(t) = \alpha(1 - e^{-t}) + u_0 e^{-t} \tag{5}$$

$$s(t) = \frac{e^{-t(1+\gamma)}\left[e^{t(1+\gamma)}\alpha(\gamma-1) + e^{t\gamma}(u_0-\alpha)\gamma + e^{t}\left(\alpha-\gamma(s_0+u_0+s_0\gamma)\right)\right]}{\gamma(\gamma-1)} \tag{6}$$

with the initial conditions $u(0) = u_0$ and $s(0) = s_0$. This solution can be used to extrapolate mRNA abundance s to a future timepoint t_1, under the assumption stated above, by entering the current state of the cell as u_0 and s_0, and then computing $s(t_1)$.

The equations above hold for a single gene. Across all genes, the same equations hold under the same assumptions, but with gene-specific rate constants. Note that setting $\beta(t) = 1$ for all genes implies that we assume a common, constant rate of splicing. This simplifying assumption reduces the number of parameters that need to be estimated, making it possible to estimate velocity from scRNA-seq data. Additional experimental data on gene-specific splicing rates, however, would allow to relax this assumption and improve extrapolation accuracy.

Note that the equations above are *rate* equations, which are deterministic and continuous-valued. The rate equations give the time-evolution of the *expectation* of the number of mRNA molecules observed, not the exact observed number at each timepoint. For example, if the rate equation gives $s(20) = 14.3$, then 14.3 is the expected number (in the statistical sense) of spliced mRNA molecules at time $t = 20$.

Master equation. In a chemical reaction system, the master equation gives the full probability distribution over the counts of all reacting species, as a function of time, often denoted Ψ. At any given point in time, $\Psi(x, y, t)$ assigns a probability to every possible configuration of molecules $u(t) = x$ and $s(t) = y$, where $\sum_x \sum_y \Psi(x, y, t) = 1$. We note that our model is an open system that contains only monomolecular reactions, for which the master equation has an exact analytical solution[24]. Informally, if the system starts with Poisson-distributed variables, they will stay Poisson. If it starts in any other state, then it will converge rapidly to a Poisson distribution. Furthermore, the Poisson-distributed variables have rate constants that are equal to the solution of the rate equations above. Thus, the master equation for our model is a product of Poisson distributions:

$$\Psi(x, y, t) \sim \mathcal{P}(x, y; u, s)$$

where *u* and *s* are the solutions (at time *t*) to the rate equation, and $\mathcal{P}$ denotes the bivariate product

Poisson distribution with probability density function

$$f(x, y; u, s) = \frac{e^u u^x}{x!} \frac{e^s s^y}{y!}$$

Estimation and extrapolation. The normalized degradation rate γ varies among genes and needs to be estimated in a gene-specifc manner. In steady-state populations, where $ds/dt = 0$, we can determine γ of a given gene as the ratio of unspliced to spliced mRNA molecules (again setting $\beta = 1$):

$$\gamma = \frac{u}{s}$$

$$\alpha = \mathrm{u}$$

The steady-state assumption may be realistic for genes expressed in populations known to be terminally differentiated. However, for genes expressed transiently during development, or in cases where the terminal population was not sampled, the steady-state assumption will fail. The subsequent sections will detail how γ can be estimated without the steady-state assumption.

More problematically, we do not know α, nor can it be easily estimated. This prevents us from extrapolating $s(t)$ into the future. Instead of assuming a constant α, we therefore estimate $s(t)$ using one of two alternative assumptions:

Model I. Constant velocity assumption:

We assume that for the purposes of $s(t)$ extrapolation, the rate of change of the spliced molecules remains constant. That is, we assume $ds/dt = v$ is constant, so that the current rate of increase or decrease in spliced mRNA molecules continues into the future. Under this assumption, extrapolation is trivial, since

$$s(t) = s_0 + vt$$

In other words, extrapolation amounts to taking the current number of mRNA molecules and adding the current rate of change multiplied by the extrapolation time step. This assumption works well in practice as long as the time step is short. For longer extrapolation, $s(t)$ can become negative if $v < 0$ (i.e. in the case of a down-regulated gene). This requires clipping the values at zero.

Model II. Constant unspliced molecules assumption:

Alternatively, we can extrapolate $s(t)$ assuming that the number of unspliced molecules stays constant, i.e. that $u(t) = u_0$. This reduces the problem to a single rate equation:

$$\frac{ds}{dt} = u_0 - \gamma s(t)$$

The solution then becomes

$$s(t) = s_0 e^{-\gamma t} + \frac{u_0}{\gamma}(1 - e^{-\gamma t})$$

In practice, we found that at short extrapolation timescales both approaches yielded very similar results. We will indicate below when we used Model I or the Model II.

Assuming gene independence, the overall RNA velocity of the cell is a multidimensional vector comprised of the individual gene velocities.

Estimation framework

In this section we give a description of the analysis framework we used the estimation of RNA velocity and the related data analysis. This analysis logic is implemented separately in R and python environments by velocyto.R and velocyto.py packages, respectively. Parameters, thresholds and other information related to the implementation of each package are described in detail in the next section, and code to reproduce our analysis is available in the companion notebooks at http://velocyto.org.

The velocity estimation procedure incorporates several features to accommodate the complexity of splicing biology. Independent normalization of spliced and unspliced counts allows to control for genome-wide variation of splicing rates between cells. Our model incorporates a gene-specific offset to account for background signals that could originate from other transcripts or alignment errors. Several further adjustments can be used to enhance RNA velocity estimation, reducing impact of single-cell measurement noise and gene-specific aberrations. The robustness of the RNA velocity estimates can be improved by pooling of transcript counts across k most similar cells (see "Cell nearest neighbor (kNN) pooling" section below). Similarly, pooling of counts can be performed across well-correlated genes, based on the assumption that such genes are also

subject to the same up-/down-regulation dynamics (see "Gene kNN pooling" section below, Supplementary Note 2 Figure 8d).

The estimation of the gene-specific equilibrium coefficient γ, a critical step for evaluating velocity, is performed using regression on the extreme expression quantiles (see next section). Such procedure ensures robust estimation even in situations where most (and sometimes all) of the observed cells are outside of the steady state (Supplementary Note 2 Section 2). This default fitting procedure, however, may systematically underestimate the velocity of genes that are observed far outside of their steady state, such as chromaffin maturation genes up-regulated at the very end of the observed differentiation, or neural-crest genes that are already being actively down-regulated in the initial Schwann cell precursor stage. To address this limitation we developed an alternative, structure-based fit to predict the steady-state relationship between spliced and unspliced RNA based on the structural parameters of the genes, such as the number of expressed exons, internal priming sites, or intronic length (Extended Data Fig. 4). The resulting velocity estimates corrected the underestimation at extremes of the chromaffin differentiation trajectory.

Estimation of RNA velocity. For each gene, the normalized degradation rate γ was determined using a least squares fit of the following linear model: $u \sim \gamma * s$, where u and s are the size-normalized unspliced and spliced abundances, respectively, observed for given gene across the cells. Note that to control for global variation of splicing efficiency and detection of unspliced molecules, the spliced and unpsliced counts are normalized separately. Specifically, in a given cell $u = U/N_u$, $s = S/N_s$, where U and S are the number of unspliced and spliced counts, respectively, and N_u and N_s are the total numbers of unspliced and spliced molecules observed in a given cell, respectively. An offset can optionally be included to account for baseline intronic counts that might be driven by unannotated transcripts.

Note that fitting γ in this manner is correct only if the cells are at steady state (and away from zero). However, using a robust quantile fit (*i.e.* including only cells near the origin and cells near the upper-right corner of the phase portrait) we found that γ could be reasonably well approximated even in situations where most (or even all) of the cells were found away from the steady state (see Supplementary Note 2 Section 2).

Under Model I, the velocity component v for a given gene in a given cell was assumed to be

constant and estimated as $v = u - \gamma s - o$, where o is the optional offset parameter accounting for contribution of extraneous transcripts. The extrapolated counts of a given gene in a given cell was then determined as $s_t = max(0, s_0 + vt)$. Where t is the extrapolation time step, that was chosen such that the total RNA count for each cell did not change substantially.

Under Model II, the displacement of spliced mRNA Δs for a given gene in a given cell was estimated assuming constant u as $\Delta s(t) = \left(\frac{\hat{u}}{\gamma} - s\right)(1 - e^{-\gamma t})$, where $\hat{u}$ is the offset-adjusted unspliced count, calculated as $\hat{u} = max(0, u - o)$, and using the default extrapolation time $t = 1$. The extrapolated counts of a given gene in a given cell was then determined as $S_t = max(0, S + \Delta s(t) N_s)$, where S is the non-normalized spliced count for a gene in a given cell. The normalized extrapolated counts were then calculated as $s_t = S_t / \widehat{N}$, where $\widehat{N}$ is the extrapolated total size of the cell $\widehat{N} = N_s + S_t - S$.

Errors due to unequal losses. Counts of spliced and unspliced mRNA molecules are subject to gene-specific losses during sample preparation. As long as these losses are equal, the observed counts U and S will be simply a linear rescaling of the true counts $\widehat{U}$ and $\hat{S}$:

$$U = k\widehat{U}$$

$$S = k\hat{S}$$

All assumptions above will still hold in the rescaled units. If the losses are uniformly random with rate *k*, and if the distributions of $\widehat{U}$ and $\hat{S}$ are Poisson, then U and S will stay Poisson (the same is true if they are Negative Binomial, since the latter can be viewed as a Poisson distribution with a Gamma-distributed rate). Thus, both the expectations of U and S, and their distributions, will be maintained in the new units. Relative variances will increase, however, since fewer molecules are observed.

However, if the gene-specific loss of unspliced molecules U differs from S, say by a factor f, then our estimates of $\frac{ds}{dt}$ will be off by the factor f. This can be seen as follows. First, at steady state, instead of estimating $\gamma = \widehat{U}/\hat{S}$, we will be estimating $f\gamma = f\widehat{U}/\hat{S}$. Second, away from steady state, our estimates of the velocity will be:

$$f\hat{U}(t) - f\gamma\hat{S}(t) = f(\hat{U}(t) - \gamma\hat{S}(t)) = f\frac{d\hat{S}}{dt}$$

Informally, the effect of this factor is to create a different timescale for each gene, compounding the effect of the assumption of constant splicing rate β (above).

Cell nearest neighbor (kNN) pooling. To improve γ estimation, we pooled count data across local neighborhoods. Specifically, we substituted *S, U* counts with the sum of the counts from the original cell and its *k* nearest neighbors. *k* was adapted to match the sparsity and size of the dataset. The estimation of velocity was carried out using pooled count data. The extrapolated state was calculated using the initial (non-pooled) count values. Pearson linear correlation distance on all genes (log scale) was used to find *k* closest cells for the SMART-seq2 datasets, while Euclidean distance in PCA space was used for the larger 10x Chromium and inDrop datasets.

Visualization of cell velocities. The extrapolated state for a cell corresponds to a vector in the same space of the original cell measurement, which can be directly visualized using linear dimensionality reduction approaches such as PCA. For PCA-based visualization (Figure 1h, Figure 2d), principal components were determined based on the observed expression space. The projection of the extrapolated state on the same eigenvectors was then used to position the tips of velocity arrows.

For non-linear, non-parametric embeddings such as t-SNE (e.g. Fig. 2h), it is challenging to project new data points into the embedding. We therefore developed an approach that places the velocity arrow in the direction in which expression difference is best correlated with the estimated velocity vector, controlling for the cell density distribution. The direction was estimated as follows. We calculated a transition probability matrix $\boldsymbol{P}$ by applying an exponential kernel on the Pearson correlation coefficient between the velocity vector and cell state difference vectors:

$$\boldsymbol{P}_{ij} = exp\left(\frac{corr(\boldsymbol{r}_{ij}, \boldsymbol{d}_i)}{\sigma}\right)$$

where $\boldsymbol{r}_{ij}$ is the difference vector between the expression vectors $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$ of cells *i* and *j* transformed with a variance-stabilizing (elementwise) transformation ϱ, and $\boldsymbol{d}_i$ is the ϱ-transformed velocity extrapolation vector of the cell *i*:

$$\boldsymbol{r}_{ij} = \varrho(\boldsymbol{s}_j - \boldsymbol{s}_i)$$

$$\boldsymbol{d}_i = \varrho(\boldsymbol{v}t)$$

$$\varrho(x) = sgn(x)\sqrt{|x|}$$

where $sgn()$ is a sign function. Other formulations of ϱ transformation are also implemented in *velocyto*, including identity and log-based functions. The transition matrix was row-normalized so that $\sum_j \boldsymbol{P}_{ij} = 1$. Then the transition probabilities P_{ij} were used as weights to compute a linear combination of the unitary displacement vectors. Given an embedding with the n positions of the cells described by a set of vectors

$$\boldsymbol{X} = [x_1, x_2, \ldots, x_{n-1}, x_n]$$

the predicted velocity displacement of a cell was calculated as:

$$\Delta x_i = \sum_j \left(P_{ij} - \frac{1}{n}\right) \frac{(x_j - x_i)}{\|x_j - x_i\|}$$

where subtracting $1/n$ corrects the estimate for the non-uniform density of points in the embedding.

Visualization of individual cell velocity arrows is not practical for large datasets. For such cases we visualized a vector field showing local group velocity evaluated on a regular grid. The grid vector field was estimated by applying Gaussian kernel smoothing to the velocity vectors of cells around each grid point:

$$\Delta x_{grid} = \sum_i K_\sigma(x_{grid}, x_i)\Delta x_i$$

where kernel function K_σ was defined as:

$$K_\sigma(a, b) = \exp\left(\frac{-\|a - b\|^2}{2\sigma^2}\right)$$

The size of the grid (number of grid points) was chosen depending on the visual scale of the figure.

Diffusion start and end-point modeling (Fig. 3).

To find the set of cells that correspond to the differentiation starting point (*e.g.* early neural progenitors) and end points (*e.g.* neurons and glial cells) we used a Markov process where respectively the transition probability matrix $\boldsymbol{T}$, where each entry was defined as:

$$\boldsymbol{T}_{ij} = p\boldsymbol{W}_{ij} + (1-p)\boldsymbol{D}_{ij}$$

where **W** (after row sum normalization) corresponds to the local Brownian motion component:

$$\boldsymbol{W}_{ij} = K_{\sigma_W}(\boldsymbol{x_i}, \boldsymbol{x_j})$$

and D (after row sum normalization) represents the local velocity-driven drift:

$$\boldsymbol{D}_{ij} = P_{ij}K_{\sigma_D}(\boldsymbol{x_i}, \boldsymbol{x_j})$$

The Gaussian kernel K_{σ_D} is used to make the diffusion process proceed gradually and locally on the embedding. p<0.5 is the mixture ratio that we set equal to 0.2. Furthermore, to avoid the influence of the local density of points on the result, we downsampled the dataset selecting the nearest neighbours of a uniform grid. σ_D was set to the average distance between neighbouring points and $\sigma_W = \sigma_D/2$. Backward diffusion (*i.e.* diffusion against the velocity bias) was performed by transposing and row-normalizing transition probability matrix $\boldsymbol{T}$. In both directions of the diffusion we started from a uniform distribution and performed 2500 iterations.

Implementation details of gene-relative estimation of RNA velocity. In the basic velocity estimation scheme, *o* was taken to be mean *u* across cells where *s=0*. For the least squares fit, cell-specific regression weights *w* were taken to be $e^4 + s^4$. Note that *o* and γ fit were estimated using a different logic when using quantile fit, spanning-read based fit, or gene *k*NN pooling. To improve stability at low counts, $\hat{u}$ was additionally calculated adding or subtracting one pseudo-count, and a minimal magnitude velocity v was reported.

To handle scenarios where most of the observed cells are outside of the steady state, we used extreme quantiles of expression values to fit the γ coefficient and offset *o*. Specifically, a linear model $u \sim \gamma * s + o$ was fit for the slope and intercept (γ and *o*) coefficients, limiting the set of cells to those with values of *s* within the top and bottom 5% for that gene (2% was used for larger datasets, such as mouse BM). No regression weights were used. The *velocyto* packages also

implement a "diagonal quantiles" option, where the extreme quantiles are determined not on the spliced expression magnitude alone, but on a normalized sum of spliced and unspliced expression magnitude ($x = s/S + u/U$), where U and S are the maximal unspliced and spliced expression of that gene, respectively.

Gene kNN pooling. Gene pooling was implemented by pooling counts across k most correlated genes. Gene correlation was assessed using Pearson linear correlation distance on $log(s+1)$ values. The resulting matrix represents counts for metagenes. The slope and offset estimates were then carried out on the pooled gene counts. The velocity of the original gene was then evaluated based on the assumption that the observed *vs.* expected (in steady-state) ratio of the unspliced signal should be similar between co-regulated genes, and therefore also similar for the pooled gene counts. The log ratio of observed to expected unspliced counts was calculated for a given gene and a given cell as $m = log_2\left(\frac{u}{\hat{\gamma}s+\hat{o}}\right)$, where $\hat{\gamma}$ and $\hat{o}$ are the the slope and offset estimates obtained using metagene matrices. Then, gene-specific γ was estimated by taking mean of $2^{log_2(\hat{u}/s)-m}$across all cells with $s > 0$. Under the assumption of constant u (Model II), $\Delta s(t)$ was then estimated as $\Delta s(t) = (s+\epsilon)[e^{-\gamma t}(1-2^m)+2^m] - s$, where $\epsilon = 10^{-4}$.

Structure-based estimation of RNA velocity (Extended Data Fig. 4). Structure-based model starts with the initial estimates (γ_r) obtained using gene-relative model described above. It then proceeds to fit a generalized additive model (gam) to predict gene-specific values of γ based on a combination of gene structure parameters, such as the number of exons, total intronic length, or the number of predicted internal priming sites. The γ values predicted by this model are then translated into the unspliced count M values (log observed over expected under steady state ratios unspliced count ratios). Unlike γ estimates, the M values can be directly compared between genes, and we expect co-regulated genes to show similar M values during up- and down-regulation. The M values were therefore refined further by applying trimmed mean procedure across closely-correlated genes, and these adjusted M values are finally used to extrapolate the future expression state of the cell.

The analysis considers only genes with more than one annotated exon ($n_e > 2$), with total exonic length l_e over 500bp, and total intronic length l_i over 3kbp. Then a global generalized additive model was constructed to fit the dependency γ_r on the structural parameters of the gene and its expression magnitude:

$$\gamma_r \sim K_2(s, l_i) + K_1\left(log_{10}\left(\frac{l_i+1}{l_e+1}\right)\right) + K_1(n_e) + K_1(n_c) + K_1(n_d)$$

where K_1 and K_2 denote 1D and 2D local smoothing kernels, respectively as implemented by the mgcv R package. Here, s denotes spliced expression magnitude of a gene, and n_e the number of expressed exons in a gene. n_c and n_d denote the number of concordant and discordant internal priming sites, respectively (see Extended Data Fig. 1 for details on the internal priming sites). The model was fit, weighting the observations of each gene by the square root of its total spliced counts across the dataset. The fitting procedure also omitted genes with disproportionate amount of unspliced counts (likely due to unannotated non-coding transcripts). To do so, total unspliced counts were modeled as $\sum_{cells} u \sim \sum_{cells} s + l_i/l_e$ using a generalized linear model with normal distribution and log link and genes with pearson deviance exceeding 3 were omitted when fitting the model.

The global model was then used to predict structure-based steady-state γ (γ_s) for all genes passing structural parameter thresholds mentioned above. γ_s was then used to estimate the log ratio of observed to expected unspliced counts as $m_s = log_2\left(\frac{u}{\gamma_s s + o_r}\right)$, where o_r is the gene specific offset determined as described in the gene-relative model. As we expect m_s estimates of individual genes to be noisy, the next step used *k* nearest genes to stabilize m estimates for each gene. Specifically, for a given gene *g*, m was estimated as a trimmed mean of the m_s values for *k* genes most correlated with gene *g* across the entire dataset. Default *k*=15 was used, with top and bottom 5 genes trimmed. These robust m estimates were then used to estimate RNA velocity as described in "Gene kNN pooling".

Metabolic labeling (Extended Data Fig. 2)

We performed metabolic labeling of Hek293 cells using 4-thiouridine as described in[22] but without fragmentation (4sU-seq), and then prepared the bulk RNA samples for sequencing using STRT[23]. We incubated cells in duplicate for 5, 15 and 30 minutes and included a no pull-down control representing the steady-state expression state. Importantly, STRT is a single-cell RNA-seq protocol based on oligo-dT priming, and thus should be representative of the protocols designed to detect poly-A+ RNA. We have previously estimated that the cross-contamination

(fraction of contaminating non-labeled RNA that is found in labeled RNA) is ~0.5%, although we could not measure it directly here as the 4sU-seq internal controls were not polyadenylated. We analyzed two independent biological samples for condition, for each samples two technical replicates (reverse transcription) were performed.

The data was analyzed using the standard *velocyto* counting pipeline with option "`-u 8b`". Genes that satisfied the following conditions were selected: (1) spliced molecules were detected in all control samples; (2) unspliced molecules were detected in at least 75% of the pull-down samples and in at least 2 of the control samples; (3) the levels of unspliced molecules in the pull-down with the higher unspliced expression were at least twice that of the control samples; (4) the coefficient of variation of the spliced fraction was lower than what would be predicted by a linear model using the spliced magnitude as a predictor (residuals to the fit above the 10^{th} percentile); (5) the time-dependent trend was significant as determined by ANOVA ($p < 0.15$). Factor analysis was performed on the genes, and the genes were clustered in the space of the two factors using a Gaussian mixture model. Of the two clusters obtained, the smaller one presented patterns showing a discordant signature (incompatible with a simple model of transcription) which were not processed further. For the remaining genes, we considered the observed fraction of spliced molecules $\frac{s}{s+u}$. The parameters beta and gamma were determined by least squares fit of the following equation, derived from the solution of time-independent rate models (note that alpha simplifies out when calculating the ratio $s/(s+u)$).

$$\frac{s}{s+u} = f(\beta,\gamma,t) = \frac{1}{\left(\frac{\beta+\gamma}{\beta} + \frac{(-e^{t\beta} + e^{t\gamma})\gamma}{e^{t\beta}\beta - e^{t\gamma}\gamma + e^{t(\beta+\gamma)}(\gamma-\beta)}\right)}$$

The time constant τ for an asymptotically increasing system is defined as the time required to reach the fraction $1-\frac{1}{e} \cong 0.632$ of its equilibrium value, that for the spliced fraction is $\lim_{t\to\infty} \frac{s}{s+u} = \frac{\beta}{\beta+\gamma}$; therefore the value of τ was obtained by solving (numerically) the following equation:

$$\frac{1}{\left(\frac{\beta+\gamma}{\beta} + \frac{(-e^{\tau\beta} + e^{\tau\gamma})\gamma}{e^{\tau\beta}\beta - e^{\tau\gamma}\gamma + e^{\tau(\beta+\gamma)}(\gamma-\beta)}\right)} - \frac{\beta\left(1-\frac{1}{e}\right)}{\beta+\gamma} = 0$$

References

24. Jahnke, T. & Huisinga, W. Solving the chemical master equation for monomolecular reaction systems analytically. *J. Math. Biol.* **54,** 1–26 (2007).