nature research

https://doi.org/10.1038/s41596-020-00469-y

Supplementary information

Revealing nascent RNA processing dynamics with nano-COP

In the format provided by the authors and unedited

Supplementary Information

Revealing nascent RNA processing dynamics with nano-COP

Heather L. Drexler, Karine Choquet, Hope E. Merens, Paul S. Tang, Jared T. Simpson and L. Stirling Churchman

Supplementary Methods

This section describes the methods employed to test other library preparation approaches during the development of nano-COP, as shown in Figure 2 of the main text, as well as to validate nanopolish-detect-polyI, as shown in Figure 4 of the main text.

1) Direct RNA sequencing of chromatin-associated RNA (Figure 2)

Cellular fractionation was performed on 10 million unlabeled human K562 cells exactly as described in steps 8-21 of Ref [1] in biological duplicate. RNA was extracted from the chromatin fraction using the Qiagen miRNeasy kit (cat. no. 217004) according to manufacturer's instructions. Ribosomal RNAs (rRNA) were depleted from the chromatin-associated RNA using RiboMinus Eukaryotic Kit v2 (ThermoFisher, cat. no. A15020) as described in Steps 82-93 of the nano-COP protocol. 3'-end tailing was performed as described in Steps 94-99 of the nano-COP protocol, with one biological replicate undergoing poly(A) tailing (step 96A), while the other replicate underwent poly(I) tailing (steps 96B). Direct RNA library preparation and sequencing was performed as indicated in steps 100-104 of the nano-COP protocol. Data analysis was performed in the same manner as nano-COP samples (steps 115-128).

2) Direct cDNA sequencing of 4sU-labeled chromatin-associated RNA (Figure 2)

4sU-labeled chromatin-associated RNA was collected from human K562 cells and rRNA-depleted following steps 1-93 of the nano-COP protocol in biological duplicate. 3'-end tailing was performed as described in Steps 94-99 of the nano-COP protocol, with one biological replicate undergoing poly(A) tailing (steps 96A), while the other replicate underwent poly(I) tailing (steps 96B). ONT direct cDNA library preparation and sequencing was performed using the SQK-DCS109 kit (Oxford Nanopore Technologies Ltd.) as described by the manufacturer, except for one modification with the poly(I)-tailed sample: the VNP primer in the kit was replaced by 2.5 µL of a custom VNP-C_{22} primer (5' -ACTTGCCTGTCGCTCTATCTTCCCCCCCCCCCCCCCCCCCCCC-3', IDT) diluted to 2 µM, with the poly(C) stretch annealing to the poly(I) tail of RNAs for reverse transcription. Data analysis was performed as for nano-COP samples except for the following modifications:

- Step 116 (conversion of RNA sequences to DNA sequences) was omitted.
- Step 117 (alignment): Reads were aligned to the reference genome using minimap2[2] with recommended parameters for ONT cDNA sequencing (-ax splice).
- Steps 121-128 (analysis of distance between transcription and splicing):
    - Step 122: For removing reads with RNA 3' ends near annotated poly(A) sites (150 nt upstream or any distance downstream) or 5' splice sites (50 nt upstream or 10 nt downstream), we used the same strategy as described above. Reads ending near poly(A) sites or splice sites and on the same strand as these features and reads starting near poly(A) sites or splice sites and on the opposite strand as these features were filtered out.

- Step 124: Using BEDTools intersect, identify reads that overlap splice sites on the same strand (option s=True) or on the opposite strand (option S=True).
- Step 125: Calculate the distance between the transcript 3' end and the 3'SS of each intron within the read. For cDNA sequencing, the transcript 3' end corresponds to the read end for sense reads and the read start for antisense reads.

3) *In vitro* transcription and direct RNA sequencing of ERCC-00048 (Figure 4)

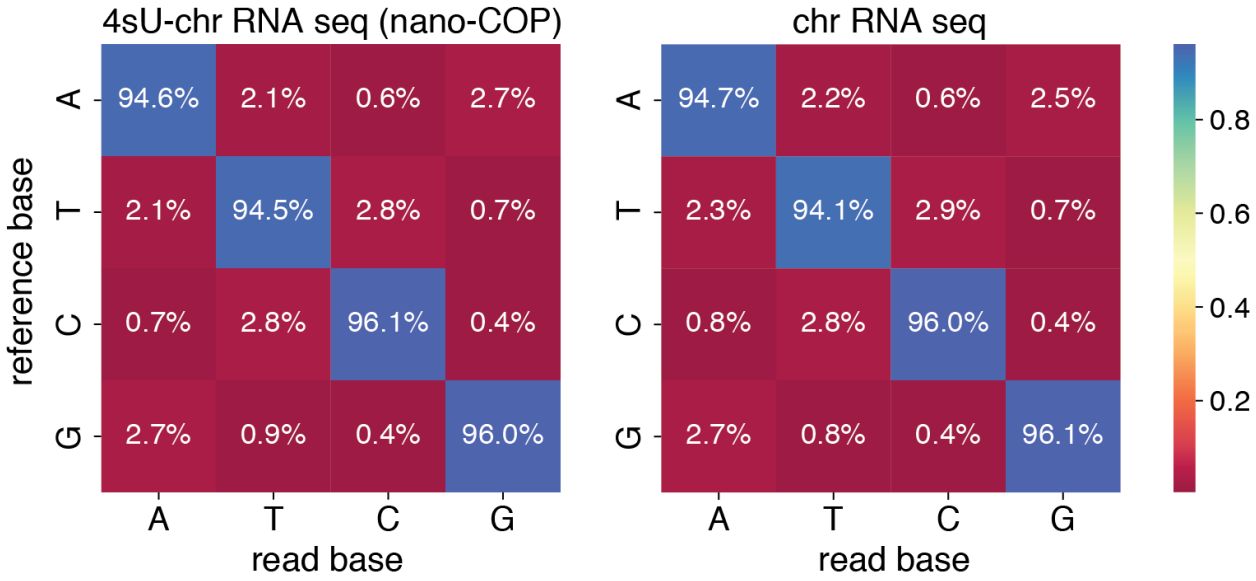
ERCC-00048 was synthesized as a G-block with a T7 promoter, *in vitro* transcribed using the HiScribe T7 High Yield RNA Synthesis kit (NEB, cat. no. E2040S) and purified by gel extraction. 3'-end tailing was performed as described in Steps 94-99 of the nano-COP protocol, with two biological replicates undergoing poly(A) tailing (steps 96A), and two other biological replicates undergoing poly(I) tailing (96B). Finally, one replicate successively underwent poly(A) tailing (steps 94-99 with option A) and then poly(I) tailing (steps 94-99 with option B). Direct RNA library preparation and sequencing was performed as indicated in steps 100-104 of the nano-COP protocol.

4) Direct RNA sequencing of polyA+ RNA (Figure 4)

Total RNA was extracted from the chromatin fraction using the Qiagen miRNeasy kit (cat. no. 217004) according to manufacturer's instructions. Polyadenylated RNA was selected using Dynabeads Oligo(dT) (ThermoFisher, cat. no. 61002). 500 ng of polyA+ RNA was mixed with 15 ng of SIRV-Set 3 (Lexogen, cat. no. 051) for poly(I) tailing (steps 94-99 with option 96B). Direct RNA library preparation and sequencing was performed as indicated in steps 100-104 of the nano-COP protocol. Data analysis was performed in the same manner as nano-COP samples (steps 115-120). SIRV-Set3 transcripts were not included in the analysis for Figure 4C.

References

1. Mayer, A. & Churchman, L. S. Genome-wide profiling of RNA polymerase transcription at nucleotide resolution in human cells with native elongating transcript sequencing. *Nat. Protoc.* **11**, 813–833 (2016).
2. Li, H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* **34**, 3094–3100 (2018).



Supplementary Figure 1. Confusion matrices of read base calls versus reference bases for one representative sample of nano-COP (with 4sU) and of direct RNA sequencing of chromatin (chr) associated RNA (no 4sU). Both libraries were prepared with poly(I) tailing. No decreased accuracy is observed for the T reference base in nano-COP, indicating that 4sU does not have a detectable effect on base calling. For both libraries, aligned bases and reference bases were recorded for all mapped regions of reads. To generate confusion matrices, the frequency of each base matching the reference was calculated for 100,000 random aligned sequence segments in each sample.

Supplementary Note: Detecting polyadenylated and polyinosinated tails in direct RNA sequencing data

1 Model Description

To detect the presence of polyadenylated and polyinosinated tails in mature and nascent RNA molecules obtained by the Oxford Nanopore direct RNA sequencing protocol, we developed a segmentation approach based on two hidden markov models, which we describe below. The first hidden markov model (HMM), which we refer to as the *Segmentation HMM*, infers a segmentation of the signal trace of an RNA molecule and isolates a region of the signal trace possessing either a polyinosine tail, in the case of nascent RNA, or a polyinosine tail and a polyadenosine tail, in the case of mature RNA.

The second hidden markov model, which we refer to as the *Bernoulli HMM*, detects the existence of a polyadenylated tail within the region isolated by the Segmentation HMM. As we further elaborate in a section below, intrinsic differences between reads and statistical similarities in the signal traces of the polyadenylated and polyinosinated regions make it necessary to use a two HMM approach.

In the rest of this supplementary note, we follow the nomenclature and conventions of the supplementary note in [3] unless otherwise specified.

1.1 Signal Segmentation via Hidden Markov Model

A hidden markov model, which we call the *Segmentation HMM*, is used to segment the squiggle of a read into distinct *regions* appearing sequentially. The state transition dynamics of the Segmentation HMM are structurally identical to that of the HMM described in the supplementary note of [3], with the following caveats:

- The *POLYA* state now represents the combined poly(I) and poly(A) regions, and its emission distribution is now given by a two-component Gaussian mixture model, which is described in the next section.
- The transition probabilities and emissions of each state have been updated to reflect changes introduced by Oxford Nanopore's *SQK-RNA-002* direct RNA sequencing kit.

A full diagram of the state transitions from [3], with updated transition probabilities, is reproduced in Figure 1.

1.1.1 Emission Distributions

Emissions are modelled with Gaussian, uniform, and Gaussian mixture distributions. The following emission distributions are used:

- *START (S)*: $0.5 \times \mathcal{N}(\mu = 70.2737, \sigma^2 = 3.7743) + 0.5 \times \mathcal{U}([40.0, 250.0])$
- *LEADER (L)*: $\mathcal{N}(\mu = 110.973, \sigma^2 = 5.237)$
- *ADAPTER (A)*: $0.874 \times \mathcal{N}(\mu = 79.347, \sigma^2 = 8.3702) + 0.126 \times \mathcal{N}(\mu = 63.3126, \sigma^2 = 2.7464)$
- *POLYA (P)*: $0.5 \times \mathcal{N}(\mu = 108.883, \sigma^2 = 3.257) + 0.5 \times \mathcal{N}(108.498, \sigma^2 = 5.257)$
- *CLIFF (C)*: $\mathcal{U}([70.0, 140.0])$
- *TRANSCRIPT (T)*: $0.346 \times \mathcal{N}(\mu = 79.679, \sigma^2 = 6.966) + 0.654 \times \mathcal{N}(\mu = 105.784, \sigma^2 = 16.022)$

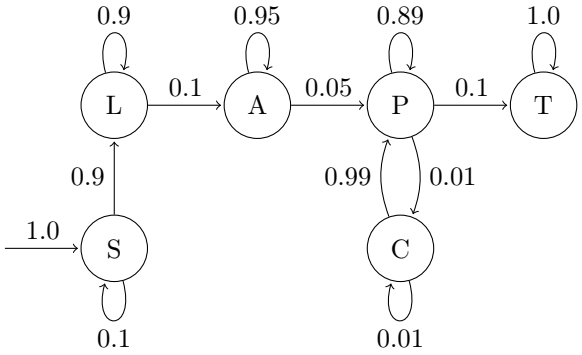


Figure 1: The state transitions of the Segmentation HMM, updated for *SQK-RNA-002*. Edges without an origin node on the left indicate the initial state probabilities.

The emission distributions were fitted following the same methodology described in the supplementary note of [3].

The emissions of the *POLYA* state — now simultaneously representing a polyinosinated region as well as a polyadenylated region — is an even mixture of two Normal distributions:

- The polyadenylated tail is represented by the distribution $f_{p(A)} = \mathcal{N}(\mu = 108.883, \sigma^2 = 3.257)$.
- The polyinosinated region is represented by the distribution $f_{p(I)} = \mathcal{N}(\mu = 108.498, \sigma^2 = 5.257)$.

Due to the similarity between the $f_{p(A)}$ and the $f_{p(I)}$ distributions as well as statistical differences between squiggles, previous attempts to model the 3'-to-5' squiggle using discrete states for each of the polyadenylated and polyinosinated regions failed to yield accurate segmentations, necessitating the development of the Bernoulli HMM described in the next section.

1.1.2 Inferring a Segmentation

Fix a read R with associated squiggle

$$\vec{s} = (s_1, \ldots, s_n).$$

A *segmentation* of R is defined as a disjoint set of contiguous intervals

$$\Sigma(R) = \langle [S], [L], [A], [P], [T] \rangle$$

where each component of the inferred segmentation represents the first and last indices of the region corresponding to the state of the Segmentation HMM, e.g. $[P] = (k_0, k_1)$ such that the subsequence $\vec{s}_P = (s_{k_0}, \ldots, s_{k_1})$ represents the region of $\vec{s}$ inferred to correspond to the combined poly(A) and poly(I) region.

To obtain the segmentation $\Sigma(R)$, we first observe that the linear-chain structure of the state space of the Segmentation HMM implies a unique ordering of the inferred states in the output sequence when running the Viterbi algorithm for the Segmentation HMM on a squiggle. If we elide the distinction between the P and C states[1], all output sequences obtained from the Viterbi algorithm must be of the form

$$(S, \ldots, S, L, \ldots, L, A, \ldots, A, P, \ldots, P, T, \ldots, T),$$

where the number of occurrences of each state varies based on the properties of the individual read. We thus define the segmentation $\Sigma(R)$ as the first and last indices of each state in the above output sequence inferred by the Viterbi algorithm.

[1] This corresponds to the notion that the C states are intended to model brief aberrations in the signal trace of the combined poly(A) and poly(I) tail.

1.2 Switchpoint Detection via Hidden Markov Model

Given a segmentation of a read R, we are now interested in inferring the switchpoint between the internal poly(I) and poly(A) states. Formally, let

$$\vec{\pi} = (\pi_1, \ldots, \pi_m)$$

be the subsequence associated to the P state in the segmentation $\Sigma(R)$. The *switchpoint* in $\vec{\pi}$ is the index $q \in [1, \ldots, m]$ such that $(\pi_1, \ldots, \pi_q)$ is biologically associated to the polyinosinated region of the squiggle and $(\pi_{q+1}, \ldots, \pi_m)$ is associated to the polyadenylated tail of the squiggle. In the case that no polyadenylated tail exists in the read — as in the case of nascent RNA — the switchpoint is defined to be $q = m$.[2]

Two related factors prevent the use of a basic HMM with Gaussian emissions for switchpoint-finding to be embedded within the Segmentation HMM, necessitating the development of a binarization process for the squiggle and a robust downstream HMM for switchpoint detection. First, the Gaussian distributions $f_{p(A)}$ and $f_{p(I)}$ are difficult to distinguish using a finite set of observables due to their highly similar location and scale parameters. Second, the emission distributions of the Segmentation HMM are trained on the regions from a pooled collection of reads; thus, the linear shifts between each read, combined with the distributional similarity of the two Gaussian distributions, are enough to induce a mis-segmentation error.

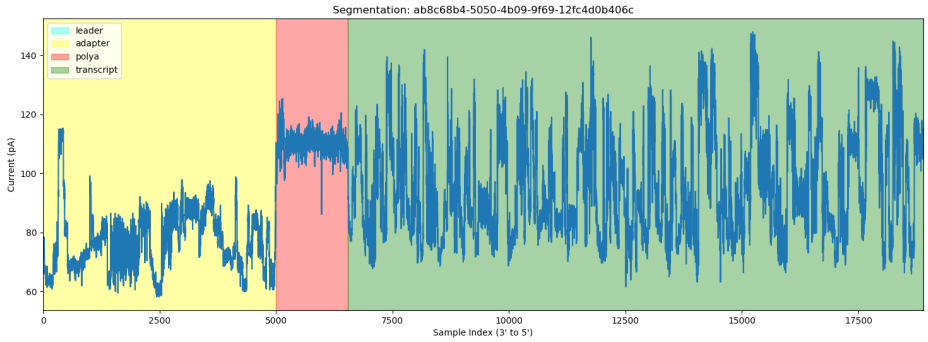


Figure 2: An example of an inferred segmentation. Note the presence of an initial spike in the POLYA region, representing polyinosination.

1.2.1 Signal Binarization via Log-Likelihood Ratios

To resolve the above issues leading to mis-segmentation and to apply the Viterbi algorithm for switchpoint detection, we developed a binarization method based on the log-likelihoods between the two components of the emissions distribution for the joint poly(I) and poly(A) region of the read. For a given squiggle $\vec{\pi}$ as defined above, we apply the following mapping to each sample π_i of $\vec{\pi}$:

$$\pi_i \mapsto \beta_i = \begin{cases} 1, & \text{if } \ell(\pi_i; p(I)) > \ell(\pi_i; p(A)), \\ 0, & \text{otherwise,} \end{cases}$$

where

$$\ell(x; p(I)) = \log \mathcal{L}(x; \mu = 108.498, \sigma^2 = 5.257) = \log f_{p(I)}(x)$$

is the log-likelihood of observing sample x for the poly(I) distribution, and $\ell(x; p(A))$ is the respective log-likelihood for the poly(A) distribution. Observing the binarized values (as in the figure below), one can visually confirm that the switchpoint in the squiggle corresponds to a sharp imbalance in the density of zeros and ones in the binarized sequence. Consequently, we can design a two-state hidden markov model with Bernoulli-distributed emissions to detect a switchpoint between the two poly(I) and poly(A) regions.

[2] In practice, even when no polyadenylated tail exists, the inferred switchpoint may be near, but not equal, to m due to the nature of numerical approximations. We employ a filtering step where reads with inferred switchpoints sufficiently close to m are classified as non-polyadenylated.

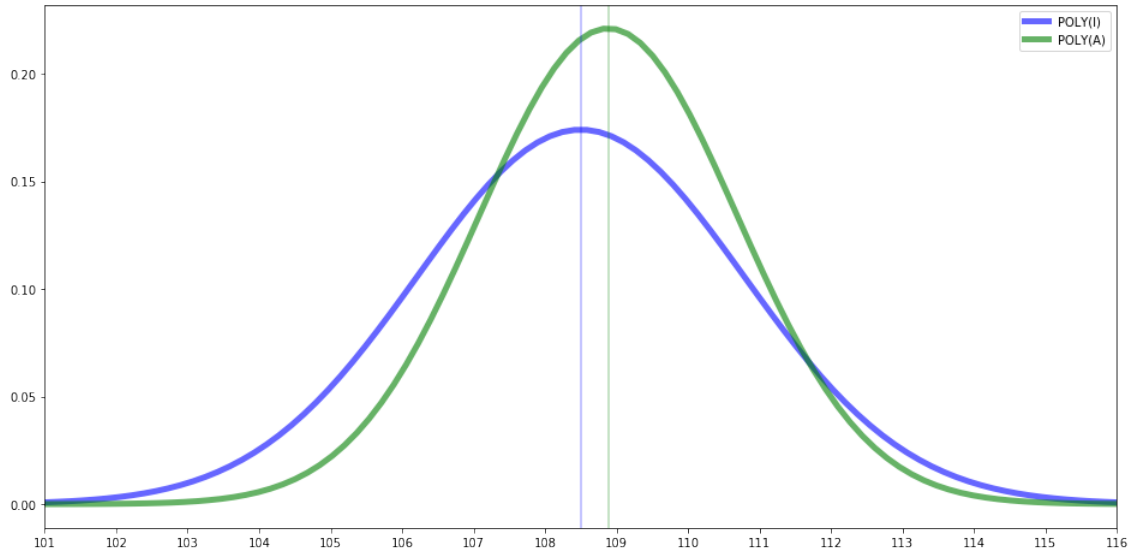


Figure 3: The Gaussian emissions for poly(I) and poly(A). The vertical green and blue lines are placed at the means of the corresponding distribution.

1.2.2 State Transitions

The Bernoulli HMM has two states, representing the poly(I) and poly(A) regions, respectively, with a single edge originating from the poly(I) state and ending in the poly(A) state. The direction of the edge represents the biological arrangement of the respective regions in the 3'-to-5' orientation of direct RNA sequencing.

1.2.3 Emission Distributions

The emission distributions of both states have Bernoulli densities with differing parameters p_I, p_A:

- The poly(I) state has a Bern($p_I = 0.72304$) density.
- The poly(A) state has a Bern($p_A = 0.92154$) density.

The emission distributions were fitted using the same workflow as the emissions of the Segmentation HMM, in which an initial estimate of the parameters was used to bootstrap valid segmentations of the squiggle, which were further used to re-fit the parameters.

1.2.4 Switchpoint Detection

The switchpoint between the poly(I) and poly(A) regions of a given read's squiggle is obtained by running the Viterbi algorithm on the binarized observation sequence $\vec{\beta}$ corresponding to the region $\vec{\pi}$ as defined above. By an argument similar to that of the segmentation obtained from the Segmentation HMM, the inferred sequence of latent states resulting from a run of the Viterbi algorithm is of the form

$$(p(I), \ldots, p(I), p(A), \ldots, p(A));$$

the index of the final $p(I)$ state in the above state sequence is the switchpoint inferred by the Bernoulli HMM.

1.3 Reproducibility

The poly(I)-poly(A) tail detection method described above is implemented as the `detect-polyi` module of nanopolish:

https://github.com/jts/nanopolish

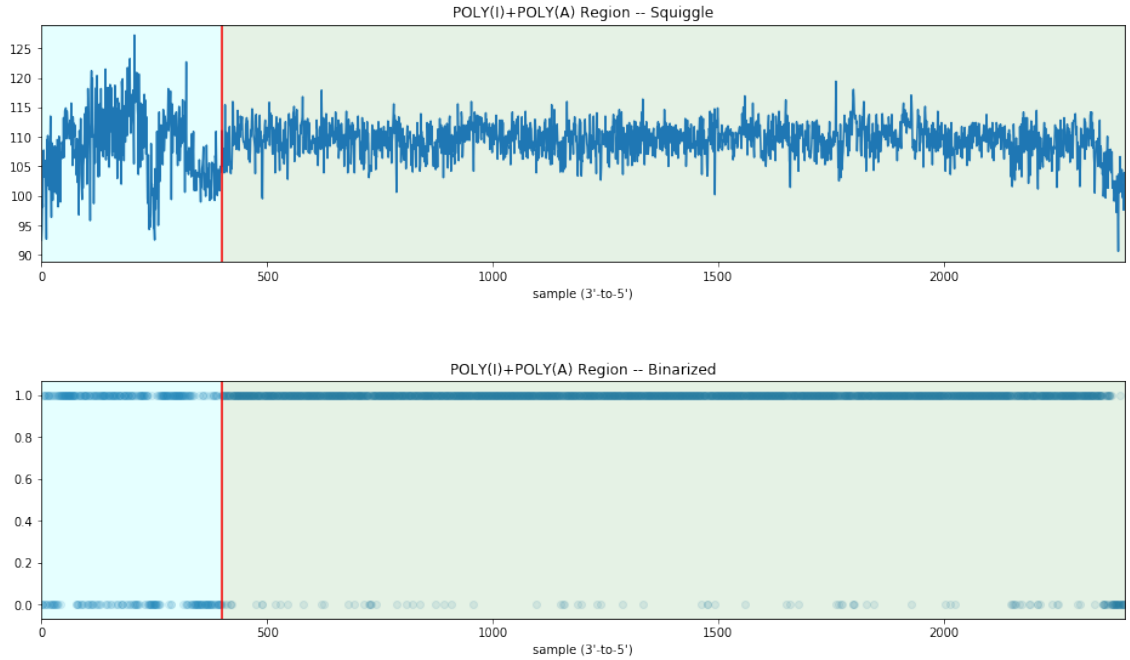


Figure 4: An example of the switchpoint (red line) inferred by the Bernoulli HMM. The difference in dispersion between the poly(I) and poly(A) regions in the signal trace result in a detectable difference in frequency in the binarized sequence.

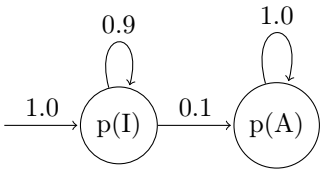


Figure 5: The state dynamics of the Bernoulli HMM. Edges without an origin node indicate the initial state probabilities.

2 References

1. Garalde, D., et al. Highly parallel direct RNA sequencing on an array of nanopores. *Nature Methods* **15**, 201-206 (2018).

2. Simpson, J.T., Workman, R.E., Zuzarte, P.C., David, M., Dursi, L.J., Timp, W. Detecting DNA cytosine methylation using nanopore sequencing. *Nature Methods* **14**, 407-410 (2017).

3. Workman, R. et al. Nanopore native RNA sequencing of a human poly(A) transcriptome. *Nature Methods* **16**, 1297–1305 (2019).

4. Drexler, H.L., Choquet, K. Churchman, L.S. Splicing kinetics and coordination revealed by direct nascent RNA sequencing. *Molecular Cell* **77**, 985-998 (2020).

Supplementary Table 1. Key characteristics of datasets used in this article.

Sample name	Replicate	Cell line	RNA purification	Library preparation + sequencing	Instrument	rRNA depletion kit	Tailing approach	GEO accession number
K562_4sUchr_ONT_1	1	K562	4sU-chr	direct RNA	MinION	Ribo-Zero	poly(A)	GSE123191
K562_4sUchr_ONT_2	2	K562	4sU-chr	direct RNA	MinION	Ribo-Zero	poly(A)	GSE123191
K562_4sUchr_ONT_3	3	K562	4sU-chr	direct RNA	MinION	Ribo-Zero	poly(A)	GSE123191
K562_4sUchr_ONT_4	4	K562	4sU-chr	direct RNA	PromethION	RiboMinus	poly(I)	GSE123191
K562_4sUchr_ONT_5a	5a	K562	4sU-chr	direct RNA	PromethION	RiboMinus	poly(I)	GSE123191
K562_4sUchr_ONT_5b	5b	K562	4sU-chr	direct RNA	MinION	RiboMinus	poly(I)	GSE154079 (nanopolish-detect-polyI dataset) and GSE123191 (all other datasets)
K562_chr_RNA_1	1	K562	chr	direct RNA	MinION	RiboMinus	poly(A)	GSE154079
K562_chr_RNA_2	2	K562	chr	direct RNA	MinION	RiboMinus	poly(I)	GSE154079
K562_4sUchr_cDNA_1	1	K562	4sU-chr	direct cDNA	MinION	RiboMinus	poly(A)	GSE154079
K562_4sUchr_cDNA_2	2	K562	4sU-chr	direct cDNA	MinION	RiboMinus	poly(I)	GSE154079
K562_mRNA_polyI	1	K562	total polyA+	direct RNA	MinION	N/A	poly(I)	GSE154079
ERCC00048_polyA_ ONT_1	1	N/A	*in vitro* transcription	direct RNA	MinION	N/A	poly(A)	GSE154079
ERCC00048_polyI_ ONT_1	1	N/A	*in vitro* transcription	direct RNA	MinION	N/A	poly(I)	GSE154079
ERCC00048_polyA_ ONT_1	2	N/A	*in vitro* transcription	direct RNA	MinION	N/A	poly(A)	GSE154079
ERCC00048_polyI_ ONT_1	2	N/A	*in vitro* transcription	direct RNA	MinION	N/A	poly(I)	GSE154079
ERCC00048_polyA_ polyI_ONT	1	N/A	*in vitro* transcription	direct RNA	MinION	N/A	poly(A)-poly(I)	GSE154079