Accuracy Assessment of Diploid Consensus Sequences

Jong Hyun Kim, Michael S. Waterman, and Lei M. Li

Abstract—If the origins of fragments are known in genome sequencing projects, it is straightforward to reconstruct diploid consensus sequences. In reality, however, this is not true. Although there are proposed methods to reconstruct haplotypes from genome sequencing projects, an accuracy assessment is required to evaluate the confidence of the estimated diploid consensus sequences. In this paper, we define the confidence score of diploid consensus sequences. It requires the calculation of the likelihood of an assembly. To calculate the likelihood, we propose a linear time algorithm with respect to the number of polymorphic sites. The likelihood calculation and confidence score are used for further improvements of haplotype estimation in two directions. One direction is that low-scored phases are disconnected. The other direction is that, instead of using nominal frequency 1/2, the haplotype frequency is estimated to reflect the actual contribution of each haplotype. Our method was evaluated on the simulated data whose polymorphism rate (1.2 percent) was based on *Ciona intestinalis*. As a result, the high accuracy of our algorithm was indicated: The true positive rate of the haplotype estimation was greater than 97 percent.

Index Terms—Haplotype, polymorphism, shotgun sequencing, diploid.

1 Introduction

In shotgun sequencing strategies, random fragments are generated from whole genomes or clones and then sequenced using Sanger four-dye dideoxy technique. In shotgun sequencing projects, the sequence coverage is usually between 5 and 10. This redundancy improves the quality of the reconstructed genome. The sequence assembly in shotgun sequencing usually follows a three-step procedure: overlap-layout-consensus. Even though the objective of shotgun sequencing has been to determine a haploid consensus sequence, fragments are from the diploid genome in a eukaryotic organism. Because the origins of fragments are unknown, the reconstruction of diploid consensus sequences (or, interchangeably, the reconstruction of haplotypes) in the consensus determination step is a challenging problem. It is necessary to differentiate polymorphisms from sequencing errors and then to infer the phases between adjacent polymorphisms. Lancia et al. [7] formulated the diploid shotgun sequencing problem by a graph theoretic approach. To deal with error seen in fragments, they defined several combinatorial problems such as Minimum Fragment Removal (MFR), Minimum Snip Removal (MSR), and Longest Haplotype Reconstruction (LHR). Further development along this direction can be found in Lippert et al. [11]. Li et al. [9] proposed a method based on a probabilistic model. In this work, the probabilities of different haplotypes (conditional on the assembly layout) were calculated. It is a computationally intensive problem to find the most likely haplotype because of the large number of possible haplotype configurations; our strategy was to consider one pair of polymorphic sites at a time and then combine the phase information between adjacent pairs to construct haplotype segments.

In this haplotype reconstruction problem, a complete solution should include an assessment of the accuracy about estimated haplotypes. To assess the accuracy of a haploid consensus sequence, it is sufficient to provide the quality measure of each consensus base. Churchill and Waterman [4] introduced a statistical method to address this problem for the case that fragments are from one target haploid chromosome. The sequence assembler, *Phrap* (http://www.phrap.org), calculates the quality values of consensus bases. However, when diploid consensus sequences are determined, it is not sufficient to provide the quality measure of each polymorphism. For a complete assessment, the phase between polymorphisms should be considered and the phase information needs to be incorporated into the accuracy assessment. But, the calculation of likelihood of an assembly is required to incorporate the phase information into the accuracy assessment of haplotype estimation. In our previous work in [9], a realistic solution to calculating the likelihood was not feasible because the number of haplotype configurations grew exponentially with respect to the number of polymorphic sites. In this paper, we present, by using a Markov structure, a fast algorithm of linear complexity with respect to the number of polymorphic sites to calculate the

- *J.H. Kim is with the Department of Computer Science and the Molecular and Computational Biology Program, Department of Biological Sciences, University of Southern California, Los Angeles, CA 90089. E-mail: jonghkim@usc.edu.*
- *M.S. Waterman is with the Molecular and Computational Biology Program, Department of Biological Sciences, University of Southern California, Los Angeles, CA 90089, and with Informatics Research, Celera Genomics, 45 West Gude Drive, Rockville, MD 20850. E-mail: msw@usc.edu.*
- *L.M. Li is with the Molecular and Computational Biology Program, Department of Biological Sciences, and the Department of Mathematics, University of Southern California, Los Angeles, CA 90089. E-mail: lilei@usc.edu.*

Manuscript received 2 July 2005; revised 17 Dec. 2005; accepted 6 Mar. 2006; published online 9 Jan. 2007.
For information on obtaining reprints of this article, please send e-mail to: tcbb@computer.org, and reference IEEECS Log Number TCBB-0072-0705.
Digital Object Indentifier No. 10.1109/TCBB.2007.1007.

likelihood of an assembly. We consider two approaches to sequencing error rates: the quality scores, if they are available, and adaptively estimated overall sequencing error rates. Based on this likelihood calculation, we define a confidence score for estimated haplotypes.

Furthermore, we use the confidence score and the likelihood calculation to improve haplotype estimation. The haplotypes with a confidence score below a specified threshold are broken to increase the accuracy. The likelihood calculation is used to estimate the sample fraction of each haplotype in the fragments. Hereafter, we refer to this fraction as haplotype frequency. If the fragments are from one individual, the nominal haplotype frequency is one-half. However, due to factors such as homozygosity, duplication, amplification bias, and misalignment, deviations may exist to some extent. A good estimate of the real haplotype frequency may help us detect abnormalities. We estimate haplotype frequency by maximizing likelihood.

In shotgun sequencing, particularly whole-genome shotgun sequencing, fragments of different sizes are prepared to provide different levels of continuity, see [1], [13], [2], [5]. The fragments are subcloned into plasmid, cosmid, or bacterial artificial chromosome (BAC) according to the size of fragments and then two-end sequenced. Our perspective is that the two-end sequencing strategy also provides long-range continuity in haplotype estimation because the two-end sequenced reads of each fragment cover more polymorphic sites. In this paper, we exploit the information from two-end sequencing to estimate longer haplotypes. In our simulation study, the polymorphism rate was based on the reported polymorphism rate (1.2 percent) of *Ciona intestinalis* in [5]; the high polymorphism rate, which is higher than other organisms such as *Fugu rubripes* 0.4 percent) and *Homo sapiens* (0.1 percent), was appropriate for evaluating the accuracy of our method, see [2], [13]. Two-end sequencing of clone libraries (plasmid, cosmid, BAC) was simulated; the fragments of variable size (1.8K bp ~ 120K bp) were generated complying with their proportions in the *Ciona intestinalis* sequencing project, see [5].

This paper is organized as follows: In Section 2, we describe our model, a Markov structure that allows us to compute the likelihood of an assembly efficiently. Then, we consider the problem of inference. And, finally, we present the algorithm of reconstruction haplotype. In Section 3, we present results based on *Ciona intestinalis* and discuss some related issues. In Section 4, we give the technical proof.

2 Methods

2.1 The Probabilistic Model

Consider n potential polymorphic sites and m fragments. Following the common practice in probability, we use uppercase letters to represent random variables and lowercase letters to represent their values. We denote two target chromosomes by $\mathbf{S} = \{S_{k1}S_{k2}\dots S_{kn}, k = 1, 2\}$. Each letter takes values from the alphabet $\mathcal{A} = \{\texttt{A}, \texttt{C}, \texttt{G}, \texttt{T}, -, \texttt{M}\}$, where $-$ denotes an internal gap, and M denotes any sequence of two or more nucleotide bases. For simplicity, we assume that the genotypes are independently and identically sampled from the composition probabilities:

Chromosome	1	A	G	C	C	M	A	G	A	T	T	C
Origin	1	A	G	A	C	M	A	G	A	ϕ	ϕ	ϕ
	2	C	C	T	A	$-$	G	C	T	A	ϕ	ϕ
	1	ϕ	G	C	C	M	A	G	A	T	T	ϕ
	2	C	C	T	A	$-$	T	C	T	A	G	T
	2	ϕ	C	T	A	$-$	G	C	T	$-$	G	T
	1	ϕ	ϕ	ϕ	C	M	A	G	A	C	T	C
Chromosome	2	C	C	T	A	$-$	G	C	T	A	G	T

Fig. 1. An illustrative example of the problem. The two target chromosomes are shown at the top and bottom, respectively. The fifth polymorphic site, "M," in this case, represents "CCC." Six fragments are aligned in the middle. In reality, the targets and origins of fragments are not observed.

$$\mu(a, b) = \Pr(S_{1,j} = a,\ S_{2,j} = b) = \Pr(S_{1,j} = b,\ S_{2,j} = a),$$
$$a,\ b \in \mathcal{A}.$$

The origins of the fragments are denoted by $\mathbf{F} = \{F_i,\ i = 1, \dots, m\}$ and they appear according to Bernoulli trials:

$$F_i = \begin{cases} 1 & \text{with prob } \lambda_1 = \lambda, \\ 2 & \text{with prob } \lambda_2 = 1 - \lambda. \end{cases}$$

We denote the true bases of the assembly matrix by $\mathbf{Y} = \{Y_{ij}, i = 1, \dots, m, j = 1, \dots, n\}$ and they relate to the target haplotype by $Y_{ij} = S_{F_i, j},\ \ i = 1, 2$. The observations $\mathbf{X} = \{X_{ij}, i = 1, \dots, m, j = 1, \dots, n\}$ are the measurement of $\mathbf{Y}$ via the following random error model:

$$\eta_{ij}(b|a) = \Pr(X_{ij} = b | Y_{ij} = a), a \in \mathcal{A}, b \in \mathcal{B},$$

where $\mathcal{B} = \{\texttt{A}, \texttt{C}, \texttt{G}, \texttt{T}, -, \texttt{M}, \phi\}$, the null symbol ϕ denotes any ambiguous determination of a base or positions beyond the ends of a fragment. The errors can be categorized as single-nucleotide replacement, single-nucleotide insertion, deletion, and errors involving multiple nucleotides. Sometimes, we drop the subscript of η when no confusion is incurred. We have assumed that measurement errors occur independently with identical distribution across the assembly because the notation is complicated without the assumption. The random fragments are generated either in the direct or reversed orientation. To deal with the issue, we introduce the complementary letters as follows: $\tilde{\texttt{A}} = \texttt{T}$, $\tilde{\texttt{T}} = \texttt{A}$, $\tilde{\texttt{G}} = \texttt{C}$, $\tilde{\texttt{C}} = \texttt{G}$, $\tilde{\texttt{M}} = \texttt{M}$, $\tilde{\phi} = \phi$, and $\tilde{-} = -$. In Fig. 1, we illustrate the data structure by a hypothetical example. For the sake of notational simplicity, we skip the issue of orientation and nonpolymorphic sites. The two target chromosomes are shown at the top and bottom, respectively. Six fragments are aligned in the middle.

In reality, only the assembly matrix $\{x_{ij}\}$ is observed while the information of $\mathbf{S}$ and $\mathbf{F}$ is missing. Thus, we need to estimate $\mathbf{S}$ and $\mathbf{F}$ based on the observations. Technically, the estimation of $\mathbf{S}$ can be based on its conditional distribution given data $\Pr(\mathbf{S}|\mathbf{X})$. According to the Bayes' rule, we have

$$\Pr(\mathbf{S}|\mathbf{X}) = \frac{\Pr(\mathbf{X}, \mathbf{S})}{\Pr(\mathbf{X})}, \tag{1}$$

where $\Pr(\mathbf{X}) = \sum_{\mathbf{S}} \Pr(\mathbf{X}, \mathbf{S})$. The formula to compute $\Pr(\mathbf{X}, \mathbf{S})$ is given by

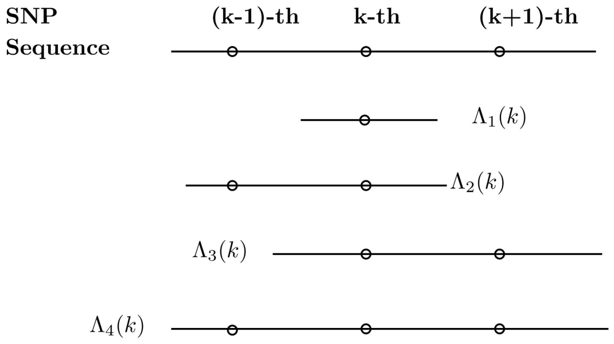


Fig. 2. Definition of four index sets. $\Gamma(k) = \Lambda_3(k) \bigcup \Lambda_4(k) = \Lambda_2(k+1) \bigcup \Lambda_4(k+1)$.

$$\begin{aligned} \Pr(\mathbf{X}, \mathbf{S}) &= \Pr(\mathbf{S})\Pr(\mathbf{X}|\mathbf{S}) \\ &= \left[\prod_{i=1}^{m} \Pr(S_{1j},\ S_{2j})\right]\left[\prod_{i=1}^{m}\sum_{k=1}^{2}\lambda_k \prod_{j=1}^{n}\Pr(X_{ij}|S_{kj})\right]. \end{aligned} \tag{2}$$

We define the most probable haplotypes by: $\max_{\mathbf{S}}\ \Pr(\mathbf{S}|\mathbf{X})$. It is possible that we cannot determine all of the phase information because the coverage and origins of fragments are not strictly uniform across the entire clone. Thus, we look for relatively shorter haplotype segments that exceed some level of confidence. Due to the computational complexity, Li et al. [9] proposed a pairwise strategy to find the most probable haplotype configuration. The calculation of the confidence score for a given haplotype configuration is also based on (1) and (2). However, the marginal probability, $\Pr(\mathbf{X})$, is the sum of joint probabilities over all haplotypes. The complexity of a straightforward algorithm is $O(5^{2n})$. Next, we develop an algorithm of linear complexity with respect to the number of polymorphic sites.

2.2 A Markov Structure

We start off with one locus and then move along the chromosome recursively. Suppose we have dealt with $k-1$ loci and are considering the kth locus. We notice that only fragments that cover the position are relevant. Denote the index set of those fragments covering the kth locus by $\Omega(k)$. We note that only these fragments are relevant for the calculation. Let $\Theta(k) = \bigcup_{j=1}^{k}\Omega(j)$. We decompose $\Omega(k)$ into four subsets: $\Lambda_1(k)$ includes those fragments covering the kth locus but neither the $(k-1)$th nor the $(k+1)$th; $\Lambda_2(k)$ includes those fragments covering both the $(k-1)$th and kth locus but not the $(k+1)$th; $\Lambda_3(k)$ includes those fragments covering both the kth and $(k+1)$th locus but not the $(k-1)$th; $\Lambda_4(k)$ includes those fragments covering the $(k-1)$th, kth and $(k+1)$th locus. We write $\Gamma(k) = \Lambda_3(k)\bigcup\Lambda_4(k)$. An illustration of the definition is shown in Fig. 2. It is easy to check that $\Gamma(k) = \Lambda_2(k+1)\bigcup\Lambda_4(k+1)$ and

$$\Theta(k+1) = \Theta(k)\bigcup\Lambda_1(k+1)\bigcup\Lambda_3(k+1).$$

Fig. 3 shows how the index sets evolve as the calculation moves along a clone. If we compute likelihood iteratively along the chromosome, then we need the dependence structure of $\Omega(k+1)$ on $\Theta(k)$. According to the definition of $\Gamma(k)$, we have the following:

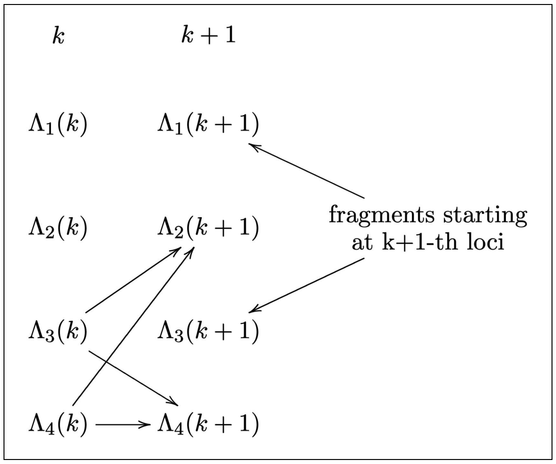


Fig. 3. An illustration of the recursive structure of the index sets. Formula (3) uses this structure.

Proposition 1. $\{S_{k,1} = a_1, S_{k,2} = a_2, F_i = f_i,\ i \in \Gamma(k)\}$ *is a Markov chain.*

It is interesting to see that the dimension of this state vector varies across loci. We define

$$\begin{aligned} &\alpha_k(a_1, a_2; f_i,\ i \in \Gamma(k)) = \Pr(\mathbf{X}_{ij} = x_{ij},\ j = 1, \ldots, k, \\ &i = 1, \ldots, m; S_{k,1} = a_1, S_{k,2} = a_2; F_i = f_i,\ i \in \Gamma(k)). \end{aligned}$$

Based on the above Markov structure, we can recursively compute $\alpha_k(a_1, a_2; f_i,\ i \in \Gamma(k))$ as follows:

Theorem 1.

$$\begin{aligned} &\alpha_{k+1}(a_1, a_2; f_i,\ i \in \Gamma(k+1)) = \mu(a_1, a_2)\left[\prod_{h=1}^{2}\lambda_h^{\sum_{j\in\Lambda_3(k+1)}\mathbf{1}(F_j=h)}\right] \\ &\left[\prod_{j\in\Lambda_1(k+1)}[\lambda_1\ \eta(x_{j,k+1}|a_1) + \lambda_2\ \eta(x_{j,k+1}|a_2)]\right] \\ &\left[\prod_{j\in\Lambda_3(k+1)}\eta(x_{j,k+1}|a_{f_j})\right]\left[\prod_{j\in\Lambda_4(k+1)}\eta(x_{j,k+1}|a_{f_j})\right] \\ &\left[\sum_{f_j=1,2,\ j\in\Lambda_2(k+1)}\left[\prod_{j\in\Lambda_2(k+1)}\eta(x_{j,k+1}|a_{f_j})\right]\right. \\ &\left.\sum_{b_1,b_2}\alpha_k(b_1, b_2; f_j,\ j \in \Gamma(k))\right]. \end{aligned} \tag{3}$$

Please notice that if $\Lambda_2(k+1)$ is empty, then we skip the corresponding summation in the formula. If, at position d, the set $\Gamma(d)$ is empty, then we have

$$\Pr(\mathbf{X}_{ij} = x_{ij},\ i = 1, \ldots, m, j = 1, \ldots, d) = \sum_{a_1,a_2}\alpha_d(a_1, a_2). \tag{4}$$

Despite the appearance of the formula, we can prove the following result:

Proposition 2. *The complexity of the algorithm defined by (3) and (4) is linear with respect to the number of polymorphic* *sites. The expected complexity is proportional to e^{κ}, where κ is the average coverage.*

This is true because, in each step of the recursion, we deal with one more locus by (3) and keep the state variables $\{\alpha_k(a_1, a_2; f_i,\ i \in \Gamma(k))\}$ in memory. The memory size is thus the state dimension, which is not constant along a chromosome. Denote the coverage variable by K. Approximately, it follows a Poisson distribution with the parameter κ. On average, the memory size is proportional to $E[2^K] = E[e^{\log\, 2\, K}] = e^{\kappa}$ according to the moment generating function of Poisson distribution.

The peak memory usage is an important issue in practice. Of course, we can keep the active coverage in some manageable range by randomly skipping some fragments. Next, we evaluate the worst case by the Poisson distribution

$$\Pr(K \geq m) = \sum_{j=m}^{\infty} \frac{\kappa^j e^{-\kappa}}{j!}.$$

Let $W_1, W_2, \cdots, W_m$ be independent and exponentially distributed random variables with parameter κ. According to the structure of the Poisson process [12], the above quantity equals

$$\begin{aligned}\Pr(K \geq m) &= \Pr(W_1 + W_2 + \cdots + W_m \leq 1)\\ &= \int_o^1 \frac{\kappa^m t^{m-1} e^{-\kappa t}}{(m-1)!}\, dt,\end{aligned}$$

namely, an incomplete Gamma integral. In the case of *Ciona intestinalis* shotgun sequencing, κ is seven and $\Pr(K \geq 20) = 0.000044$, $\Pr(K \geq 23) \leq 10^{-6}$. Thus, it is very unlikely that the memory requirement exceeds 2^{20}. Our simulations justify this analysis.

2.3 Sequencing Error Rates and Quality Scores

The values of $\eta_{ij}(b|a)$ are crucial in our reconstruction procedure. If we assume that base-calling is independent and identically distributed across the assembly and $\eta_{ij}(b|a) = \eta(b|a)$, then we can apply an E-M procedure sketched in [9] to estimate them.

Another approach makes use of the quality scores provided by some base-calling algorithms ([6]). We can connect these scores to our model if a valid probabilistic interpretation is available. First, we consider the cases of SNPs. Write

$$\varepsilon_{ij} = \Pr(\mathrm{Base-call} \neq a | \mathrm{True\ base} = a).$$

The quality scores are usually given in the following transformed form:

$$q_{ij} = -10 \log_{10} \varepsilon_{ij}.$$

Approximately, we have

$$\eta_{ij}(b|a) = \begin{cases} 1 - \varepsilon_{ij} & a = b, \\ \varepsilon_{ij}\omega(b|a) & a \neq b. \end{cases}$$

The error-bias parameters $\{\omega(b|a),\ a \neq b,\ a \in \mathcal{A}, b \in \mathcal{B}\}$ can either be set to some constants or can be estimated from data. In the case of complex indels, we align an observed sequence with a template and calculate $\eta_{ij}(b|a)$ by multiplying scores from each position. In the haploid case, Li et al. [10] provided a method to calibrate quality scores and estimate conditional error probability using a mixture of logistic regressions.

2.4 Inference of Haplotype Frequency

Next, our focus turns to the issue of haplotype frequency. For any fixed value of λ, the recursive algorithm, (3) and (4), allows us to efficiently compute the probability of the fragment assembly, or the likelihood as it is called in inference theory. Denote the log-likelihood of the observed assembly by

$$L(\lambda) = \log \Pr(\mathbf{X}; \lambda)\ ,$$

where $\mathbf{X}$ is the assembly matrix. By maximizing the log-likelihood with respect to the haplotype frequency parameter in the range from 0 to 1/2, we can obtain its estimate. In general, the maximum likelihood estimate is asymptotically efficient under regular conditions. In other words, it is one of the most accurate estimates in the large sample scenario. Alternatively, if we have several hypothetical values for the haplotype frequency known from a genomic or genetic context, say, $\lambda = 0, 1/4, 1/3, 1/2$, then we can select the value that achieves the largest likelihood. In real genome assembly, we can use the estimate of haplotype frequency to monitor the existence of misalignment of "bad" fragments.

2.5 Reconstruction of Haplotype

The conditional probability $\Pr(\mathbf{S}|\mathbf{X})$ plays a key role in our reconstruction and we term it a confidence score. The reconstruction procedure is a generalization of that given in [9]. We start by considering each adjacent pair. To determine the haplotype for two loci, we check the odds ratio of the two most probable states and the pairwise confidence score. In addition to forward linking of adjacent loci, we also check confidence scores and adjust solutions in a backward fashion.

Algorithm 1.

1. For each locus, we report the most probable genotype according to the single confidence score.
2. For each adjacent pair, we report the most probable haplotypes according to the odds ratio and pairwise confidence score.
3. Link the haplotype phases obtained in Step 2 and construct haplotype segments. If inconsistent adjacent pairs occur, then we consider these sites jointly.
4. Evaluate the overall confidence score for each haplotype segment.
5. If the confidence score is below a threshold, we check the pair of loci with the smallest pairwise confidence score obtained in Step 2. Then, we compute the overall confidence score by flipping the phase between these two loci. If the score exceeds the threshold, we stop; otherwise, we break the segment into two. In this case, we repeat Steps 4 and 5, respectively, to these two segments.

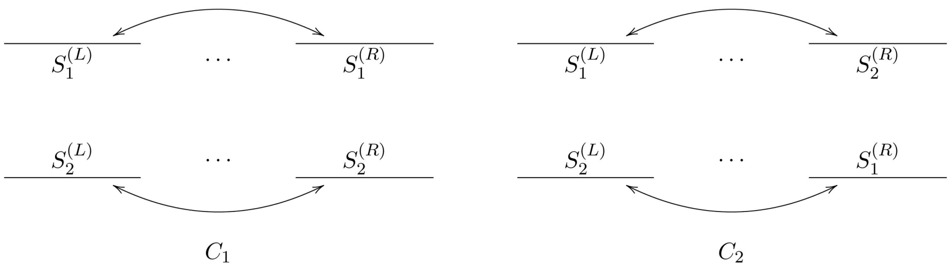


Fig. 4. Bridging two contigs by mate-pair fragments. Two possible configurations are shown.

In the case where the nominal frequency value is known, we can still use its estimated value in the calculation of confidence scores to achieve adaptive reconstruction and, consequently, to improve accuracy of reconstruction.

2.6 Mate-Pair Information and Second-Stage Bridging

The above model applies to the case of two-end sequencing if we assign the letter ϕ to those uncalled bases in the middle of each fragment. However, when we evaluate the overall confidence score by (3) and (4), a clone remains active even at those polymorphic sites between the two ends. This may increase the coverage severalfold. We notice that the mate-pair information does not provide much extra help in regions (scaffolds) of high coverages. Thus, we skip the mate-pair information in the first round of Algorithm 1. If two contigs are connected by at least one clone through mate-pair fragments after the first round, then we apply the algorithm to the two contigs and "bridging" clones, trying to determine the phase. We present some details in what follows. Suppose one contig consists of two haplotype segments $S_1^{(L)}$ and $S_2^{(L)}$, another contig consists of $S_1^{(R)}$ and $S_2^{(R)}$. Two possible phase configurations between the two nonoverlapping contigs are shown in Fig. 4 and we denote them by C_1 and C_2, respectively. Suppose some clones overlap with one contig at one end and overlap with another at the other end. Denote these clones by $\mathbf{Z} = \{Z_i, i \in I\}$. For each clone Z_i, denote by $J_{i,L}$ the index set of those polymorphic sites that overlap with $S_1^{(L)}$ and $S_2^{(L)}$. Similarly, we define $J_{i,R}$ for the polymorphic sites on Z_i at the other end. Thus, $Z_i = \{Z_{ij}, j \in J_{i,L}\} \cup \{Z_{ij}, j \in J_{i,R}\}$. Then,

$$\Pr(\mathbf{Z}|C_1) = \prod_{i \in I} \Pr(Z_i|C_1), \Pr(\mathbf{Z}|C_2) = \prod_{i \in I} \Pr(Z_i|C_2),$$

where

$$\begin{cases} \Pr(Z_i|C_1) = & \frac{1}{2}\prod_{j\in J_{i,L}} \Pr(Z_{ij}|S_{1j}^{(L)}) \prod_{j\in J_{i,R}} \Pr(Z_{ij}|S_{1j}^{(R)}) \\ & +\frac{1}{2}\prod_{j\in J_{i,L}} \Pr(Z_{ij}|S_{2j}^{(L)}) \prod_{j\in J_{i,R}} \Pr(Z_{ij}|S_{2j}^{(R)}) \\ \Pr(Z_i|C_2) = & \frac{1}{2}\prod_{j\in J_{i,L}} \Pr(Z_i j|S_{1j}^{(L)}) \prod_{j\in J_{i,R}} \Pr(Z_{ij}|S_{2j}^{(R)}) \\ & +\frac{1}{2}\prod_{j\in J_{i,L}} \Pr(Z_{ij}|S_{2j}^{(L)}) \prod_{j\in J_{i,R}} \Pr(Z_{ij}|S_{1j}^{(R)}). \end{cases}$$

Based on the calculation, we make the following decision according to a threshold larger than one:

$$\begin{cases} \text{accept } C_1 & \text{if} \quad \frac{\Pr(\mathbf{Z}|C_1)}{\Pr(\mathbf{Z}|C_2)} > \text{threshold}, \\ \text{accept } C_2 & \text{if} \quad \frac{\Pr(\mathbf{Z}|C_2)}{\Pr(\mathbf{Z}|C_1)} > \text{threshold}, \\ \text{no decision} & \text{if} \quad \text{otherwise}. \end{cases}$$

We iterate this bridging step to extend haplotype segments.

To evaluate the confidence score of any extended contig, we regard a "bridging" clone as one single fragment by including its mate-pair information. We emphasize that the two mate-pair fragments of a clone are not considered jointly if they fall in the same contig because the extra phase information is negligible in this case. Only the two mate-pair fragments from a "bridging" clone are treated as linked in (3). Thus, coverage is not an issue anymore.

3 Results and Discussion

Ciona intestinalis is an important organism to study the origins of chordates and vertebrates. A draft of its protein-coding portion has been reported ([5]). Its high polymorphism rate, about 1.2 percent as reported, makes it an ideal case for reconstructing haplotypes from shotgun sequencing. To evaluate the proposed methodology, we simulated contigs according to the parameters obtained from *Ciona* sequencing.

The simulation was based on the stochastic model proposed in [8]. Denote the clone length by H. According to the random model, the number of polymorphic sites in the clone, denoted by $N(H)$, is a Poisson random variable with the parameter λH, where $1/\lambda$ measures the average interarrival distance between adjacent polymorphic sites. Conditional on the total number, the positions of polymorphic sites are uniformly distributed along the interval $[0, H]$. We generated random fragments (1.8K bp ~ 120K bp) according to their proportions in the *Ciona intestinalis* ([5]) and simulated two-end sequencing of the fragments. The average sequencing read was 650 bp and coverage was seven. To match the polymorphism rate of *Ciona intestinalis*, the expected interarrival time between potential adjacent loci was set to be 66 bp. The sequencing error rates were about 4 percent.

We are applying the method to reconstruct the diploid genome using the whole-genome shotgun sequencing data; see http://genome.jgi-psf.org/ciona4/ciona4.download.ftp.html. Namely, we first align reads to the published reference sequence and then apply Algorithm 1 to each scaffold. Fig. 5 shows a part of one scaffold in which nine fragments were aligned; the two targets are shown at the top. Four polymorphic sites, including an indel CCC/- - -, were observed in this region of about 40 nucleotides. This is quite typical in the *Ciona intestinalis* genome ([5]).

```
scaffold_1611 #1 aaCgagataatagaatTagaagtgt---atcttcccca-Ccct-t
              #2 aaTgagataatagaatAagaagtgtCCCatcttcccca-Acct-t

                 aaCgagataatagaatTagaagtgt---atcttcccca-Ccct-t
                 aaCgagataatagaatTagaagtgt---atcttcccca-Ccct-t
                 aaCgagataatagaatTagaagtgt---atcttcccca-Ccct-t
                 aaCgagataatagaatTagaagtgt---atcttcccca-Ccct-t
                 aaTgagataatagaatAagaagtgtCCCatcttcccca-Acct-t
                 aaCgagataatagaatTagaagtgt---atcttccccacCcct-t
                 aaTgagataatagaatAagaagtgtCCCatcttcccca-Acct-t
                 -aCgagataatagaatTagaagtgt---atcttcccca-Ccct-t
                               atTagaagtgt---atcttcccca-Ccctgt
```

Fig. 5. Part of a scaffold from *Ciona intestinalis*. Nine fragments were aligned and four polymorphic sites were observed. Nonpolymorphic and polymorphic sites are represented by small and large letters, respectively. The two targets are shown at the top.

3.1 Reconstruction of Haplotype Segments

We reconstructed haplotype segments by applying Algorithm 1. In Step 2, we scanned each adjacent pair of loci for significant haplotypes. The threshold was set as follows: The pairwise confidence score is larger than 0.5 and the odds ratio of the top two most probable cases is larger than 1.1. We reported outcomes under two haplotype frequencies, 0.5 and 0.25. The results are shown in Table 1. In the case of $r = 0.5$, the true positive rate was 97.05 percent. The percentage of correctly detected pairs among all is 88.96 percent. We also include results for the case of $r = 0.5$ without mate-pair information. In the case of haplotype frequency $r = 0.25$, the performance was still satisfactory considering the coverage and sequencing error rates.

The performance of the method on genomes of less dense polymorphisms is tested by another simulation. We simulate a situation of a polymorphism rate of 0.3 percent, which can be found in some regions of *Homo sapiens*. The results are shown in Table 2.

3.2 Length of Haplotype Segment and Two-End Sequencing

The gain of two-end sequencing of variable size fragments and the proposed bridging strategy using mate-pair information can be measured by the average length of haplotype segment. In the simulation of the *Ciona intestinalis* genome, on average, each haplotype segment contains 70.36 polymorphic sites while it contains only 45.43 polymorphic sites without mate-pair information. In the case of 0.3 percent polymorphic rates, on average, each haplotype segment contains 33.87 polymorphic sites while it contains only 5.06 polymorphic sites without mate-pair information. This shows that two-end sequencing strategy offers significant haplotype information and the bridging strategy works well.

3.3 Error Patterns

We categorize false positive errors in Table 3. As we can see, phase errors are rare. Some errors are of partial genotypes. Namely, one base in a genotype is mistaken and the other one is correct.

3.4 Confidence Scores for Haplotype Segments

The accuracy assessment of haplotype estimation is exemplified in Fig. 6. The estimated haplotype segments can be evaluated by the scores coupled with them. We checked the consistency of observed probability scores versus nominal scores calculated by (4) and (3) in Fig. 7. When the confidence score is larger than 0.5, the empirical

TABLE 1
The Reconstruction Result from a Simulation Based on *Ciona intestinalis*

haplotype frequency	$\lambda = 1/2$	$\lambda = 1/2$	$\lambda = 1/4$
mate-pair information	w/o mate-pair	w/i mate-pair	w/i mate-pair
total # polymorphism	674246	674246	671359
# polymorphism reported (including singleton)	618034	618034	554442
true positive rate (all reported)	97.51%	97.05%	96.08%
percentage of correctly detected sites	89.38%	88.96%	79.35%
average segment length	45.43	70.36	31.10

The polymorphism rate is 1.2 percent. The total size of scaffolds is about 60M bp. To determine the significance of pairwise comparison, we set the thresholds for pairwise confidence to be 0.5 and the odds ratio of the two most probable cases to be 1.1. The number of polymorphisms in the final report includes singletons, namely, those single sites that cannot be connected to others. In this case, we report their genotypes. The true positive rates are for those reported sites, either genotypes or haplotypes. The last two accounts are the percentage of correctly detected pairs among all the polymorphic sites generated and the average lengths of haplotype segments.

TABLE 2
The Reconstruction Result from a Simulation of a Polymorphism Rate 0.3 percent, cf. Table 1

haplotype frequency	$\lambda = 1/2$	$\lambda = 1/2$	$\lambda = 1/4$
mate-pair information	w/o mate-pair	w/i mate-pair	w/i mate-pair
total # polymorphism	179686	179686	180294
# polymorphism reported (including singleton)	165188	165188	151501
true positive rate (all reported)	98.23%	96.0%	94.49%
percentage of correctly detected sites	90.31%	88.25%	79.40%
average segment length	5.06	33.87	19.39

TABLE 3
Error Patterns

Error type		$\lambda = 1/2$	$\lambda = 1/4$
Truth	Reconstructed	percentage	
heterozygote	heterozygote, one match	1.23	2.77
heterozygote	wrong phase	1.72	1.15
Total		2.95	3.92

The polymorphism rate is 1.2 percent.

results are quite consistent with the expected ones. When the confidence score is smaller than 0.5, the empirical error rates are slightly higher than the nominal ones. This is due to the fact that we reject some phases in the pairwise comparison step using a threshold of 0.5. We can correct the bias in the range of low probability by a straightforward empirical method. We also check the confidence scores of each haplotype segment versus number of polymorphic sites. Most of the errors occurred in short segments with less than four polymorphic sites due to factors such as low coverage and relatively large distance from other polymorphic sites.

3.5 MLE of Haplotype Frequency

A theoretical assessment of the estimation is hard to establish due to the complicated data structure. To study its statistical performance, once again we used the simulation explained earlier except that the scaffold size was 120K bp. The left side of Fig. 8 shows the histogram of MLE when the true frequency is one-quarter. It is approximately a normal distribution. The right side of Fig. 8 shows the histogram of MLE when the true frequency is one-half. The averages and standard deviations of the estimates are shown in Table 4. The larger the scaffold size, the more accurate the MLE.

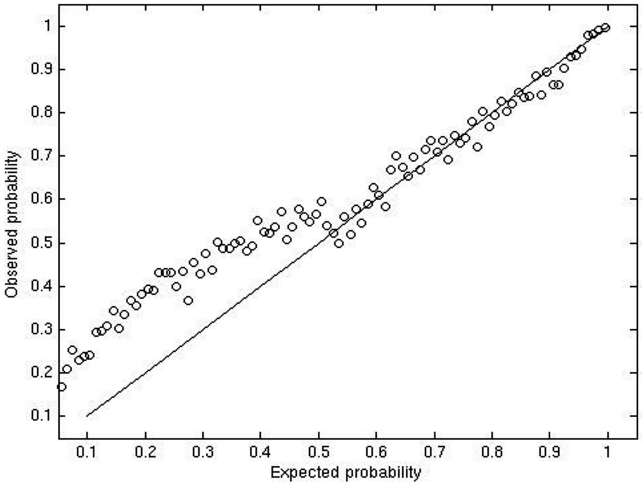


Fig. 7. Observed error rates versus nominal scores calculated by (4) and (3).

The estimation of haplotype frequency can be applied to the detection of duplications in shotgun sequencing. Fig. 9 shows the log-likelihood as a function of the haplotype frequency, respectively, for the heterozygous and homozygous cases. The scaffold size was 120k bp. As expected, the likelihoods were maximized, respectively, at around one-half and zero. In comparison, the change of coverage has been used as an indication of duplication [3]. In practice, we estimate haplotype frequency for each contig and a significant change in value may indicate that an abnormality, such as misalignment, has occurred.

3.6 Composition Probabilities

Other than sequencing error rates and haplotype frequency, the parameters in our model also include the compositional probabilities. Churchill and Waterman [4] derived an E-M procedure to estimate the compositional probabilities, sequencing error rates from the assembly for the one-chromosome problem. In principle, similar ideas apply to our case. If we assume independence of composition between two haplotypes, then we have a more parsimonious representation: $\mu(a,b) = \mu(a)\mu(b)$. With the information of S, the estimates of composition probabilities are simply frequencies. The iterative E-M algorithm is based on estimation of S. We note that the conditional probabilities $\Pr(\mathbf{S}|\mathbf{X})$ are no longer independent among polymorphic sites and we have to estimate them jointly. Computationally, this is a challenging problem due to the large size of the space $\mathbf{S}$. The following strategy is easy to implement:

- Reconstruct the most likely haplotype under the current parameter values.

```
Segment 7 starts at 144 and ends at 201 - size: 58 - confidence score: 0.9960188778
Template 1:       ATACCTCGTTCCGAATGCGAATACCGCTCAATACTGAACCTGTAAACCAACGCGTAGA
Template 2:       TCCATAATGGTAC-TACACCGGTGTATCTCCGGTACTCGAGAAGTCTGTGAATTGTAG

Segment 8 starts at 202 and ends at 230 - size: 29 - confidence score: 0.9752074893
Template 1:       TAATTACCATAGTGACATCAGTTCAATTT
Template 2:       ACGCCGTACCCTCCTA-AGCAAAACGACA

Segment 9 starts at 231 and ends at 254 - size: 24 - confidence score: 0.6125626073
Template 1:       ATGCCAACATTCTCCCCGCAGCTA
Template 2:       -GAAATTGTGGTGTTAATTGAGAT
```

Fig. 6. Examples of accuracy assessment. The haplotype segments with positions, sizes, and confidence scores are produced by our program.

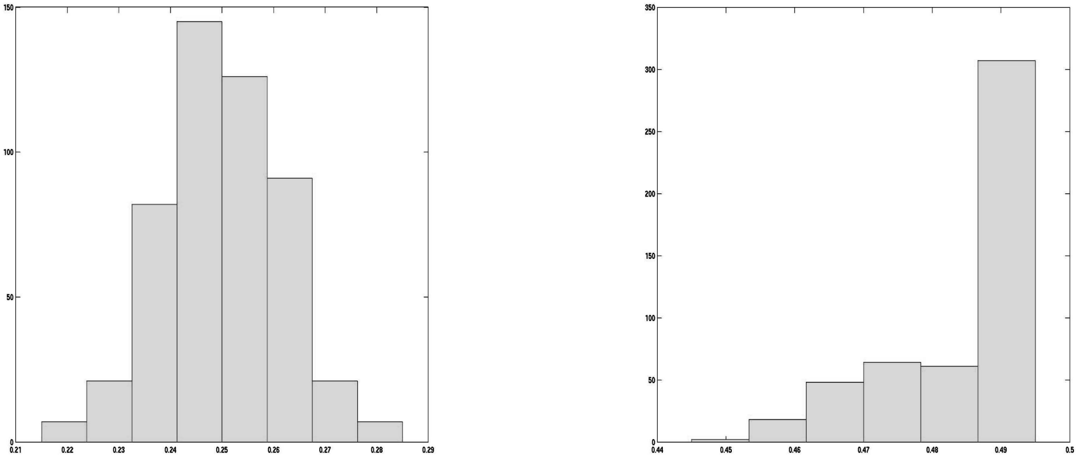


Fig. 8. Histograms of MLE estimates. Fragments were generated from scaffolds of size 120K bp. Left: The true haplotype frequency is one-quarter. Right: The true haplotype frequency is one-half.

- Update the frequency of each of the positive nucleotide bases.

3.7 Code

We have coded the algorithms presented in the paper in C on the Linux platform and have tested them on various simulation examples. We will be happy to send our code to readers at their request.

Appendix

Proof of Theorem 1. Because the derivation is quite lengthy, we include some explanatory remarks after each step. Without loss of clarity, we use the same letter in different subscripts in the same line. The scope of each subscript is restricted by its domain. These domains are separated by semicolons or can be identified from the context. According to the definition of α_k, only the fragments in $\Theta(k)$ are relevant. Therefore, we have

$$\alpha_{k+1}(a_1, a_2; f_i,\ i \in \Gamma(k+1))$$

$$= \Pr(\mathbf{X}_{ij} = x_{ij},\ \ j = 1, \ldots, k,\ i \in \Theta(k+1);$$

$$S_{k+1,1} = a_1, S_{k+1,2} = a_2; F_i = f_i, i \in \Gamma(k+1))$$

(next sum over cases of $\Lambda_1(k+1)$ and $\Lambda_2(k+1)$ since $\Gamma(k+1) = \Lambda_3(k+1) \cup \Lambda_4(k+1)$)

$$= \sum_{b_1,b_2} \sum_{f_i=1,2,\ i\in\Lambda_1(k+1)\cup\Lambda_2(k+1)} \{\Pr(\mathbf{X}_{ij} = x_{ij},\ j = 1, \ldots, k+1,$$

$$i \in \Theta(k+1); S_{k,1} = b_1, S_{k,2} = b_2;$$

$$S_{k+1,1} = a_1, S_{k+1,2} = a_2; F_i = f_i, i \in \Omega(k+1))\}$$

(next apply the Markov structure)

$$= \sum_{b_1,b_2} \sum_{f_i=1,2,\ i\in\Lambda_1(k+1)\cup\Lambda_2(k+1)}$$

$$\{\Pr(\mathbf{X}_{i,k+1} = x_{i,k+1},\ i \in \Omega(k+1) | S_{k+1,1} = a_1, S_{k+1,2} = a_2;$$

$$F_i = f_i, i \in \Omega(k+1))$$

$$\Pr(\mathbf{X}_{ij} = x_{ij},\ j = 1, \ldots, k,\ i \in \Theta(k); S_{k,1} = b_1, S_{k,2} = b_2;$$

$$S_{k+1,1} = a_1, S_{k+1,2} = a_2; F_i = f_i, i \in \Omega(k+1))\}$$

(notice that $\Gamma(\mathrm{k}) = \Lambda_2(\mathrm{k}+1) \bigcup \Lambda_4(\mathrm{k}+1)$)

$$= \sum_{b_1,b_2} \sum_{f_i=1,2\ i\in\Lambda_1(k+1)} \sum_{f_i=1,2,\ i\in\Lambda_2(k+1)}$$

$$\{\Pr(\mathbf{X}_{i,k+1} = x_{i,k+1},\ i \in \Omega(k+1) | S_{k+1,1} = a_1, S_{k+1,2} = a_2;$$

$$F_i = f_i, i \in \Omega(k+1))$$

$$\Pr(\mathbf{X}_{ij} = x_{ij},\ j = 1, \ldots, k,\ i \in \Theta(k); S_{k,1} = b_1, S_{k,2} = b_2;$$

$$S_{k+1,1} = a_1, S_{k+1,2} = a_2; F_i = f_i, i \in \Gamma(k);$$

$$F_i = f_i, i \in \Lambda_1(k+1); F_i = f_i, i \in \Lambda_3(k+1))\}$$

(next factorize some quantities by independence of fragments and base compositions)

$$= \sum_{b_1,b_2} \sum_{f_i\ i\in\Lambda_1(k+1)} \sum_{f_i,\ i\in\Lambda_2(k+1)}$$

$$\left\{\left[\prod_{i\in\Omega(k+1)} \Pr(\mathbf{X}_{i,k+1} = x_{i,k+1} | S_{k+1,f_i} = a_{f_i})\right]\right.$$

$$\Pr(\mathbf{X}_{ij} = x_{ij},\ i \in \Theta(k), j = 1, \ldots, k; S_{k,1} = b_1, S_{k,2} = b_2;$$

$$F_i = f_i, i \in \Gamma(k))\Pr(S_{k+1,1} = a_1, S_{k+1,2} = a_2)$$

$$\left.\Pr(F_i = f_i, i \in \Lambda_1(k+1))\Pr(F_i = f_i, i \in \Lambda_3(k+1))\right\}$$

(notice that $\Omega(k+1) = \Lambda_1(k+1) \bigcup \Lambda_2(k+1) \bigcup \Lambda_3(k+1) \bigcup \Lambda_4(k+1)$ and move common terms out of summation)

TABLE 4
Statistics of the Maximum Likelihood Estimates of Haplotype Frequency

	mean	STD
r=0.5	0.4867	0.0122
r=0.25	0.2501	0.0134

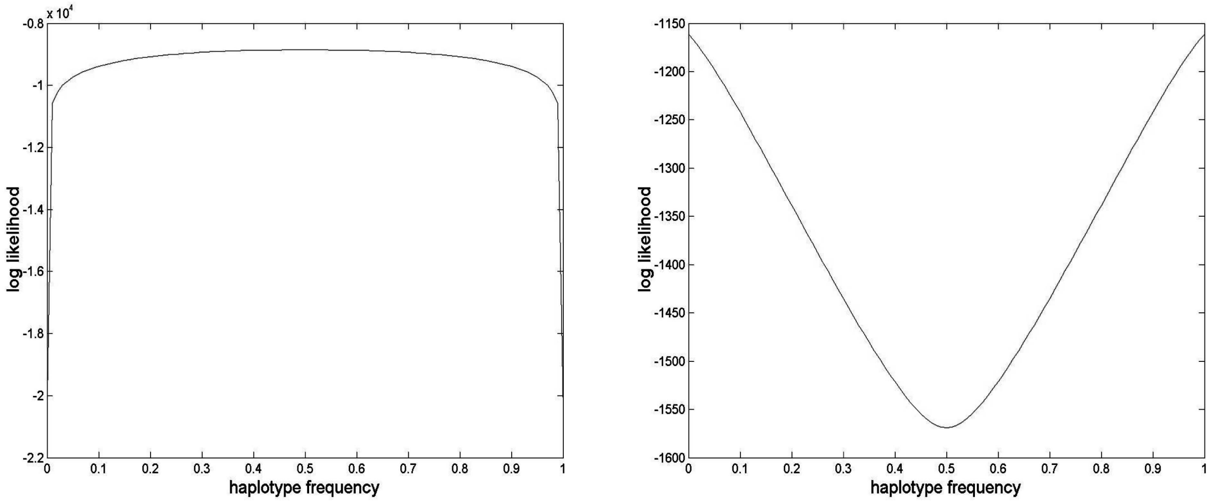


Fig. 9. The log-likelihood curve—logarithm of probability of observations as a function of haplotype frequency. The scaffold size is 120K bp. Left: The true haplotype frequency is one-half. Right: The true haplotype frequency is one—homozygous case.

$$= \Pr(S_{k+1,1} = a_1, S_{k+1,2} = a_2)\Pr(F_i = f_i, i \in \Lambda_3(k+1))$$

$$\left[\prod_{i\in\Lambda_3(k+1)\cup\Lambda_4(k+1)} \Pr(\mathbf{X}_{i,k+1} = x_{i,k+1} | S_{k+1,F_i} = a_{f_i})\right]$$

$$\sum_{b_1,b_2}\sum_{f_i,\, i\in\Lambda_2(k+1)} \Bigg\{ \alpha_k(b_1, b_2; f_i, i \in \Gamma(k))$$

$$\left[\prod_{i\in\Lambda_2(k+1)} \Pr(\mathbf{X}_{i,k+1} = x_{i,k+1} | S_{k+1,F_i} = a_{f_i})\right]$$

$$\Bigg[\sum_{f_i,\, i\in\Lambda_1(k+1)} \Pr(F_i = f_i, i \in \Lambda_1(k+1))$$

$$\prod_{i\in\Lambda_1(k+1)} \Pr(\mathbf{X}_{i,k+1} = x_{i,k+1} | S_{k+1,F_i} = a_{f_i})\Bigg]\Bigg\}$$

$$\text{(The last line is } \left[\prod_{i\in\Lambda_1(k+1)} [\lambda_1\eta(x_{i,k+1}|a_1) + \lambda_2\eta(x_{i,k+1}|a_2)\right]\text{)}$$

$$= \Pr(S_{k+1,1} = a_1, S_{k+1,2} = a_2)\prod_{h=1}^{2}\left[\lambda_h^{\sum_{j\in\Lambda_3(k+1)} \mathbf{1}(F_j=h)}\right]$$

$$\left[\prod_{i\in\Lambda_3(k+1)\cup\Lambda_4(k+1)} \eta(x_{i,k+1}|a_{f_i})\right]$$

$$\sum_{b_1,b_2}\sum_{f_i,\, i\in\Lambda_2(k+1)} \Bigg\{ \alpha_k(b_1, b_2; f_i, i \in \Gamma(k)))\left[\prod_{i\in\Lambda_2(k+1)} \eta(x_{i,k+1}|a_{f_i})\right]$$

$$\left[\prod_{i\in\Lambda_1(k+1)} [\lambda_1\eta(x_{i,k+1}|a_1) + \lambda_2\eta(x_{i,k+1}|a_2)]\right]\Bigg\}$$

(move the terms relating to $\Lambda_1(k+1)$ out of summation and rearrange summations)

$$= \Pr(S_{k+1,1} = a_1, S_{k+1,2} = a_2)\left[\prod_{h=1}^{2} \lambda_h^{\sum_{j\in\Lambda_3(k+1)} \mathbf{1}(F_j=h)}\right]$$

$$\left[\prod_{j\in\Lambda_1(k+1)} [\lambda_1\ \eta(x_{j,k+1}|a_1) + \lambda_2\ \eta(x_{j,k+1}|a_2)\right]$$

$$\left[\prod_{j\in\Lambda_3(k+1)\cup\Lambda_4(k+1)} \eta(x_{j,k+1}|a_{f_j})\right]$$

$$\Bigg\{\sum_{f_j=1,2,\ j\in\Lambda_2(k+1)}\left[\prod_{j\in\Lambda_2(k+1)} \eta(x_{j,k+1}|a_{f_j})\right]$$

$$\sum_{b_1,b_2} \alpha_k(b_1, b_2; f_j,\ j \in \Gamma(k))\Bigg\}.$$

□

Acknowledgments

This work was supported by the Center of Excellence in Genome Science at the University of Southern California, US National Institutes of Health (NIH) P50 HG002790 CEGS grant. The real sequence data of *Ciona intestinalis* were produced by the US Department of Energy Joint Genome Institute, http://www.jgi.doe.gov/. Lei M. Li's work is partially supported by grant R01 GM75308-01 from the NIH.

References

[1] M.D. Adams et al., "The Genome Sequence of *Drosophila Melanogaster*," *Science*, vol. 287, pp. 2185-2195, 2000.

[2] S. Aparicio et al., "Whole-Genome Shotgun Assembly and Analysis of the Genome of *Fugu Rubripes*," *Science*, vol. 297, pp. 1301-1310, 2002.

[3] J.A. Bailey et al., "Recent Segmental Duplications in the Human Genome," *Science*, vol. 297, pp. 1003-1007, 2002.

[4] G.A. Churchill and M.S. Waterman, "The Accuracy of DNA Sequences: Estimating Sequence Quality," *Genomics*, vol. 14, pp. 89-98, 1992.

[5] P. Dehal et al., "The Draft Genome of *Ciona Intestinalis*: Insights into Chordate and Vertebrate Origins," *Science,* vol. 298, pp. 2157-2167, 2002.

[6] B. Ewing and P. Green, "Base-Calling of Automated Sequencer Traces Using *phred*. 2. Error Probabilities," *Genome Research,* vol. 8, pp. 186-194, 1998.

[7] G. Lancia, V. Bafna, S. Istrail, R. Lippert, and R. Schwartz, "SNPs Problems, Complexity, and Algorithms," *Proc. European Symp. Algorithms,* pp. 182-193, 2001.

[8] E.S. Lander and M.S. Waterman, "Genomic Mapping by Fingerprinting Random Clones," *Genomics,* vol. 2, pp. 231-239, 1998.

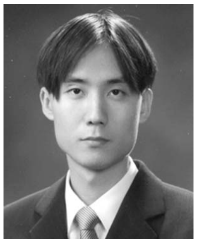
[9] L.M. Li, J.H. Kim, and M.S. Waterman, "Haplotype Reconstruction from SNP Alignment," *J. Computational Biology,* vol. 11, pp. 505-516, 2004.

[10] M. Li, M. Nordborg, and L.M. Li, "Adjusted Quality Scores from Alignment and Improve Sequencing Accuracy," *Nucleic Acid Research,* vol. 32, pp. 5183-5191, 2004.

[11] R. Lippert, R. Schwartz, G. Lancia, and S. Istrail, "Algorithmic Strategies for the SNP Haplotype Assembly Problem," *Briefings in Bioinformatics,* vol. 3, pp. 1-9, 2002.

[12] S.M. Ross, *Introduction to Probability Models,* fourth ed. Academic Press, 1989.

[13] J.C. Venter et al., "The Sequence of the Human Genome," *Science,* vol. 291, pp. 1304-1351, 2001.



Jong Hyun Kim received the bachelor's degree from Yonsei University in 2001. He is a PhD candidate in the Department of Computer Science at the University of Southen California. He is interested in applying computational methods to the fields of genomics and systems biology.

Michael S. Waterman is a professor of biological sciences, mathematics, and computer science at the University of Southern California. He is a founding editor of the *Journal of Computational Biology* and is coauthor of the texts *Computational Genome Analysis: An Introduction* and *Introduction to Computational Biology: Maps, Sequences and Genomes*.

Lei M. Li received the BS degree in mathematics and the MS degree in statistics in 1988 and 1991, respectively, from Peking University, China, and the PhD degree in statistics in 1998 from the University of California at Berkeley. He is currently an associate professor of computational biology and mathematics at the University of Southern California. His research interests include computational biology and bioinformatics, statistical modeling, and computation.