Segmentation and HMMs for nanopore data

http://users.soe.ucsc.edu/~karplus/papers/

segmentation-2014-nov-6.pdf

Kevin Karplus

Biomolecular Engineering Department
University of California, Santa Cruz
karplus@soe.ucsc.edu

2014 Nov 6

Nanopore sequencing

- Thin membrane separating salt-water baths, with a tiny hole.
- Ionic current through hole partially blocked by DNA translocating.
- Exact current depends on which bases currently in “reading window”.
- DNA motors used to get slow step-wise motion of DNA.

Noise

- Simplest model of nanopore is as a 10GΩ variable resistor.
- Thermal noise: $i_{\text{thermal}} = \sqrt{\frac{4k_B T I \Delta f}{V}}$
- Shot noise: $i_{\text{shot}} = \sqrt{2qI\Delta f}$
- Amplifier noise is also thermal noise and shot noise, but based on temperature of resistors and other parts—the largest contributor to noise in present system.
- Tiny currents (10–100 pA), even smaller changes in current (0.2–10pA), but noise is large (2–5pA).

`http://gasstationwithoutpumps.wordpress.com/2013/04/21/noise-in-nanopores/`

Noise Spectrum

- Noise is “white”—equal power at all frequencies, total power proportional to bandwidth.
- Moving DNA infrequently means our signal is mainly low frequencies.
- We improve signal-to-noise ratio with low-pass filter, removing high-frequency noise.

Segmentation problem

- DNA moves stepwise (long pauses between rapid movements).
- Current should change in steps (though obscured by noise).
- We need to re-create stepwise current from noisy data.
- Low-pass filters can blur steps into slow ramps (step signals have high-frequency components).
- If we can identify when steps happen, and how big they are, we can use just summary information about segments (duration, mean current, amount of noise) in later processing.

Filtered Derivative

- Our first segmentation methods were all variants on "filtered derivative" techniques: low-pass filter the signal to remove noise, take the derivative, and look for peaks.
- We couldn't get reliable behavior—either small transitions or short segments were missed (blurred out by filter) or noisy regions resulted in over-segmentation.
- Can't just tweak parameters—both missing short segments and over-segmenting happen in the same event.

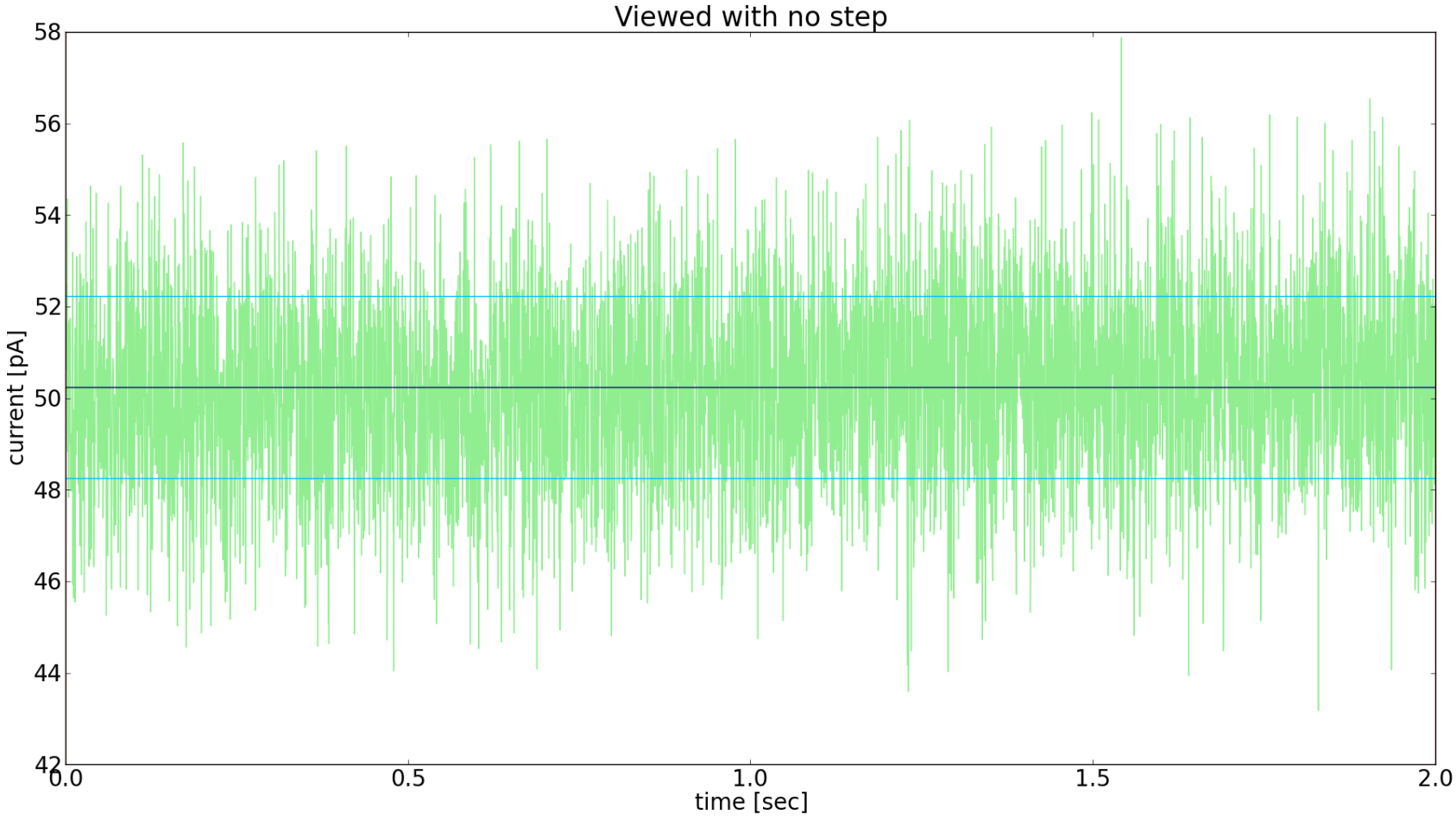
Statistical change detection

Let's simplify the problem temporarily: look for a single step in an interval with duration T. Is there one here?

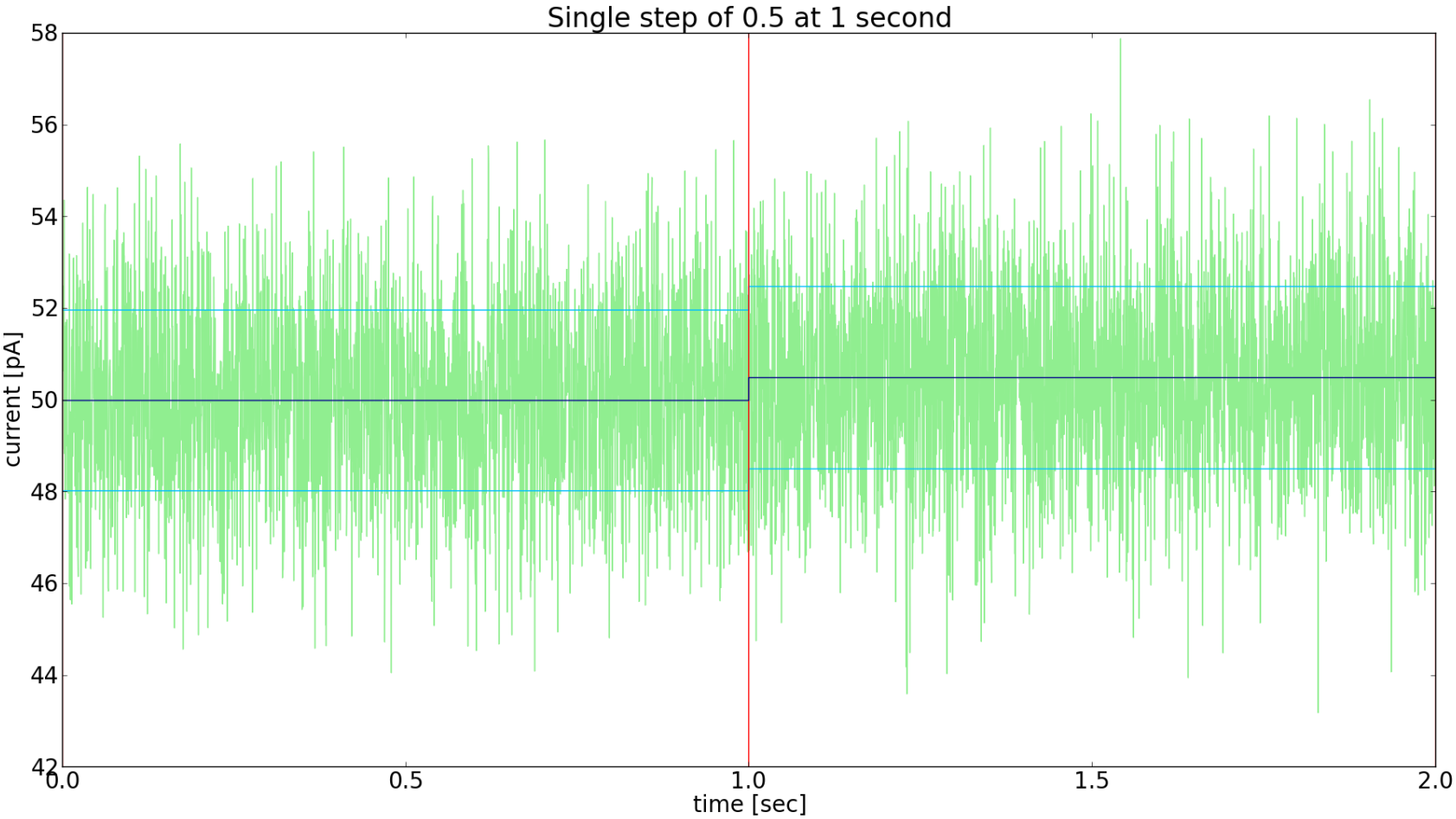
Model 1: Gaussian noise: $x(t) \in G(\mu_0, \sigma_0)$

Model 2: Step at time τ: $\begin{cases} x(t) \in G(\mu_1, \sigma_1) & \text{, for } 0 \leq t < \tau \\ x(t) \in G(\mu_2, \sigma_2) & \text{, for } \tau \leq t < T \end{cases}$

No step


Viewed with no step
current [pA]
time [sec]
58
56
54
52
50
48
46
44
42
0.0
0.5
1.0
1.5
2.0

With step


Single step of 0.5 at 1 second
current [pA]
time [sec]
58
56
54
52
50
48
46
44
42
0.0
0.5
1.0
1.5
2.0

Statistical change detection

Look at log-odds ratio: $\ln \frac{\text{Prob}_{M2}(x)}{\text{Prob}_{M1}(x)}$

$$= \sum_{0 \le t < \tau} \ln \text{Prob}_1(x(t)) + \sum_{\tau \le t < T} \ln \text{Prob}_2(x(t)) - \sum_{0 \le t < T} \ln \text{Prob}_0(x(t))$$

Log probability of a single sample from a Gaussian distribution:

$$\ln\left(\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right) = -0.5\ln(2\pi) - \ln(\sigma) - \frac{(x-\mu)^2}{2\sigma^2}$$

Simplifying sums

$0.5\ln(2\pi)$ is constant, so adding T times and subtracting T times cancels.
We choose μ and σ optimally for each interval:

$$\sigma^2 = E((x-\mu)^2)$$

and so $\sum_{0\le t<T}(x(t)-\mu)^2/(2\sigma^2) = T/2$, which also cancels.
Thus the log-odds ratio simplifies to

$$T\ln\sigma_0 - \tau\ln\sigma_1 - (T-\tau)\ln\sigma_2\ ,$$

which is always non-negative.

Choosing best step

Modeling with a step is always better than without, and we can choose the best τ to maximize the log-odds ratio.
If that is above some threshold, we accept the step.
The threshold is equivalent to having prior odds of there being a step, and using log posterior odds ratio.

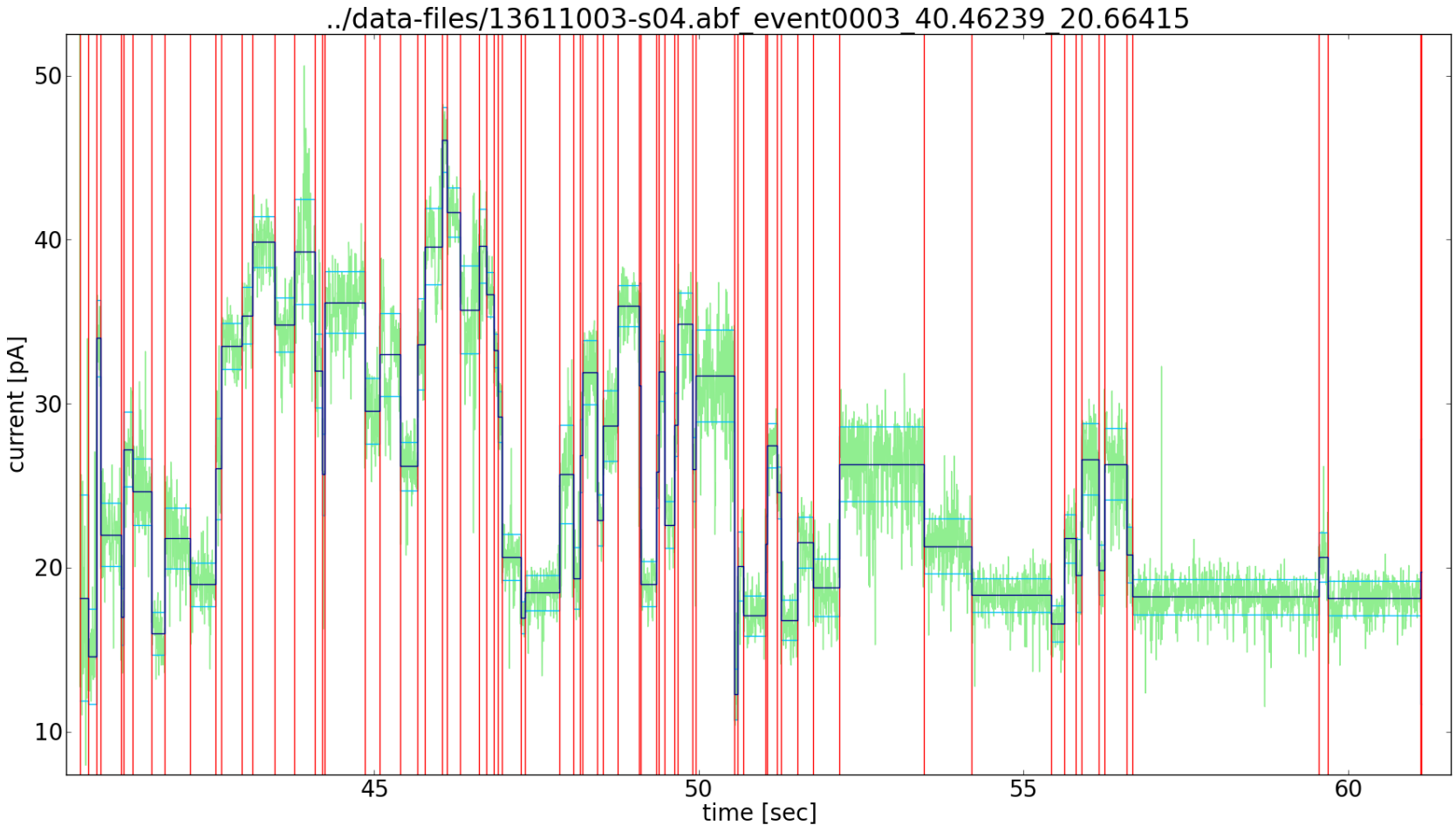
Many steps

For multiple steps, find best step in window, then recursively apply to both subwindows.
Stop when reaching minimum segment duration or best step is not good enough to pass threshold.
Efficiency hack: apply to 1-second window past last known step time, rather than to entire duration of event. If no step found, advance by half window width and try again.

`http://gasstationwithoutpumps.wordpress.com/2013/08/10/segmenting-noisy-signals-from-nanopores/`

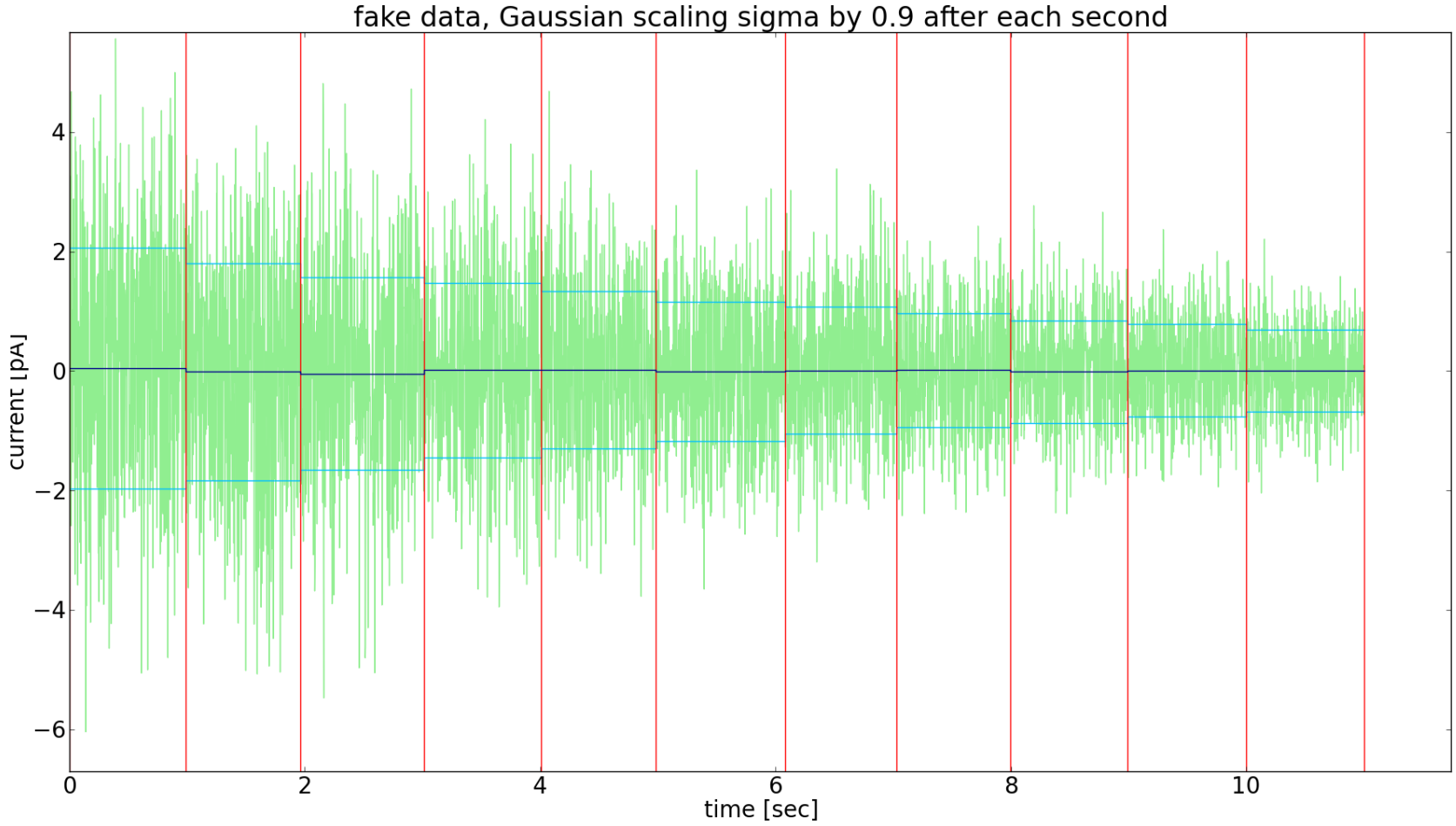
`http://gasstationwithoutpumps.wordpress.com/2014/02/01/more-on-segmenting-noisy-signals/`

Example on real data



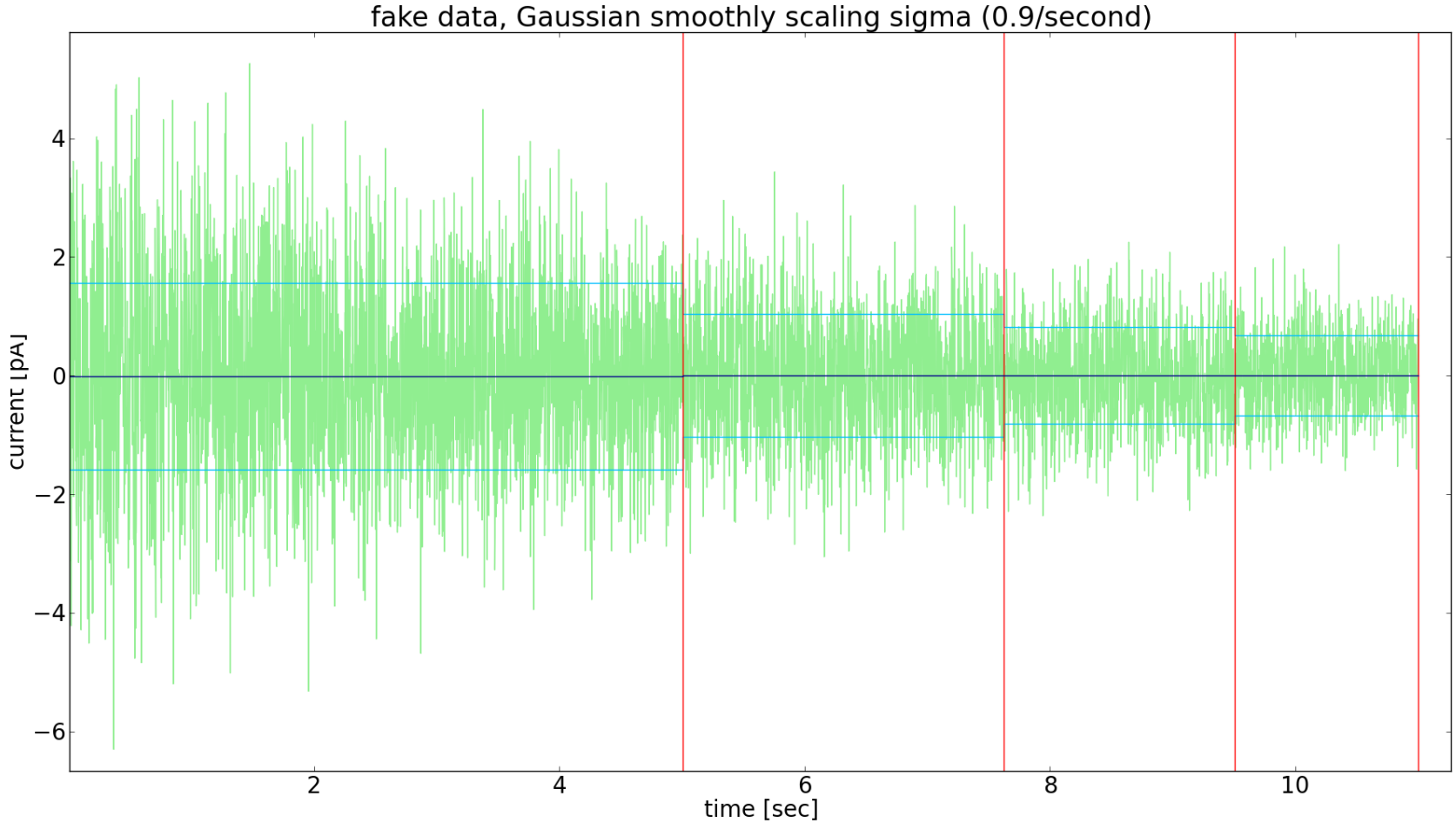
Not limited to steps (1)

The “steps" do not require a change in mean—just that the split into 2 models fits enough better than a single model.



Introduced steps

But finding a step doesn't mean that there really was one in the underlying data.



Filtered signals

The analysis above assumed that samples were independent, but low-pass filtering ensures that they are not!
Low-pass filtering makes a sample very much like the preceding one.
We can adjust for the low-pass filtering by scaling the log-odds score.
Multiplying the score by the filter cutoff frequency over the Nyquist frequency does a pretty good job of keeping the scaled score having the same cumulative distribution over a wide range of filter frequencies.

`http://gasstationwithoutpumps.wordpress.com/2014/05/24/segmenting-filtered-signals/`

Threshold matters

- If we set the threshold for the scores too low, we get over-segmentation, with minor fluctuations in mean value due to noise causing segments to break up.
- If we set the threshold too high, we get under-segmentation, with multiple DNA bases being lumped into a single segment.

Setting threshold

Keep the split if

$$k \ln \frac{\text{Prob}_{\text{step}}}{\text{Prob}_{\text{no step}}} > \ln \frac{1-s}{s} - \ln F \,,$$

where k is filter cutoff over Nyquist frequency,
s is expected number of segments per sample, and
F is allowable false positives per sample.

In practice only one of s or F needs to be specified, and we usually specify in per-second terms, rather than per-sample.)
`http://gasstationwithoutpumps.wordpress.com/2014/06/17/segmenting-noisy-signals-revisited/`

Interpreting sequences

If we have a sequence of segments, how do we convert them into biologically interpretable results?

We're at UCSC, so the answer is aligning them to a stochastic model—most likely a hidden Markov model or some variant. Each state of an HMM is either a silent state or one that emits a segment (we may need to generalize to a contiguous sequence of segments).

Scoring segments

What is emission probability for a segment?

- simplest: distribution for mean value of segment (normal? kernel density estimate from training set?)
- more complex: product of independent distributions for mean value and for duration. (need to merge successive segments for same state—log-normal distribution)
- how can noise level be part of scoring?

HMM structure

Several HMM structures possible:

- simple profile for recognizing a single molecule type.
- extra states to correspond to under-segmentation
- extra states for “blips” (brief excursions in the current trace)
- extra transitions for back slips
- profile with branching to recognize molecule with limited variation (used for initial methylation studies)
- complete de Bruijn graph connecting kmers (for base callers)

Future Work: Theory

We empirically observed that the log-likelihood score L_i applied to a pure Gaussian signal is exponentially distributed:

$$\text{Prob}(L_i > x) = e^{-x} ,$$

but we have no proof.

We observed that L_i applied to low-pass filtered Gaussian with filter cutoff k times the Nyquist frequency is

$$\text{Prob}(L_i > x) = e^{-kx} ,$$

but we have no proof.

Future Work: Theory 2

The log-likelihood score is not distributed purely according to the exponential distribution on the previous slide for all interval lengths.

Cutting up a short interval results in slightly higher scores than cutting up long intervals, so there is a tendency for cut points to cluster around previously identified cut points.

Better theory could let us compensate and avoid over-segmenting near transitions.

Future Work: Speed

My code (in Python) is barely fast enough to be useful.
Jacob Schreiber has recoded it in Cython (a minimal change), for substantial speed up.
We probably want it recoded in c, to run fast enough to do a lot more computation after segmentation to recognize DNA sequences in real time on a MinION with 100s of events happening in parallel.

Future Work: Online implementation

The current code does batch processing, taking an entire event and subdividing it.

But we want to be able to recognize DNA molecules as they go through the nanopore, to re-read parts of them or kick them out if they are uninteresting.

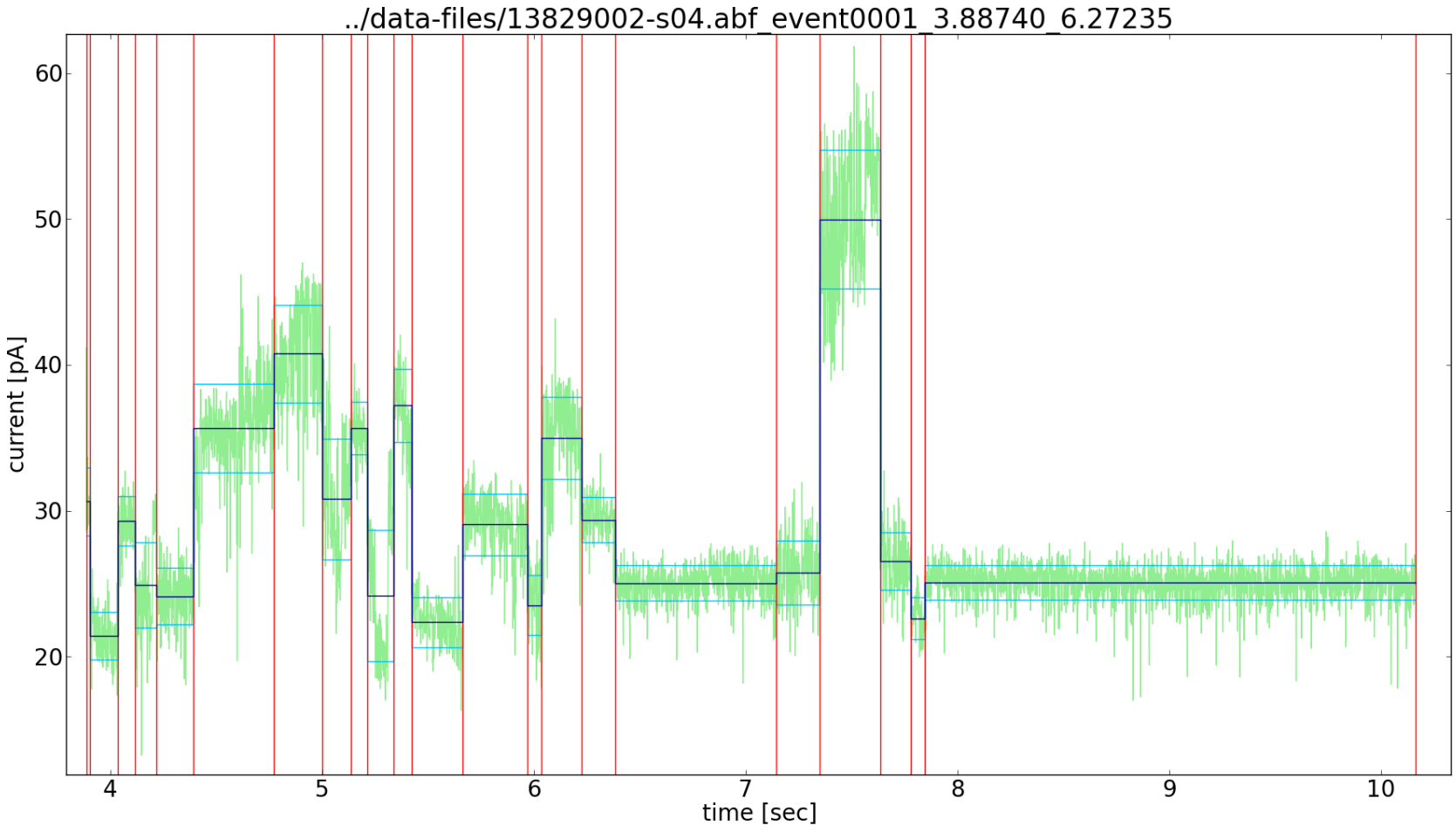
This means segmenting the data with only a few seconds of delay.

The algorithm is easy to change to being online, but the code needs to be completely rewritten, particularly the I/O, which will be a real-time stream from the digitizer, rather than a file.

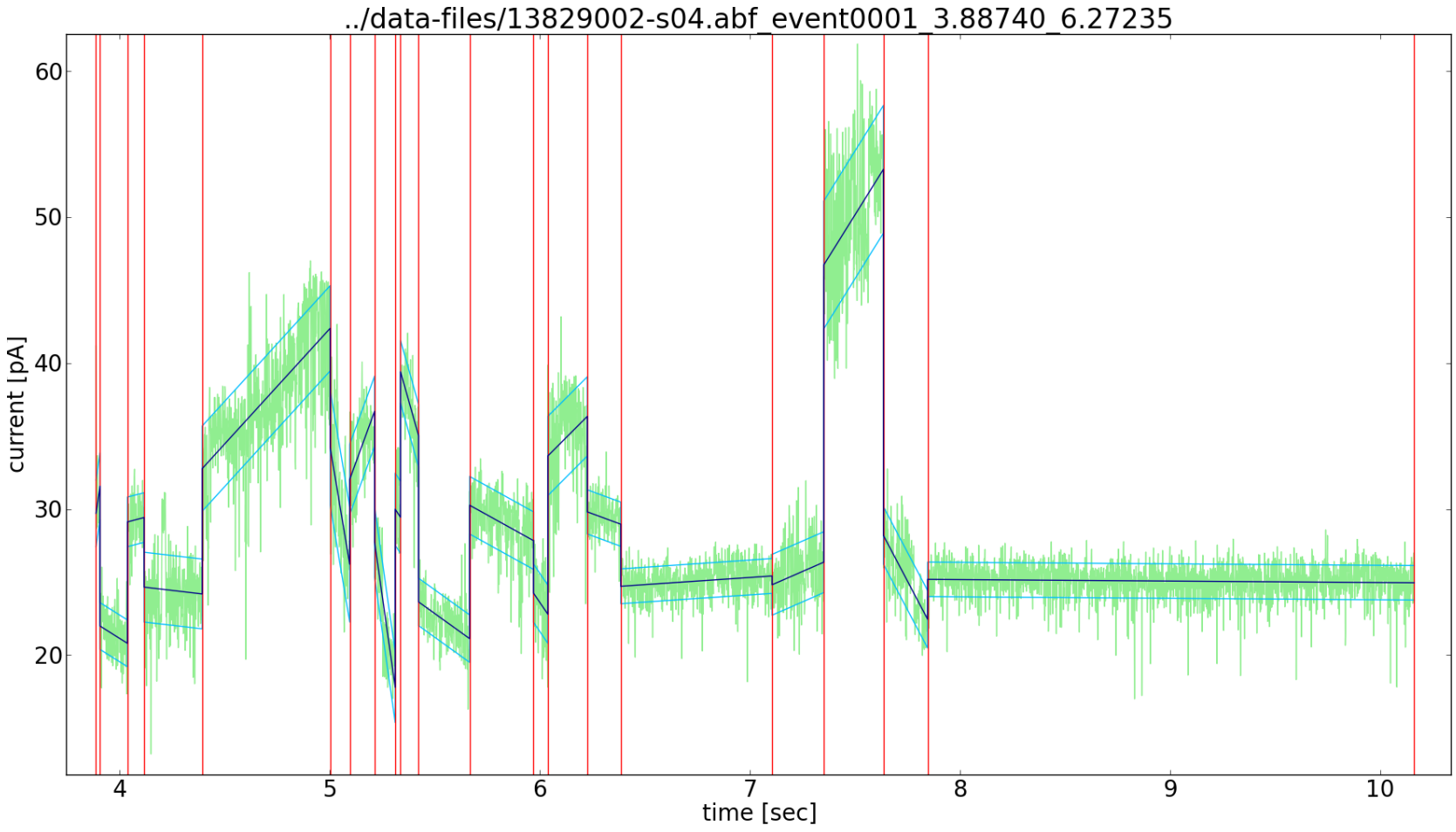
Future Work: stepwise slanted segments

We can use any model that results in Gaussian distributed errors around a fitted curve, not just stepwise constants. The log-likelihood ratio is still based on logs of the standard deviations of the errors.
For example, we can use linear regression to get stepwise slanted segments.

Protein trace with steps



Protein trace with slanted segments



Future Work: Testing

The algorithm has been tested on only a few hundred real examples, but only a handful of synthetic examples where the correct answer is known.
More extensive testing and comparison with filtered-derivative algorithms is desirable.

Future Work: HMMs

- A base caller that uses Viterbi path (or full forward-backward) through an HMM.
- A sequence recognizer that knows particular DNA sequences and recognizes where in the sequence the nanopore is reading.
- Sequence recognizer with variants to call DNA modifications (like 5mC, 5hmC, . . .)
- Control techniques to change movement of DNA based on what is recognized.

Web sites

These slides:

`http://users.soe.ucsc.edu/~karplus/papers/segmentation-2014-nov-6.pdf`

Blog posts:

`http://gasstationwithoutpumps.wordpress.com/tag/nanopore/`

Textbook: *Detection of Abrupt Changes: Theory and Application* Michèle Basseville and Igor V. Nikiforov

`ftp://ftp.irisa.fr/local/as/mb/k11.pdf`