MODELING AND BASE-CALLING FOR DNA SEQUENCING-BY-SYNTHESIS

Helmy Eltoukhy and Abbas El Gamal

Department of Electrical Engineering,
Stanford University, Stanford, CA 94305

ABSTRACT

The process of DNA sequencing-by-synthesis and its non-idealities are modeled as a noisy switched linear system parameterized by the unknown DNA sequence. The base-calling problem is then formulated as a parameter detection problem. As this system can have long memory, performing exact maximum-likelihood decoding is computationally prohibitive. An approximate ML method applied to experimental Pyrosequencing data demonstrates reliable read lengths exceeding 200 bases, which is significantly longer than that achieved by current methods.

I. INTRODUCTION

Sequencing-by-synthesis methods perform DNA sequencing by building up the complement of a single-stranded template DNA base-by-base. Thus this process converts a single-stranded molecule into its double-stranded counterpart. A prime example of such methods is Pyrosequencing [1], which offers the promise of high throughput and low cost sequencing via efficient mass parallelization and relative simplicity. However, DNA read lengths obtained using commercial Pyrosequencing machines remain below 30-40 bases, which are too short for such applications as whole genome de novo sequencing via shot-gun assembly. Moreover, in contrast to the tremendous and very fruitful efforts devoted to developing sophisticated base-calling techniques for Sanger sequencing, e.g., [2], there has been little such work to date for sequencing-by-synthesis.

In this paper we model the sequencing-by-synthesis process and its non-idealities as a noisy switched linear system parameterized by the unknown DNA sequence, where the switching is performed by the input test sequence. The base-calling problem is then formulated as a parameter detection problem: Given a test sequence and its corresponding noisy output sequence, determine the system parameters, i.e., the DNA sequence that minimizes the probability of decoding error. As this system can have long memory, performing exact maximum-likelihood (ML) decoding is computationally prohibitive. Applying existing approximate ML (AML) methods to experimental Pyrosequencing data, we demonstrate close to an order of magnitude increase in read length. We also provide bounds on the probability of correct decoding.

In the next section we describe the Pyrosequencing process and its non-idealities. In Section III, we develop the model specifically for Pyrosequenicng, though it should apply to other sequencing-by-synthesis methods. In Section IV, we briefly describe the AML method used. In Section V, we provide experimental results.

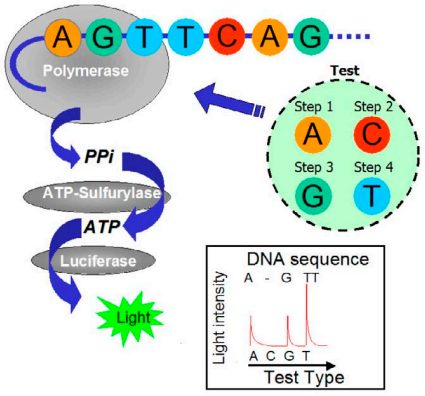

Fig. 1. DNA template undergoing Pyrosequencing reaction. Using testing order A, C, G, T, we obtain a signal proportional to 1A-0C-1G-2T (shown in inset box), and infer that the first four bases of the template sequence are AGTT.

II. PYROSEQUENCING

In Pyrosequencing, a DNA template is sequenced by repeatedly cycling through the tests for the 4 base types, A, C, G, and T. When a test is successful, the base that is added as part of the test procedure is *incorporated* at the current position on the DNA template, producing a light signal proportional to the length of the homopolymeric region, i.e., stretch of identical bases. Figure 1 illustrates this process for the DNA template AGTTCAG.

The above description details the ideal outcome of the Pyrosequencing chemistry, whereby the length of the DNA template to be sequenced can be arbitrarily long. In practice, there are several non-idealities that limit realizable read lengths [3], including:

Incomplete Incorporation: To obtain high enough signal levels, multiple copies of the template must be simultaneously sequenced. Since the incorporation reaction is stochastic in nature, incomplete incorporation occurs, i.e., some copies do not incorporate the added base when they should. This leads to *desynchrony* in the read position of different strands. Analogously we can view the process of sequencing multiple copies of the DNA template as reading multiple identical tapes using tape heads with slightly varying read rates and observing only the sum of their signals.

Misincorporation: Misincorporation occurs when in some strands the wrong base incorporates. This again leads to de-

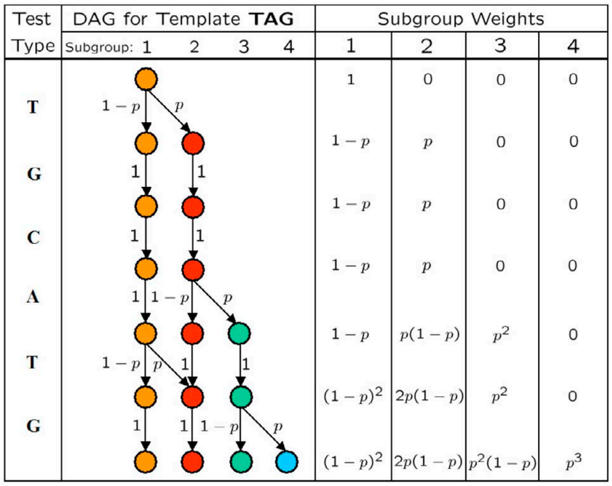

Fig. 2. Directed acyclic graph showing the evolution of the DNA template TAG subgroups in response to the tests. The corresponding fraction of the initial strand population at each node is highlighted in the table and are denoted as subgroup weights.

synchrony in the read position. This phenomenon, however, is less likely to occur than incomplete incorporation.

Read noise: Read noise occurs as a result of the thermal and shot noise of the imaging system as well as test-to-test variations in the reagent handling and delivery apparatus.

III. PYROSEQUENCING MODEL

For clarity of presentation, we first develop the model assuming only incomplete incorporation. We then show how read noise and misincorporation can be included. We ignore the stochastic nature of the incorporation rate, and only consider its mean value $p \in (0, 1)$, which we define as the average fraction of template strands that incorporate the added base, when incorporation should ideally occur.

Incomplete incorporation gives rise to a population of template strands with subgroups classified by a representative lag from the ideal subgroup. The process of subgroup formation and evolution can be represented by a weighted directed acyclic graph as illustrated in Figure 2. The nodes, which represent the subgroups, are organized in levels each corresponding to a given test. When incorporation occurs, a fraction p of the strands in a subgroup advance to the succeeding subgroup, while a fraction $1-p$ remains. When no incorporation occurs, the subgroup position remains unchanged, i.e., does not advance to the right, and a weight of 1 is assigned to the corresponding edge. From Figure 2, it is clear that the initial population of template strands becomes distributed over many different subgroups, with laggard subgroups contributing to the overall signal at each test thereby distorting signal quality at longer reads.

Before describing our model, we introduce some needed definitions. Consider a DNA strand with M homopolymeric regions. Since each region can assume one of four possible base types, we represent it by a a 4-tuple $\mathbf{s}_n$ whose entries correspond to the number of A, C, G, and T bases in the region, respectively. Thus each vector has a single nonzero integer valued entry. Denote by S, the $4 \times (M+1)$ *DNA sequence matrix* whose first M columns are the $\mathbf{s}_n$ vectors and the last column $\mathbf{s}_{M+1} = \mathbf{0}$. For example, for the oligonucleotide TAGCGG,

$$S = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 2 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Recall that each test aims at detecting the presence of a specific base at the current position in the template. Accordingly, we represent each test by a binary 4-tuple $\mathbf{t}_n$, $1 \le n \le N$, whose entries correspond to the type of test performed. So, for example, $[1\,0\,0\,0]^T$ corresponds to a test for base A, $[0\,1\,0\,0]^T$ corresponds to C, etc. Denote by T, the $4 \times N$ *DNA test sequence matrix* whose columns are the $\mathbf{t}_n$ vectors.

Finally, for each test $1 \le n \le N$, we define the column *weight vector* $\mathbf{w}_n$ of length $M+1$ to consist of the fractions of the total template population in each possible subgroup after the nth test, beginning with $\mathbf{w}_0 = [1\;0\;0\cdots0]^T$. Thus, by definition, for each n, $\mathbf{w}_{n,i} \ge 0$, for all i, and $\sum_i w_{n,i} = 1$. For the example DNA template sequence in Figure 2, the weight vectors are the rows of subgroup weights, i.e., $\mathbf{w}_1 = [1-p\;p\;0\;0]^T, \ldots, \mathbf{w}_4 = [1-p\;p(1-p)\;p^2\;0]^T$.

From the above discussion, it follows that the ith component of the weight vector at time n can be expressed as

$$w_{n,i} = a_{i-1,n}\, w_{n-1,i-1} + (1 - a_{i,n})\, w_{n-1,i},$$

where $a_{i,j} = p\,\mathrm{I}((S^T\mathbf{t}_j)_i)$, $(S^T\mathbf{t}_j)_i$ refers to the ith element in the vector $S^T\mathbf{t}_j$, and $\mathrm{I}(a) = 1$ if $a > 0$ and zero, otherwise. Thus, the evolution of the weight vector can be described by the switched linear relation

$$\mathbf{w}_n = A_{\mathbf{t}_n}\mathbf{w}_{n-1}, \text{ for } 1 \le n \le N, \tag{1}$$

where the state-transition matrix

$$A_{\mathbf{t}_n} = \begin{bmatrix} 1-a_{1,n} & 0 & \ldots & 0 \\ a_{1,n} & 1-a_{2,n} & \ldots & 0 \\ 0 & a_{2,n} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1-a_{M+1,n} \end{bmatrix}.$$

Hence, each test vector $\mathbf{t}_n$ selects (or switches to) the appropriate state-transition matrix $A_{\mathbf{t}_n}$ for the current test.

Define the noiseless output x_n to be the sum of the contributions from each subgroup of the template population that “tests positive” in response to test $\mathbf{t}_n$, i.e.,

$$x_n = p\,[S\mathbf{w}_{n-1}]^T\mathbf{t}_n. \tag{2}$$

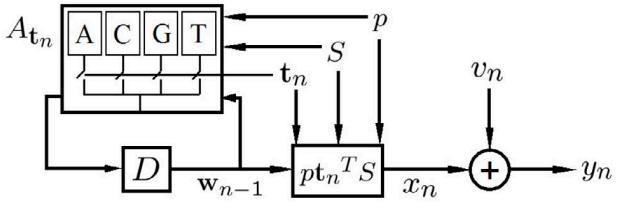

Fig. 3. Block diagram of the noisy switched linear model for Pyrosequencing or, more generally, sequencing-by-synthesis.

Each term in the inner product of $S^T\mathbf{t}_n$ and $\mathbf{w}_{n-1}$ represents a contribution from a subgroup.

To include read noise, we assume it is additive White Gaussian Noise (WGN) that is independent of the signal and system parameters. With noise added, the output of the system is given by

$$y_n = x_n + v_n, \text{ for } 1 \le n \le N,$$

where v_n, $1 \le n \le N$ are independent, identically distributed $\mathcal{N}(0, \sigma^2)$ random variables. Figure 3 shows the overall system model, where the input $\mathbf{t}_n$ dictates the evolution of the system with parameters S and p.

Misincorporation, if non-negligibly present, can be modeled by introducing a misincorporation rate, $0 \le \epsilon \le 1$ akin to the incorporation rate, p. Depending on the outcome of each test, subgroups either advance to the succeeding subgroup with weight p upon success or with weight ϵ upon failure. Since ϵ is typically small ($< .02$), as a good approximation and to simplify the analysis, we only consider one subgroup advancement per test.

Explicitly, to include the effect of misincorporation, we replace $a_{i,j}$ in the definition of $A_{\mathbf{t}_n}$ by $\tilde{a}_{i,j} = I_p((S^T\mathbf{t}_j)_i)$, where $I_p(s) = p$ if $s > 0$ and ϵ, otherwise.

IV. APPROXIMATE ML BASE-CALLING

From the results of the previous section, it is clear that given a DNA sequence matrix S, a test matrix, and p, we can solve for the expected value of the output signal $E(y_n) = x_n$, at each time n. However, it is the reverse procedure, i.e., *base-calling*, that is of practical interest to genomics: Given T, p, and $y_1, y_2, \ldots, y_N$, estimate the sequence matrix S? Casting this problem as a parameter detection problem we can use maximum-likelihood detection (MLD) to obtain

$$\begin{aligned} S^* &= \underset{S}{\text{argmax}}\, f(y_1, \ldots, y_N | S, T, p) \\ &= \underset{S}{\text{argmin}} \sum_{n=1}^{N} (y_n - p\,[S\mathbf{w}_{n-1}]^T\mathbf{t}_n)^2. \end{aligned}$$

Thus, we only need to minimize the l_2 distance between the observed sequence and the chosen sequence over the search space of all possible DNA sequences. The computational complexity of this approach is daunting, since even if we limit the length of homopolymeric region to no more than K bases, we must search through $\frac{4}{3}(3K)^M$ possible sequences.

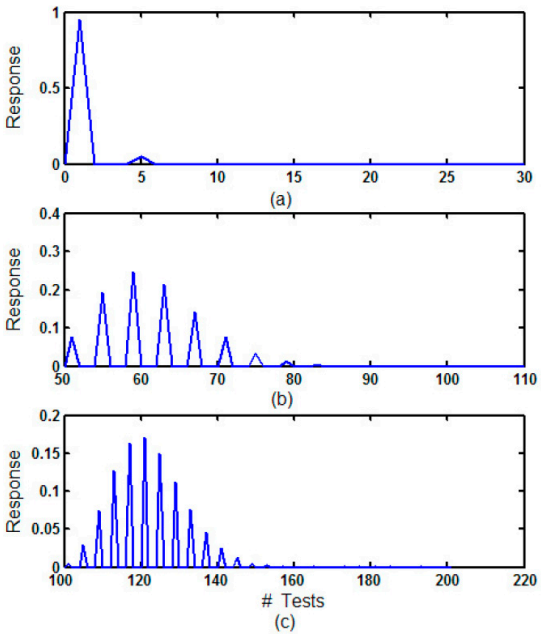

Fig. 4. Evolution of the pulse response for $p = 0.95$ and $\epsilon = 0$. Initially, (a) ISI is negligible for the first base, but becomes more severe after (b) 50 tests and (c) 100 tests.

Clearly, we need to take better advantage of the problem structure to reduce the computational complexity.

Communication Channel Analogy: To gain better understanding of the special structure of the model, we view each homopolymeric region as producing a particular "channel" pulse in response to a given set of tests over time. Accordingly, we decompose each x_n into the sum of the shifted pulse responses due to each individual homopolymeric region as

$$[x_1\, x_2\, \cdots\, x_N]^T = H_T \cdot [111 \cdots 111]^T,$$

where

$$H_T = p \cdot \begin{bmatrix} b_{1,1} & 0 & \cdots & 0 \\ b_{2,1} & b_{2,2} & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ b_{N,1} & b_{N,2} & \cdots & b_{N,M} \end{bmatrix},$$

and

$$b_{i,j} = w_{i-1,j}(S^T\mathbf{t}_i)_j.$$

Each column of H_T represents the contribution from an individual homopolymeric region to the resultant $x_1, x_2, \ldots, x_N$. Figure 4 plots the pulse response as a function of the number of tests. Note that as long as $p < 1$, each homopolymeric region contributes to succeeding outputs even for very large N, although possibly negligibly, or in communication theory parlance, the channel exhibits severe intersymbol-interference (ISI).

Iterative Partial-MLSD: The above discussion suggests two special features of the model that can be exploited to

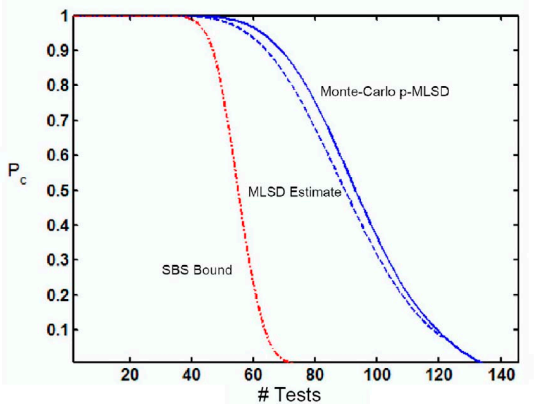

Fig. 5. Plot of MLSD P_c estimate versus partial-MLSD Monte Carlo simulation and lower bound for symbol-by-symbol detection for $p = 0.95$ and $\epsilon = 0$.

reduce the computational complexity of base-calling. The first is that only a limited portion of the pulse response is significant. The second, and perhaps more important, is that tests of different base types are reasonably uncorrelated for p close to one. Taking these properties into consideration, we appropriately modify the soft-input Viterbi algorithm (VA) [4] to perform base-calling and, hence, approximate ML sequence detection (MLSD). We use 4 trellises (each one with states corresponding to only one test type) and perform symbol-by-symbol detection to obtain a rough estimate of the sequence to initialize the algorithm. We then perform the standard VA on each of the four trellises, with the best surviving paths from the other three trellises filling in the gaps. We extend the paths one path length at a time and iterate through the trellises based on the order specified by the test matrix T. In this manner, we approximate MLSD using an iterative, partial-MLSD approach. Note that to generate a confidence score for each base, as is commonly done in Sanger sequencing [2], a soft-output VA (SOVA) [5] or List VA (LVA) [6] can be used.

The above communication analogy can also be exploited to derive bounds and estimates on the probability of correct decoding (P_c). Figure 5 compares the lower bound on P_c obtained assuming symbol-by-symbol (SBS) detection (i.e., with no lookahead) and a conservative estimate assuming MLSD with worst case ISI to P_c obtained using a Monte-Carlo simulation of the partial-MLSD algorithm. As can be seen, the performance of the partial-MLSD algorithm is on par with the MLSD estimate.

V. EXPERIMENTAL RESULTS

Before applying the iterative partial-MLSD algorithm to experimental Pyrosequencing data, several preprocessing steps are needed. A baseline correction must be performed due to the changing chemical background signal present as well as integration of the total photoemission from each test. Normalization of integrated signals also needs to be

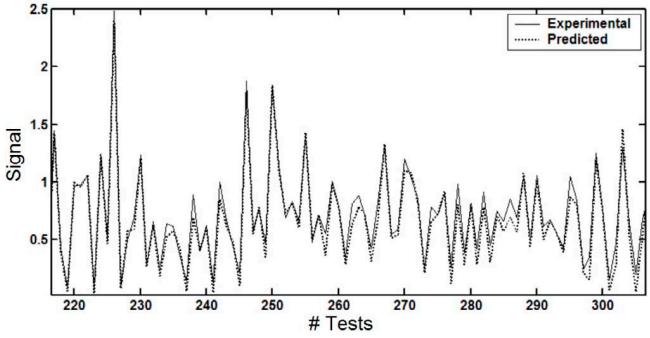

Fig. 6. Plot of predicted vs. experimental data (for $p = 0.995$ and $\epsilon = 0.018$).

performed to account for chemical variations from run to run. The scaling factor required for such normalization is automatically extracted from the histogram of the photoemission values. Additionally, we need to estimate the parameters of our model, e.g., p, base specific gains, etc. These parameters are similarly extracted from a subset of the data itself through an iterative fitting procedure.

We applied the iterative partial-MLSD algorithm to 5 Pyrosequencing datasets ranging in length between 55 and 224 bases, correctly decoding the first 170 out of 208 and 205 out of 224 bases of the longest two templates as well as correctly decoding all shorter sequences. Figure 6 highlights the model fit for the dataset of the longest template. This demonstrates a clear improvement over the 30-40 base read lengths reliably sequenced by commercially available Pyrosequencing machines. This suggests using existing Pyrosequencing systems in conjunction with our proposed base-calling algorithm, that read lengths well over 200-300 bases are attainable.

VI. REFERENCES

[1] M. Ronaghi, S. Karamohamed, B. Pettersson, M. Uhlen and P. Nyren, "Real-Time DNA Sequencing using Detection of Pyrophosphate Release," *Analytical Biochemistry*, vol. 242, pp. 84-89, 1996.

[2] B. Ewing, P. Green, "Basecalling of automated sequencer traces using phred. II. Error probabilities," *Genome Research* 8, pp. 186-194, 1998.

[3] B. Gharizadeh, T. Nordstrom, A. Ahmadian, M. Ronaghi, and P. Nyren, "Long read Pyrosequencing using pure 2'-deoxyadenosine-5'-0'-(1-thiotriphosphate) Sp-isomer," *Analytical Biochemistry*, vol. 301, pp. 82-90, 2002.

[4] G. D. Forney, "The Viterbi algorithm," *Proc. IEEE*, vol. 61, N. 3, pp. 268-278, Mar. 1973.

[5] J. Hagenauer, P. Hoeher, "A Viterbi Algorithm With Soft Decision Outputs and Its Applications," *Proceedings of IEEE GLOBECOM*, pp. 47.1.1-47.1.6., November 1989.

[6] N. Seshadri, C-E.W. Sundberg, "List Viterbi Decoding Algorithms with Applications," *IEEE Transactions on Communications*, vol. 42, no. 2/3/4, pp. 313-323, 1994.