seq2ribo: Structure-aware integration of machine learning and simulation to predict ribosome location profiles from RNA sequences

Gün Kaynar[1] and Carl Kingsford[1]

[1]Ray and Stephanie Lane Computational Biology Department, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA

February 9, 2026

Abstract

Motivation: Ribosome dynamics are vital in the process of protein expression. Current methods rely on ribosome profiling (Ribo-seq), RNA-seq profiles, and full genomic context. This restricts their use in de novo sequence design, like messenger RNA (mRNA) vaccines. Simulation-only approaches like the Totally Asymmetric Simple Exclusion Process (TASEP) oversimplify translation by focusing solely on codon elongation times.
Results: We present seq2ribo, a hybrid simulation and machine learning framework that predicts ribosome A-site locations using only an mRNA sequence as input. Our method first employs a novel structure-aware TASEP (sTASEP), which models translation using a comprehensive set of fitted parameters that include codon wait times and structural features, such as local angles, base-pairing, and discrete positional buckets. The ribosome locations generated by sTASEP are then processed by a polisher model, which learns to refine the simulated ribosome distributions. seq2ribo provides high-fidelity predictions of ribosome locations across diverse cell types (iPSC, HEK293, LCL, and RPE-1), significantly outperforming baselines. When benchmarked against sequence-only Translatomer, seq2ribo achieves reductions in transcript-level error up to 35.8%, while simultaneously attaining the highest Pearson and Spearman correlations in every cell line and reducing structural errors between 43.3% and 97.3%. By adding a task-specific head, seq2ribo achieves Spearman correlations up to 0.795 with experimental translation efficiency (TE) across several cell lines, and 0.689 with measured protein expression. By operating from sequence alone, seq2ribo provides a new tool for synthetic biology, enabling the rational design and optimization of mRNA sequences without the need for expression-level data or genomic context.
Availability: seq2ribo is available at https://github.com/Kingsford-Group/seq2ribo.
Contact: gkaynar@cs.cmu.edu, carlk@cs.cmu.edu.
Supplementary information: Supplementary data are available.

1 Introduction

The translation of messenger RNA (mRNA) into protein is a fundamental biological process and a primary determinant of protein expression levels in the cell. Translation does not proceed as a uniform, steady-state production line. It is a dynamic system governed by initiation, elongation, and termination (Hershey et al., 2012). Ribosomes move at variable speeds, pause at specific

sequences (Yu et al., 2015), and form traffic jams that influence protein output, co-translational folding, and mRNA stability (Jia et al., 2024; Ingolia et al., 2009, 2012). Quantitative models of these ribosome dynamics are therefore central to molecular biology and synthetic biology.

Translation rate and translation efficiency (TE) are key quantitative outcomes of this process (Ingolia et al., 2009; Schwanhäusser et al., 2011; Zarai et al., 2017). They partially emerge from the combined effects of codon usage and mRNA structure. The mRNA folds into structured elements such as hairpins, loops, and pseudoknots (Rouskin et al., 2014). These structures act as physical barriers that slow or stall ribosomes (Caliskan et al., 2014; Bao et al., 2020; Lin et al., 2022). At the same time, the degeneracy of the genetic code means that many different coding sequences can encode the same protein. These synonymous sequences fold into distinct structures and produce very different ribosome traffic patterns and expression levels (Mauro and Chappell, 2014). This makes sequence-level prediction of translation dynamics both challenging and attractive, especially for the rational design of mRNA therapeutics such as vaccines and protein replacement therapies (Pardi et al., 2018; Verbeke et al., 2022; Leppek et al., 2022; Jin et al., 2025). Sequence-only deep learning models have also been developed for related tasks such as translation initiation site prediction (Zhang et al., 2017) and RNA subcellular localization (Yan et al., 2019; Cao et al., 2018).

Ribosome profiling (Ribo-seq) experiments produce codon-level ribosome occupancy profiles by sequencing ribosome-protected mRNA fragments (Ingolia et al., 2012; Spealman et al., 2016; Ingolia et al., 2009; Ahmed et al., 2019; Ingolia et al., 2019; Wang et al., 2016). These data have enabled analysis tools that detect translated open reading frames, normalize ribosome profiles, and quantify translation from existing experiments (Calviello et al., 2016; Xiao et al., 2018; O'Connor et al., 2016). Most computational models use these experimental profiles as input rather than predicting them de novo. Translatomer (He et al., 2024) predicts Ribo-seq signal by integrating genomic sequence with cell-type-specific RNA-seq coverage. It also includes a sequence-only variant, but this simplified model underperforms the full sequence-plus-RNA-seq approach.

Sequence-based TE models such as RiboNN learn to predict TE directly from RNA sequence, using experimental TE labels only during training (Zheng et al., 2025). These approaches demonstrate that sequence carries a strong signal about translational output, but they still require experimental labels for each transcript during model training or rely on additional expression measurements, and they do not produce codon-level ribosome profiles for novel sequences. At the other end of the spectrum, simulation-based approaches such as the totally asymmetric simple exclusion process (TASEP) model ribosome traffic on a one-dimensional lattice (Spitzer, 1970; MacDonald et al., 1968; Zia et al., 2011). Classical implementations treat resistance along the mRNA as a function of codon-specific elongation times and tRNA availability (Dong et al., 2007; von der Haar, 2012) and do not account for mRNA structure, even though structural elements and sequence motifs are known to induce strong ribosome pausing and stalling (Qu et al., 2011; Bao et al., 2020).

We develop `seq2ribo` (Figure 1), a hybrid framework that combines mechanistic simulation with deep learning to predict high-fidelity ribosome dynamics from nucleotide sequence alone. `seq2ribo` consists of two components. First, we introduce sTASEP, a structure-aware totally asymmetric simple exclusion process that models translation with a fitted parameter set that includes codon wait times and additional wait times for local base pairing, structural angle changes, and positional buckets along the coding sequence (CDS). Second, we pass the ribosome distributions that sTASEP generates through a polisher based on the Mamba (Gu and Dao, 2024) structured state space model, which refines the simulated A-site profiles into final ribosome count predictions.

We demonstrate the generalizability of our framework by training and evaluating `seq2ribo` on four distinct cell lines: induced pluripotent stem cells (iPSC), human embryonic kidney 293 (HEK293), lymphoblastoid cell lines (LCL), and hTERT-immortalized retinal pigment epithelial cells (RPE-1). We show that sTASEP consistently improves on a classical TASEP baseline, and that the full

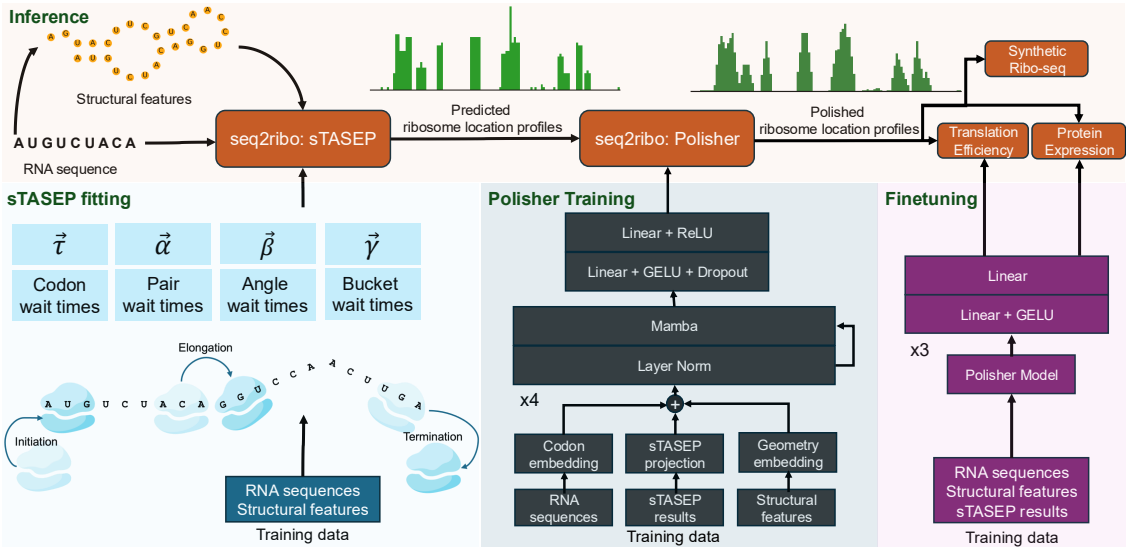


Figure 1: Overview of `seq2ribo` during inference. Given an input RNA sequence, we first derive codon-level structural features and pass the sequence and features into the sTASEP module of `seq2ribo`, which simulates a raw ribosome location profile. The polisher module of `seq2ribo` then combines the codon sequence, structural features, and simulated profile to produce a polished ribosome location profile. We reuse this shared polished profile for two sequence-only downstream tasks. We predict translation efficiency (TE) for human transcripts and predict protein expression, and we can also use the predicted ribosome profiles to generate synthetic ribosome profiling data. The bottom panels summarize how we fit the sTASEP parameters, train the polisher model, and finetune `seq2ribo` for the downstream regression tasks.

seq2ribo pipeline substantially outperforms a sequence-only Translatomer baseline. Specifically, seq2ribo reduces per-transcript mean absolute error (MAE) by 27.3% to 35.8%. Consistent with this, seq2ribo achieves the highest Pearson and Spearman correlations in every cell line, indicating improved location-specific agreement with measured ribosome occupancy even when MAE alone is not fully informative. We also find that seq2ribo improves coarse-grained structural agreement with observed profiles, reducing structural errors relative to Translatomer across all four cell lines, including large drops of 97.3% in RPE-1 and 92.7% in iPSC. We further evaluate seq2ribo as a sequence-only predictor for downstream tasks. For translation efficiency (TE), we outperform the sequence-based RiboNN baseline across all datasets, increasing Spearman correlation with experimental measurements to 0.769 on the mean label set and up to 0.795 on cell-type-specific sets. For protein expression, we achieve a Spearman correlation of 0.689 on an external expression dataset. Together, these results show that predicted ribosome count profiles from seq2ribo serve as strong features for TE and expression. Moreover, seq2ribo functions as a synthetic Ribo-seq data generator, enabling the in silico profiling of arbitrary sequences and providing a practical tool for de novo mRNA design.

2 Methods

2.1 Dataset

We obtain global aggregate A-site ribosome counts for iPSC (Chen et al., 2020), LCL (Cenik et al., 2015), HEK293 (Wagner et al., 2020), and RPE-1 (Haas et al., 2020) cell lines from the GWIPS-viz Ribo-seq resource (Michel et al., 2014). We restrict all analyses to coding sequences and apply the preprocessing and filtering pipeline described in detail in Section 1.1 of the Supplementary Text. For each retained transcript, we have a codon sequence and a matched A-site ribosome count sequence of the same length. For each CDS, we also construct three codon-level structural feature sequences that have the same length as the codon sequence. We generate these structural features using ViennaRNA (Lorenz et al., 2011) and describe the folding and feature construction procedure in Section 1.2 of the Supplementary Text. To verify that the test splits assess generalization to novel sequences rather than memorization of near-neighbors, we perform a train-test similarity analysis in Section 1.16 of the Supplementary Text.

For translation efficiency (TE), we define four distinct prediction tasks based on experimental values from Zheng et al. (2025). We use three cell-line-specific TE measurements (HEK293, RPE-1, and LCL) and a mean TE measurement. We match these labels to the coding sequences in our dataset to create matched datasets of transcripts that possess codon sequence, A-site ribosome counts, and the corresponding experimental TE label. For each task, we split the matched transcripts into training (80%), validation (10%), and test (10%) sets using stratification by the total A-site ribosome count per transcript. The exact transcript counts for each task are provided in Section 1.1 of the Supplementary Text.

For experimental protein expression, we use a protein (mRFP) expression dataset of Nieuwkoop et al. (2023). This dataset contains 1,459 coding sequences with measured protein production levels and a predefined split into 1,021 training sequences, 219 validation sequences, and 219 test sequences (Li et al., 2024). For each mRFP codon sequence, we compute the same codon-level pair, angle, and bucket features and retain the codon sequence, structural feature sequences, and ground truth protein expression label.

2.2 Problem formulation

The inputs and outputs share the same structure across cell lines (iPSC, HEK293, LCL, RPE-1). We define the notation for a generic dataset D. Let $D = \{(C_i, P_i, A_i, B_i, Y_i)\}_{i=1}^{|D|}$ denote the full dataset for the given cell line, where $|D|$ is the sample size.

We partition D into three disjoint subsets: the training set D_{train}, the validation set D_{val}, and the test set D_{test}. Each sample i corresponds to a coding sequence (CDS) with variable length N_i codons. We denote the codon sequence by

$$C_i = \left(C_i^1, \ldots, C_i^{N_i}\right),$$

where C_i^j is the j-th codon in the CDS of sample i. For the same sample i, we define three codon-level structural feature sequences. We denote the pair count sequence by

$$P_i = \left(p_i^1, \ldots, p_i^{N_i}\right), \quad p_i^j \in \{0, 1, 2, 3\}.$$

The value p_i^j records how many nucleotides in codon C_i^j participate in base pairing in the predicted RNA structure. We denote the angle sequence by

$$A_i = \left(a_i^1, \ldots, a_i^{N_i}\right), \quad a_i^j \in \{0, 1, 2, 3\}.$$

The value a_i^j quantifies the change in local backbone angle at codon C_i^j relative to the previous codon in the predicted two-dimensional structure. We discretize continuous angle changes into four bins that represent no change, small change, medium change, and large change. We derive these bins from the distribution of angle changes in the training set and describe this procedure in Section 1.3 of the Supplementary Text. We denote the bucket sequence by

$$B_i = \left(b_i^1, \ldots, b_i^{N_i}\right), \quad b_i^j \in \{1, 2, 3\}.$$

The value b_i^j encodes the relative position of codon C_i^j along the CDS. Codons in the first third of the sequence receive bucket value 1. Codons in the middle third receive bucket value 2. Codons in the final third receive bucket value 3. For each sample i, we observe A-site ribosome counts at each codon position, where each count is an integer in $\{0, 1, 2, \ldots\}$. We denote the target count sequence by $Y_i = \left(y_i^1, \ldots, y_i^{N_i}\right)$, where y_i^j is the observed A-site ribosome count at codon C_i^j. We use an integer $K \geq 1$ to denote the number of independent simulation runs that we generate for each sample. We implement a stochastic simulator

$$f : (C_i, P_i, A_i, B_i; \xi_{i,r}) \longmapsto \hat{Y}_i^{(r)}, \quad r = 1, \ldots, K,$$

that maps the codon sequence and all structural features to a sequence of simulated A-site counts, where $\xi_{i,r}$ denotes simulator randomness for run r. For each sample i, we obtain K simulator outputs

$$\hat{Y}_i^{(r)} = \left(\hat{y}_i^{1,(r)}, \ldots, \hat{y}_i^{N_i,(r)}\right), \quad r = 1, \ldots, K.$$

Each $\hat{Y}_i^{(r)}$ is the output of one independent simulation run, where $\hat{y}_i^{j,(r)}$ represents the simulated A-site count at the j-th codon of sample i in run r. We report the value of K used in each experiment in the corresponding sections. We collect the K simulated sequences for sample i into the set

$$\widehat{\mathcal{Y}}_i = \{\hat{Y}_i^{(r)}\}_{r=1}^{K}.$$

We train a polisher that refines a single simulator output at a time. We learn a function

$$g : (C_i, P_i, A_i, B_i, \hat{Y}_i^{(r)}) \longmapsto \tilde{Y}_i^{(r)},$$

that refines the simulator outputs. The polisher receives the same codon and structural features, and produces polished predictions

$$\tilde{Y}_i^{(r)} = g\big(C_i, P_i, A_i, B_i, \hat{Y}_i^{(r)}\big) = \big(\tilde{y}_i^{1,(r)}, \ldots, \tilde{y}_i^{N_i,(r)}\big).$$

For polisher training, we treat each pair (i, r) as an independent example.

We evaluate models with eight metrics that summarize agreement between predicted and observed A-site counts at different levels of aggregation. elemwiseMAE measures the average absolute error across all codon positions. pertranscriptMAE computes an absolute error for each transcript and then averages across transcripts so each transcript contributes equally. codonMAE summarizes the error in total counts for each codon identity. pairMAE aggregates error by base pairing category. angleMAE aggregates error by discretized local backbone angle change. bucketMAE aggregates error across the first, middle, and final thirds of each CDS. We also report elementwise Pearson and Spearman correlations, denoted r and ρ, computed between predicted and observed A-site counts across all codon positions in all transcripts. Because we compute both pair and angle errors over four discrete groups, pairMAE and angleMAE can occasionally take identical values when the per-group aggregated errors coincide. We describe the calculation of each metric in Section 1.4 of the Supplementary Text. Lower values are better for the six MAE metrics, and higher values are better for p and ρ.

2.3 The Structure-Aware TASEP (sTASEP)

We implement the simulator f as a structure-aware totally asymmetric simple exclusion process (sTASEP). For each transcript i, we simulate ribosomes that move along a one-dimensional lattice of codon positions $j = 1, \ldots, N_i$. Each ribosome occupies a single codon index and excludes other ribosomes within three codons upstream and three codons downstream along the mRNA (Brar and Weissman, 2015). At each simulation step, we sample three types of stochastic events. **Initiation** introduces a new ribosome at the first codon if codon positions 1 through 4 are free. **Elongation** moves a ribosome from position j to $j+1$ only if the downstream exclusion zone is empty and proceeds with an elongation rate

$$r_i^j = \frac{1}{\max\big(w_i^j, \varepsilon\big)}, \qquad \varepsilon = 10^{-12},$$

where w_i^j is a codon-specific wait time that combines sequence and structural effects at that position,

$$w_i^j = \tau_{(C_i^j)} + \alpha_{p_i^j} + \beta_{a_i^j} + \gamma_{b_i^j}.$$

Here $\tau_{(C_i^j)}$ is the wait time for the non-stop codon at position j, $\alpha_{p_i^j}$ is the wait time for the pair category p_i^j, $\beta_{a_i^j}$ is the wait time for the angle category a_i^j and $\gamma_{b_i^j}$ is the wait time for the bucket category b_i^j. We maintain one parameter for each of the 61 non-stop codons, one parameter for each of the four pair categories, one parameter for each of the four angle categories, and one parameter for each of the three bucket categories. **Termination** removes a ribosome when it reaches the stop codon and records a termination event.

We fit a distinct set of wait time parameters for each cell line. These parameters are optimized by iterating Monte Carlo simulations and group-wise parameter updates that align aggregated simulated A-site counts with the observed A-site counts for that specific cell line. We describe the full simulation schedule, parameter initialization, and update algorithm in Section 1.5 of the Supplementary Text. As an alternative initialization strategy, we also derive analytical first estimates of the wait time parameters directly from observed ribosome occupancy, as described in Section 1.10 of the Supplementary Text.

2.4 Structured state space polisher

We implement the polisher g as a structured state space model based on Mamba (Gu and Dao, 2024). The model refines the simulated A-site counts produced by sTASEP and outputs polished codon-level predictions. For a transcript i with length N_i, we construct five input channels. We embed codon identity, the pair sequence, the angle sequence, the bucket sequence, and the simulated A-site counts into a shared latent dimension and sum these embeddings position-wise to obtain an initial representation $H_i^{(0)}$ of length N_i.

The core of the polisher consists of stacked Mamba blocks. Each block applies layer normalization (LN), a Mamba state space layer, and a residual connection,

$$Z_i^{(\ell)} = \mathrm{LN}\big(H_i^{(\ell-1)}\big), \quad U_i^{(\ell)} = \mathrm{Mamba}\big(Z_i^{(\ell)}\big), \quad H_i^{(\ell)} = H_i^{(\ell-1)} + U_i^{(\ell)},$$

for block index ℓ. The Mamba layers operate along the sequence dimension and share parameters across positions. After the final Mamba block, we apply a position-wise feed-forward network and a final linear layer to map each position to a single non-negative scalar. We select the output activation function based on validation performance for each cell line. For iPSC and HEK293, we use a standard ReLU nonlinearity. For LCL and RPE-1, we observed that ReLU produced an excessive number of zero-valued predictions; for these datasets, we therefore substitute ReLU with a Softplus nonlinearity to ensure smoother non-negative counts.

We train a separate polisher g for each cell line using the corresponding training set D_{train}. Each training example consists of $(C_i, P_i, A_i, B_i, \hat{Y}_i^{(r)})$ as inputs and the observed A-site counts Y_i as targets. We minimize a composite loss that combines six MAE metrics, and we select the best checkpoint by validation loss. We provide embedding dimensions, the exact number of layers, and all optimization hyperparameters in Section 1.6 of the Supplementary Text. We define the composite loss and describe the choice of the six loss weights in Section 1.7 of the Supplementary Text. We use the resulting polisher together with the fitted sTASEP simulator as the `seq2ribo` pipeline in all experiments. We provide a detailed analysis of the computational hardware, theoretical complexity, and runtime estimates for both simulation and training in Section 2 of the Supplementary Text.

2.5 Downstream tasks

To demonstrate that the ribosome count profiles produced by `seq2ribo` form a good basis for downstream tasks, we place two regression heads on top of the polisher output. One head predicts TE, and one head predicts protein expression. In both cases, the polisher first produces a codon-level count profile for each transcript and a task-specific head then aggregates this profile to a single scalar prediction.

2.5.1 Translation efficiency prediction

We address the four TE tasks described in Section 2.2 by training a specific prediction head for each task. For each transcript, we treat the polished codon-level profile $\tilde{Y}_i = (\tilde{y}_i^1, \ldots, \tilde{y}_i^{N_i})$ as a sequence of features. We apply masked mean pooling over codon positions to obtain a fixed-length transcript representation that ignores padded positions, and we pass this representation through a small feed-forward TE head that outputs a single scalar prediction. We train each head with a mean squared error (MSE) loss between predicted and experimental TE values. We average predictions over multiple simulated and polished profiles per transcript to reduce variance. We describe the head architecture, the number of simulations, and all training details in Section 1.8 of the Supplementary Text.

2.5.2 Protein expression prediction

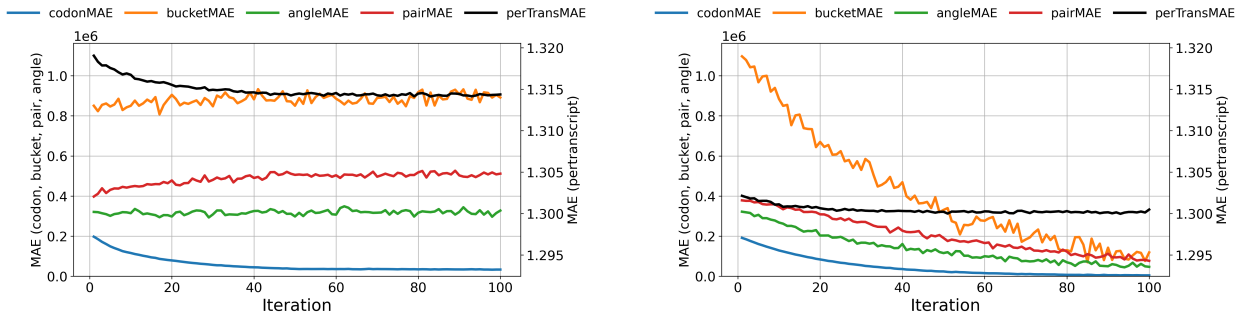
For the mRFP protein expression dataset, we apply the same strategy. We run sTASEP on each sequence, polish the simulated A-site counts to obtain a codon-level profile, apply masked mean pooling to get a fixed-length transcript representation, and pass this representation through an expression head with the same architecture as the TE head. We finetune the full model on the mRFP training split, use the validation split for early stopping, and evaluate on the test split. We again average predictions over many simulated and polished profiles per sequence. We provide the exact simulation counts and training setup in Section 1.8 of the Supplementary Text.

2.6 Baseline models

We benchmark both components of `seq2ribo` against internal and external baselines. We first compare the sTASEP simulator to a classical codon-level TASEP model that fits only codon-specific wait times. We fit separate TASEP baselines for each cell line. We then assess the benefit of the Mamba-based polisher by replacing the structured state space backbone with a transformer-based architecture that uses the same input embeddings and a comparable number of parameters. We also train a variant of the Mamba polisher that optimizes only the pertranscriptMAE. We describe the classical TASEP formulation, the transformer architecture, and the pertranscriptMAE-only polisher ablation in Sections 1.9, 1.13, and 1.14 of the Supplementary Text.

As an external baseline for ribosome profile prediction, we use Translatomer (He et al., 2024). In its original form, Translatomer ingests genomic DNA in a fixed-width window together with RNA-seq coverage and predicts ribosome footprint density. In contrast, `seq2ribo` ingests CDS only and predicts A-site ribosome counts. To adapt Translatomer to our setting, we train a sequence-only Translatomer variant that uses the same genomic DNA window input but omits RNA-seq coverage from both the architecture and the inputs, and instead predicts A-site ribosome counts. We train four separate Translatomer (sequence-only) models, one per cell line, and evaluate each model on the test set from its corresponding cell line. We provide training details for sequence-only Translatomer in Section 1.11 of the Supplementary Text.

For TE prediction, we compare the `seq2ribo` TE head with RiboNN (Zheng et al., 2025). RiboNN ingests RNA sequences and outputs predicted TE values. We train separate RiboNN models for each of the four TE tasks using the same train, validation, and test splits and the same labels that we use for the `seq2ribo` TE head so that both models see the same data. We give architecture and training details for RiboNN in Section 1.12 of the Supplementary Text.



(a) iPSC TASEP

(b) iPSC sTASEP

Figure 2: Loss convergence for classical TASEP and sTASEP fits on the iPSC dataset.

3 Results

3.1 `seq2ribo`

We fit kinetic parameters for both the classical TASEP model and our sTASEP simulator separately for each cell line dataset. We evaluate both simulators with eight metrics, namely pertranscript-MAE, elemwiseMAE, codonMAE, pairMAE, angleMAE, and bucketMAE, as well as Pearson and Spearman correlations. Figure 2 shows training dynamics for the iPSC dataset. Training dynamics for HEK293, LCL, and RPE-1 appear in Supplementary Figures S19–S24. Table 1 reports test set losses for TASEP, sTASEP, the sequence-only Translatomer baseline, and `seq2ribo` for each cell line.

Across all cell lines, sTASEP improves simulator accuracy over the classical TASEP baseline on the primary objectives and on most structural metrics. Regarding the primary objectives, sTASEP lowers both pertranscriptMAE and elemwiseMAE in every dataset, with the LCL model achieving a 7.6% reduction in per transcript error. We also observe consistent gains in correlation, where sTASEP increases Pearson and Spearman correlations in every cell line.

The improvements are even more pronounced in the structural metrics, where sTASEP exploits its explicit parameterization to capture patterns that TASEP misses. In the iPSC and LCL datasets, sTASEP reduces codonMAE by 90.9% and 95.6%, respectively. Similarly, for RPE-1, sTASEP reduces codonMAE by 89.7% and pairMAE by 74.7%. While classical TASEP retains an advantage in specific metrics for HEK293 (pairMAE), LCL (angleMAE), and RPE-1 (bucketMAE), sTASEP yields consistent improvements across the majority of structural metrics in these lines as well. For instance, despite the exception in RPE-1, sTASEP substantially improves bucketMAE in the other three cell lines, reducing error by 37.3% in iPSC. These results demonstrate that adding explicit structural terms to the simulator improves agreement with observed ribosome profiles across diverse experimental settings, and we therefore adopt sTASEP as the simulator f for each cell line.

Using the fitted sTASEP simulator, we generate 32 independent simulation runs per transcript and treat each run as a separate training example for the Mamba-based polisher g for each cell line. We train the polisher with the composite loss described in Section 1.7 of the Supplementary Text, and we select checkpoints using the validation split for each cell line. We report full pipeline losses for each cell line in Table 1, and we use the resulting simulator and polisher as the `seq2ribo` pipeline in subsequent experiments.

Table 1: Test set metrics across cell lines. Lower is better for MAE metrics, while higher is better for Pearson correlation (r) and Spearman correlation (ρ). Bold indicates the best performance within each cell line and metric.

Cell line	Metric	TASEP	sTASEP	Translatomer (seq.)	`seq2ribo`
iPSC	pertranscriptMAE	1.339	1.308	1.532	**0.996**
iPSC	elemwiseMAE	4.895	4.561	6.638	**4.306**
iPSC	Pearson (r)	0.474	0.538	0.211	**0.551**
iPSC	Spearman (ρ)	0.538	0.577	0.121	**0.592**
iPSC	codonMAE	40893.8	3725.2	5915.5	**3137.8**
iPSC	pairMAE	40181.2	**15120.1**	94648.3	17083.3
iPSC	angleMAE	16181.6	**12608.3**	94648.3	17083.3
iPSC	bucketMAE	43280.0	27123.1	311640.3	**22777.7**
HEK293	pertranscriptMAE	1.811	1.807	**1.001**	1.123
HEK293	elemwiseMAE	0.225	0.222	**0.150**	0.197
HEK293	Pearson (r)	0.409	0.453	0.018	**0.488**
HEK293	Spearman (ρ)	0.205	0.270	0.005	**0.333**
HEK293	codonMAE	244.2	**183.4**	4859.0	232.6
HEK293	pairMAE	**300.4**	317.9	77743.7	3069.7
HEK293	angleMAE	430.6	**258.7**	77743.7	3069.7
HEK293	bucketMAE	7403.0	6684.8	103658.3	**4092.9**
LCL	pertranscriptMAE	1.642	1.518	1.599	**1.162**
LCL	elemwiseMAE	32.927	28.956	32.164	**28.659**
LCL	Pearson (r)	0.150	0.299	0.152	**0.319**
LCL	Spearman (ρ)	0.436	0.581	0.265	**0.594**
LCL	codonMAE	800628.0	35579.3	**33723.5**	41800.8
LCL	pairMAE	298006.2	**248120.9**	539576.3	305794.6
LCL	angleMAE	**119976.5**	155087.8	562606.9	167966.6
LCL	bucketMAE	464613.4	**299511.7**	789397.8	312387.0
RPE-1	pertranscriptMAE	1.411	1.369	1.590	**1.020**
RPE-1	elemwiseMAE	3.750	3.506	5.100	**3.285**
RPE-1	Pearson (r)	0.447	0.507	0.179	**0.558**
RPE-1	Spearman (ρ)	0.430	0.551	0.071	**0.595**
RPE-1	codonMAE	23033.5	**2383.0**	4739.5	3516.7
RPE-1	pairMAE	27097.7	**6857.8**	75832.2	9331.1
RPE-1	angleMAE	17322.9	**4286.2**	76335.5	10707.8
RPE-1	bucketMAE	15732.9	22854.4	497259.1	**13242.2**

3.2 `seq2ribo` benchmark against baselines

We benchmark the complete `seq2ribo` pipeline against Translatomer (sequence only) across four distinct cell lines: iPSC, HEK293, LCL, and RPE-1. As shown in Table 1, the `seq2ribo` pipeline achieves the lowest pertranscriptMAE in three of four cell lines, yielding reductions of 35.0% for iPSC, 27.3% for LCL, and 35.8% for RPE-1 relative to Translatomer. In HEK293, Translatomer attains lower MAE values because it predicts all zero counts, which can produce deceptively small absolute errors while failing to capture positional variation.

`seq2ribo` achieves the highest Pearson and Spearman correlations in every cell line, including HEK293, where Translatomer yields near zero correlation due to its constant zero output. This indicates that `seq2ribo` better captures the location specific structure of ribosome profiles even when MAE alone is not fully informative.

In terms of structural objectives, `seq2ribo` improves coarse-grained agreement with observed profiles relative to Translatomer. Notably, `seq2ribo` reduces bucketMAE relative to Translatomer across all four cell lines, including large reductions in RPE-1 (from 497259.1 to 13242.2, a 97.3% drop) and iPSC (from 311640.3 to 22777.7, a 92.7% drop). We also observe a trade-off between transcript-level fidelity and fine-scale structural fit. While the full `seq2ribo` pipeline achieves the best transcript-level performance and the best location-level correlations, the intermediate sTASEP simulator often achieves lower error on fine-scale structural metrics such as pairMAE and angleMAE than the final polished output. This suggests the polisher g prioritizes transcript-wise correctness and correlation, occasionally smoothing over specific structural features captured by the simulator. However, despite this trade-off, the full `seq2ribo` pipeline still improves over Translatomer on structural metrics in the majority of cases, confirming that `seq2ribo` captures both local and global aspects of ribosome occupancy across diverse cellular contexts.

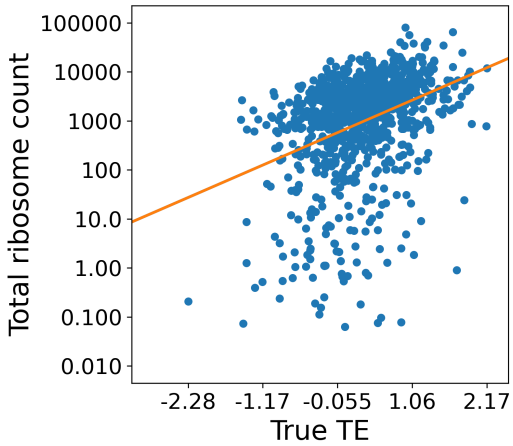
3.3 Translation efficiency prediction

Table 2: Translation efficiency prediction results across translation efficiency label sets on the test set. We report Spearman correlation ρ. Bold indicates the best performance within each label set.

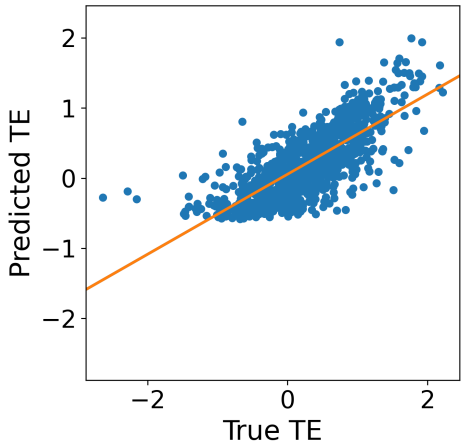
TE label set	Translatomer (seq.)	sTASEP	`seq2ribo`	RiboNN	Finetuned `seq2ribo`
Mean	0.282	0.385	0.394	0.695	**0.769**
HEK293	0.441	0.494	0.581	0.596	**0.704**
LCL	0.162	0.307	0.432	0.700	**0.795**
RPE-1	0.282	0.453	0.555	0.657	**0.785**

We evaluate `seq2ribo` on the TE prediction tasks defined in Methods, covering both mean TE and cell-type-specific efficiency for HEK293, LCL, and RPE-1. We first treat the total predicted ribosome load (sum of codon-level counts per transcript) as a simple summary feature. On the mean TE test set, the sum of counts from the fitted sTASEP simulator yields a Spearman correlation of 0.385 with experimental TE. Using the `seq2ribo` polisher output increases this correlation slightly to 0.394 (Figure 3a). While the scatter plots for the polisher output without finetuning are displayed in log-transformed space for visualization, the reported correlations are calculated in the original non-transformed space. As shown in Table 2, this trend is consistent across cell lines, with the polished total load consistently showing higher correlation than the raw simulation load.

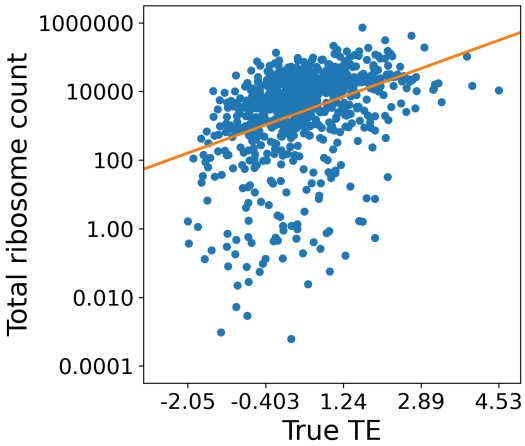
We compare these baselines to two external models. Translatomer (sequence-only) predictions show modest correlations with TE, with $\rho = 0.282$ on the mean set and ρ ranging from 0.162



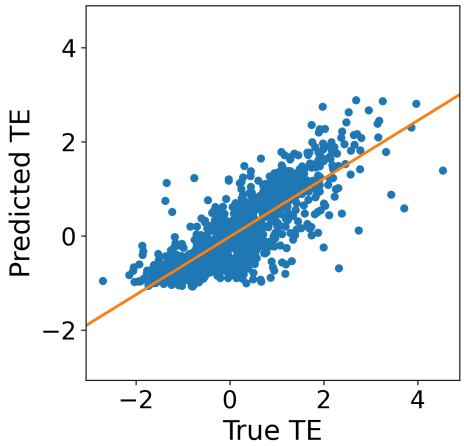
(a) Mean TE: `seq2ribo` ribosome count predictions



(b) Mean TE: Finetuned `seq2ribo` predictions



(c) LCL: `seq2ribo` ribosome count predictions



(d) LCL: Finetuned `seq2ribo` predictions

Figure 3: Correlation of `seq2ribo` predictions with experimental TE on the test set. Top row (a, b) shows results for the Mean TE task. Bottom row (c, d) shows results for the LCL cell-line-specific task. Left panels display the correlation of total ribosome load from the fixed polisher; right panels display the correlation after finetuning the task-specific head. For visualization purposes, the scatter plots in (a) and (c) are displayed in log-transformed space (y-axis is transformed).

to 0.441 across cell lines. However, Translatomer remains below the performance of `seq2ribo` total load in every label set. We train RiboNN (Zheng et al., 2025), a sequence-only TE model, on the same train and test splits that we use for `seq2ribo`. RiboNN performs substantially better, achieving correlations of 0.695 for Mean, 0.596 for HEK293, 0.700 for LCL, and 0.657 for RPE-1. See Supplementary Figures S45–S56 for scatter plots of TE versus Translatomer and RiboNN predictions.

Finetuning the `seq2ribo` pipeline with the TE head yields the best performance across all four tasks. The finetuned model outperforms RiboNN substantially, increasing Spearman correlation by 10.6% on the mean set, 18.1% on HEK293, 13.6% on LCL, and 19.5% on RPE-1. As shown in Figure 3, the finetuned `seq2ribo` model displays a strong monotonic trend and a much narrower spread for the mean and LCL TE label sets. We observe the same trend for the HEK293 and RPE-1 label sets in Supplementary Figures S41–S44.

These results show that a direct sequence-to-TE model captures an important signal, but that the structured ribosome count profiles generated by `seq2ribo` provide a richer representation that a task-specific head can exploit to reach higher accuracy.

3.4 Protein expression prediction

We evaluate `seq2ribo` on a protein expression task using an mRFP expression dataset (Nieuwkoop et al., 2023). As described in Methods, Translatomer cannot be applied in this setting because it requires genomic context and RNA-seq coverage that are not available.

We first apply the `seq2ribo` pipeline trained on the iPSC dataset to the mRFP test sequences without expression-specific supervision and summarize each transcript by its total predicted ribosome load. This yields a Spearman correlation of 0.270 with measured expression (Figure 4a), which shows that the predicted ribosome profiles already capture an intrinsic signal that relates to protein production.

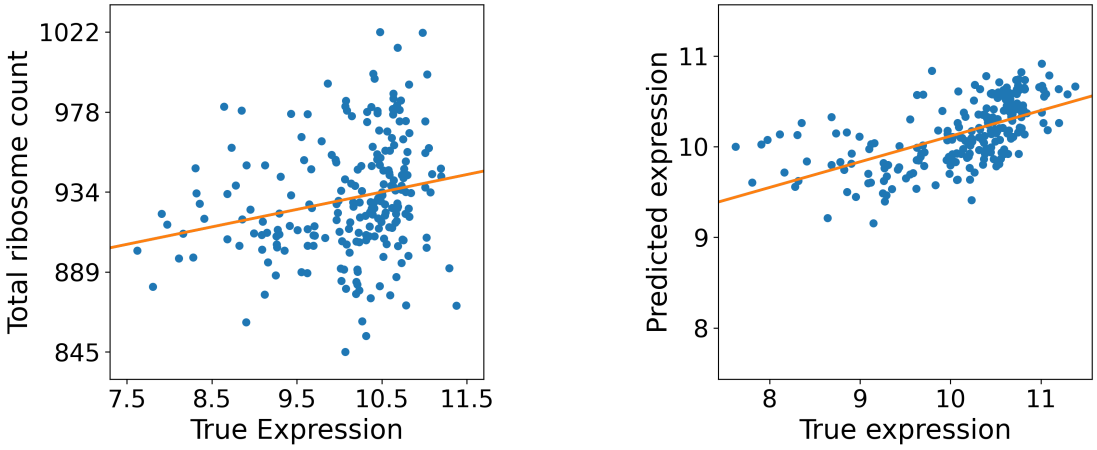
We then add the expression head described in Methods and finetune this head on the mRFP training split, using the setup described there (including 1,000 sTASEP simulations per transcript and validation-based early stopping). This finetuning dramatically improves performance, with the test Spearman correlation rising to 0.689. The scatter plot in Figure 4b shows a much tighter relationship between predicted and measured expression.

These results confirm that the `seq2ribo` pipeline provides a strong foundation for downstream predictive tasks. The simulated ribosome count profiles, when combined with a simple task-specific head and supervised finetuning, can more accurately predict protein expression from sequence alone on an independent experimental dataset.

3.5 Ablation studies on `seq2ribo` components

To deconstruct the `seq2ribo` pipeline and quantify the contribution of its core inputs, we conduct a set of ablation studies on the polisher module using the iPSC dataset. We systematically remove key inputs, namely the sTASEP simulation counts, the structural features and the sequence, and measure the effect on predictive accuracy. All models use the same architecture and optimization settings as the full polisher and are trained with the composite loss on the iPSC training split. Detailed ablation procedures appear in Section 1.15 of the Supplementary Text. Table 3 summarizes the test set performance across all six error metrics, and reports Pearson and Spearman correlations.

The model that receives only sequence performs poorly with a pertranscriptMAE of 6.342, while the structural-only model performs worst with a pertranscriptMAE of 45.831. Using sTASEP alone is substantially stronger, achieving a pertranscriptMAE of 1.022. Among the two-input ablations,



(a) seq2ribo ribosome count predictions

(b) Finetuned seq2ribo expression predictions

Figure 4: Correlation of seq2ribo predictions with measured protein expression on the mRFP test set. Left panel (a) displays the correlation of total ribosome load from the fixed polisher; right panel (b) displays the correlation after finetuning the task-specific expression head.

adding sTASEP to the sequence yields a pertranscriptMAE of 1.042, while adding structural features to sTASEP yields 1.040, and adding structural features to the sequence yields only limited benefit, reaching 6.194. The full seq2ribo model achieves the best overall performance, reducing pertranscriptMAE to 0.996 and improving correlation to $r = 0.551$ and $\rho = 0.592$. Together, these results support the hybrid two-stage design. The sTASEP simulation is essential for providing a high-quality initial ribosome A-site profile, and the polisher then refines this profile using sequence and explicit structural context.

Table 3: Summary of polisher feature ablation results on the iPSC test set across six error metrics and two correlation metrics. Lower is better for all MAE metrics, while higher is better for Pearson and Spearman. Bold indicates the best performance in each row.

Metric	Seq.	sTASEP	Struc.	Seq+sTASEP	Seq+Struc.	sTASEP+Struc.	seq2ribo
pertranscriptMAE	6.342	1.022	45.831	1.042	6.194	1.040	**0.996**
elemwiseMAE	8.479	4.498	11.163	4.469	8.882	4.501	**4.306**
Pearson (r)	0.113	0.544	-0.023	0.540	0.033	0.545	**0.551**
Spearman (ρ)	0.127	0.575	-0.029	0.581	0.088	0.579	**0.592**
codonMAE	28002.5	4286.2	97543.2	3943.8	23610.2	3415.2	**3137.8**
pairMAE	449412.9	48998.1	1554340.0	40391.1	348100.0	27496.9	**17083.3**
angleMAE	449412.9	48998.1	1554340.0	40391.1	353951.8	18631.9	**17083.3**
bucketMAE	864091.3	65330.8	4701181.3	53854.8	464133.3	56802.5	**22777.7**

4 Discussion

We introduce seq2ribo as a hybrid simulation and machine learning framework. Our central contribution is the integration of a structure-aware TASEP simulator with a modern sequence model. sTASEP provides an explicit mechanistic prior on ribosome traffic that depends on codon identity, local base pairing, structural angles, and positional buckets. The Mamba-based polisher

then refines these simulated profiles and learns residual patterns that the additive simulator cannot capture. This two-stage design lets us keep a physically interpretable core while still matching the accuracy of large data-driven models. The integration is essential for performance across diverse biological contexts. We show that sTASEP consistently improves on a classical TASEP model that only fits codon occupancy, reducing structural errors by over 90%. When we add the Mamba polisher, the full `seq2ribo` pipeline further reduces error and improves positional agreement relative to a sequence-only baseline. `seq2ribo` proves particularly effective at transcript-level prediction, achieving a 35.0% reduction in per-transcript MAE in iPSC and a 35.8% reduction in RPE-1 relative to the state-of-the-art. Consistent with this, `seq2ribo` achieves the highest Pearson and Spearman correlations in every cell line, indicating improved location-specific agreement with measured ribosome occupancy. Simultaneously, it drastically lowers structure-aware errors, achieving up to a 97.3% reduction.

A second key feature of `seq2ribo` is that it uses sequence-only input. This makes the method suitable for de novo sequence design where experimental covariates are unavailable. We show that the predicted ribosome profiles serve as strong features for downstream tasks. With a simple regression head, we reach Spearman correlations up to 0.795 across four distinct translation efficiency tasks, outperforming the sequence-based RiboNN baseline by up to 19.5%. On an external protein expression dataset, we obtain a Spearman correlation of 0.689. These results indicate that `seq2ribo` captures the aspects of translation dynamics that matter for protein output, not only ribosome positions.

Beyond these tasks, `seq2ribo` can generate synthetic Ribo-seq experiments. Because the framework operates on nucleotide sequence alone, it can simulate high-fidelity ribosome profiles for arbitrary inputs, including de novo designed sequences or isoforms for which no experimental data exists. This capability enables in silico ribosome profiling, allowing researchers to screen libraries of candidate mRNA sequences to optimize ribosome traffic patterns, such as maximizing ribosome load or minimizing collision-prone motifs before experimental synthesis. By providing a virtual platform for testing translational hypotheses, `seq2ribo` bridges the gap between sequence design and functional validation in synthetic biology.

Our study has limitations that suggest directions for future work. Currently, we derive structural features from predicted two-dimensional RNA secondary structures, which do not fully capture the complex three-dimensional folding and tertiary interactions that influence ribosome movement in vivo. Future iterations could incorporate 3D structural predictions or co-transcriptional folding dynamics to improve fidelity. Additionally, we train separate models for each cell line to maximize specificity. In future work, we aim to develop a unified model capable of zero-shot transfer across tissues and to extend the framework to other species to test how well the same integration of simulation and sequence modeling transfers across diverse translational regimes.

Funding

This work was supported in part by the US National Science Foundation [III-2232121] and the US National Institutes of Health [R01HG012470].

Data availability

The ribosome profiling data used in this study were obtained from zthe GWIPS-viz database (https://gwips.ucc.ie/). Specifically, we used aggregate A-site coverage tracks for the iPSC, HEK293, LCL, and RPE-1 cell lines. Gene annotations were obtained from GENCODE release v49 (https://www.gencodegenes.org/). The translation efficiency labels were obtained from Zheng

et al. (2025). The mRFP protein expression dataset was obtained from Nieuwkoop et al. (2023). All processed datasets, including the matched sequences, structural features, and ribosome counts used for training and testing, are available at https://github.com/Kingsford-Group/seq2ribo.

Code availability

The source code for seq2ribo, including the sTASEP simulator, the Mamba-based polisher, the task-specific finetuning pipelines for translation efficiency and protein expression, and scripts for data preprocessing, training, and evaluation, is available on GitHub at https://github.com/Kingsford-Group/seq2ribo. The repository also contains the pretrained model checkpoints for all four cell lines and the notebooks used to generate the figures in this manuscript.

Author contributions statement

G.K. and C.K. conceived the study and designed the algorithms. G.K. implemented the seq2ribo framework, performed the data processing, trained the models, and conducted the experiments. G.K. and C.K. analyzed the results. G.K. wrote the manuscript with input from C.K.

Acknowledgments

We thank Dr. David Koes for valuable discussions, Dr. Guillaume Marçais for help with system setup and careful review of the manuscript, and Jialin He for her feedback on training the sequence-only Translatomer model.

Conflicts of interest

C.K. is a co-founder of Ocean Genomics, Inc.

References

N. Ahmed, P. Sormanni, P. Ciryam, M. Vendruscolo, C. M. Dobson, and E. P. O'Brien. Identifying A-and P-site locations on ribosome-protected mRNA fragments using Integer Programming. *Scientific Reports*, 9(1):6256, 2019.

C. Bao, S. Loerch, C. Ling, A. A. Korostelev, N. Grigorieff, and D. N. Ermolenko. mRNA stem-loops can pause the ribosome by hindering A-site tRNA binding. *Elife*, 9:e55799, 2020.

G. A. Brar and J. S. Weissman. Ribosome profiling reveals the what, when, where and how of protein synthesis. *Nature Reviews Molecular Cell Biology*, 16(11):651–664, 2015.

N. Caliskan, V. I. Katunin, R. Belardinelli, F. Peske, and M. V. Rodnina. Programmed–1 frameshifting by kinetic partitioning during impeded translocation. *Cell*, 157(7):1619–1631, 2014.

L. Calviello, N. Mukherjee, E. Wyler, H. Zauber, A. Hirsekorn, M. Selbach, M. Landthaler, B. Obermayer, and U. Ohler. Detecting actively translated open reading frames in ribosome profiling data. *Nature Methods*, 13(2):165–170, 2016.

Z. Cao, X. Pan, Y. Yang, Y. Huang, and H.-B. Shen. The lncLocator: a subcellular localization predictor for long non-coding RNAs based on a stacked ensemble classifier. *Bioinformatics*, 34 (13):2185–2194, 2018.

C. Cenik, E. S. Cenik, G. W. Byeon, F. Grubert, S. I. Candille, D. Spacek, B. Alsallakh, H. Tilgner, C. L. Araya, H. Tang, et al. Integrative analysis of RNA, translation, and protein levels reveals distinct regulatory variation across humans. *Genome Research*, 25(11):1610–1621, 2015.

J. Chen, A.-D. Brunner, J. Z. Cogan, J. K. Nuñez, A. P. Fields, B. Adamson, D. N. Itzhak, J. Y. Li, M. Mann, M. D. Leonetti, et al. Pervasive functional translation of noncanonical human open reading frames. *Science*, 367(6482):1140–1146, 2020.

J. Dong, B. Schmittmann, and R. K. Zia. Inhomogeneous exclusion processes with extended objects: The effect of defect locations. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 76 (5):051113, 2007.

A. Gu and T. Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024.

A. Haas, B. S. Nilges, S. A. Leidel, and K.-H. Klempnauer. PDCD4 controls the G1/S-phase transition in a telomerase-immortalized epithelial cell line and affects the expression level and translation of multiple mRNAs. *Scientific Reports*, 10(1):2758, 2020.

J. He, L. Xiong, S. Shi, C. Li, K. Chen, Q. Fang, J. Nan, K. Ding, Y. Mao, C. A. Boix, et al. Deep learning prediction of ribosome profiling with Translatomer reveals translational regulation and interprets disease variants. *Nature Machine Intelligence*, 6(11):1314–1329, 2024.

J. W. Hershey, N. Sonenberg, and M. B. Mathews. Principles of translational control: an overview. *Cold Spring Harbor Perspectives in Biology*, 4(12):a011528, 2012.

N. T. Ingolia, S. Ghaemmaghami, J. R. Newman, and J. S. Weissman. Genome-wide analysis in vivo of translation with nucleotide resolution using ribosome profiling. *Science*, 324(5924):218–223, 2009.

N. T. Ingolia, G. A. Brar, S. Rouskin, A. M. McGeachy, and J. S. Weissman. The ribosome profiling strategy for monitoring translation in vivo by deep sequencing of ribosome-protected mRNA fragments. *Nature Protocols*, 7(8):1534–1550, 2012.

N. T. Ingolia, J. A. Hussmann, and J. S. Weissman. Ribosome profiling: global views of translation. *Cold Spring Harbor Perspectives in Biology*, 11(5):a032698, 2019.

X. Jia, X. He, C. Huang, J. Li, Z. Dong, and K. Liu. Protein translation: biological processes and therapeutic strategies for human diseases. *Signal Transduction and Targeted Therapy*, 9(1):44, 2024.

L. Jin, Y. Zhou, S. Zhang, and S.-J. Chen. mRNA vaccine sequence and structure design and optimization: Advances and challenges. *Journal of Biological Chemistry*, 301(1):108015, 2025.

K. Leppek, G. W. Byeon, W. Kladwang, H. K. Wayment-Steele, C. H. Kerr, A. F. Xu, D. S. Kim, V. V. Topkar, C. Choe, D. Rothschild, et al. Combinatorial optimization of mRNA structure, stability, and translation for RNA-based therapeutics. *Nature Communications*, 13(1):1536, 2022.

S. Li, S. Moayedpour, R. Li, M. Bailey, S. Riahi, L. Kogler-Anele, M. Miladi, J. Miner, F. Pertuy, D. Zheng, et al. CodonBERT large language model for mRNA vaccines. *Genome Research*, 34(7): 1027–1035, 2024.

J. Lin, Y. Chen, Y. Zhang, H. Lin, and Z. Ouyang. Deciphering the role of RNA structure in translation efficiency. *BMC Bioinformatics*, 23(Suppl 3):559, 2022.

R. Lorenz, S. H. Bernhart, C. Höner zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, and I. L. Hofacker. ViennaRNA Package 2.0. *Algorithms for Molecular Biology*, 6(1):26, 2011.

C. T. MacDonald, J. H. Gibbs, and A. C. Pipkin. Kinetics of biopolymerization on nucleic acid templates. *Biopolymers: Original Research on Biomolecules*, 6(1):1–25, 1968.

V. P. Mauro and S. A. Chappell. A critical analysis of codon optimization in human therapeutics. *Trends in Molecular Medicine*, 20(11):604–613, 2014.

A. M. Michel, G. Fox, A. M. Kiran, C. De Bo, P. B. O'Connor, S. M. Heaphy, J. P. Mullan, C. A. Donohue, D. G. Higgins, and P. V. Baranov. GWIPS-viz: development of a ribo-seq genome browser. *Nucleic Acids Research*, 42(D1):D859–D864, 2014.

T. Nieuwkoop, B. R. Terlouw, K. G. Stevens, R. A. Scheltema, D. De Ridder, J. Van der Oost, and N. J. Claassens. Revealing determinants of translation efficiency via whole-gene codon randomization and machine learning. *Nucleic Acids Research*, 51(5):2363–2376, 2023.

P. B. O'Connor, D. E. Andreev, and P. V. Baranov. Comparative survey of the relative impact of mRNA features on local ribosome profiling read density. *Nature Communications*, 7(1):12915, 2016.

N. Pardi, M. J. Hogan, F. W. Porter, and D. Weissman. mRNA vaccines—a new era in vaccinology. *Nature Reviews Drug Discovery*, 17(4):261–279, 2018.

X. Qu, J.-D. Wen, L. Lancaster, H. F. Noller, C. Bustamante, and I. Tinoco. The ribosome uses two active mechanisms to unwind messenger RNA during translation. *Nature*, 475(7354):118–121, 2011.

S. Rouskin, M. Zubradt, S. Washietl, M. Kellis, and J. S. Weissman. Genome-wide probing of RNA structure reveals active unfolding of mRNA structures in vivo. *Nature*, 505(7485):701–705, 2014.

B. Schwanhäusser, D. Busse, N. Li, G. Dittmar, J. Schuchhardt, J. Wolf, W. Chen, and M. Selbach. Global quantification of mammalian gene expression control. *Nature*, 473(7347):337–342, 2011.

P. Spealman, H. Wang, G. May, C. Kingsford, and C. J. McManus. Exploring ribosome positioning on translating transcripts with ribosome profiling. In *Post-Transcriptional Gene Regulation*, pages 71–97. Springer, 2016.

F. Spitzer. Interaction of Markov processes. *Advances in Mathematics*, 5(2):246–290, 1970.

R. Verbeke, M. J. Hogan, K. Loré, and N. Pardi. Innate immune mechanisms of mRNA vaccines. *Immunity*, 55(11):1993–2005, 2022.

T. von der Haar. Mathematical and computational modelling of ribosomal movement and protein synthesis: an overview. *Computational and Structural Biotechnology Journal*, 1(1):e201204002, 2012.

S. Wagner, A. Herrmannová, V. Hronová, S. Gunišová, N. D. Sen, R. D. Hannan, A. G. Hinnebusch, N. E. Shirokikh, T. Preiss, and L. S. Valášek. Selective translation complex profiling reveals staged initiation and co-translational assembly of initiation factor complexes. *Molecular Cell*, 79 (4):546–560, 2020.

H. Wang, J. McManus, and C. Kingsford. Isoform-level ribosome occupancy estimation guided by transcript abundance with Ribomap. *Bioinformatics*, 32(12):1880–1882, 2016.

Z. Xiao, R. Huang, X. Xing, Y. Chen, H. Deng, and X. Yang. De novo annotation and characterization of the translatome with ribosome profiling data. *Nucleic Acids Research*, 46(10):e61–e61, 2018.

Z. Yan, E. Lécuyer, and M. Blanchette. Prediction of mRNA subcellular localization using deep recurrent neural networks. *Bioinformatics*, 35(14):i333–i342, 2019.

C.-H. Yu, Y. Dang, Z. Zhou, C. Wu, F. Zhao, M. S. Sachs, and Y. Liu. Codon usage influences the local rate of translation elongation to regulate co-translational protein folding. *Molecular Cell*, 59 (5):744–754, 2015.

Y. Zarai, M. Margaliot, and T. Tuller. Ribosome flow model with extended objects. *Journal of the Royal Society Interface*, 14(135):20170128, 2017.

S. Zhang, H. Hu, T. Jiang, L. Zhang, and J. Zeng. TITER: predicting translation initiation sites by deep learning. *Bioinformatics*, 33(14):i234–i242, 2017.

D. Zheng, L. Persyn, J. Wang, Y. Liu, F. Ulloa-Montoya, C. Cenik, and V. Agarwal. Predicting the translation efficiency of messenger RNA in mammalian cells. *Nat. Biotechnol.*, pages 1–14, July 2025. ISSN 1546-1696. doi: 10.1038/s41587-025-02712-x.

R. K. Zia, J. Dong, and B. Schmittmann. Modeling translation in protein synthesis with TASEP: A tutorial and recent developments. *Journal of Statistical Physics*, 144(2):405–428, 2011.