Improving genome assemblies using multi-platform sequence data

Pınar Kavak [1,2,*], Bekir Ergüner [1], Bayram Yüksel [3] , Duran Üstek [4], Mahmut Şamil Sağıroğlu [1], Tunga Güngör [2], and Can Alkan [5,*]

[1]Advanced Genomics and Bioinformatics Research Group (İGBAM), BİLGEM, TÜBİTAK, 41470 Gebze, Kocaeli, Turkey, pinar.kavak@tubitak.gov.tr; [2]Department of Computer Engineering, Boğaziçi University, 34342 Bebek, İstanbul, Turkey
[3]Advanced Genomics and Bioinformatics Research Group (İGBAM), MAM, TÜBİTAK, 41470 Gebze, Kocaeli, Turkey
[4] Department of Medical Genetics, İstanbul Medipol University, 34810 Beykoz, İstanbul, Turkey

[5]Department of Computer Engineering, Bilkent University, 06800 Bilkent, Ankara, Turkey, calkan@cs.bilkent.edu.tr

Abstract

De novo assembly using short reads generated by next generation sequencing technologies is still an open problem. Although there are several assembly algorithms developed for data generated with different sequencing technologies, and some that can make use of hybrid data, the assemblies are still far from being perfect. There is still a need for computational approaches to improve draft assemblies. Here we propose a new method to correct assembly mistakes when there are multiple types of data obtained using different sequencing technologies that have different strengths and biases. We apply our method to Illumina, 454, and Ion Torrent data, and also compare our results with existing hybrid assemblers, Celera and Masurca.

1 Introduction

Since the introduction of high throughput next generation sequencing (NGS) technologies, traditional Sanger sequencing is being abandoned especially for large-scale sequencing projects. Although cost effective for data production, NGS also imposes increased cost for data processing and computational burden. In addition, the data quality is in fact lower, with greater error rates, and short read lengths for most platforms. One of the main algorithmic problems to analyze NGS data is the *de novo* assembly: i.e. "stitching" billions of short DNA strings into a collection of larger sequencees, ideally the size of chromosomes. However, "perfect" assemblies with no gaps and no errors are still lacking due to many factors, including the short read and fragment (paired-end) lengths, sequencing errors in basepair level, and the complex and repetitive nature of most genomes. Some of these problems in *d*e novo assembly can be ameliorated through using data generated using different sequencing platforms, where each technology has "strengths" that may be used to fix biases introduced by others.

*to whom correspondence should be addressed. pinar.kavak@tubitak.gov.tr, calkan@cs.bilkent.edu.tr

In this work, we propose to improve draft assemblies (i.e. produced using a single data source, and/or single algorithm) by incorporating data generated using different NGS technologies, and applying novel correction methods. To achieve better improvements, we exploit the advantages of both short but low-error and long but erroneous reads. We show that correcting the contigs built by assembling long reads through mapping short (and high quality) read contigs produce the best results, compared to the assemblies generated by algorithms that use hybrid data.

2 Methods

We first cloned a bacterial artificial chromosome (BAC) from human chromosome 13. We then sequenced this BAC separately using Illumina, Roche/454, and Ion-Torrent platforms. Illumina data is paired-end, where the others are single-end. The read lengths are 101bp for Illumina, 10bp-1Kbp for Rochhe/454, and 5bp-201bp for Ion Torrent. We also obtained a "gold standard" reference assembly using template-based assembly with Mira [7] with Roche/454, which is then corrected with the Illumina reads. Since Roche/454 and Ion Torrent platforms have similar sequencing biases (i.e. problematic homopolymers), we worked on two separate groups: Illumina & 454 and Illumina & Ion-Torrent, which gives us an opportunity to compare Roche/454 and Ion-Torrent.

Pre-processing: We first discarded the reads that has low average quality value (phred score 17, i.e. $\geq$2% error rate). Next, we removed the reads with high N-density (with >10% of the read consisting of Ns). We then trimmed groups of bases that seem to be non-uniform according to sequence base content. We also inevitably applied each assembler's pre-processing operations.

Assembly: We used several assembly tools: Velvet[3], a de Bruijn graph based assembler to assemble the short reads; and two different overlap-layout-consensus (OLC) assemblers: Celera [1], and SGA [2] to assemble the long read data sets (Roche/454 and Ion Torrent) separately. Finally, we also used a de Bruijn based assembler, SPAdes[4] on the long read data. We then mapped all draft assemblies to the E. coli reference sequence to identify and discard E. coli contamination due to the cloning process. At the end, we obtained one short read, and three long read assemblies.

Correction: We mapped the contigs obtained with the short reads onto the contigs generated by assembling long reads using BLAST[8]. Since BLAST may report multiple

mapping locations due to repeats, we accepted only the "best" map locations. Reasoning from the fact that the short reads show less sequencing errors, we opted for the sequence reported by the short read based contigs over the long read contigs assemblies when there are disagreements between the pair, and patched the "less fragmented" long read assemblies. We repeated this process for each of the three long read assembly data sets.

Evaluation: We mapped each of the final corrected assemblies onto the reference genome we constructed, calculated various statistics based on the comparisons, and estimated assembly qualities (Table 1). We also used two hybrid assemblers, Celera-CABOG [5] and Masurca [6] on the same data to compare our correction methodology with those of hybrid assembly algorithms.

3 Results and Conclusion

We present a summary of the results in Table 1. Briefly, the Velvet assembly using only the Illumina reads showed better coverage (99%) and high average identity (97.5%) rates compared to Celera assembly using Celera. Correcting the Celera assembly with our method improves both coverage and average identity rates, which are then further improved by reiteratively applying our method. We also observe that the hybrid assemblers used in this study did not produce better results with this data set.

Here we presented a new method to improve draft assemblies by correcting high contiguity assemblies using high quality short read contigs. However, the need to develop new methods that exploit different data properties of different NGS technologies remains.

Funding The project is supported by the Republic of Turkey Ministry of Development Infrastructure Grant (no: 2011K120020), BİLGEM - TÜBİTAK (The Scientific and Technological Research Council of Turkey) grant (no: T439000), and a TÜBİTAK grant to C.A.(112E135).

References

[1] E.W.Myers *et al* (2000) A Whole-Genome Assembly of Drosophila, *Science*, **287**:2196-2204.

[2] J.Simpson *et al* (2012) Efficient de novo assembly of large genomes using compressed data structures, *Genome Research*, **22**:549-556.

[3] D.Zerbino, E.Birney (2000) Velvet: Algorithms for de novo short read assembly using de Bruijn graphs, *Genome Research*, **18(5)**:821-829.

[4] A.Bankevich *et al* (2012) SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing, *Journal of Computational Biology*, **19(5)**:455-477.

[5] J.R.Miller *et al* (2008) Aggressive assembly of pyrosequencing reads with mates, *Bioinformatics*, **24(24)**:2818-2824.

[6] A.Zimin *et al* (2013) The MaSuRCA genome Assembler, *Bioinformatics*, **29(21)**:2669-2677.

[7] B.Chevreux *et al* (1999) Genome sequence assembly using trace signals and additional sequence information, *Computer Science and Biology:Proceedings of the German Conference on Bioinformatics (GCB)*, **99**:45-56.

[8] S.Altschul *et al* (1990) Basic local alignment search tool, *Journal of Molecular Biology*, **215(3)**:403-410.

Table 1: Results of assembly correction method on BAC data.

Name	Length	# of Contigs	# of Mapped Contigs	# of Covered bases	Coverage	Avg. Identity	# of Gaps	Size of Gaps
Reference	*176.843*							
Velvet								
Ill. Velvet	197,040	455	437	175,172	0.99055	0.97523	39	1,671
Celera								
454 Celera	908,008	735	735	172,563	0.97580	0.92599	18	4,280
Ion Celera	39,347	27	27	47,638	0.26938	0.96932	47	129,205
Corrected Celera								
Ill-454 Celera	4,945,785	895	270	176,368	0.99731	0.94370	5	475
Ill-454 Celera^{2*}	5,078,059	890	265	176,640	0.998852	0.944527	4	203
Ill-454 Celera^{3}	5,086,627	890	265	176,640	0.998852	0.944560	4	203
Ill-Ion Celera	93,909	30	28	81,819	0.46267	0.96327	36	95,024
Ill-Ion Celera^{2}	145,262	30	28	91,962	0.52002	0.97412	33	84,881
Ill-Ion Celera^{2}	216,167	30	28	99,645	0.56347	0.98066	34	77,198
SGA								
454 SGA	62,909,254	108,095	101,514	176,546	0.99832	0.97439	1	297
Ion SGA	842,997	6,417	6,122	153,092	0.86569	0.99124	197	23.751
Corrected SGA								
Ill-454 SGA	295,009	335	335	176,757	0.99951	0.96823	5	86
Ill-454 SGA^{2}	279,034	305	305	176,757	0.99951	0.96769	5	86
Ill-Ion SGA	197,509	291	291	175,052	0.98987	0.97501	45	1,791
Ill-Ion SGA^{2}	203,064	291	291	175,676	0.99340	0.97413	34	1,167
Ill-Ion SGA^{2}	204,524	291	291	175,677	0.99341	0.97405	34	1,166
SPADES								
454 SPADES	12,307,761	49,824	49,691	176,843	1.0	0.98053	0	0
Ion SPADES	176,561	110	107	167,890	0.94937	0.92909	9	8,953
Corrected SPADES								
Ill-454 SPADES	290,702	298	298	176,454	0.99780	0.96538	5	389
Ill-454 SPADES^{2}	290,917	297	297	176,454	0.99780	0.96530	5	389
Ill-Ion SPADES	198,665	52	52	171,977	0.97248	0.94215	4	4,866
Ill-Ion SPADES^{2}	200,307	52	52	172,101	0.97319	0.94230	2	4,742
Masurca								
Ill-454 Masurca	380	1	0	0	0	0	0	0
Ill-Ion Masurca	2,640	8	8	1,952	0.01104	0.98223	9	174,891
Celera-CABOG								
Ill-454 Celera	1,101,716	891	891	174,330	0.98579	0.92452	12	2,513
Ill-Ion Celera	0	0	0	0	0.0	0.0	0	0.0

Name: the name of the data group that constitute the assembly; # of contigs: the number of contigs that belong to the resulting assembly; # of Mapped Contigs: the number of contigs that successfully mapped onto the reference sequence; # of Covered bases: the number of bases on the reference sequence that are covered by the assembly; Coverage: percentage of covered reference; Avg. identity: percentage of the correctly predicted reference bases; # of Gaps: The number of gaps that cannot be covered on the reference genome; Size of Gaps: total number of bases on the gaps.

* "2" represents the results of the second cycle of correction, "3" represents the third cycle.