BMC Bioinformatics

BioMed Central

Proceedings

Open Access

Duration learning for analysis of nanopore ionic current blockades

Alexander Churbanov*[1], Carl Baribault[2] and Stephen Winters-Hilt[1,2]

Address: [1]The Research Institute for Children, 200 Henry Clay Ave., New Orleans, LA 70118, USA and [2]Department of Computer Science, University of New Orleans, New Orleans, LA, 70148, USA

Email: Alexander Churbanov* - atchourb@cs.uno.edu; Carl Baribault - cbaribault@bellsouth.net; Stephen Winters-Hilt - winters@cs.uno.edu

* Corresponding author

from Fourth Annual MCBIOS Conference. Computational Frontiers in Biomedicine
New Orleans, LA, USA. 1–3 February 2007

Published: 1 November 2007

BMC Bioinformatics 2007, **8**(Suppl 7):S14 doi:10.1186/1471-2105-8-S7-S14

This article is available from: http://www.biomedcentral.com/1471-2105/8/S7/S14

Abstract

Background: Ionic current blockade signal processing, for use in nanopore detection, offers a promising new way to analyze single molecule properties, with potential implications for DNA sequencing. The alpha-Hemolysin transmembrane channel interacts with a translocating molecule in a nontrivial way, frequently evidenced by a complex ionic flow blockade pattern. Typically, recorded current blockade signals have several levels of blockade, with various durations, all obeying a fixed statistical profile for a given molecule. Hidden Markov Model (HMM) based duration learning experiments on artificial two-level Gaussian blockade signals helped us to identify proper modeling framework. We then apply our framework to the real multi-level DNA hairpin blockade signal.

Results: The identified upper level blockade state is observed with durations that are geometrically distributed (consistent with an a physical decay process for remaining in any given state). We show that mixture of convolution chains of geometrically distributed states is better for presenting multimodal long-tailed duration phenomena. Based on learned HMM profiles we are able to classify 9 base-pair DNA hairpins with accuracy up to 99.5% on signals from same-day experiments.

Conclusion: We have demonstrated several implementations for *de novo* estimation of duration distribution probability density function with HMM framework and applied our model topology to the real data. The proposed design could be handy in molecular analysis based on nanopore current blockade signal.

Background

In its quest for survival the bacterium *Staphylococcus aureus* secretes α-hemolysin monomers that bind to the outer membrane of susceptible cells, where seven such units can oligomerize to form a water-filled transmembrane channel [1-4]. The channel can cause death to the target cell by rapidly discharging vital molecules (such as ATP) and disturbing the membrane potential.

Suspended in lipid bilayer [see Additional File 1] the α-hemolysin channel becomes a sensor when large molecules interact with the nanopore and modulate uniform ionic flow through the channel. Driven by transmembrane potential, single stranded DNA or RNA molecules translocate through the nanopore [5,6], while more complex hairpins either unzip and translocate [7,8] or toggle in the channel's vestibule [8,9] [see Additional File 1]. The durations of ionic flow blockade events in these experiments are important signatures of interacting nucleic acid fragments composition [7,10] or in certain cases characterize the molecular length [11].

Two distinct approaches of duration modelling have been proposed for HMM framework by speech recognition community, based on *explicit* duration modelling, which is normally implemented with histograms or parametric distributions, and *implicit* modeling based on set of geometrically distributed self-recurring nodes [12]. The most common way of implementing explicit duration model is Generalized Hidden Markov Model (GHMM), where each state can emit more than one symbol at a time [13]. Following [14], the optimal GHMM parse could be expressed by the following equation

$$\begin{aligned}\phi_{optimal} &= \arg\max_{\phi} P(\phi \mid S) \\ &= \arg\max_{\phi} \frac{P(\phi, S)}{P(S)} \\ &= \arg\max_{\phi} P(\phi, S) \\ &= \arg\max_{\phi} P(S \mid \phi) P(\phi) \\ &= \arg\max_{\phi} \prod_{i=1}^{n} P_e(S_i \mid q_i, d_i) P_t(q_i \mid q_{i-1}) P_d(d_i \mid q_i)\end{aligned} \quad (1)$$

where φ is a parse of the sequence consisting of a series of states q_i and state durations d_i, $0 \le i \le n$, with each state q_i emitting subsequence S_i of length d_i, so that the concatenation of all $S_0 S_1 \ldots S_n$ produces the complete output sequence S. $P_e(S_i|q_i, d_i)$ denotes the probability that state q_i emits subsequence S_i of duration d_i. $P_t(q_i|q_{i-1})$ is GHMM transition probability from state q_{i-1} to state q_i and $P_d(d_i|q_i)$ is the probability that state q_i has duration d_i. The primary objective, expressed in (1), is to combine probability returned by content probabilistic model (such as HMM) with duration probability for optimal parse. The GHMM implementation, as well as HMM-with-Duration approach mentioned in [15], require explicit assignment of duration histogram to run Viterbi decoding.

When we try to classify single DNA base pair by nanopore ionic flow blockade signal processing [16], we frequently have to deal with a sequence of blockades resulting from complex molecular interactions with unknown states. For this reason, we are interested in *de novo* learning of emission content and duration distributions corresponding to these stationary blockade states. In this study we research several approaches to the problem of duration and content sensor learning in the context of nanopore ionic flow blockades analysis.

Results and discussion

Explicit duration model learning experiment

The objective of this experiment was to evaluate the ability of a randomly initialized explicit Duration HMM (DHMM), as described [see Section *The explicit duration HMM implementation*], to learn the original duration phenomena present in artificial data. We use the Expectation Maximization (EM) training procedure, as discussed [see *Appendix D*], to iteratively reinforce the network structure to match the test data set topology. For the model training purposes we have generated 120 sequences of 1,000 emissions each with the maximum state durations of 30 according to protocol discussed [see Section *Running original explicit DHMM in generative mode*].

First, we learned randomly initialized geometric model, as described [see Section *Geometric duration distribution and convolution of geometric states*], for 200 iterations to reliably recover the two major Gaussian emitting components and roughly estimate the average duration for two states. An accuracy of 85.75% has been achieved by Viterbi decoding [see *Appendix D*] on the learned geometric duration model for the test set, which constitutes 95.72% performance of the original explicit DHMM run on the same test set. Here and further we identify accuracy as the ratio of correctly decoded emissions to the entire number of emissions in the given time series

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN},$$

where True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) are among the classified data points.

We use the recovered Gaussian emissions and initial probabilities to initialize the explicit DHMM, and we use learned average state duration as the expected prior for the explicit duration histogram initialization. We freeze the emission Gaussian Probability Density Functions (PDFs) so that they don't change. Upon convergence of the explicit DHMM to a local likelihood maximum we record 88.98% Viterbi decoding accuracy on the test set, which is 99.33% of the original explicit DHMM performance, i.e. we were able to recover the duration phenomena with performance almost identical to the original explicit

DHMM. Figure 1 shows histograms obtained for the state durations. Although their shape approximates the original duration PDF pretty well, the recovered histograms experience substantial abrupt and unwanted variations. Another shortcomings of this duration learning strategy is that it is extremely slow and requires simultaneous estimation of large number of parameters. Therefore, in the next section we present an approach based on a convolution chain of geometric duration states.

Convolution of states learning experiment

In this experiment we construct aggregate model states as convolution chain of three geometric distributions. The convolution chain for identical geometric distributions can be represented as a bell-shaped Negative Binomial discrete PDF, as discussed [see Section *Geometric duration distribution and convolution of geometric states*].

The resulting convolution model trained with EM algorithm [see *Appendix D*] on an artificial nanopore signal with maximum state duration of 30 of 120 sequences 1,000 emissions each, generated according to protocol discussed [see Section *Running original explicit DHMM in generative mode*], is shown in Figure 2. In this learning experiment we use known emissions $\mathcal{N}(45, 20)$ and $\mathcal{N}(50, 20)$, that do not change in the process of learning, and initialize the prior probabilities and transitions with the expected state durations. Interestingly, direct use of the learned convolution model in Viterbi decoding produces reconstruction fidelity substantially inferior to the simple geometrically-distributed model. The convolution chain has full power only for forward-backward procedure [13] for likelihood estimation [see *Appendix D*] and does not work for representing duration phenomena in case of Viterbi decoding. Therefore, we use the histograms shown in Figure 2 to initialize explicit DHMM transitions.

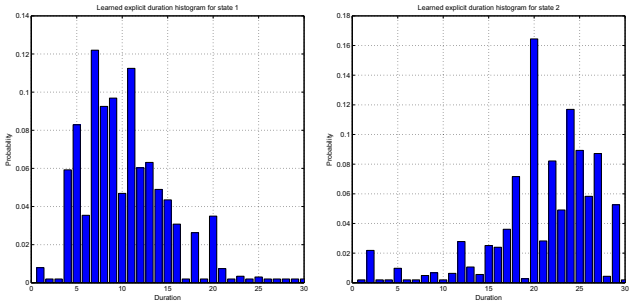


Figure 1
Recovered duration histograms by learning the randomly initialized explicit duration DHMM for the maximum state duration of 30.

Such hybrid explicit DHMM achieves on Viterbi decoding accuracy of 90.20%, which is 99.69% performance of original explicit DHMM on the same test set. Therefore, by learning chains of convolving geometrically-distributed components we achieve similar or better performance as compared to direct learning of explicit DHMM, in much shorter time period. The experiment clearly demonstrates the ability of a convolution chain to learn the complex duration phenomena in the data, outperforming the simple geometric duration model. Convolving states cannot generate or model duration phenomena shorter than the chain length (three in our case), therefore the number of convolving states should be used conservatively.

Performance of Viterbi decoding depending on blockade maximum duration

We run Viterbi decoding for the original models presented [see Sections *The explicit duration HMM implementation* and *Geometric duration distribution and convolution of geometric states*] with results shown in Table 1. In case of the geometric model we used transitions assigned according to simple maximum likelihood formula [see Section *Geometric duration distribution and convolution of geometric states*] estimated on the test set emissions of the original explicit DHMM.

The geometric distribution model runs faster, but decoding performance is always inferior compared to the explicit DHMM. The geometric HMM appears to be simple and crude approximation to the duration signal. Explicit DHMM runs D times slower than simple geometric model, but produces superior results to any other types of implementations, given the exact duration histogram. The higher the maximum state duration D and the longer the average phenomena duration is, the better decoding quality we can obtain for all the models.

Learning durations on the real blockade signal

Here we analyze the ionic flow blockade signal resulting from the nine base pair "upper level toggler" hairpin DNA molecule CGTGGAACGTTTTCGTTCCACG, generated according to protocol described in [9] (signal was filtered at 50 kHz bandwidth using an analog low pass Bessel filter with the 20 μs analog-to-digital sampling). Due to its unique sequence in the stem region and its interactions with the channel's vestibule it produces a rich signal (the upper level toggle) [17].

From the physical perspective the hairpin molecule can undergo different modes of capture blockade, such as Intermediate Level (IL), Upper Level (UL), Lover Level (LL) conductance states and spikes (S) [18]. When a 9 bp DNA hairpin initially enters the pore, the loop is perched in the vestibule mouth and the stem terminus binds to

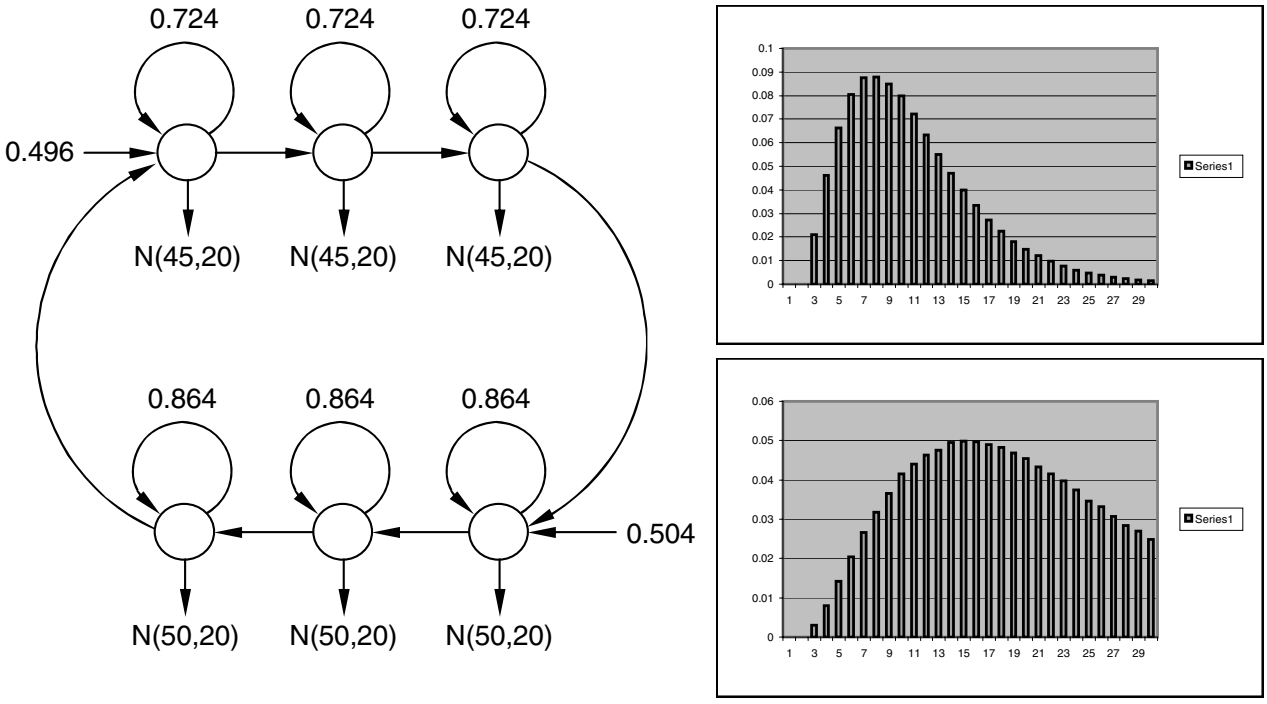


Figure 2
Learned convolution model for the two known Gaussian emitting components with maximum state duration of 30. Discrete duration distribution histograms are put next to each aggregate state.

amino acid residues near the limiting aperture. This results in the IL conductance level. When the terminal basepair desorbs from the pore wall, the stem and loop may realign, resulting in a substantial current increase to UL. Interconversion between the IL and UL states may occur numerous times with UL possibly switching to the LL state. This LL state corresponds to binding of the stem terminus to amino acids near the limiting aperture but in a different manner from IL. From the LL bound state, the duplex terminus may fray, resulting in extension and capture of one strand in the pore constriction resulting into short term S state. The allowed transition events between these levels $IL \Leftrightarrow UL \Leftrightarrow LL \Leftrightarrow S$ can happen at any time during the analysis procedure.

For the purposes of *de novo* emission levels detection we have learned the mixture of six Gaussian (MoG) components [see *Appendix B*] on the raw ionic flow blockade signals. EM learning step converged to the following mixture of six Gaussian components

$$\begin{aligned} p(x) &= 0.228\times\mathcal{N}(x\mid 52.26,1.18) + 0.08\times\mathcal{N}(x\mid 62.35,13.57) + 0.18\times\mathcal{N}(x\mid 55.05,6.13) \\ &+ 0.09\times\mathcal{N}(x\mid 42.29,3.92) + 0.28\times\mathcal{N}(x\mid 38.82,1.68) + 0.14\times\mathcal{N}(x\mid 59.87,1.83) \end{aligned} \quad (2)$$

as could be seen in Figure 3, where $\mathcal{N}(x|\mu, \sigma^2)$ is normal PDF.

We took four primary components from the recovered MoG (2) and applied them as emissions to the convolution model [see Section *Geometric duration distribution and convolution of geometric states*] for four aggregate states. We learn the model on the ionic flow blockade signal of size

Table 1: Test set decoding performance for various aggregate state sizes. Here we show percentage of states recovered correct in Viterbi decoding for various methods.

Max. state duration	Explicit DHMM	Geometric duration HMM
6	81.04%	72.70%
10	83.36%	74.43%
24	88.80%	82.18%
30	90.06%	87.94%

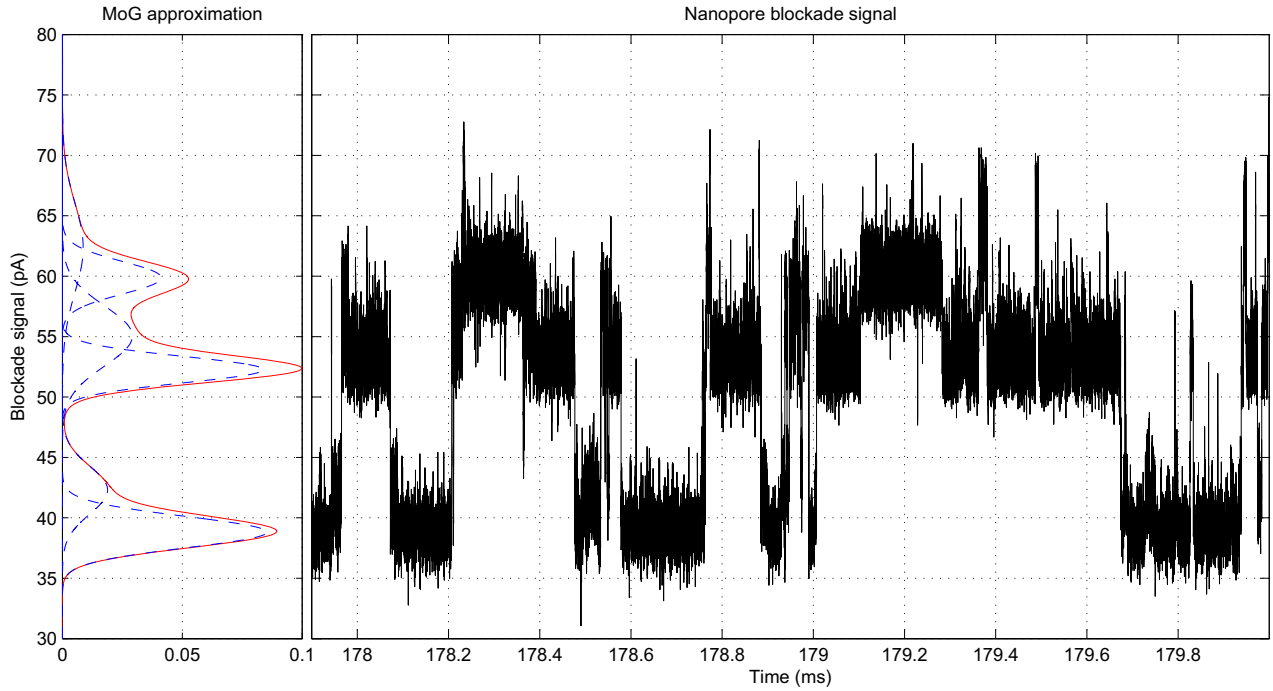


Figure 3
Resulting PDF for learning the mixture of six Gaussian components on ionic flow blockade signal.

173,000 samples with the recovered topology shown in Figure 4. The graph of transition nodes connecting the learned aggregate states appears to be sparse with nonzero transitions [see Additional File 2]. This analysis shows that not all transitions between molecular interaction states are allowed. Interestingly enough, the second state has transitions to other three states. According to the interaction physical model discussed above the molecule should bounce back and fourth between the deeper blockade levels, thus components $\mathcal{N}(52.26, 1.18)$ and $\mathcal{N}(38.82, 1.68)$ dominate. The recovered geometric distribution of the blockade events (a classic physical decay phenomenon), indicate that upper level toggler molecule has constant state-dependent probability to dissociate from one interaction state and transit to another physically feasible conformation.

Aggregate states 1, 3 and 4 converged to pure geometric distributions with no apparent bell-shaped duration phenomena, as could be seen in Figure 4. However, as could be seen in Figures 5(a) and 5(d), the long-tailed distribution does not fall nicely in the framework of geometric duration. The geometric durations learned on these long-tailed distributions does not approximate well neither the initially sharp peak nor the long tail in duration histograms and therefore should really be treated as multimodal distribution approximated by mixture of geometric components. On the other hand, the histograms shown in Figures 5(b) and 5(c) fall perfectly into framework of geometric PDF.

We improve the histograms for multimodal long-tailed distributions by training the mixture of two convolution chains. Resulting convolution mixture generative histograms present the original phenomena much better as could be seen in Figure 6. Upon introduction of convolution mixture the model logarithmic likelihood [see *Appendix B*], given training sequence, has increased from -420068.73 to -418636.5 for the fully trained topologies which indicated better model fit to data. Mixing more than two components per aggregate state did not provide apparent improvements and unnecessarily complicated the model.

System performance on the 9CG versus 9TA data

We took 3,460 ms ionic flow blockade signals for 9GC and 9TA molecules, recorded according to protocol described in [9], and automatically learned the convolution topology according to the strategy [see Section *Learning durations on the real blockade signal*]. The remaining sequences, generated the same day, were used as a test set. We have split the test set sequences into chunks of 100 ms each to investigate short-term classification performance,

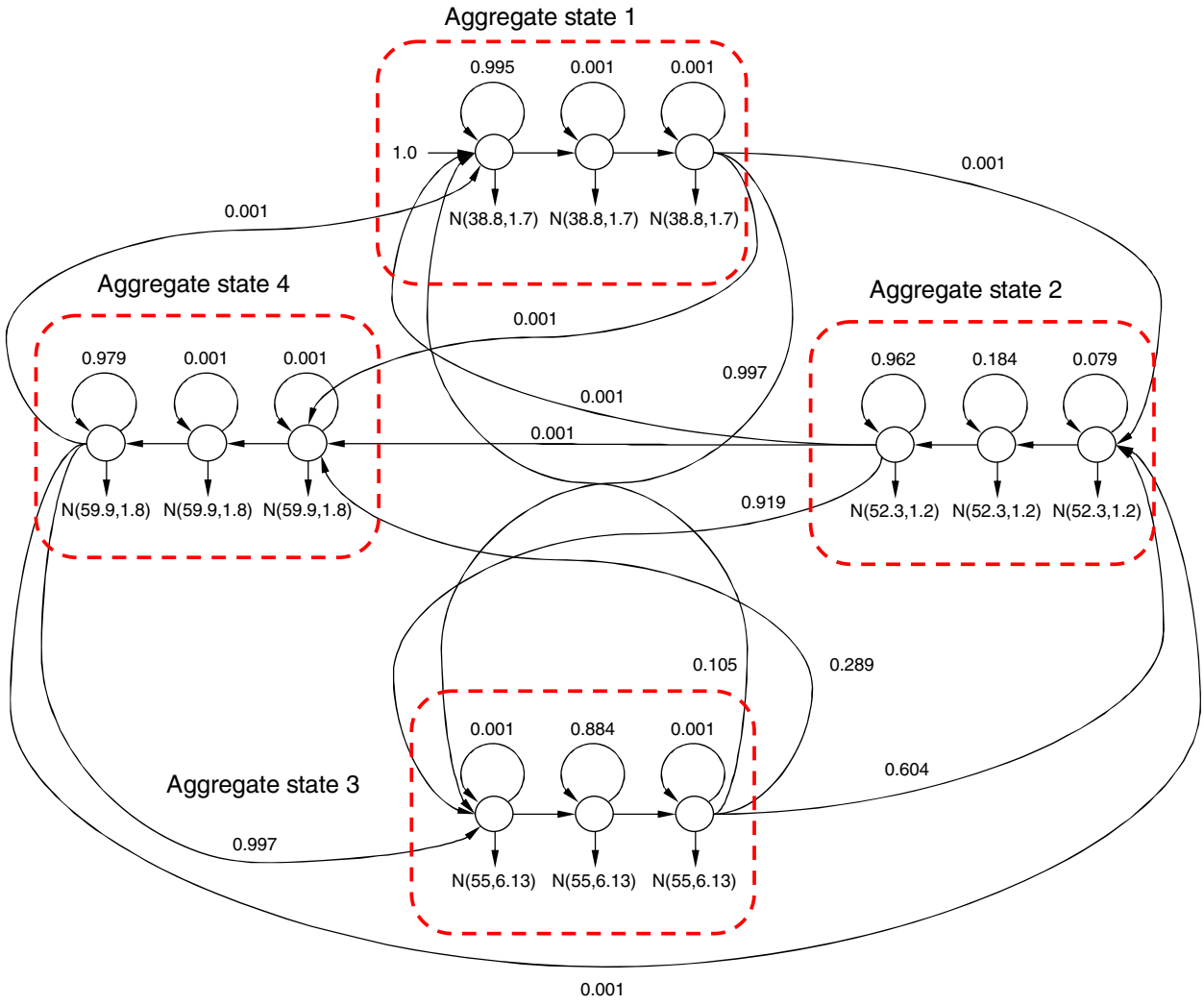


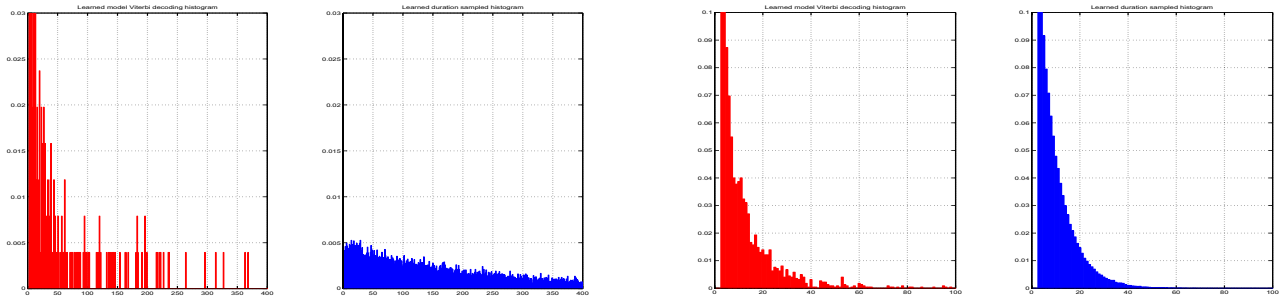
Figure 4
Learned convolution model for the four major MoG components recovered. Transitions with weight 0.001 are negligible and were forcefully assigned by learning algorithms not to cause underflow in forward-backward procedure.

resulting into 13,753 test fragments for 9TA signal and 15,652 9GC test fragments as could be seen in Figure 7. We run both 9GC and 9TA learned HMM profiles on the test sequences and classify them according to maximum likelihood. We achieve classification accuracy of 99.56% on the 9GC test set and 97.87% on the 9TA test set.

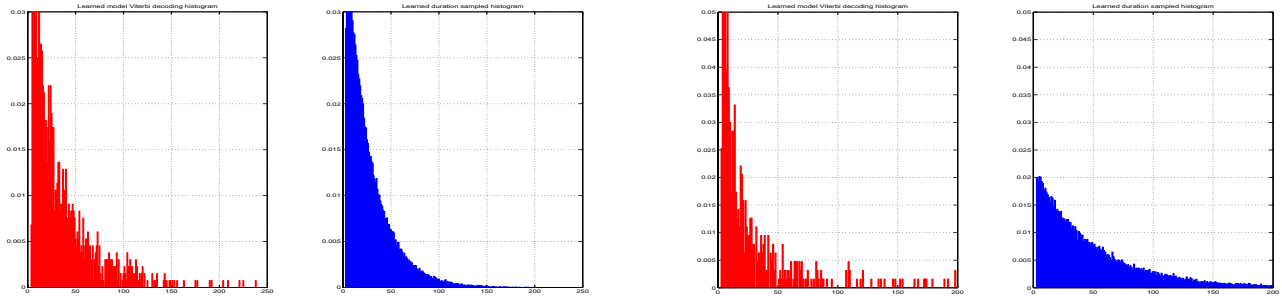
Conclusion

Although running slowly, the explicit DHMM design has many advantages over other duration representation methods for HMMs, such as using unmodified Viterbi decoding algorithm and possibility for exact representation of any duration phenomena. Original explicit DHMM produced the best results in all artificial test categories. However, learning of such topology can quickly turn into a grim experience, since too many parameters need to be learned with noisy data.

The geometric duration distribution HMM is simple, but is not well suited to complex duration data analysis [see Section *Performance of Viterbi decoding depending on blockade maximum duration*]. Convolution of geometric states, especially a mixture of such aggregate states, is a much more robust and powerful method of interpolating noisy multimodal duration phenomena encountered in ionic flow blockade time series analysis. The software used to



(a) Component $\mathcal{N}(38.82, 1.68)$ (c) Component $\mathcal{N}(55.05, 6.13)$



(b) Component $\mathcal{N}(52.26, 1.18)$ (d) Component $\mathcal{N}(59.87, 1.83)$

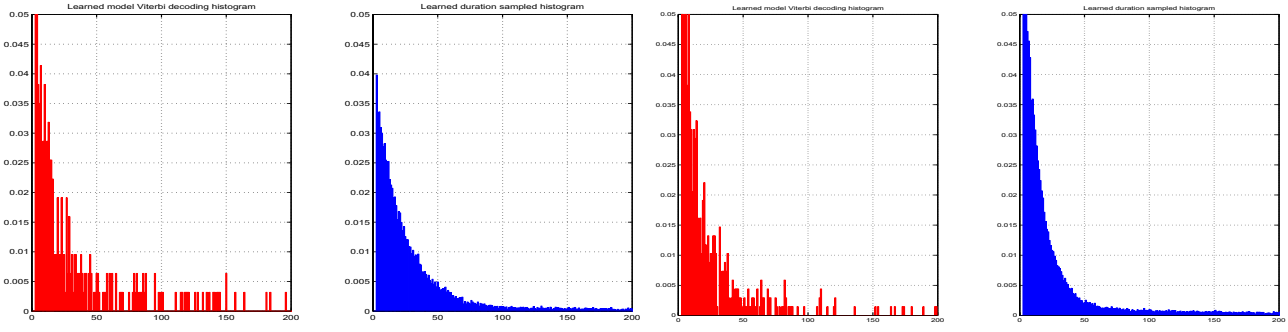
Figure 5
The duration histograms recovered. Histograms recovered by running Viterbi decoding of learned convolution model on the ionic flow blockade signal are shown in red. Blue histograms (to the right in each subfigure) are produced by running learned convolution model in generative mode.

conduct experiments in this report is freely available at http://logos.cs.uno.edu/~achurban.

Methods

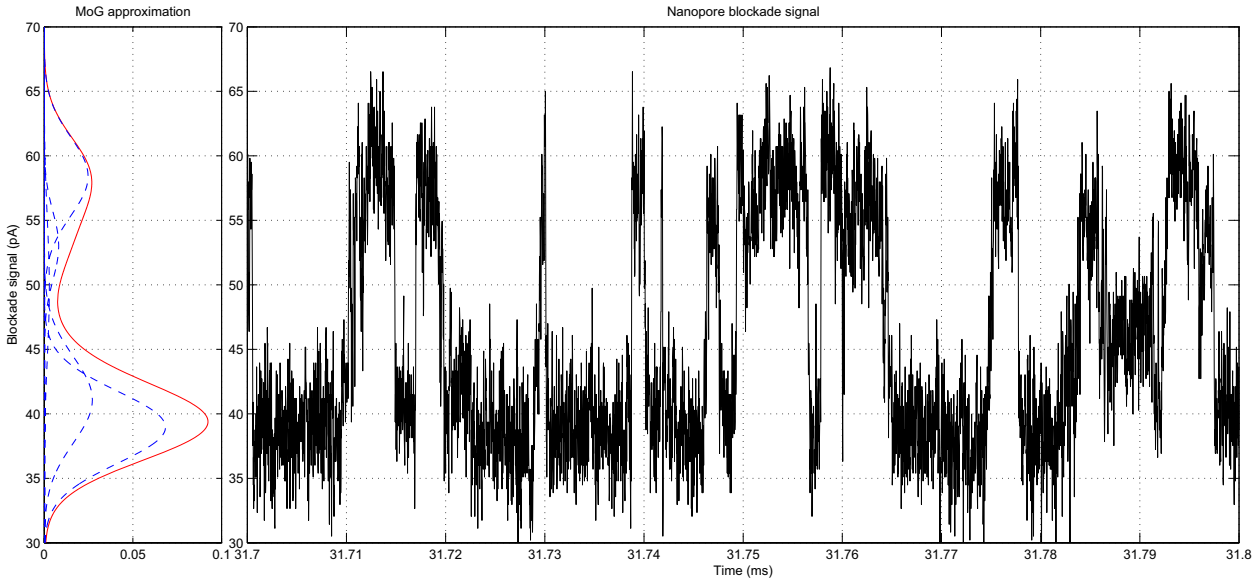
The explicit duration HMM implementation

In Figure 8 we show the explicit DHMM topology we use to combine duration with content sensors, which we refer to as the original explicit DHMM throughout the manu-

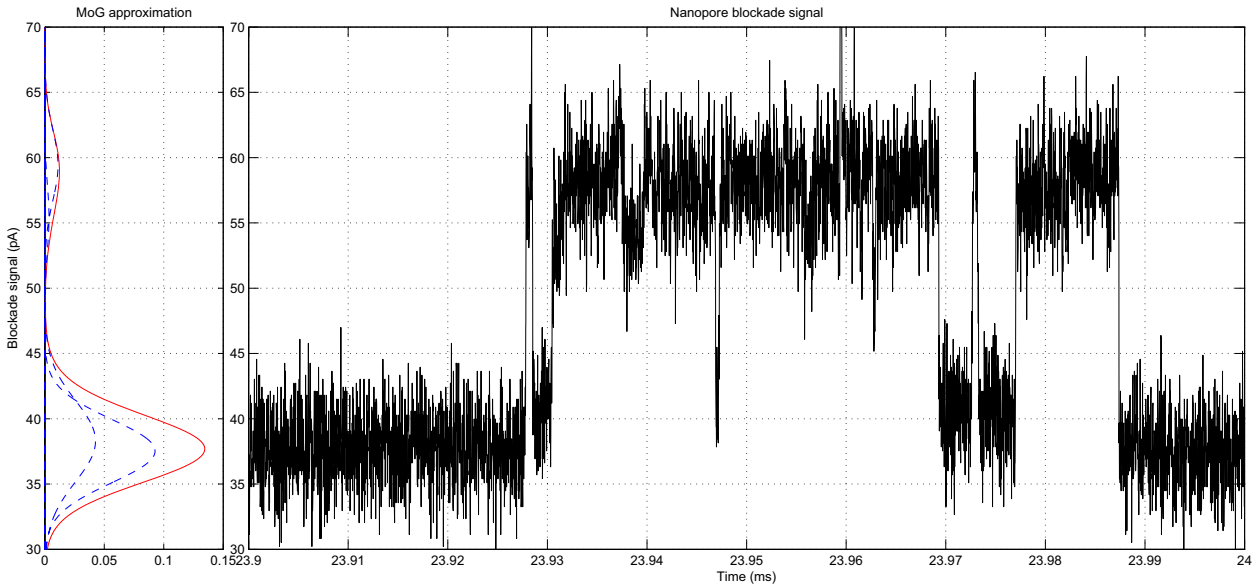


(a) Component $\mathcal{N}(38.82, 1.68)$ (b) Component $\mathcal{N}(59.87, 1.83)$

Figure 6
The duration histograms recovered. In this case we approximate long-tailed histogram by mixture of two convolution chains, which produces better fit as compared to Figures 7a and 7d.



(a) 9TA series 100 ms example with corresponding MoG density



(b) 9GC series 100 ms example with corresponding MoG density

Figure 7
Sample 100 ms nanopore blockade signals for 9TA and 9GC molecules with corresponding MoG densities.

script. This model follows the topology discussed in [19] for exact duration implementation and is similar in computational complexity to a more common explicit duration modeling of GHMM [13,14,20]. Our implementation takes advantage of intuitive duration presentation, instead of using disjoint parametric distributions or histograms for duration modeling that complicate decoding algorithm well beyond standard Viterbi procedure.

Our model uses standard Viterbi decoding algorithm [see *Appendix D*], which we implemented in linear memory using linked list of back pointers in addition to implementation of Forward-Backward algorithm [21] for EM

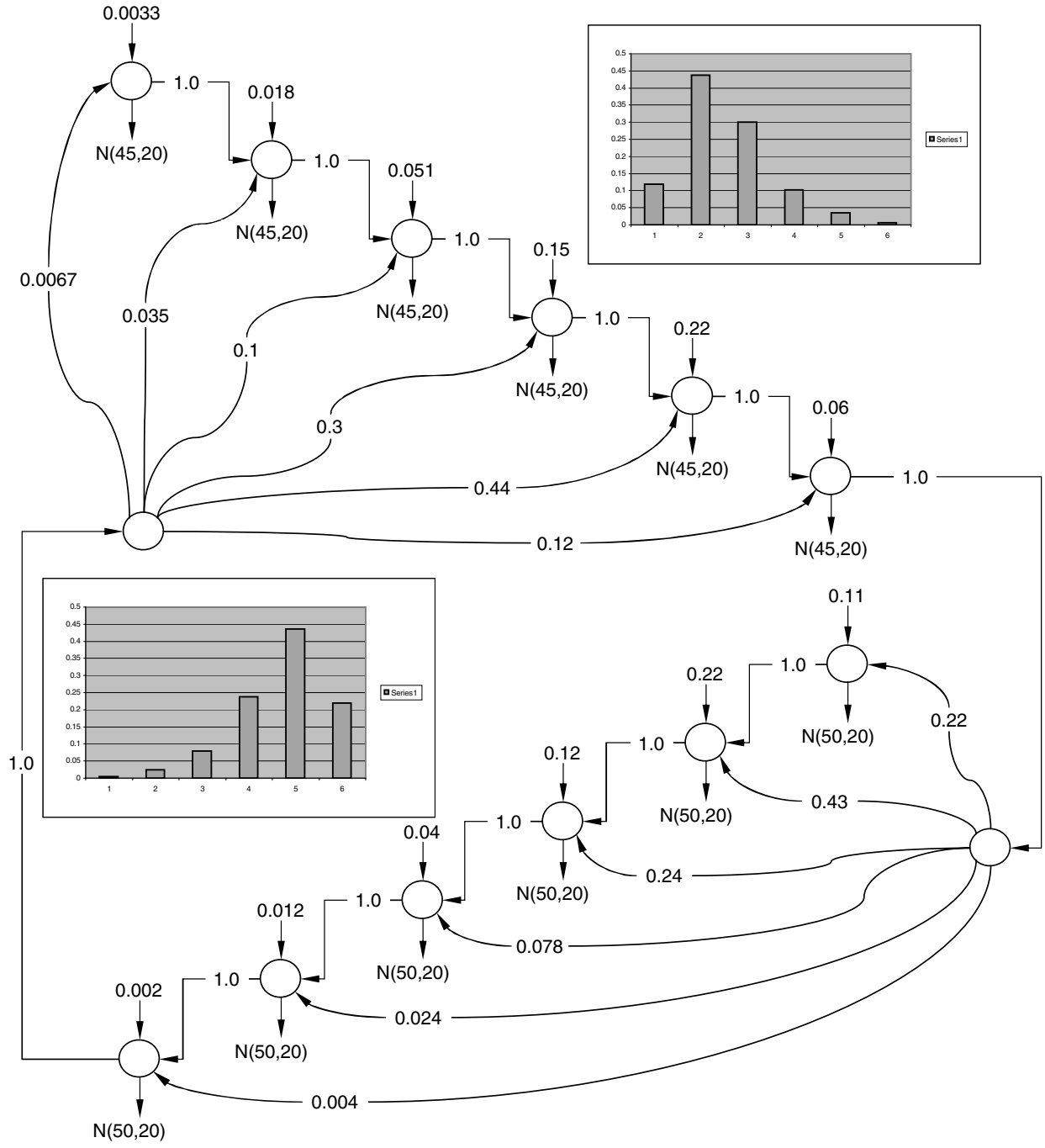


Figure 8
The explicit DHMM topology we use with the maximum state duration of 6. Discrete duration distribution histograms are put next to each aggregate state.

learning [see *Appendix D*] with memory use proportional to the number of states. The maximum state duration D has to be imposed on each duration histogram in this model which might seem as a limitation in case of long-tailed distribution. This deficiency could easily be resolved by adding the geometrically distributed states to explicit DHMM, that are capable of modeling simple infinite long tailed durations [see Section *Geometric duration*

distribution and convolution of geometric states], and use explicit part of the model to catch only the initial complex duration phenomena.

In this study we use two aggregate groups of states with corresponding discrete duration PDF obtained by discretizing to continuous PDFs [see Additional File 3], denoted as first and second states. The thermal noise of ionic flow at certain blockade level is approximated extremely well by the Gaussian PDF emission from HMM hidden states [see *Appendix C*]. The aggregate states are formed by lossless chains of transitions between hidden states, where we sacrifice the probability score only to enter the chain. We use Gaussian emissions $\mathcal{N}(45, 20)$ and $\mathcal{N}(50, 20)$ in the first and second aggregate states, correspondingly. Initial probabilities correspond to 50% chance to begin decoding in the first aggregate state and 50% for the second aggregate state.

Running original explicit DHMM in generative mode

We run original explicit DHMM [see Section *The explicit duration HMM implementation*] running in generative mode to get the test set of 1,000,000 sample points of artificial nanopore blockade signal [see Additional File 4]. In order to generate the test set we simply traverse the HMM graph in stochastic fashion according to transition probabilities assigned to edges, where each transition culminates in emission from PDF assigned to a state [see *Appendix C*]. Along with the emissions we record the known emitting hidden states for performance testing and parameter estimates of geometrically distributed HMM [see Section *Geometric duration distribution and convolution of geometric states*]. We use the test set to evaluate performance of various HMM implementations and learning techniques [see Sections *Explicit duration model learning experiment*, *Convolution of states learning experiment* and *Performance of Viterbi decoding depending on blockade maximum duration*].

Geometric duration distribution and convolution of geometric states

The geometric duration distribution is implemented as a self-recurring hidden state in the HMM framework and there are many merits of such duration modeling. The geometric duration distribution is modeled by only one state, which results in very compact probability tables for forward-backward and Viterbi decoding algorithms. Random variable x is distributed according to geometric law $p_x(k) = p(1 - p)^{k-1}$ where $k = 1, 2, 3...$ and $1 - p$ is the probability to stay in the same state. Parameter p fully characterizes this distribution and could be easily estimated by maximum likelihood, which is calculated as following

$$\hat{p} = \frac{1}{\text{Expected duration}} = \frac{1}{\frac{1}{N}\sum_{i=1}^{N} k_i}$$

where N is the number of discrete duration samples $k_1,..., k_N$. The topology of the two state model with duration distributed according to geometric law [see Additional File 5].

The chain of consecutive identical geometrically distributed states could represent bell-shaped *Negative binomial* duration distributions [19], as discussed [see *Appendix A*]. In the case of non-identical geometrically distributed connected states the PDF remains bell-shaped since the number of possible paths through the model $\binom{k-1}{n-1}$ increases as the number of trials k grows, but the total sum of probabilities attributed to all these paths through n geometric components decreases. The mixture of aggregate states distributed according to Negative binomial law, as shown in Figure 9, can interpolate duration distribution even better, especially in case of multimodal distributions. A nice attribute of the duration representation, with geometrically distributed states, is that we are able to interpolate the noisy duration histogram, common for ionic flow time series, with much smoother discrete distribution.

Competing interests

The authors declare that they have no competing interests.

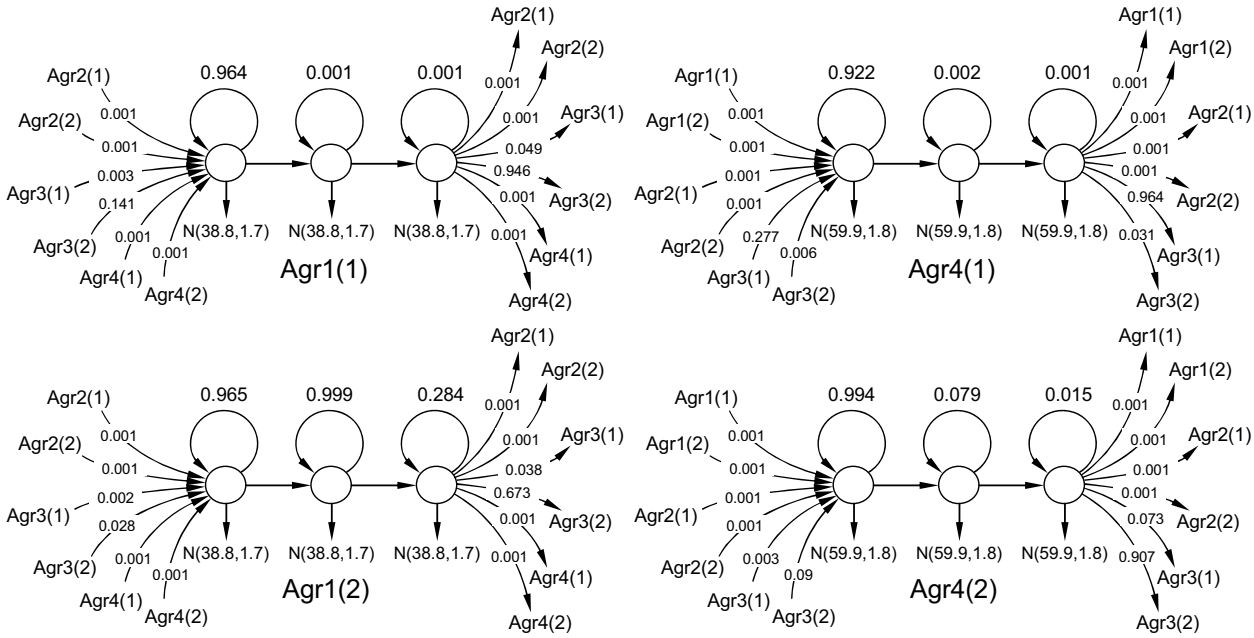
Authors contributions

AC conceptualized the use of explicit duration HMM and convolution of geometric duration states. AC has implemented the system prototype, learned the models and drafted the manuscript. CB helped with implementing HMM-with-Duration, conducting performance tests on artificial and real nanopore blockade signal. SWH helped with writing up the manuscript and provided many valuable suggestions throughout the study. All authors read and approved the final manuscript.

Appendices

Appendix A – Convolution of geometric distributions

In statistics, the probability distribution of the sum of several independent random variables is the convolution of their individual distributions. Suppose random variable x is distributed according to geometric law $p_x(k) = p\, q^{k-1}$ where $k = 1, 2, 3...$ is the number of trials to exit the state and $q = 1 - p$ is the probability to stay in the same state. The moment generating function for geometric distribution is



(a) Mixture of two convolution chains for first aggregate state. (b) Mixture of two convolution chains for fourth aggregate state.

Figure 9
Mixture of convolutions for Aggregate states 1 (Agr1) and 4 (Agr4) where in brackets we include mixture component number. Transitions with weight 0.001 are negligible and were forcefully assigned by learning algorithms not to cause underflow in forward-backward procedure.

$$G_x(s) = \frac{ps}{1-qs} \quad \text{if} \quad |s| < q^{-1}.$$

If random variable x is distributed according to *Negative binomial*, i.e. $x \sim NegBin(n, p)$, then the moment generating function is written as

$$G_x(s) = \sum_{k=0}^{\infty} \binom{k-1}{n-1} p^n q^{k-n} s^k = \left(\frac{ps}{1-qs}\right)^n \quad \text{if} \quad |s| < q^{-1}.$$

The Negative binomial moment generating function is a product of n geometric distribution moment generating functions, which corresponds to convolution [19] of n identical geometric distributions with parameter p [see Additional File 6]. Distinct bell-shaped plot of Negative binomial distribution PDF with parameters $p = 0.99$ and $n = 1,..., 5$ presented [see Additional File 7].

Appendix B – Learning the mixture models

The one-dimensional MoG model [22] of M components is a degenerate case of HMM

$$p(o \mid \Theta) = \sum_{i=1}^{M} \alpha_i \mathcal{N}(o \mid \Theta_i) = \sum_{i=1}^{M} \alpha_i \mathcal{N}(o \mid \mu_i, \sigma_i^2) \text{ with } \sum_{i=1}^{M} \alpha_i = 1,$$

where α_i is mixing proportions.

The objective of learning is to maximize likelihood function $p(O|\Theta) = \prod_{i=1}^{N} p(o_i \mid \Theta) = \mathcal{L}(\Theta|O)$, i.e. we wish to find locally optimal set of parameters $\Theta^* = \underset{\Theta}{argmax}\, \mathcal{L}(\Theta|O)$ by using Expectation Maximization (EM) iterative procedure given the set of data points O.

Expectation step in mixture fitting algorithm is made through computing responsibility matrix of the components given data points

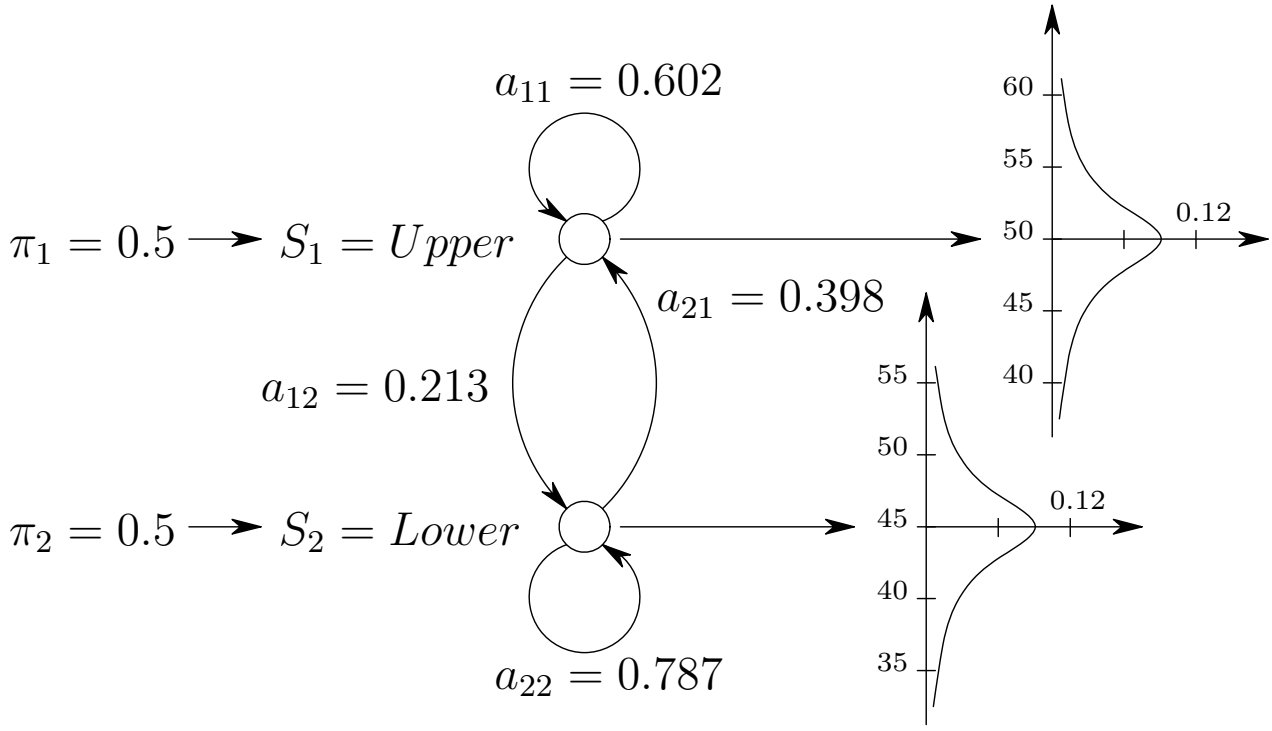


Figure 10
Simple HMM topology with emissions drawn from $\mathcal{N}$ (45, 20) and $\mathcal{N}$ (50, 20).

$$\left.\begin{array}{cccc} p(\Theta_1 \mid o_1, \mathbf{\Theta}) & \cdots & p(\Theta_M \mid o_1, \mathbf{\Theta}) \\ p(\Theta_1 \mid o_2, \mathbf{\Theta}) & \cdots & p(\Theta_M \mid o_2, \mathbf{\Theta}) \\ p(\Theta_1 \mid o_3, \mathbf{\Theta}) & \cdots & p(\Theta_M \mid o_3, \mathbf{\Theta}) \\ \cdots & \cdots & \cdots \\ \underbrace{p(\Theta_1 \mid o_N, \mathbf{\Theta}) \quad \cdots \quad p(\Theta_M \mid o_N, \mathbf{\Theta})}_{M \text{ mixture components}} \end{array}\right\} N \text{ data points}$$

We use Bayesian rule to find posterior probability (responsibility) of a mixture component with parameters Θ_i for data point o_j

$$p(\Theta_i \mid o_j, \mathbf{\Theta}) = \frac{\alpha_i \mathcal{N}(o_j \mid \Theta_i)}{\sum_{k=1}^{M} \alpha_k \mathcal{N}(o_j \mid \Theta_k)}.$$

Expectation step is followed my maximization step where we re-estimate parameters

(a) Mixture proportions $\hat{\alpha}_k = \frac{1}{N} \sum_{i=1}^{N} p(\Theta_k \mid o_i, \mathbf{\Theta})$,

(b) Mean $\hat{\mu}_k = \frac{\sum_{i=1}^{N} o_i p(\Theta_k \mid o_i, \mathbf{\Theta})}{\sum_{i=1}^{N} p(\Theta_k \mid o_i, \mathbf{\Theta})}$,

(c) Variance $\hat{\sigma}_k^2 = \frac{\sum_{i=1}^{N} p(\Theta_k \mid o_i, \mathbf{\Theta})(o_i - \hat{\mu}_k)(o_i - \hat{\mu}_k)}{\sum_{i=1}^{N} p(\Theta_k \mid o_i, \mathbf{\Theta})}$.

Appendix C – Definition of Hidden Markov Model

The Hidden Markov Model (HMM) is a widely accepted stochastic modelling tool [23] used in various domains, such as speech recognition [24] and bioinformatics [25]. HMM is a stochastic finite state machine where each transition between hidden states is culminated by a symbol emission. The HMM could be represented as a directed graph with *N* states where each state could emit either discrete character or continuous value drawn from PDF. In order to describe HMM we need the following parameters

• Set of states, we label individual states as $S = \{S_1, S_2, ..., S_N\}$, and denote the state visited at time t as q_t,

• Set of PDFs from where emission is drawn, in our case we use Normal distributions $B = \{\mathcal{N}(\mu_1, \sigma_1^2), ..., \mathcal{N}(\mu_N, \sigma_N^2)\}$,

• The state-transmission probability matrix $A = \{a_{ij}\}$, where $a_{ij} = p(q_{t+1} = j | q_t = i)$,

• The initial state distribution vector $\prod = \{\pi_1, ..., \pi_N\}$.

Set of parameters $\lambda = (\prod, A, B)$ completely specifies HMM. A simple example of HMM with two states where emissions are drawn from normal distributions $\mathcal{N}$ (45, 20) and $\mathcal{N}$ (50, 20) is shown in Figure 10.

Appendix D – HMM forward-backward algorithm and Viterbi decoding

Here we adopt notation from [13] and report final HMM parameters update rules for EM learning algorithm rigorously derived in [22].

Viterbi algorithm for finding optimal parse

The Viterbi algorithm is a dynamic programming algorithm that runs on HMM for finding the most likely sequence of hidden states, called the Viterbi path, that result in an observed sequence.

1. Initially $\delta_1(i) = \pi_i \mathcal{N}(o_1 | \Theta_i)$, $\psi_1(i) = 0$ for $1 \le i \le N$,

2. $\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] \mathcal{N}(o_t | \Theta_j), \psi_t(j) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}]$ for t = 2,..., T and $1 \le j \le N$,

3. Finally $q_T^* = \arg\max_{1 \le i \le N} [\delta_T(i)]$, trace back $q_t^* = \psi_{t+1}(q_{t+1}^*)$ for $t = T - 1, T - 2, ..., 1$ with optimal decoding $Q^* = \{q_1^*, q_2^*, ..., q_T^*\}$.

HMM expectation step

We need to find expected probabilities of being at a certain state at a certain moment of time with forward-backward procedure.

Forward procedure By definition $\alpha_t(i) = p(o_1, o_2, ..., o_t, q_t = S_i | \lambda)$ is calculated the following way

1. Initially $\alpha_1(i) = \pi_i \mathcal{N}(o_1 | \Theta_i)$ for $1 \le i \le N$,

2. $\alpha_t(j) = \left[\sum_{i=1}^N \alpha_{t-1}(i) a_{ij} \right] \mathcal{N}(o_t | \Theta_j)$ for t = 2, 3,..., T and $1 \le j \le N$,

3. Finally $p(O | \lambda) = \sum_{i=1}^N \alpha_T(i)$ is the sequence *likelihood* according to model.

Backward procedure By definition $\beta_t(i) = p(o_{t+1}, o_{t+2}, ..., o_T, q_t = S_i | \lambda)$ is calculated the following way

1. Initially $\beta_T(i) = 1$ for $1 \le i \le N$,

2. $\beta_t(i) = \sum_{j=1}^N a_{ij} \mathcal{N}(o_{t+1} | \Theta_j) \beta_{t+1}(j)$ for $t = T - 1, T - 2, ..., 1$ and $1 \le i \le N$,

3. Finally $p(O | \lambda) = \sum_{i=1}^N \pi_i \mathcal{N}(o_1 | \Theta_i) \beta_1(i)$.

By definition $\xi_t(i, j)$ is the probability of being in state i at time t, and state j at time $t + 1$, given the model and the observation sequence

$$\xi_t(i,j) = p(q_t = S_i, q_{t+1} = S_j | O, \lambda) = \frac{\alpha_t(i) a_{ij} \mathcal{N}(o_{t+1} | \Theta_j) \beta_{t+1}(j)}{p(O | \lambda)} = \frac{\alpha_t(i) a_{ij} \mathcal{N}(o_{t+1} | \Theta_j) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} \mathcal{N}(o_{t+1} | \Theta_j) \beta_{t+1}(j)}.$$

By definition $\gamma_t(i)$ as the probability of being in state i at time t, given the observation sequence and the model

$$\gamma_t(i) = p(q_t = S_i | O, \lambda) = \sum_{j=1}^N \xi_t(i, j).$$

HMM maximization step

We update HMM parameters according to their expected utilization

(a) Initial state probabilities estimate $\hat{\pi}_i = \gamma_1(i)$ for $1 \le i \le N$,

(b) State-transition probabilities estimate $\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$ for $1 \le i, j \le N$,

(c) Gaussian output probabilities estimate $\hat{\mu}_j = \frac{\sum_{t=1}^{T} o_t \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}$, $\hat{\sigma}_j^2 = \frac{\sum_{t=1}^{T} (o_t - \hat{\mu}_j)(o_t - \hat{\mu}_j) \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}$ for $1 \le j \le N$.

Additional material

Additional file 1

DNA hairpin molecule toggles in the α-hemolysin nanopore vestibule.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-8-S7-S14-S1.eps]

Additional file 2

Nonzero transitions between blockade levels.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-8-S7-S14-S2.eps]

Additional file 3

Artificial duration distributions represented as continuous PDFs of Beta mixtures. By discretizing these densities we can get duration histograms for any size of aggregate states used in our experiments. Here we use the following PDFs for the first state $\mathrm{Mix}_1(x) = 0.1874 \times \mathrm{Beta}(x|3.0315, 3.0097) + 0.8126 \times \mathrm{Beta}(x|3.9944, 9.4049)$ *and* $\mathrm{Mix}_2(x) = 0.1583 \times \mathrm{Beta}(x|3.0446, 2.6063) + 0.8417 \times \mathrm{Beta}(x|8.0777, 2.8867)$ *for the second state.*
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-8-S7-S14-S3.eps]

Additional file 4

Gaussian PDFs and corresponding emissions for DHMM model [see Section The explicit duration HMM implementation*] running in generative mode. Here the maximum duration of a state is 480* μs *with 20* μs *sampling rate.*
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-8-S7-S14-S4.eps]

Additional file 5

The HMM with geometric duration distribution corresponding to the maximum state duration of 6. Discrete duration distribution histograms are put next to each state.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-8-S7-S14-S5.eps]

Additional file 6

Convolution example of three consecutive geometric distributions.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-8-S7-S14-S6.eps]

Additional file 7

Bell-shaped plots for $\mathrm{NegBin}(n, p)$ *PDF. Distributions for* $n = 1$ *follows geometric law.*
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-8-S7-S14-S7.eps]

Acknowledgements

Federal funding was provided by an NIH K-22 (SWH PI, 5K22LM008794), an NIH NNBM R-21 (SWH co-PI), and LA Board of Regents Enhancement, RCS, and LaSPACE grants (SWH PI). Funding also provided by New Orleans Childrens Hospital and the University of New Orleans Computer Science Department. The authors are grateful for many constructive suggestions made by anonymous reviewers.

This article has been published as part of *BMC Bioinformatics* Volume 8 Supplement 7, 2007: Proceedings of the Fourth Annual MCBIOS Conference. Computational Frontiers in Biomedicine. The full contents of the supplement are available online at http://www.biomedcentral.com/1471-2105/8?issue=S7.

References

1. Song L, Hobaugh M, Shustak C, Cheley S, Bayley H, Gouaux J: **Structure of Staphylococcal alpha-Hemolysin, a Heptameric Transmembrane Pore.** *Science* 1996, **274(5294):**1859-1865.
2. Bhakdi S, Tranum-Jensen J: **Alpha-toxin of Staphylococcus aureus.** *Microbiol Rev* 1991, **55(4):**733-751.
3. Walker B, Bayley H: **Key Residues for Membrane Binding, Oligomerization, and Pore Forming Activity of Staphylococcal α-Hemolysin Identified by Cysteine Scanning Mutagenesis and Targeted Chemical Modification.** *Journal of Biological Chemistry* 1995, **270(39):**23065-23071.
4. Gouaux J, Braha O, Hobaugh M, Song L, Cheley S, Shustak C, Bayley H: **Subunit Stoichiometry of Staphylococcal alpha-Hemolysin in Crystals and on Membranes: A Heptameric Transmembrane Pore.** *PNAS* 1994, **91:**12828-12831.
5. Kasianowicz J, Brandindagger E, Brantondagger D, Deamer D: **Characterization of individual polynucleotide molecules using a membrane channel.** *PNAS* 1996, **93:**13770-13773.
6. Akeson M, Branton D, Kasianowicz J, Brandin E, Deamer D: **Microsecond time-scale discrimination among polycytidylic acid, polyadenylic acid, and polyuridylic acid as homopolymers or as segments within single RNA molecules.** *Biophysical Journal* 1999, **77:**3227-3233.
7. Mathé J, Visram H, Viasnoff V, Rabin Y, Meller A: **Nanopore Unzipping of Individual DNA Hairpin Molecules.** *Biophysical Journal* 2004, **87:**3205-3212.
8. Vercoutere W, Winters-Hilt S, Olsen H, Deamer D, Haussler D, Akeson M: **Rapid discrimination among individual DNA hairpin molecules at single-nucleotide resolution using an ion channel.** *Nature Biotechnology* 2001, **19:**248-252.
9. Vercoutere W, Winters-Hilt S, DeGuzman V, Deamer D, Ridino S, Rodgers J, Olsen H, Marziali A, Akeson M: **Discrimination among individual Watson-Crick base pairs at the termini of single DNA hairpin molecules.** *Nucleic Acids Research* 2003, **31(4):**1311-1318.
10. Akeson M, Branton D, Kasianowicz J, Brandin E, Deamer D: **Microsecond time-scale discrimination among polycytidylic acid, polyadenylic acid, and polyuridylic acid as homopolymers or as segments within single RNA molecules.** *Biophysical Journal* 1999, **77(6):**3227-3233.
11. Kasianowicz J, Brandin E, Branton D, Deamer D: **Characterization of individual polynucleotide molecules using a membrane channel.** *PNAS* 1996, **93(24):**13770-13773.
12. Mitchell C, Helzerman R, Jamieson L, Harper M: **A Parallel Implementation of a Hidden Markov Model with Duration Modeling for Speech Recognition.** *Digital Signal Processing, A Review Journal* 1995, **5:**298-306 [http://citeseer.ist.psu.edu/mitchell95parallel.html].
13. Rabiner L: **A tutorial on Hidden Markov Models and Selected Applications in Speach Recognition.** *Proceedings of IEEE* 1989, **77:**257-286.
14. Majoros W, Pertea M, Delcher A, Salzberg S: **Efficient decoding algorithms for generalized hidden Markov model gene finders.** *BMC Bioinformatics* 2005, **6(16):**.
15. Winters-Hilt S: **Hidden Markov Model Variants and their Application.** *BMC Bioinformatics* 2006, **7(Suppl 2):**S14.
16. Winters-Hilt S, Landry M, Akeson M, Tanase M, Amin I, Coombs A, Morales E, Millet J, Baribault C, Sendamangalam S: **Cheminformatics Methods for Novel Nanopore analysis of HIV DNA termini.** *BMC Bioinformatics* 2006, **7(Suppl 2):**S22.
17. DeGuzman V, Lee C, Deamer D, Vercoutere W: **Sequence-dependent gating of an ion channel by DNA hairpin molecules.** *Nucleic Acids Research* 2006, **34(22):**6425-6437.

18. Winters-Hilt S, Vercoutere W, DeGuzman V, Deamer D, Akeson M, Haussler D: **Highly Accurate Classification of Watson-Crick Basepairs on Termini of Single DNA Molecules.** *Biophysical Journal* 2003, **84:**967-976.
19. Durbin R, Eddy S, Krogh A, Mitchison G: *Biological sequence analysis Volume chap 3. Cambridge University press*; 1998:69.
20. Mitchell C, Helzerman R, Jamieson L, Harper M: **A Parallel Implementation of a Hidden Markov Model with Duration Modeling for Speech Recognition.** *Digital Signal Processing, A Review Journal* 1995, **5:**298-306 [http://citeseer.ist.psu.edu/mitchell95parallel.html].
21. Miklós I, Meyer I: **A linear memory algorithm for Baum-Welch training.** *BMC Bioinformatics* 2005, **6:**231.
22. Bilmes J: **A Gentle Tutorial of the EM algorithm and its application to parameter Estimation for Gaussian mixture and Hidden Markov Models.** In *Tech Rep TR-97-021 International Computer Science Institute*; 1998.
23. Bilmes J: **What HMMs can do.** In *Tech rep University of Washington, Seattle*; 2002.
24. Rabiner L, Juang BH: *Fundamentals of speech recognition Printice Hall*; 1993.
25. Durbin R, Eddy S, Krogh A, Mitchison G: *Biological sequence analysis Cambridge University press*; 1998.