

Stochastic Simulation to Visualize Gene Expression and Error Correction in Living Cells

Kevin Y. Chen^{1,*}, Daniel M. Zuckerman², Philip C. Nelson³

¹Department of Chemistry, University of Cambridge, Lensfield Road, Cambridge CB2 1EW, UK

²Department of Biomedical Engineering, Oregon Health & Science University, Portland, OR 97239, USA

³Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, PA 19104, USA

ABSTRACT Stochastic simulation can make the molecular processes of cellular control more vivid than the traditional differential equation approach by generating typical system histories, instead of just statistical measures such as the mean and variance of a population. Simple simulations are now easy for students to construct from scratch—that is, without recourse to black-box packages. In some cases, their results can also be compared directly with single-molecule experimental data. After introducing the stochastic simulation algorithm, this article gives two case studies involving gene expression and error correction, respectively. For gene expression, stochastic simulation results are compared with experimental data, an important research exercise for biophysics students. For error correction, several proofreading models are compared to find the minimal components necessary for sufficient accuracy in translation. Animations of the stochastic error correction models provide insight into the proofreading mechanisms. Code samples and resulting animations showing results are given in the online Supplemental Material.

KEY WORDS kinetic proofreading; stochastic simulation; undergraduate instructors; graduate instructors

I. INTRODUCTION

Physical processes unfold over time. Our minds grasp physical mechanisms largely by narrative, so it is not surprising that some of the most vivid physics demonstrations also play out over time. *Simulations* of physics that unfold over time are similarly powerful; simulations created by the student can be best of all. This view is gaining ground in introductory courses (1), but the benefits of animated simulation extend farther than this. Here we wish to show that the behavior of strongly nonequilibrium statistical systems can be illustrated via stochastic simulations that are simple enough to serve as undergraduate projects. Recently developed, free, open source programming resources sidestep some of the laborious coding chores that were once required for such work. In particular, we believe that the error correction mechanism known as kinetic proofreading can be more clearly understood when a student views typical stochastic temporal sequences, as opposed to solving a set of deterministic rate equations (or even more complicated approaches involving first-passage times, etc.). Coding this and other

“*” corresponding author

Received: 26 June 2019

Accepted: 27 September 2019

Published: 14 February 2020

© 2020 Biophysical Society.

simple processes also opens the door for the student to study systems that are too complex for the rate-equation approach to yield insight.

II. SCIENTIFIC AND PEDAGOGICAL BACKGROUND

A. Double-well hopping

Students are often told that a simple chemical reaction, for example isomerization of a macromolecule, can be regarded as a barrier-passing process. A micrometer-size bead in a double optical trap serves as a mesoscopic model system with this character (2), and it is worthwhile for students to watch it undergo a few dozen sharp transitions in between episodes of constrained Brownian motion near its two stable positions (see Video S1 in the Supplemental Material). A simple model for this behavior states that the hopping transitions occur at random times drawn from an exponential distribution; that is, many rapid transitions are interspersed with a few long pauses. Section III.A below discusses a simple simulation of this process.

B. Birth–death process

Section III.B generalizes from situations with essentially only one kind of transition (or two symmetric kinds) to the more interesting case in which several inequivalent choices are possible, and where the relevant transition probabilities depend on the current state. This general situation can describe a chemical reaction that requires, and depletes, molecules of some substrate.

Most science students know that living cells synthesize each of their messenger RNAs (mRNAs) from a single copy (or a small fixed number) of the corresponding gene. Some genes are constitutive (unregulated), and we model these with a constant rate of mRNA synthesis. Once an mRNA transcript has formed, it has a limited lifetime until it is degraded by other cellular machinery. We assume that this process, too, relies on chance encounters with degradation enzymes. Each of many species of mRNA must all share the attentions of a limited number of degradation enzymes, so each

mRNA copy has a fixed probability per unit time to be removed from the system.

The physical hypotheses in the preceding paragraph amount to a model called the *birth–death process*, which has many other applications in physics and elsewhere. Similar to a one-dimensional (1D) random walk, we characterize the system’s state by an integer, in this case the population size of the mRNA of interest. Synthesis is a transition that increases this number, with a fixed probability per unit time k_s (called the *mean rate of synthesis*). Degradation is a transition that decreases it, with a probability per unit time that is the current population n times another constant k_d (the rate constant for degradation). Section IV.B describes the insights students can get from simulating this model and extend them to describe the stochastic phenomenon of gene bursting.

C. Proofreading

Ask a student, “What is the big secret of life?” and the answer will probably be “DNA,” or perhaps “evolution by natural selection.” Indeed, DNA’s high, but not perfect, degree of stability underlies life’s ability to replicate with occasional random modifications. However, it is less well appreciated that the *stability* of a molecule of DNA does not guarantee the *accuracy* of its replication and transcription. There is another big secret here, just as essential to life as the well known ones. In fact, a wide range of molecular recognition events must have extremely high accuracy for cells and their organisms to function. Think of our immune cells, which must ignore the vast majority of antigens they encounter (from “self”), yet reliably attack a tiny subpopulation of foreign antigens differing only slightly from the self.

Translation of mRNA into proteins is an emblematic example of such a puzzle. It is true that artificial machines now exist that can read the sequence of mRNA. Then, another artificial machine can take the resulting sequence of base triplets, decode it, and synthesize a corresponding polymer of amino acids (a polypeptide), which in some cases will then

fold into a functional protein without further help. But the cells in our bodies, and even bacteria, do these jobs reliably *without* those huge and expensive machines, despite the incessant nanoscale thermal motion!

Merely intoning that a wonderful molecular machine called the *ribosome* accomplishes this feat doesn't get us over the fundamental problem: At each step in translation, the triplet codon at the ribosome's active site fits a few of the 41 distinct *Escherichia coli* transfer RNA (tRNA) isoacceptors *somewhat* better than it fits the others. But the binding energy difference, which quantifies "somewhat better," only amounts to two or three hydrogen bonds. This translates into a fraction of time spent bound to the wrong tRNAs that is about 1/100 times as great as the corresponding quantity for the correct amino acid (3). If the fraction of incorrect amino acids incorporated into a polypeptide chain were that high, then *every* protein copy longer than a few hundred amino acids would be defective!

In fact, the error fraction of amino acid incorporation is more like 10^{-4} . The fact that this figure is so much smaller than the one seemingly demanded by thermodynamics remained puzzling for decades. After all, the ribosome is rather complicated, but it is still a nanoscale machine. *Which* of its features could confer this vast improvement in accuracy?

Hopfield and Ninio proposed an elegant physical mechanism for the ribosome's surprising accuracy (3–5). To explore it, we begin by paraphrasing a metaphor from Alon (6). Imagine that you run an art museum and want to find a mechanism that picks out Picasso lovers from among all your museum's visitors. You could open a door from the main hallway into a room with a Picasso painting. Visitors would wander in at random, but those who do not love Picasso would not remain for as long as those who do. Thus, the concentration of Picasso lovers in the room would arrive at a steady value (with fluctuations) that is enriched for the desired subpopulation.

To improve the enrichment factor further, you could hire an employee who occasionally closes the door to the main hallway, stopping

the dilution of your enriched group by random visitors, then open a new exit doorway onto an empty corridor. Some of the trapped visitors will gratefully escape, but die-hard Picasso lovers will remain, leading to a second level of enrichment. After an appropriate time has elapsed, you can then reward everyone still in the room with, say, tickets to visit the Picasso museum in Paris.

In the ribosome, the initial, reversible binding of a tRNA is followed by a transformation analogous to closing the door in the preceding metaphor. This transformation involves hydrolysis of a guanosine triphosphate (GTP) molecule complexed with the tRNA; hence, it is nearly *irreversible* because of the highly non-equilibrium concentration of GTP compared with the hydrolysis products guanosine diphosphate (GDP) and P_i (inorganic phosphate). Such hydrolysis reactions are well known to supply the free energy needed to drive otherwise unfavorable reactions in cells, but here their role is more subtle.

Hopfield (4) knew that after hydrolysis, incorporation of the amino acid was delayed and could still be preempted by unbinding of the tRNA complex. The existence of this pathway was previously known but had seemed wasteful: an energy-rich GTP had been spent without anything useful (protein synthesis) being done. On the contrary, however, this second step implements the mechanism in the art museum metaphor, giving the ribosome an independent second chance to dismiss a wrong tRNA that accidentally stayed bound long enough to progress to this stage. Spending some GTPs may be a modest price to pay compared with creating and then having to detect and recycle an entire defective protein.

Hopfield coined the name *kinetic proofreading* for this mechanism, but we will refer to it as the classic Hopfield–Ninio (HN) mechanism because the original term is somewhat misleading. In chemical reaction contexts, a kinetic mechanism generally implies bias toward a product with lower activation barrier, even if it is less stable than another product with a higher barrier. This preference is most pronounced at high, far-from-equilibrium catalytic

rates (7). In contrast, the classic HN proofreading mechanism involves two sequential thermodynamic (quasi-equilibrium) discriminations. Moreover, these discriminations take place before reading even the very next codon, in contrast to editorial proofreading, which generally happens after an entire manuscript is written. (Our choice of term also distinguishes the classic scheme from later models that are sometimes also called kinetic proofreading.)

In a nutshell, we will outline an exploration of Hopfield's conclusion (4) that *an effectively irreversible step, or at least a step far from equilibrium, can give rise to enhanced accuracy. The free energy of GTP hydrolysis is the price paid for this accuracy.*

III. METHODS

A. Double-well hopping

With the physical motivation from section II.A, students can explore how to generate simulated waiting times. Any computer math system has a pseudorandom number generator that generates floating-point numbers uniformly distributed between 0 and 1. Many students are surprised (and some are intrigued) to learn that applying a nonlinear function to samples from a random variable yields samples with a different probability density function, and in particular that $y = -\tau \ln x$ is exponentially distributed, with mean τ , if x is uniform on $(0,1]$ (8).

Starting from that insight, it takes just one line of code to generate a list of simulated waiting times for transitions in a symmetric double well; finding the cumulative sums of that list gives the actual transition times (see Computer Code S1 in Supplemental Material and Kinder and Nelson (9)).

The freely accessible VPython programming system (or its web-based version GlowScript) makes it very easy to create an animation of an object whose spatial position is supplied as a function of time (10). The only challenging part is to pass from a list of irregularly spaced transition times t_m to particle positions x at each of many (regularly spaced) video frames. Computer Code S1 accomplishes this transformation in a few lines summarized here:

- (a) Divide the desired total movie duration into real time steps according to the desired frame rate. Initialize real time to the end of the first video frame. Initialize transition number to $m = 0$.
- (b) Step forward in transition number m until the time t_m of the most recent transition exceeds current real time. Switch states for each transition encountered (if any) and save the state at the end of the video frame.
- (c) Update the real time to the end of the next video frame and repeat step 2 until the whole animation has been generated.

The payoff is immediate: visually, the simulated trajectories have a very similar character to the actual hopping of a bead in a double trap (Video S2).

B. The stochastic simulation algorithm

Gillespie extended, validated, and popularized a simple but powerful method, the stochastic simulation algorithm, which generalizes the preceding idea to describe systems that make transitions between multiple discrete states (11). For any allowed state i , we have been given the probability per time $k_{j \leftarrow i}$ to transition to every other allowed state j . The algorithm repeatedly executes the following steps:

- (a) The system begins in some initial state i . It waits for a certain time, then transitions to a new state. To find the waiting time and the new state:
 - (1) Determine the probability per time k_{tot} for *any* of the allowed transitions to occur by summing all the mean rates, $k_{j \leftarrow i}$ over j .
 - (2) Draw a waiting time from the exponential distribution with mean given by the reciprocal of k_{tot} by the method in section III.A: $\Delta t = -(k_{\text{tot}})^{-1} \ln x$.
 - (3) Determine which of the M allowed processes happens at that transition time by drawing from a nonuniform discrete distribution on M objects. The probabilities for the choices j are proportional to the corresponding $k_{j \leftarrow i}$.

- (b) Update the current state according to that decision. Update the current time by adding the waiting time Δt . Return to step 1 and repeat until the required total elapsed time is reached.

The beauty of this algorithm, besides its correctness (12), is that no computation is wasted on time steps at which nothing happened: by construction, there is a state transition at every chosen time.

Two-state hopping is a particularly simple example of this algorithm: here, the state variable is position $x = \pm 1$. The transition rate depends on the system's state only in the trivial sense that each state can hop only to the other one; the numerical value of the rate constant is always the same.

C. Birth–death process

Computer Code S2 implements a stochastic simulation of the birth–death process introduced in section II.B. Here, the state is specified by an integer n , the population of the molecular species of interest. In each transition, n can either increase or decrease by one.

The probability per time to make a transition is not a constant; rather, $k_{\text{tot}} = k_s + nk_d$, where k_s is the probability per time for synthesis and k_d is the rate constant for degradation. To decide what reaction occurs, we make a Bernoulli trial with probability $p = k_s/k_{\text{tot}}$ to increase population n by one, and $1 - p$ to decrease it.

D. Proofreading

1. Simulation goals

The qualitative word model given at the start of section II.C may seem promising, but the corresponding kinetic equations make for difficult reading and understanding. Better intuition could emerge from a presentation that stays closer to the concrete ideas of discrete actors randomly arriving, binding, unbinding, and so on, visibly implementing the ideas behind the museum metaphor. The following sections argue that stochastic simulation can realize that goal.

Besides confirming the predicted accuracy payoff, we will explore the claim that the

enhancement of accuracy depends on GTP, GDP, and P_i being held far from chemical equilibrium, so that the hydrolysis step is nearly irreversible (the “door shuts tightly” in the museum metaphor). In fact, the model predicts *no enhancement* of accuracy compared with the one-step model when this chemical driving force is low (4). Far from equilibrium, however, we will confirm Hopfield's observation that the predicted error fraction can be as low as the *square* of the equilibrium value (or even a higher power if multiple rounds of sequential testing are employed).

In addition to checking Hopfield's prediction, section IV.C explores speed–accuracy tradeoffs in different minimal models of translation. The classic HN mechanism relies only on off-rate differences between right and wrong tRNA. However, differences in on-rates have also been observed between right and wrong tRNA (13–16), so we also explore a model relying solely on on-rate differences. Finally, we explore a model in which all kinetic parameters are based on in vitro translation experiments, which includes both on-rate and off-rate differences. The speed and error rate of translation obtained in the three different models will be compared with the speed and error rate measured in vitro, which, respectively, are about 0.25–8 amino acids/s (17, 18) depending on the in vitro system used and $1.6 \cdot 10^{-3}$ for Zaher and Green's (13) in vitro system. Translation rates in vitro are typically slower than in vivo *E. coli* translation rates of 10–20 amino acids/s (18). Zaher and Green's in vitro study did not directly measure translation rates, but from the assumption that peptide bond formation is the rate-limiting step, the upper limit for translation rates in their system should be 7 amino acids/s. The actual translation rate is likely lower because of tRNA rejections at the proofreading steps.

This exercise will provide insight into the contribution of off-rate and on-rate differences to the speed and accuracy of translation. All these variations are readily implemented by changing a handful of constants in a simulation code.

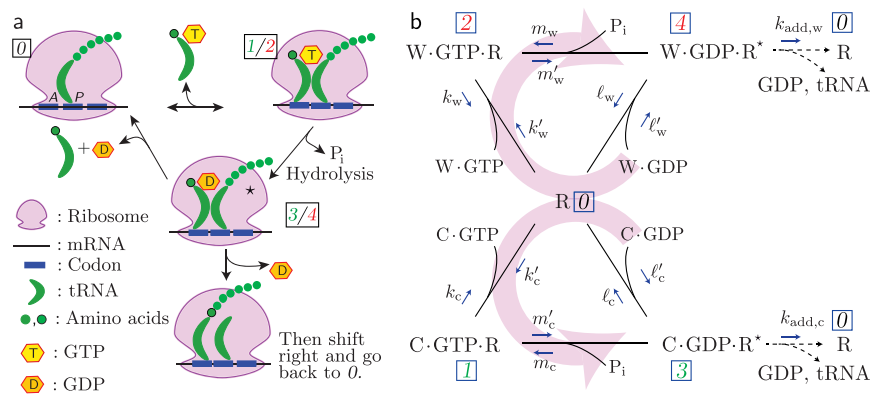


Fig 1. Two representations of the classic Hopfield–Ninio (HN) mechanism. (a) Traditional cartoon expressing the catalytic cycle of the ribosome [after Banerjee et al. (20)]. (b) Corresponding kinetic diagram that forms the basis for a stochastic simulation with states labeled 0–4. The large pale arrows indicate the net circulation in each cycle under cellular conditions, where guanosine triphosphate (GTP) is maintained out of equilibrium with guanosine diphosphate (GDP) and inorganic phosphate (P_i). The symbol R denotes a ribosome complexed with mRNA; R^* is the corresponding complex activated by GTP hydrolysis. At far right, R indicates the ribosome with one additional amino acid added to the nascent polypeptide chain. The classic HN mechanism assumes that unbinding rates k_c and l_c are smaller than their mismatched counterparts k_w and l_w , but that other constants are all equal for correct (C) and wrong (W) tRNA.

2. A single ribosome in a bath of precursors

This section's goal is to formulate the word model of section II.C in the context of mRNA translation, then set up a stochastic simulation [section III.B; see also Zuckerman (19)]. Later sections will show how students can explore the expectations raised at the end of the preceding section.

We will assume that a single ribosome is complexed with a single mRNA and has arrived at a particular codon. This complex sits in a bath containing the following free, dissolved species at fixed concentrations (Fig 1a):

- C denotes correct tRNA (i.e., the species that matches the codon currently being read), loaded with its appropriate amino acid. We neglect the possibility of a tRNA being incorrectly loaded; accurate loading is the concern of a separate proofreading mechanism that we are not studying now (21, 22).
- W is similar to C, but refers to the wrong tRNA for the codon under study.
- Other reactions supply complexes of tRNA with guanosine phosphates: C-GTP, C-GDP, W-GTP, and W-GDP. (For simplicity, we suppress any mention of elongation factors, one of which, EF-Tu (elongation factor

thermo unstable), is also included in these complexes but is only implicit in the classic HN mechanism.)

Computer Code S3 implements a stochastic simulation on the five states shown in Figure 1b. The figure represents the ribosome–mRNA complex by ribosome R. In state 0, this complex is not bound to any tRNA. (More precisely, no tRNA is bound at the A site of the ribosome; a previously bound tRNA, together with the nascent polypeptide chain, is bound at another site [labeled P in Fig 1a], which we do not explicitly note.) Surrounding this state, Figure 1b shows four other states 1–4 in which the ribosome is bound to the complexes introduced earlier. The upper part of the figure describes wrong tRNA binding and possible incorporation; the lower part corresponds to the correct tRNA. Horizontal arrows at the top and bottom denote hydrolysis of GTP, which is coupled to a transformation of the ribosome into an activated state, R^* .

Although any chemical reaction is fundamentally reversible, under cellular conditions the concentration ratio $[P_i][C-GDP]/[C-GTP]$ is far below the equilibrium value, so that the reactions in Figure 1b are predominantly in the direction shown by the pale purple arrows, which was one of the conditions in Hopfield's

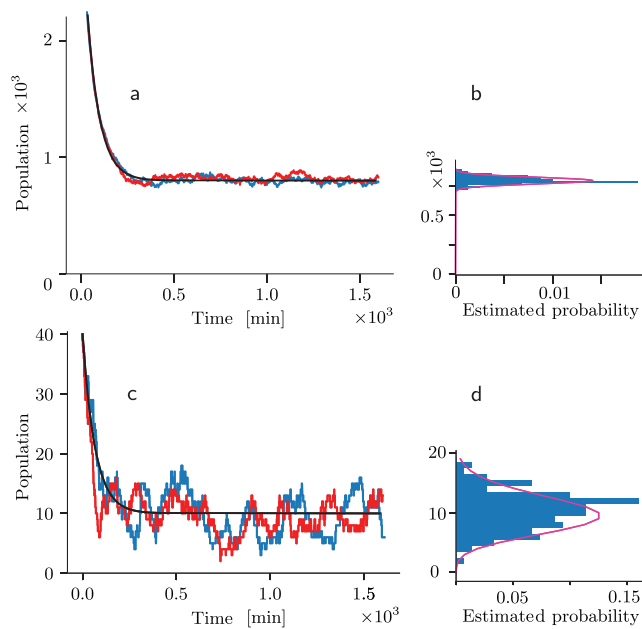


Fig 2. Behavior of a birth–death process. Figure generated by Computer Code S2. (a) The bumpy traces show two instances of simulated time series with $k_s = 12 \text{ min}^{-1}$, $k_d = 0.015 \text{ min}^{-1}$, and hence $n_* = 800$ (see Eq. 1). The initial population was $n_0 = 3200$. The smooth trace shows the exponential relaxation predicted by the continuous, deterministic approximation. (b) After the system comes to steady state, there is a tight distribution of n values across 100 runs of the simulation (bars). The curve shows the Poisson distribution with mean n_* for comparison. (c, d) The same as in panel b, but with $k_s = 0.15 \text{ min}^{-1}$ and $n_0 = 40$. Although individual instances (runs of the simulation) deviate strongly from the continuous, deterministic approximation (solid curve in panel c), nevertheless the sample mean of the population $n(t)$ over many runs does follow that prediction (not shown). The distribution of steady-state fluctuations is again Poisson (curve in panel d).

original proposal and corresponds to the usual biochemical assumption of irreversibility. (Section IV.C.5 will explore relaxing it.)

Again, we are assuming that a *single* ribosome bounces around this state diagram in the presence of fixed concentrations of feedstocks either imposed in vitro by the experimenter or supplied by a cellular milieu. After hydrolysis, the ribosome can reject its tRNA–GDP complex with probability per unit time ℓ . In the metaphor of section II.C, this option corresponds to “exiting the museum exhibit by the second door.” Or, with probability per unit time k_{add} , the ribosome complex can add its amino acid to the nascent polypeptide and translocate the tRNA to the *P* binding site, ejecting any tRNA already bound

there. Either way, the *A* binding site becomes vacant and, for the purposes of this state diagram, the ribosome returns to state 0.

IV. RESULTS AND DISCUSSION

A. Multiwell hopping

Section III.A described a simulation of double-well hopping. A small modification of the algorithm described there gives an interesting extension: Instead of hopping between two wells (reversing direction on every step), consider one-dimensional diffusion on a symmetric many-well potential, for example, one of the form $U(x) = \sin(x)$. In such a potential, for each transition the system must also make a random decision whether to increase or decrease a position coordinate x by 2π . The resulting random walk will display the same long-time scaling behavior as any unbiased 1D walk, but with trajectories that hop at random times, not periodic steps as in the simplest realization (8).

B. Birth–death model

1. Convergence to the continuous, deterministic approximation

Students will probably find it reasonable that, when the mRNA population size n is sufficiently large, we may neglect its discrete character. Students who have been exposed to probability ideas may also find it reasonable that, in this case, the relative fluctuations of n from one realization to the next will be small, and so n effectively behaves as a continuous, deterministic variable, subject to the differential equation $dn/dt = k_d n + k_s$. That equation predicts exponential relaxation from an initial value n_0 to the steady value $n_* = k_s / k_d$ with e-folding time $1/k_d$:

$$n(t) = n_* + (n_0 - n_*)e^{-k_d t}. \quad (1)$$

The simulation bears out this expectation (Fig 2a, b).

However, mRNA populations in living cells are often *not* large. Nevertheless, although individual realizations of $n(t)$ may differ significantly, the *ensemble average* of many such trajectories does follow the prediction of the

continuous/deterministic idealization (Fig 2c). Within individual cells, however, there will be significant deviation around that mean behavior (Fig 2c). Specifically, for this case (representing a simple constitutive promoter), the steady state will have fluctuations of mRNA count n that follow a Poisson distribution (Fig 2d). That key result is more memorable for students when they discover it empirically in a simulation than it would be if they just watched the instructor prove it with abstract mathematics (by solving a master equation) (8).

State fluctuations of the sort just mentioned may suffice to pop a more complex system out of one steady state and into a very different one. Indeed, even the simplest living cells do make sudden, random state transitions of this sort. Such unpredictable behavior, not seen in the differential equation approach, can potentially be useful to bacteria, implementing a population-level bet-hedging strategy (23–26).

A real bacterium is not simply a beaker of reagents. Bacteria periodically divide, partitioning a randomly chosen subset of each mRNA species into each daughter cell. That extra level of realism is difficult to introduce into an analytical model, but straightforward in a simulation. The results are similar to those just described, with a larger effective value of the clearance rate constant (8).

2. Upgrade to cover bursting processes

Bacteria are supposedly simple organisms. The birth–death process is simple, too, and it fits with the cartoons we see in textbooks, so it is interesting for students to follow the recent discovery that the model makes quantitative predictions for mRNA production that in some cases were experimentally *disproven* (8, 27, 28).

Indeed, most bacterial genes are not constitutively expressed; for example, many are regulated by transcription factors, whose binding/unbinding can introduce additional variation in mRNA copy numbers. Recent advances in single-molecule imaging permit the direct measurement of $n(t)$ in individual cells and disagree with the birth–death model’s prediction that the distribution of n in the steady state should be Poisson; typically, the ratio of variance to mean was found to be 5 or more

(not 1 as in a Poisson process). Researchers found, however, that a simple modification of the birth–death model could accommodate this and other discrepant data. The required extension amounts to assuming that mRNA transcripts are generated in *bursts*, that the bursts themselves are initiated with a fixed probability per unit time, and that once initiated, a burst is also terminated with fixed probability per unit time. For example, a burst could begin with unbinding of a repressor and end when it rebinds. Many other regulatory architectures are probably realized in cells, but the model just outlined is simple and tractable and, hence, serves as a phenomenological representative for what lies beyond constitutive gene expression.

We can readily upgrade the birth–death simulation by supplementing the state variable n with a binary on/off variable; see the second code of Computer Code S2. Although the new model has two additional parameters compared with the original birth–death model, nevertheless it was overconstrained by the experimental data, so its success was a nontrivial test (Fig 3) (8, 28). Later work confirmed its predictions for how mRNA statistics should change when the level of a transcription factor was changed (29, 30). More broadly, it is instructive to point out to students that had the original authors been content with a reasonable fit to averaged data (Fig 3a), they might have accepted the birth–death model. Only when fluctuation information is included do we see that that model cannot explain the experiment (Fig 3b). Eventually, analytic results were also obtained for this model (31–35), but the original approach of Golding et al. (27) was via numerical simulations, like the one described here.

C. Proofreading

We now return to the process described in section III.D, described by models of the sort shown in Figure 1.

1. Visualization of the simulation results

To keep the project (stochastic simulation and visualization of various proofreading models) modular, we constructed simulation code

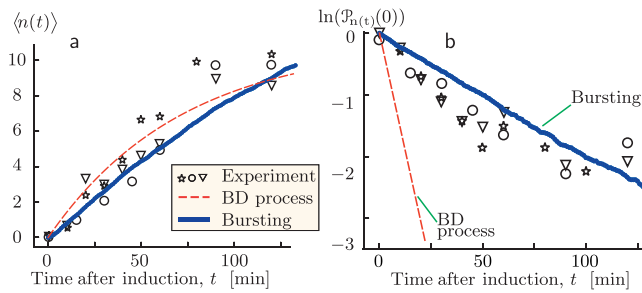


Fig 3. Evidence for transcriptional bursting. The figure was generated by the second code of Computer Code S2. (a) The time course of the number of mRNA transcripts in a cell $n(t)$, averaged over 50 or more cells in each of three separate, identical experiments, is shown with three different symbols. All of the cells were induced to begin gene expression at time 0. The dashed curve shows a fit of the birth–death (BD) process (Eq. 1) to data, determining the apparent synthesis rate $k_s \approx 0.15/\text{min}$ and clearance rate constant $k_d \approx 0.014/\text{min}$. The solid curve shows the corresponding result from a computer simulation of the bursting model discussed in section IV.B.2 [see also Nelson (8)]. (b) Semilog plot of the fraction of observed cells that have zero copies of mRNA versus elapsed time. Symbols: Data from the same experiments as in panel a. Dashed line: The birth–death process that fit (a) predicts that initially $\mathcal{P}_{n(t)}(0)$ falls with time as $\exp(-k_s t)$, where k_s has the value found by fitting the data in (a). The experimental data instead yield the initial slope $-0.028/\text{min}$. Solid line: Corresponding result from the bursting model (section IV.B.2). [Experimental data from Golding et al. (27).]

that writes its state trajectory to a file (Computer Code S3). A second code then reads that file and creates a visual output (Computer Code S4). The first of these codes operates similarly to section III.C, but with a four-way choice of what transition to make after each waiting interval (Fig 1b). The second code can be almost as simple as the one described in section III.A. However, students with more time (perhaps in a capstone project) can make a more informative display with a reasonable additional effort, as follows.

The supplemental videos not only show the state that is current at the end of each video frame, they also animate the pending arrivals of new complexes that are about to bind and the departures of old ones that have unbound without incorporation. By this means, the videos give a rough sense of the narrative in the trajectory being shown (Fig 4). These improvements are not difficult to add once the basic code is working. Alternatively, stu-

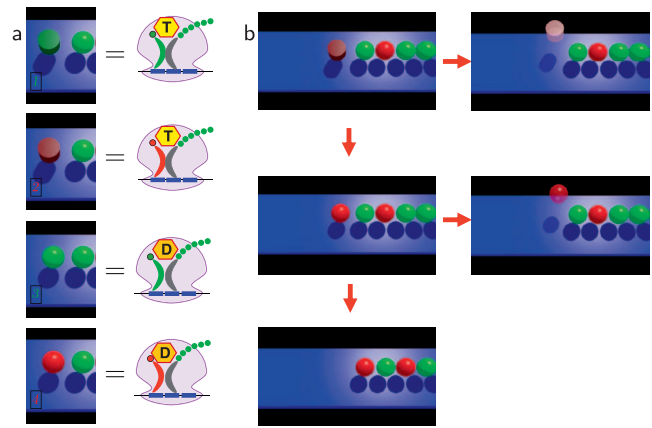


Fig 4. Conventions used in Videos S3–S5. Green objects represent correct transfer RNA (tRNA)/amino acid complexes; red objects represent complexes bearing an incorrect amino acid. Spheres resting on the blue plane represent the nascent chain of incorporated amino acid residues. (a) Four states with a tRNA complex bound to the A binding site and their corresponding cartoon representations. (b) Still frames from an animation. An incorrect complex has bound at top left. It can unbind (top right) or proceed to hydrolyze its guanosine triphosphate (center left). Thereafter, it can unbind (center right), or be (wrongly) incorporated (bottom).

dents can construct the basic version, then be shown these videos.

The exponential distribution of waiting times implies that there will be episodes with several events happening rapidly, interspersed with long pauses. For this reason, it is useful to view the simulation in two ways: Once with a shorter time step that resolves most individual events but covers only a limited time interval (Video S3), and then with a coarser time step to see the entire synthesis trajectory (Video S4).

We also found it useful (solely for visualization purposes) to alter the distribution of waiting times in a simple way that relieves visual congestion without, we think, too much damage to the realism of the simulation. Our modification, shown in the supplemental videos, was simply to add a small fixed delay, for example one half of one video frame, to every transition waiting time (Fig 5).

2. Classic Hopfield–Ninio mechanism

Following Hopfield (4), we initially assume that the rate constant for incorporation k_{add} is the same regardless of whether the tRNA is correct or incorrect. We also suppose that the binding rates $k'_c = k'_w$ and $\ell'_c = \ell'_w$ have this

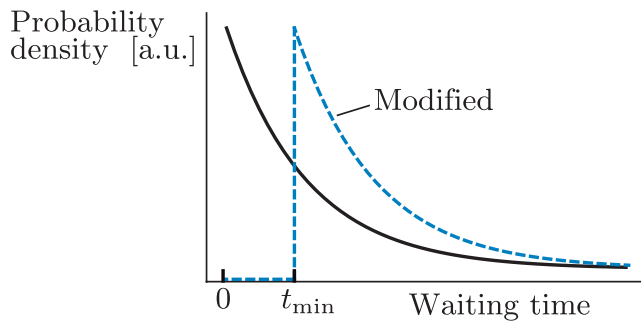


Fig 5. Modified waiting times. Solid line: Exponential distribution of waiting times. Dashed line: Shifted exponential distribution obtained by adding the constant t_{\min} to each sample.

property; for example, they may all be diffusion-limited (3). Only the *unbinding* rates differ in the classic HN mechanism:

$$k_w = \phi_{-1}k_c \text{ and } \ell'_w = \phi_3\ell_c. \quad (2)$$

Here, $\phi_{-1} = 94$ and $\phi_3 = 7.9$ are the preference factors for unbinding the wrong tRNA relative to the correct one before and after GTP hydrolysis, respectively. These values were taken from in vitro measurements (13). Again following Hopfield, in this section we also take the hydrolysis rate constants to be equal: $m'_w = m'_c$ (and $m_w = m_c$). The remaining factor, ϕ_2 for condensation, was set by kinetic consistency (see Data S1.1).

For the rate constants, themselves, each is either a constant probability per unit time (unbinding and hydrolysis) or a probability per unit time with the substrate concentration already lumped in (binding and condensation). The values we chose were appropriate for the concentrations of reactants present in Zaher and Green's (13) in vitro stopped flow experiments. (See Table 1 and Data S1.1 in the Supplemental Material for a list of parameters and an explanation of the rate constants chosen.)

From the preference factors and rate constants mentioned above (summarized in Table 1), the classic HN ribosome had a simulated error of $2/1000 = 0.002$ wrong incorporations, close to the in vitro measured error rate of 0.0016. This result also roughly matches the analytic predicted fraction $(\phi_{-1}\phi_3)^{-1}/(1 + (\phi_{-1}\phi_3)^{-1}) = 0.0013$, where $\phi_{-1}\phi_3 = (94)(7.9) \approx 740$ is the predicted preference factor on the basis of Hopfield's analysis. However, the simulated speed was $\approx 10^{-4}$ amino acids/s, much lower than the in vitro speeds of $\approx 0.25\text{--}8$ amino acids/s. The slow translation rate is a result of the two sequential quasi-equilibrium proofreading steps. When the speed of translation is increased, the error

Table 1. Illustrative rate values. The symbols in the second column are defined in Figure 1. The third column is classic Hopfield–Ninio (HN) proofreading; that is, the only differences between right and wrong tRNA are the off-rates. For comparison, the fourth column is a model that discriminates mainly by forward rates (FRD, section IV.C.3). The fifth column is a ribosome model that uses in vitro measured rates (section IV.C.4) (13, 20, 36). The last column is the same as the HN ribosome, except that guanosine triphosphate (GTP) hydrolysis is now at equilibrium (section IV.C.5). See Data S1 for discussion of numerical values.

Description	Symbol	Value, HN model	FRD	Realistic	Equil.
Binding GTP complex (s^{-1})	k'_c	40	40	40	40
Unbinding GTP complex (s^{-1})	k_c	0.50	0.50	0.50	0.50
Binding GDP complex (s^{-1})	ℓ'_c	0.001	0.001	0.001	0.26
Unbinding GDP complex (s^{-1})	ℓ_c	0.085	0.085	0.085	0.085
Hydrolysis and P_i release (s^{-1})	m'_c	0.01	25	25	0.01
Condensation/ P_i binding (s^{-1})	m_c	0.001	0.001	0.001	0.26
Binding GTP	$k'_w = \phi_1 k'_c$	$\phi_1 = 1$	0.68	0.68	1
Unbinding GTP	$k_w = \phi_{-1} k_c$	$\phi_{-1} = 94$	1	94	94
Binding GDP	$\ell'_w = \phi_{-3} \ell'_c$	$\phi_{-3} = 1$	0.0027	0.0027	1
Unbinding GDP	$\ell_w = \phi_3 \ell_c$	$\phi_3 = 7.9$	1	7.9	7.9
Hydrolysis	$m'_w = \phi_2 m'_c$	$\phi_2 = 1$	0.048	0.048	1
Condensation	$m_w = \phi_{-2} m_c$	$\phi_{-2} = 0.084$	12.09	1	0.084
Incorporation (s^{-1})	$k_{\text{add},c}$	0.001	4.14	4.14	0.001
Incorporation	$k_{\text{add},w} = \phi_{\text{add}} k_{\text{add},c}$	$\phi_{\text{add}} = 1$	0.017	0.017	1

Abbreviations: GDP, guanosine diphosphate; GTP, guanosine triphosphate; P_i , inorganic phosphate.

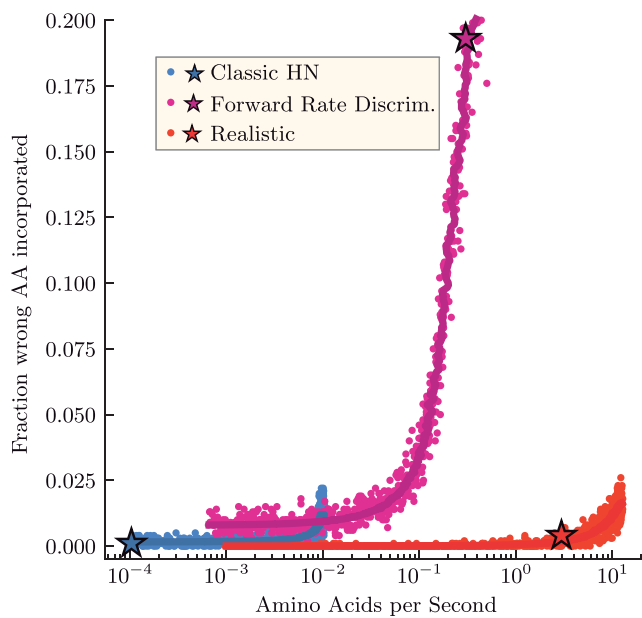


Fig 6. Error and speed tradeoffs for classic Hopfield-Ninio, forward-rate discrimination (FRD, section IV.C.3), and realistic (section IV.C.4) ribosome models. For each ribosome model, the final incorporation rate $k_{\text{add},c}$ was varied from 10^{-3} s^{-1} to 10^3 s^{-1} (1000 amino acids simulated for each $k_{\text{add},c}$). Note that the subscript $k_{\text{add},c}$ is not the same as the average translation rate, which is plotted on the horizontal axis. The average translation rate was determined empirically as the number of amino acids simulated divided by the total time for synthesis. Stochastic simulation results are shown as dots, and theoretical first-passage predictions from Banerjee et al. (36) are shown as solid lines. The stochastic simulation and analytical results agree nicely. To obtain the analytical curves, the error fractions were calculated by Eq. 13 in SI of Banerjee et al. (36) for each $k_{\text{add},c}$ and each $k_{\text{add},c}$ was mapped to a translation speed by stochastic simulation. The resulting curves were smoothed with a moving average (interval of 3). The simulation results and theoretical curve for the HN ribosome does not extend past 0.01 amino acids/s because the translation speed plateaus at that point as $k_{\text{add},c}$ is increased. For reference, the simulation results from the sets of parameters in Table 1 are shown as stars.

fraction also increases as the quasi-equilibrium is broken (Fig 6).

For visualizing the HN model in an animation, we raised the probability of incorrect choices so that wrong incorporations could be seen within a reasonable time frame. To do this, the preference ratios ϕ_{-1} and ϕ_{-3} were both lowered to 5. Videos S3 and S4 show the resulting behavior. Perhaps the most important impression we get from viewing these animations is that *the cell is a busy place*. The riot of activity, the constant binding events that end with no progress (and often not even GTP

hydrolysis), are hallmarks of chemical dynamics that are hard to appreciate in textbook discussions, yet vividly apparent in the simulation. This is especially clear in Video S4, which shows a typical run of 25 amino acid incorporations. Because the simulation shows many unproductive binding and unbinding events, not every event is shown in detail in Video S4. Focusing on just the GDP-tRNA rejections shows that more correct tRNAs than incorrect tRNAs make it past GTP hydrolysis, and that the few incorrect tRNAs that do make it past are mostly rejected in the second proofreading step. However, we also see many *correct* complexes bind and get rejected, before or after GTP hydrolysis. This is the price paid for accuracy in the classic HN mechanism. In the instance shown, only one incorrect amino acid was incorporated out of 25 incorporations, much lower than the expected error fraction of $\phi^{-1}/(1 + \phi^{-1}) = 0.17$ from single-step equilibrium binding (where $\phi = 5$ is the preference factor) and fortuitously close to Hopfield's predicted value $\phi^{-2}/(1 + \phi^{-2}) = 0.038$.

Video S3 provides a more detailed look at this process. The videos also show clearly the jerky, nonuniform progress of synthesis, with some amino acid incorporations happening after much longer delays than others. That feature is by now well documented by single-molecule experiments.

3. Forward-rate discrimination model

Much has been learned about ribosome dynamics (37, 38) after the original insights of Hopfield (4) and Ninio (5). We now know that each step in our model consists of substeps. For example, GTP hydrolysis is subdivided into GTPase activation followed by actual hydrolysis—the latter step probably depends on a rearrangement of monitoring bases in the ribosomal RNA—and so on (39).

The simulation described in section IV.C.2 was designed to show the HN mechanism in its classic, or pure, form and how it can enhance fidelity even without help from the effects just described. For example, we assumed that the only dependence on right versus wrong tRNA was via unbinding rates. Indeed, such depen-

dence was later seen at the single-molecule level (40), but it now appears that some of the forward rates also depend on the identity of the tRNA (13, 41, 42), an effect we will call *forward-rate discrimination*. This forward-rate discrimination (FRD) is in part driven by an induced-fit mechanism, whereby correct codon–anticodon pairing causes conformational changes that accelerate EF-Tu GTP hydrolysis and tRNA accommodation in site A (14–16, 43). To quantify its contribution to the error fraction of the ribosome, we simulated a ribosome with all forward rate ratios the same as that determined *in vitro*, but with backward rate ratios set to 1 (see Table 1, FRD ribosome, and Data S1.4). With these rates, the error was $182/1000 = 0.182$, much higher than the HN ribosome error of 0.002, and the speed was 0.32 amino acids/s, which is also higher than the HN ribosome speed of 10^{-4} amino acids/s and closer to the range of *in vitro* translation rates (0.25–8 amino acids/s). Figure 5 shows that, similar to the classic HN ribosome, the FRD-only ribosome achieves lower error at slower translation rates, but at all speeds, the FRD ribosome has higher error than the HN ribosome. The FRD-only ribosome also must trade off a significant amount of accuracy to achieve *in vitro* translation speeds, with the proportion of wrong incorporations >0.3 at 1–10 amino acids/s.

Overall, comparing the FRD-only and HN-only models reveals that off-rate differences account for almost all of the translation accuracy, whereas on-rate differences increase the efficiency and speed of translation.

4. More realistic model

The realistic ribosome with all rates and ratios set by Zaher and Green's (13) *in vitro* measurements combines the off-rate and forward-rate discrimination described above (Table 1, fifth column; Data S1.2). The realistic model demonstrates that a combination of both strategies allows the ribosome to achieve both low error and high speed. For a simulation of 10 000 amino acids, the error was $18/10\ 000 = 0.0018$, which is similar to the *in vitro* ribosome error, but the speed was 2.89 amino acids/s, faster than both the HN and FRD

ribosome and within the range of *in vitro* translation rates. Figure 6 shows that the realistic ribosome does not have the speed limits of the HN ribosome or the costly tradeoff of accuracy for speed of the FRD ribosome.

Video S5 shows an animation of a simulation with the realistic ribosome rates. We see both a bias for correct tRNA binding/hydrolysis and a bias for rejection of wrong tRNAs before GTP hydrolysis. Of the 26 correct tRNA binding events in this run, 25 resulted in successful incorporation. This is much more efficient compared with the fraction $24/10\ 245 = 0.002$ of productive correct tRNA binding events in the classic HN ribosome simulation in Videos S3 and S4. Additionally, of the 30 incorrect tRNA binding events on the realistic ribosome, all 30 resulted in rejection.

The real ribosome uses both the classic HN mechanism (quasi-equilibrium, energetic proofreading) and forward-rate discrimination; our results show that both are required to optimize speed, efficiency, and accuracy. Despite this, simulating the classic HN mechanism is still a valuable exercise for students. Most of the discrimination power is accounted for by the HN-only ribosome (Fig 6), and by visualizing discrimination via only a difference in unbinding rates, students see the minimal components necessary to attain high accuracy in a broad class of biological reactions. The classic HN mechanism illustrates an essential part of biological proofreading that fundamentally relies on nonequilibrium physics. Recent evidence also points to *two* kinetic proofreading steps—that is, two sequential, nearly irreversible steps, each of which can be followed by unbinding of tRNA (44, 45). Our simulation could be extended to include such effects, whereas analytic methods would quickly become intractable.

5. Role of thermodynamic driving force

Next, we return to the classic HN mechanism, this time operating at nearly equilibrium concentrations of GTP, GDP, and P_i to demonstrate the importance of the one-way door (the nonequilibrium GTP hydrolysis step). Table 1 summarizes the rates we chose for this undriven model (see Data S1.3 for more

detailed description). Before a discussion, students can be given these values and asked what behavior they find.

With these rates, the reaction still creates a peptide chain, because we assumed a fixed probability per time to add an amino acid irreversibly whenever the ribosome visits its activated state. However, this time, the simulation gave an error of $1683/15\,000 = 0.1122$ and a speed of $\approx 4 \times 10^{-5}$ amino acids/s. The error fraction of 0.1122 is from GDP-tRNA directly binding the ribosome, which is consistent with an expected error fraction of $\phi^{-1}/(1 + \phi^{-1}) = 0.1124$, where $\phi = 7.9$. This large error illustrates the significance of the thermodynamic driving force, as the ribosome in equilibrium is both slow and error-prone.

To gain more insight into the role of the irreversible GTP hydrolysis step, some students may wish to rerun the simulation with different incorporation and hydrolysis rates. For example, a simulation with $m'_w = 25$ and $k_{\text{add}} = 4.14$ results in a simulation with many tRNAs flipping between GDP and GTP states, another way in which the two discrimination steps effectively merge into one.

V. CONCLUSION

The models described here show fairly elementary physical principles that lie at the heart of cell biology. Specifically, gene expression and kinetic proofreading are two important, fundamental topics that are within reach of undergraduates.

A module that introduces stochastic simulation need not dominate a semester course: one class week is enough for the first exposure. Indeed, the entire simulation plus visualization in Computer Code S1 consists of just *seven short lines* of code, and yet it creates a valuable educational experience not available in a static textbook. Moreover, the framework is not specifically biological in character; it can serve as a stepping stone to more complex simulations relevant for a variety of courses.

Finally, the stochastic stimulations used here allowed comparison of different models for error correction, and the comparison of differ-

ent minimal models is an important skill for biophysics students.

SUPPLEMENTAL MATERIAL

Supplemental videos for Scientific and Pedagogical Background and Results and Discussion, computer codes for Methods, and data for Results and Discussion are available at: <https://doi.org/10.35459/tbp.2019.000101>.

ACKNOWLEDGMENTS

We are grateful to Ned Wingreen and Anatoly Kolomeisky for correspondence and to Bruce Sherwood for help with software. This work was partially supported by the United States National Science Foundation (NSF) under grants PHY-1601894 and MCB-1715823. Some of the work was done at the Aspen Center for Physics, which is supported by NSF grant PHY-1607611, and at the Physical Biology of the Cell summer school (Marine Biological Laboratory). The authors declare no conflicts of interest.

AUTHOR CONTRIBUTIONS

KYC and PCN designed the research. KYC and PCN carried out all simulations based in part on suggestions in unpublished work by DMZ. KYC and PCN wrote the article, and all authors edited it.

REFERENCES

- Chabay, R. W., and B. A. Sherwood. 2015. *Matter and Interactions*. 4th edition. Wiley, New York.
- Simon, A., and A. Libchaber. 1992. Escape and synchronization of a Brownian particle. *Phys Rev Lett* 68:3375–3378.
- Phillips, R., J. Kondev, J. Theriot, and H. Garcia. 2012. *Physical Biology of the Cell*. 2nd edition. Garland Science, New York.
- Hopfield, J. J. 1974. Kinetic proofreading: a new mechanism for reducing errors in biosynthetic processes requiring high specificity. *Proc Natl Acad Sci U S A* 71:4135–4139.
- Ninio, J. 1975. Kinetic amplification of enzyme discrimination. *Biochimie* 57:587–595.
- Alon, U. 2006. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Chapman and Hall/CRC Press, Boca Raton, FL.
- Sartori, P., and S. Pigolotti. 2013. Kinetic versus energetic discrimination in biological copying. *Phys Rev Lett* 110:188101.
- Nelson, P. 2015. *Physical Models of Living Systems*. W. H. Freeman and Co., New York.
- Kinder, J. M., and P. Nelson. 2018. *A Student's Guide to Python for Physical Modeling*. Updated edition. Princeton University Press, Princeton, NJ.
- Morgan, W. A., and L. Q. English. 2019. VPython for Introductory Mechanics: Complete Version. Accessed May 2018. <https://scholar.dickinson.edu/vpythonphysics/1>.
- Gillespie, D. T. 1976. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J Comput Phys* 22:403–434.
- Gillespie, D. T. 1977. Exact stochastic simulation of coupled chemical reactions. *J Phys Chem* 81:2340–2361.
- Zaher, H. S., and R. Green. 2010. Hyperaccurate and error-prone ribosomes exploit distinct mechanisms during tRNA selection. *Mol Cell* 39:110–120.
- Gromadski, K. B., and M. V. Rodnina. 2004. Kinetic determinants of high-fidelity tRNA discrimination on the ribosome. *Mol Cell* 13:191–200.
- Ogle, J. M., F. V. Murphy, M. J. Tarry, and V. Ramakrishnan. 2002. Selection of tRNA by the ribosome requires a transition from an open to a closed form. *Cell* 111:721–732.

16. Pape, T., W. Wintermeyer, and M. Rodnina. 1999. Induced fit in initial selection and proofreading of aminoacyl-tRNA on the ribosome. *EMBO J* 18:3800–3807.
17. Wagner, E. G., P. C. Jelenc, M. Ehrenberg, and C. G. Kurland. 1982. Rate of elongation of polyphenylalanine in vitro. *Eur J Biochem* 122:193–197.
18. Capece, M. C., G. L. Kornberg, A. Petrov, and J. D. Puglisi. 2015. A simple real-time assay for in vitro translation. *RNA* 21:296–305.
19. Zuckerman, D. M. 2018. Physical lens on the cell: active (“kinetic”) proofreading, accessed May 2018. <http://physicallensonthecell.org/cell-biology-phenomena/active-kinetic-proofreading>.
20. Banerjee, K., A. B. Kolomeisky, and O. A. Igoshin. 2017. Elucidating interplay of speed and accuracy in biological error correction. *Proc Natl Acad Sci U S A* 114:5183–5188.
21. Hopfield, J. J., T. Yamane, V. Yue, and S. M. Coutts. 1976. Direct experimental evidence for kinetic proofreading in amino acylation of tRNA^{le}. *Proc Natl Acad Sci U S A* 73:1164–1168.
22. Yamane, T., and J. J. Hopfield. 1977. Experimental evidence for kinetic proofreading in the aminoacylation of tRNA by synthetase. *Proc Natl Acad Sci U S A* 74:2246–2250.
23. Choi, P. J., L. Cai, K. Frieda, and X. S. Xie. 2008. A stochastic single-molecule event triggers phenotype switching of a bacterial cell. *Science* 322:442–446.
24. Eldar, A., and M. B. Elowitz. 2010. Functional roles for noise in genetic circuits. *Nature* 467:167–173.
25. Lidstrom, M. E., and M. C. Konopka. 2010. The role of physiological heterogeneity in microbial population behavior. *Nat Chem Biol* 6:705–712.
26. Carey, J. N., E. L. Mettert, M. Roggiani, K. S. Myers, P. J. Kiley, and M. Goulian. 2018. Regulated stochasticity in a bacterial signaling network permits tolerance to a rapid environmental change. *Cell* 173:196–207.e14.
27. Golding, I., J. Paulsson, S. M. Zawilski, and E. C. Cox. 2005. Real-time kinetics of gene activity in individual bacteria. *Cell* 123:1025–1036.
28. Golding, I., and E. C. Cox. 2008. Chapter 8. Spatiotemporal dynamics in bacterial cells: real-time studies with single-event resolution. *Methods Cell Biol* 89:223–251.
29. Jones, D. L., R. C. Brewster, and R. Phillips. 2014. Promoter architecture dictates cell-to-cell variability in gene expression. *Science* 346:1533–1536.
30. Phillips, R., N. M. Belliveau, G. Chure, H. G. Garcia, M. Razo-Mejia, and C. Scholes. 2019. Figure 1 theory meets figure 2 experiments in the study of gene expression. *Annu Rev Biophys* 48:121–163.
31. Raj, A., C. S. Peskin, D. Tranchina, D. Y. Vargas, and S. Tyagi. 2006. Stochastic mRNA synthesis in mammalian cells. *PLOS Biol* 4:e309.
32. Sanchez, A., and J. Kondev. 2008. Transcriptional control of noise in gene expression. *Proc Natl Acad Sci U S A* 105:5081–5086.
33. Iyer-Biswas, S., F. Hayot, and C. Jayaprakash. 2009. Stochasticity of gene products from transcriptional pulsing. *Phys Rev E Stat Nonlin Soft Matter Phys* 79:031911.
34. Iyer-Biswas, S. 2009. Applications of methods of non-equilibrium statistical physics to models of stochastic gene expression. PhD diss., Ohio State University: Columbus.
35. Mugler, A., A. M. Walczak, and C. H. Wiggins. 2009. Spectral solutions to stochastic models of gene expression with bursts and regulation. *Phys Rev E Stat Nonlin Soft Matter Phys* 80:041921.
36. Banerjee, K., A. B. Kolomeisky, and O. A. Igoshin. 2017. Accuracy of substrate selection by enzymes is controlled by kinetic discrimination. *J Phys Chem Lett* 8:1552–1556.
37. Rodnina, M. V. 2011. Mechanisms of decoding and peptide bond formation. In *Ribosomes: Structure, Function, and Dynamics*. M. V. Rodnina, W. Wintermyer, and R. Green, editors. Springer, New York, pp. 199–212.
38. Bahar, I., R. L. Jernigan, and K. A. Dill. 2017. *Protein Actions: Principles and Modeling*. Garland Science, New York.
39. Satpati, P., J. Sund, and J. Aqvist. 2014. Structure-based energetics of mRNA decoding on the ribosome. *Biochemistry* 53:1714–1722.
40. Blanchard, S. C., R. L. Gonzalez, H. D. Kim, S. Chu, and J. D. Puglisi. 2004. tRNA selection and kinetic proofreading in translation. *Nat Struct Mol Biol* 11:1008–1014.
41. Prabhakar, A., J. Choi, J. Wang, A. Petrov, and J. D. Puglisi. 2017. Dynamic basis of fidelity and speed in translation: coordinated multistep mechanisms of elongation and termination. *Protein Sci* 26:1352–1362.
42. Rodnina, M. V., N. Fischer, C. Maracci, and H. Stark. 2017. Ribosome dynamics during decoding. *Philos Trans R Soc Lond B Biol Sci* 372:20160182.
43. Gromadski, K. B., T. Daviter, and M. V. Rodnina. 2006. A uniform response to mismatches in codon-anticodon complexes ensures ribosomal fidelity. *Mol Cell* 21:369–377.
44. Jeong, K.-W., U. Uzun, M. Selmer, and M. Ehrenberg. 2016. Two proofreading steps amplify the accuracy of genetic code translation. *Proc Natl Acad Sci U S A* 113:13744–13749.
45. Chen, J., J. Choi, S. E. O’Leary, A. Prabhakar, A. Petrov, R. Grosely, E. V. Puglisi, and J. D. Puglisi. 2016. The molecular choreography of protein synthesis: translational control, regulation, and pathways. *Q Rev Biophys* 49:e11.