Segmenting nanopore traces

Kevin Karplus

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

14 Nov 2013

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).

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, std. deviation) 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 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

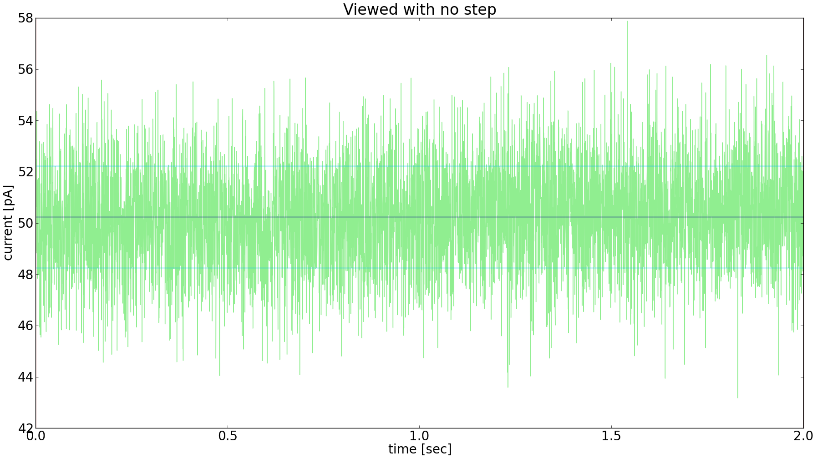
Let's simplify the problem temporarily: look for a single step in an interval with duration T. Is there one here?

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

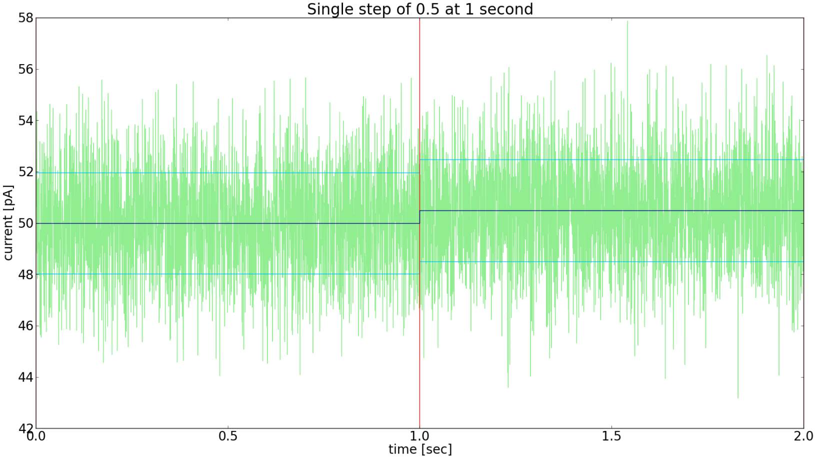
Model 2: Step at time τ:

$$\begin{cases} x(t) \in G(\mu_1, \sigma_1) & \text{, for } 0 \leq t < \tau \\ x(t) \in G(\mu_2, \sigma_2) & \text{, for } \tau \leq t < T \end{cases}$$

No step


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

With step


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

Statistical change detection

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

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

$$- \sum_{0 \le t < T} \ln \mathrm{Prob}_0(x(t))$$

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 $\sum_{0\leq t<T}(x(t)-\mu)^2/(2\sigma^2) = T/2$
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.

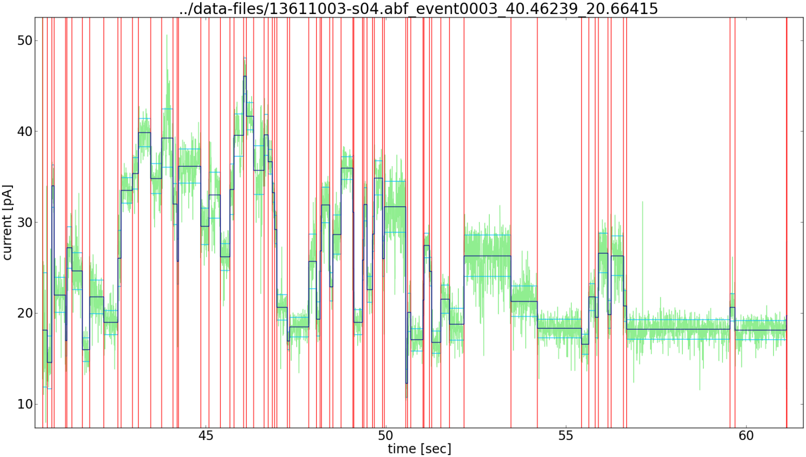
Many steps

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.

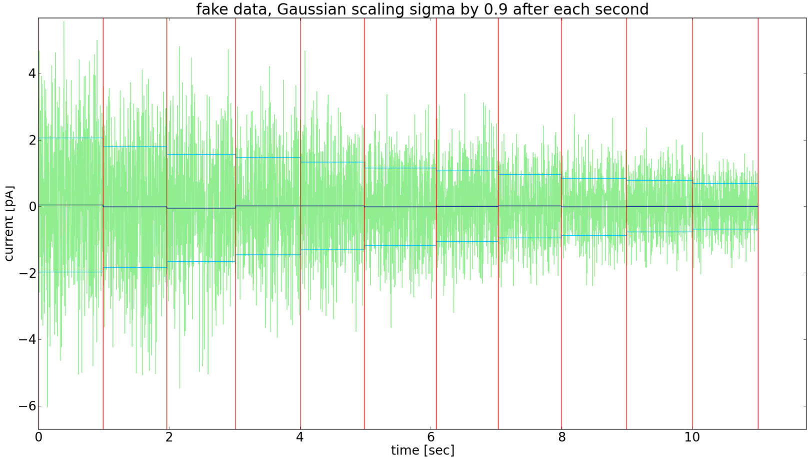
Efficiency hack: apply to 1-second window past last known step time, rather than to entire duration of event. If no step found, advance by half window width and try again.

Example on real data



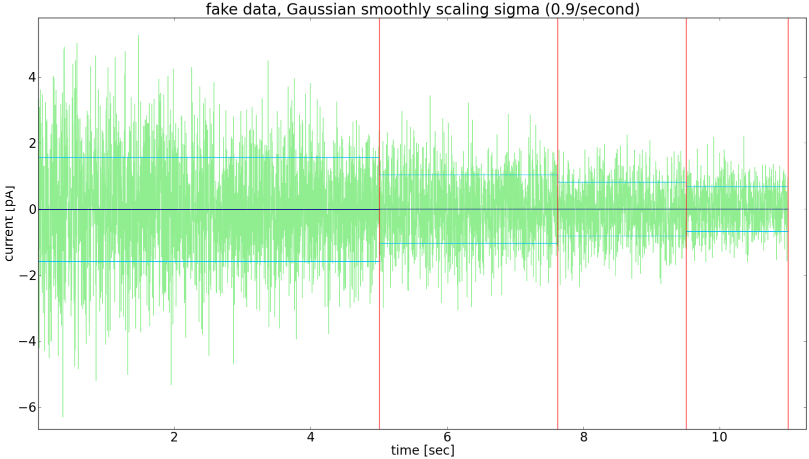
Not limited to steps (1)

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.



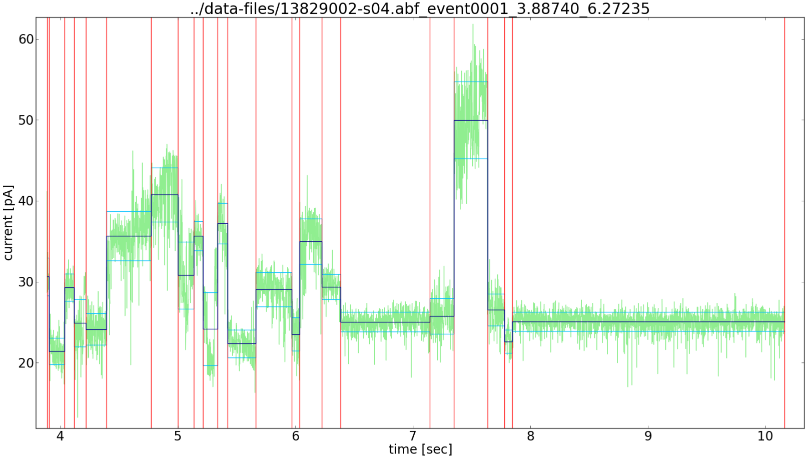
Not limited to steps (2)

We can use any model that results in Gaussian distributed errors around a fitted curve, not just stepwise constants.

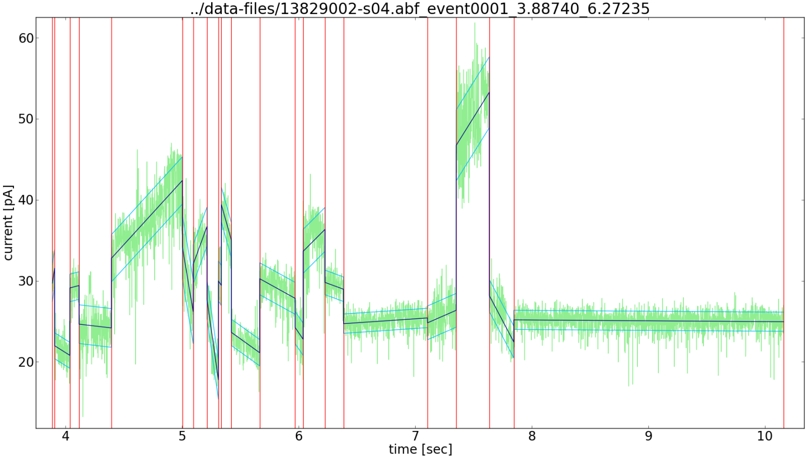
The log-likelihood ratio is still based on logs of the standard deviations of the errors.

For example, we can use linear regression to get stepwise slanted segments.

Protein trace with steps



Protein trace with slanted segments



Future Work: Theory

The current specification of the threshold is unintuitive, and it might be good to be more formal about thresholds and priors.
Could we, for example, specify the prior in terms of expected number of steps per second?

Future Work: Testing and publishing

The algorithm has been tested on only a few dozen real examples, and only a handful of synthetic examples where the correct answer is known.

More extensive testing and comparison with filtered-derivative algorithms is needed.

The method has not yet been published in a place where nanopore and nanopipette researchers are likely to find it.

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: Speed

My code (in Python) is barely fast enough to do online segmentation. 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 and still interact with the nanopore controller with only a few seconds lag.

Future Work: Slanted segments

The algorithm can do stepwise slanted segments, but we've not yet shown that this is useful.

We hope that they can provide a better summary of what we see running proteins through a nanopore, but no higher-level analytic framework has been set up.

Future Work: Aligners

- An aligner that can align events (as segment sequences) from copies of the same DNA. (Kevin and Jacob)
- A repeat finder that looks for repeated sequences of segments to determine when the DNA motor has fallen off and another one taken over. (Jacob)
- A backslip finder that recognizes when the DNA has slipped backwards a base. (no one working on general case)

Future Work: HMMs

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

Web sites

These slides: http://users.soe.ucsc.edu/~karplus/papers/segmentation-2013-nov-14.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