Pacific Symposium on Biocomputing 7:199-210 (2002)

EULER-PCR: FINISHING EXPERIMENTS FOR REPEAT RESOLUTION

ZUFAR MULYUKOV, PAVEL A. PEVZNER
Department of Computer Science and Engineering,
University of California, San Diego, CA 92093, USA

Genomic sequencing typically generates a large collection of unordered contigs or scaffolds. Contig ordering (also known as gap closure) is a non-trivial algorithmic and experimental problem since even relatively simple-to-assemble bacterial genomes typically result in large set of contigs. Neighboring contigs maybe separated either by *gaps* in read coverage or by *repeats*. In the later case we say that the contigs are separated by *pseudogaps*, and we emphasize the important difference between gap closure and pseudogap closure. The existing gap closure approaches do not distinguish between gaps and pseudogaps and treat them in the same way. We describe a new fast strategy for closing pseudogaps (repeat resolution). Since in highly repetitive genomes, the number of pseudogaps may exceed the number of gaps by an order of magnitude, this approach provides a significant advantage over the existing gap closure methods.

1 Introduction

Large scale sequencing projects always require a *finishing* phase, i.e., designing and conducting additional experiments for closing gaps and establishing the overall order of contigs. The design of such finishing experiments still requires extensive human intervention using interactive tools, such as sequence editors (Gordon et al., 1998 [1]). A typical DNA sequencing project generates a large collection of unordered contigs or scaffolds. Ordering such contigs is a major effort and often a bottleneck in sequence finishing. Contig ordering is usually done by PCR experiments that correspond to the queries "Are the contigs A and B neighbors?" A naive approach to such "The twenty questions game" requires PCR experiments for every pair of contigs and is very time-consuming. Sorokin et al., 1996, [2] Tettelin et al., 1999, [3] and Beigel et al., 2001 [4] suggested *multiplex PCR* approach that uses pooling strategy to ask more complicated queries "Given sets of contigs $\mathcal{A}$ and $\mathcal{B}$, do they contain contigs $A \in \mathcal{A}$ and $B \in \mathcal{B}$ that are neighbors?".

Contig ordering is closely related to *gap closure*. Neighboring contigs maybe separated either by *gaps* in read coverage or by *repeats*. In the later case of repeat-induced gaps we say that the contigs are separated by *pseudogaps*. For example, in the *Neisseria meningitidis* (NM) project (Parkhill et al., 2000 [5]), Phrap generates 160 contigs, but only half of them are separated by gaps, while the other half is separated by pseudogaps (Pevzner et al.,

2001 [6]). The existing contig ordering algorithms do not distinguish between gaps and pseudogaps and treat them in the same way. This approach is inefficient, since it ignores information available for pseudogaps, such as repeat length and contig sequences adjacent to a particular repeat. Therefore, an algorithm that employs a separate approach to resolving pseudogaps provides a significant advantage over the existing gap closure methods. We describe a new algorithm, EULER-PCR, that significantly reduces the number of finishing experiments for repeat resolution. EULER-PCR software is available by contacting Z.M.

2 Repeat graph

Long repeats present a problem in DNA sequencing since they often lead to multiple solutions of the fragment assembly problem. Figure 1(a) illustrates the "repeat problem" caused by perfect triple repeat that leads to two possible sequence assemblies. The classical "overlap-layout-consensus" approach (Kececioglu and Myers, 1995 [7]) to the assembly problem is based on the notion of the *overlap graph* (Fig. 1(b)). Every read corresponds to a vertex in the overlap graph and two vertices are connected by an edge if the corresponding reads overlap. The DNA sequence corresponds to a path traversing the consecutive reads in this graphs. The fragment assembly problem is thus cast as finding a path in the overlap graph visiting every vertex exactly once, a *Hamiltonian path* problem. However, repeats complicate the overlap graph since repeated regions create edges between non-consecutive reads. The Hamiltonian path problem is NP-complete and the efficient algorithms for solving this problem in large graphs are unknown. This is the reason why fragment assembly of highly repetitive genomes is a notoriously difficult problem. Myers et al., 2000 [8] suggested to mask most of multi-copy repeats, thus breaking the assembly into a large number of contigs. A better approach would be to use the information about repeated regions and try to reduce the number of contigs.

Pevzner et al. 2001, [9] and Pevzner and Tang 2001 [6] developed a new fragment assembly algorithm (EULER) based on the Eulerian path approach. Instead of masking repeats and breaking DNA sequence into a set of contigs, EULER constructs a *repeat graph*, which represents the repeat structure better than the overlap graph does. Given a DNA sequence, the repeat graph can be visualized by glueing together all identical repeated regions (Fig.1(c)). One can see that the repeat graph (Fig.1(c)) is a much simpler representation of repeats than the overlap graph (Fig.1(b)).

To construct the repeat graph from the set of sequencing reads, EULER

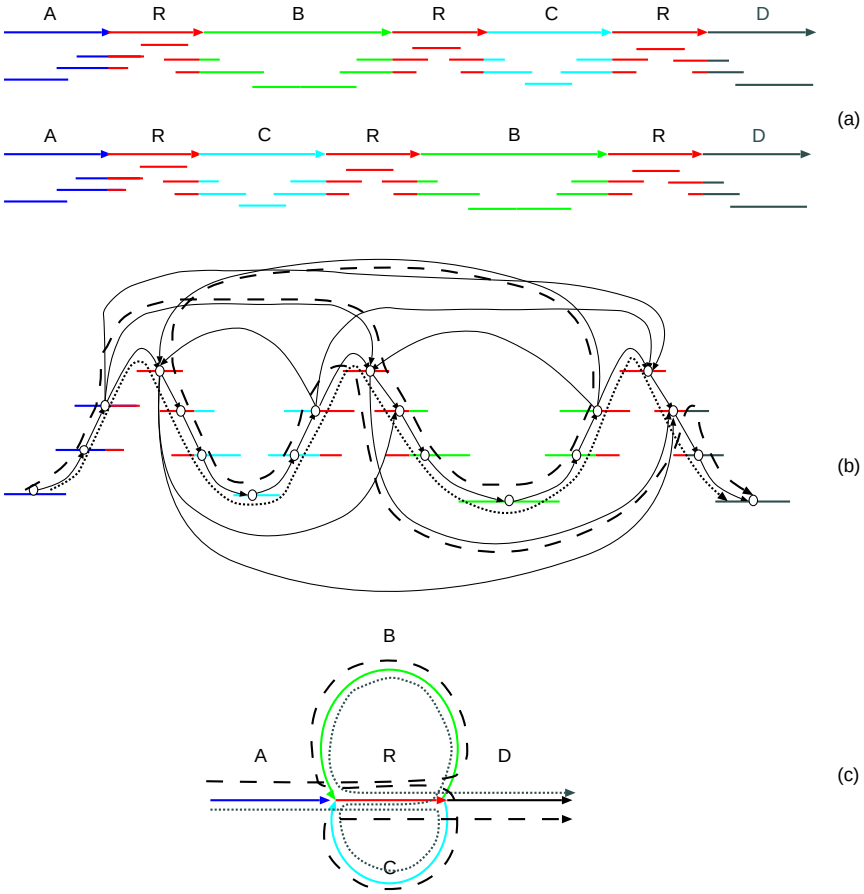

Figure 1: **(a)** DNA sequence with a triple repeat R and four unique segments A, B, C, D. Due to the repeat R, the same set of sequencing reads (shown by short lines under assembled DNAs) can be assembled either as ARBRCRD (upper assembly) and ARCRBRD (lower assembly), which differ by transposition of B and C. **(b)** Overlap graph for "overlap-layout-consensus" approach. Two Hamiltonian paths, corresponding to two possible fragment assemblies, are shown by dashed (for ARBRCRD) and dotted (for ARCRBRD) lines. **(c)** Repeat graph where three copies of the repeat R are "glued" into a single edge. Every Eulerian path in this graph corresponds to a valid solution of the fragment assembly problem. Two Eulerian paths, corresponding to two possible fragment assemblies, are shown by dashed (for ARBRCRD) and dotted (for ARCRBRD) lines.

breaks the reads into short k-mers (continuous strings of length k). One can view such k-mers as a result of hybridization of reads with a very large virtual DNA chip. These k-mers are represented by edges of the *de Bruijn* graph, while the set all $(k-1)$-mers from set of sequencing reads are represented by vertices of the graph. Two vertices v and w are joined by a directed edge if there is a k-mer in which first $(k-1)$ nucleotides coincide with v, and last $(k-1)$ nucleotides coincide with w (see example in Fig.2). We emphasize that the fragment assembly is now cast as finding a path visiting every *edge* of the graph exactly once, an *Eulerian path* problem. In contrast to the Hamiltonian path problem, the Eulerian path problem is easy to solve even for graphs with millions of vertices since there exist linear-time Eulerian path algorithms. This is the fundamental difference between the EULER algorithm (Pevzner et al., 2001) and the "overlap-layout-consensus" approach. The repeat graph is obtained from de Bruijn graph by collapsing paths in the de Bruijn graph into single edges (see Pevzner et al. 2001 [9] for details).

In this new approach contigs, which would be disconnected if repeats were masked, are represented by edges of a connected graph. We can compute repeat copy numbers by assigning minimal (nonzero) multiplicities to the graph edges that balance in-flow and out-flow on every vertex (Pevzner and Tang, 2001 [6]). For example in Fig. Fig.1(c), repeat edge R has multiplicity 3. Edges with multiplicity higher than one represents a repeat, while edges with unit multiplicity represent conventional contigs. Note, that every repeat corresponds to a single edge in the repeat graph rather than to a collection of vertices in the layout graph. The DNA sequence in Fig.1(a) consists of four unique segments A,B,C,D and one triple repeat R. The corresponding repeat graph (Fig.1(c)) consists of 4+1=5 edges. Two edges X and Y in the repeat graph follow each other if and only if segment X follow segment Y in the DNA sequence. For a repeat edge $e=(v,w)$, edges entering the vertex v are called *entrances* into a repeat, and edges leaving the vertex w are called *exits* from a repeat.

Gaps in read coverage break DNA sequence into a set of Lander-Waterman islands (Lander and Waterman, 1988 [10]) and cause sequencing reads to be assembled into a set of contigs. The repeat graph for a continuous DNA sequence has a single source and a sink vertices, which correspond to the beginning and the end of the DNA sequence. On the other had, the repeat graph for a set of contigs has multiple sources and sinks, corresponding to contig end-points. Fig.3 shows the fragment assembly problem similar to Fig.1 with some reads missing thus leading to two islands in the read coverage. In this case the the repeat graph corresponds to two contigs and there are two possible solutions of the fragment assembly problem: contigs ARB and CRD, or contigs ARD and CRB. A single finishing PCR experiment would resolve

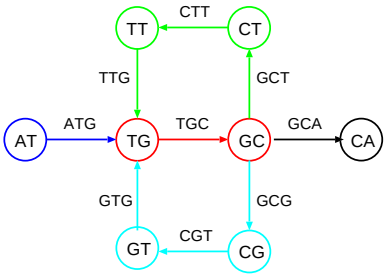

Figure 2: Example of a de Bruijn graph for sequence ATGCTTGCGTGCA, with edges being all 3-mers from this sequence, and vertices being all 2-mers. Due to the repeat TGC there is another sequence ATGCGTGCTTGCA corresponding to another Eulerian path in same de Bruijn graph (compare with Fig.1).

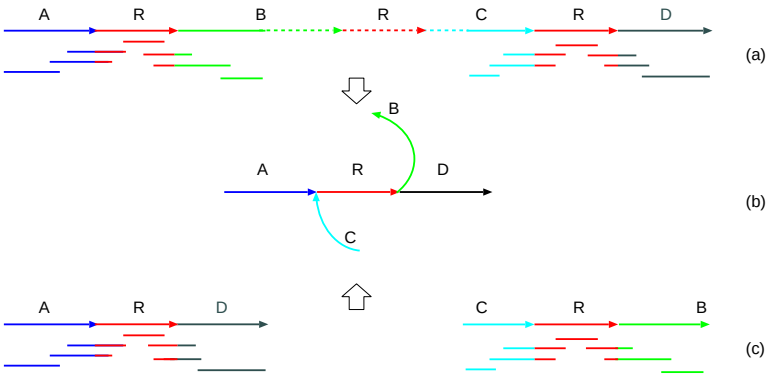

Figure 3: (a) Gap in read coverage breaks DNA sequence into two islands ARB and CRD (B and C correspond to the shortened versions of segments B and C from Fig.1). (b) The repeat graph for these two islands is similar to the repeat graph in Fig.1 with "broken" edges B and C. (c) Another solution of the fragment assembly problem with islands ARD and CRB (instead of ARB and CRD).

the repeat and delineate the correct assembly.

The above examples illustrate that the repeat graphs are valuable tools providing important new insights into the repeat structure, as well as guiding finishing experiments.

3 Repeat resolution

EULER (Pevzner et al., 2001[9]) typically resolves all repeats except long perfect ones that are not contained inside any sequencing read and therefore cannot be resolved without double barreled data. Similarly EULER-DB (Pevzner and Tang, 2001 [6]) typically resolves all repeats that are shorter than the clone length. In a repeat graph such repeats are represented by edges with multiplicities greater than one. Multiplicities of the repeat edges define the repeat copy numbers. Figure 4(a) shows the largest connected component of the repeat graph for the *Neisseria meningitidis* (NM) sequencing project. It is not obvious which edges in this graph correspond to repeats and what are their multiplicities. Pevzner and Tang, 2001[6] described EULER-CN algorithm that find multiplicities of edges in the repeat graph by iteratively balancing the Kirchhoff flow on every vertex.

While Fig. 4(a) looks complicated, it tells us what contigs are possible neighbors and how they are oriented with respect to each other. The traditional sequence assembly algorithms do not output this information, leaving finishers in the dark during the gap closure process. Comparison of figures 4(a) and 4(b) illustrates the advantages of generating a repeat graph instead of a large set of disconnected contigs. For a set of disconnected contigs (Fig. 4(b)), a straightforward way to order them is to conduct PCR experiments for all possible pairs of contigs (*combinatorial PCR*). This results in an extensive finishing effort requiring over 30,000 PCR experiments. The repeat graph in Fig. 4(a) eliminates the need for exhaustive test of every contig pair and suggests conducting PCR experiments only for edges entering and exiting a repeat. Even such simple approach, which tests all possible pairs of edges entering and exiting a repeat one-by-one (*graph-based combinatorial PCR*), requires only 195 PCR experiments to resolve all repeats in the graph in Fig. 4(a) for NM sequencing data.

Tettelin et al., 1999 [3] suggested optimized primer pooling for multiplex PCR to minimize number of experiments for gap closing. However unlike in the case of gap closure, in the case of pseudogap closure the repeat graph provides information about the length of a repeat to be resolved, as well as entrance and exit sequences for the repeat. This information enables us to resolve all pseudogaps in sequencing data in significantly smaller number of

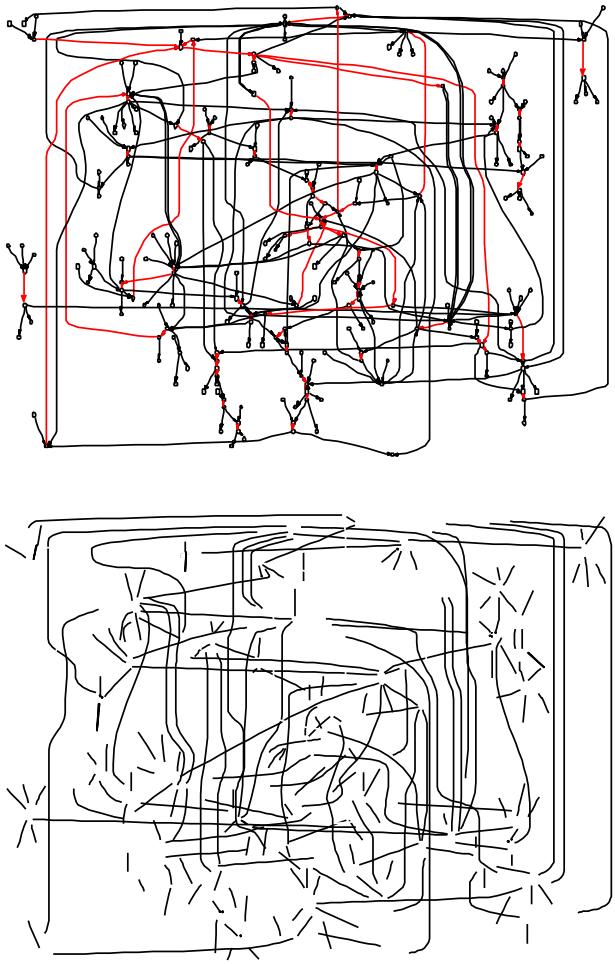

Figure 4: (a) Largest connected component of the repeat graph for *Neisseria meningitidis* project (Parkhill et al., 2001 [5]) as assembled by EULER (Pevzner et al., 2001 [9]). Red edges indicate repeats as determined by EULER-CN. (b) Masking repeats breaks the repeat graph into an unordered set of contigs.

multiplex PCR experiments as compared to resolution of gaps. EULER-PCR uses the repeat graph to select primers pools for multiplex PCR experiments. For NM project, EULER-PCR reduces number of PCR experiments to 21 reactions, which can be run concurrently.

Fig.5 shows an example of a repeat of multiplicity 3, with 3 edges entering and 3 edges leaving the repeat. In this case the sequencing reads do not provide information on which of exits X, Y, Z follow the entrances A, B, C. The sequence reconstruction requires determining 3 correct pairings among 9 possibilities: A-X, A-Y, A-Z, B-X, B-Y, B-Z, C-X, C-Y, and C-Z. This can be accomplished by generating PCR products spanning the repeat. To generate such products, one has to choose unique PCR primers on entrance and exit edges. If we are able to choose such positions for forward and reverse primers, so that *all possible* PCR products will have lengths that are sufficiently different from each other, we can deduce the correct pairings from a single multiplex reaction by measuring the PCR products lengths. Assuming a conservative estimate of the length measurement accuracy for long-range PCR product to be about 10%, the relative pairwise length differences of possible PCR products should be at least 10%. Fig.5 demonstrates that a repeat with 3 entrances and 3 exits can be resolved in a single multiplex PCR experiment using only 3 forward and 3 reverse primers. A single reaction tests all 9 possible pairings between entrance and exit edges.

Repeats can follow each other in the graph, therefore a divide-and-conquer strategy is employed to find edges on which primers to be placed. The repeat graph is partitioned into set of smaller subgraphs which contain one or more repeats as follows: if all entrance and exit edges of a repeat have unit multiplicity, such repeat, along with entrance and exit edges constitutes a single simple subgraph; if some entrance of exit edges or a repeat are repeats on their own, subgraph is expanded to include entrance and exit edges of all repeats in the subgraph until every terminal edge in the subgraph has unit multiplicity. PCR primer pairs will be placed only on edges with unit multiplicity (the terminal edges of the subgraph). This procedure minimizes the number of multiplex PCRs, as well as number of primers, necessary to resolve the repeats.

An example of a repeat subgraph from NM sequencing data is given in Fig.6. The structure of this repeat subgraph reveals that central region of the repeats overlap, while some smaller repeats are completely contained inside larger ones. Thus the repeat graph generated by EULER can provide insights into the history of duplication events in genomes.

After finding the set of repeat subgraphs, EULER-PCR selects such set of primers per reaction tube that will test maximal number of pairings between entrance and exit edges of the repeat subgraphs given the constraint on maxi-

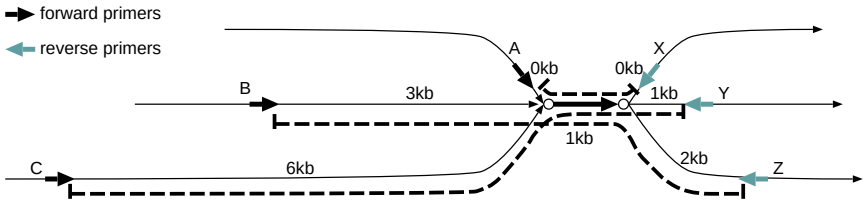

Figure 5: An example of primer placement on edges entering and exiting a repeat. Length of the repeat is 1 kb. Forward primers A, B, and C are placed on distances 0 kb, 3 kb, and 6 kb, respectively, "upstream" from the vertex starting the repeat. Reverse primers X, Y, and Z are placed on distanced 0 kb, 1 kb, and 2 kb, respectively "downstream" from the vertex ending the repeat. With such primer arrangement all nine possible PCR products will have length differing by 10% or more from each other. Lengths of PCR product between primer pairs: A-X, A-Y, A-Z, B-X, B-Y, B-Z, C-X, C-Y, and C-Z varies from 1 kb to 9 kb respectively. One possible outcome of multiplex PCR is shown by dashed lines.

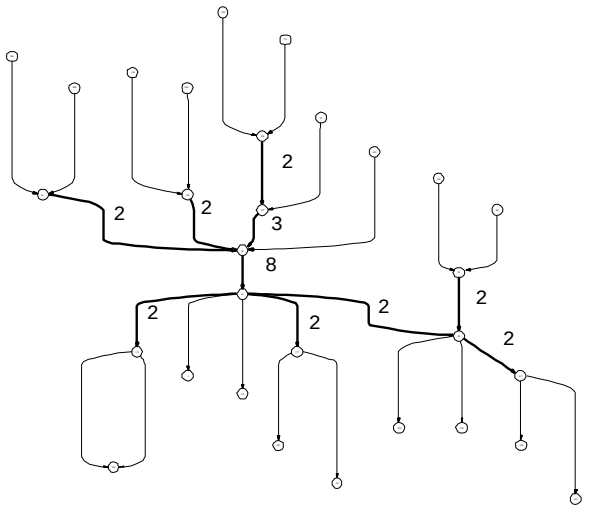

Figure 6: A complex repeat subgraph containing multiple repeats (thick edges). Numbers next to thick edges indicate repeat multiplicities. Thin edges have unit multiplicity.

Table 1: "Combinatorial PCR" and POMP columns show the number of finishing PCR experiments for gap closure. "Graph-based Combinatorial PCR" and EULER-PCR columns show the number of repeat resolution experiments.

Genome	Number of contigs before repeat resolution	Number of contigs after repeat resolution	Number of experiments			
			Combinatorial PCR	Multiplex PCR with primer pooling (POMP)	Graph-based Combinatorial PCR	Multiplex EULER-PCR
CJ	18	15	630	90	16	3
LL	53	8	5,565	490	305	24
NM	123	69	30,135	1800	195	21

mal length of PCR product L_{max}. Primers sequences chosen by EULER-PCR are of length 20 bases or longer. Those sequences are selected in accordance with standard requirements for primer selection (Haas et al., 1998 [11]) regarding uniqueness in the genomic sequence, melting temperature requirements, G or C for 3' base, etc.

Last two columns in the Table 1 show the number of repeat resolution experiments by straightforward graph-based combinatorial PCR and by optimized EULER-PCR. For comparison, though indirect, the table also presents the number of finishing experiments for gap closure by combinatorial PCR and by pipette optimized multiplex PCR (POMP) suggested by Tettelin et al.,1999. [3] The results are presented for *Campylobacter jejuni* (Parkhill et al., 2001 [12]) , *Lactococcus lactis* (Bolotin et al., 2001 [13]), and *Neisseria meningitis* (Parkhill et al., 2000 [5]) sequencing projects. First two methods deal with disconnected contigs and, while closing both gaps and pseudogaps these methods require fairly large number of experiments. For N contigs, combinatorial PCR simply tests all $\binom{2N}{2}$ primer pairs. Optimized multiplex PCR method tests for presence of PCR products between pairs of primer pools, instead of pairs of primers. The number of initial reactions using POMP with pool size $\sqrt{2N}$ if given by $\binom{\sqrt{2N}}{2}$ (Tettelin et al.,1999 [3]). However, up to $2\sqrt{2N}$ additional reactions are needed for each initial PCR product to determine which primers in the pair of pools are mates. Thus, number of POMP reactions is estimated as $\sqrt{2N}\binom{\sqrt{2N}}{2}$. Note, that even straightforward approach to conduct PCRs, using the repeat graph and choosing primers on all possible pairs of edges entering and exiting repeats, requires relatively small number of reactions to resolve repeats. Results for EULER-PCR are generated for long-range PCR setup with relative pairwise difference between PCR products $\epsilon = 0.1$, and maximal PCR product length $L_{max} = 10kb$. Average number of primers per reaction suggested by EULER-PCR is 6 for CJ, 11 for LL, and 9 for NM.

Resolving repeats before gap closure significantly reduces number of finishing experiments. This reduction is especially significant for highly repetitive genomes like *Lactococcus lactis*. Number of contigs after repeat resolution reduces from 53 to only 8 for LL, thus requiring only few gap closing experiments.

Acknowledgments

This paper would not be possible without input, both with data and with discussions, provided by Haixu Tang. We thank David Harper, Julian Parkhill and Alexei Sorokin for providing sequencing data. We also thank Uri Keich and Steffen Heber for feedback and useful discussions. This work was supported by NIH grant 1 R01 HG02366-01 NCHGR.

1. D. Gordon, C. Abajian, and P. Green. Consed: A graphical tool for sequence finishing. *Genome Research*, 8:195–202, 1998.
2. A. Sorokin, A. Lapidus, V. Capuano, N. Galleron, P. Pujic, and S. Ehrlich. A new approach using multiplex long range accurate PCR and yeast artificial chromosomes for bacterial chromosome mapping and sequencing. *Genome*, 6:448–453, 1996.
3. H. Tettelin, D. Radune, S. Kasif, H. Khouri, and S. L. Salzberg. Optimized multiplex PCR: Efficiently closing a whole-genome shotgun sequencing project. *Genomics*, 62:500–507, 1999.
4. R. Beigel, N. Alon, M. Apaydin, L. Fortnow, and S. Kasif. An optimal procedure for gap closing in whole genome shotgun sequencing. In *Proceedings of the Fifth Annual International Conference in Computational Molecular Biology (RECOMB-01)*, Montreal, Canada, April, 2001. ACM Press.
5. J. Parkhill, M. Achtman, K. D. James, S. D. Bentley, C. Churcher, S. R. Klee, G. Morelli, D. Basham, D. Brown, T. Chillingworth, R. M. Davies, P. Davis, K. Devlin, T. Feltwell, N. Hamlin, S. Holroyd, K. Jagels, S. Leather, S. Moule, K. Mungall, M. A. Quail, M. A. Rajandream, K. M. Rutherford, M. Simmonds, J. Skelton, S. Whitehead, B. G. Spratt, and B. G. Barrell. Complete dna sequence of a serogroup a strain of Neisseria meningitidis Z2491. *Nature*, 404:502–506, 2000.
6. P. Pevzner and H. Tang. Fragment assembly with double barreled data. *Bioinformatics*, 17:S225–S233, 2001.
7. J.D. Kececioglu and E.W. Myers. Combinatorial algorithms for DNA sequence assembly. *Algorithmica*, 13:7–51, 1995.
8. E.W. Myers, G.G. Sutton, A.L. Delcher, I.M. Dew, D.P. Fasulo, M.J. Flanigan, S.A. Kravitz, C.M. Mobarry, K.H.J. Reinert, K.A. R., E.L. Anson, R.A. Bolanos, H. Chou, C.M. Jordan, A.L. Halpern, S. Lonardi,

E.M. Beasley, R.C. Brandon, L. Chen, P.J. Dunn, Z. Lai, Y. Liang, D.R. Nusskern, M. Zhan, Q. Zhang, X. Zheng, G.M. Rubin, M.D. Adams, , and J.C. Venter. A whole-genome assembly of drosophila. *Science*, 287:2196–2204, 2000.

9. P. Pevzner, H. Tang, and M. Waterman. An eulerian path approach to DNA fragment assembly. *Proceedings of National Academy of Sciences*, 98:9748–9753, 2001.
10. E.S. Lander and M.S. Waterman. Genomic mapping by fingerprinting random clones: a mathematical analysis. *Genomics*, 2:231–239, 1988.
11. S. Haas, M. Vingron, A. Poustka, and S. Wiemann. Primer design for large scale sequencing. *Nucleic Acids Research*, 26:3006–3012, 1998.
12. J. Parkhill, B.W. Wren, K. Mungall, J. M. Ketley, C. Churcher, D. Basham, T. Chillingworth, R. M. Davies, T. Feltwell, S. Holroyd, K. Jagels, A.V. Karlyshev, S. Moule, M. J. Pallen, C. W. Penn, Q. A. Quail, M. A. Rajandream, K. M. Rutherford, A. H. van Vliet, S. Whitehead, and B. G. Barrell. The genome sequence of the food-borne pathogen Campylobacter jejuni reveals hypervariable sequences. *Nature*, 403:665–668, 2000.
13. A. Bolotin, P. Wincker, S. Mauger, O. Jaillon, K. Malarme, J. Weissenbach, S. D. Ehrlich, and A. Sorokin. The complete genome sequence of the lactic acid bacterium Lactococcus lactis ssp. lactis IL1403. *Genome Research*, 11:731–753, 2001.