J. Microbiol. Biotechnol. (2006), **16**(5), 683–688

JOURNAL OF MICROBIOLOGY AND BIOTECHNOLOGY

FASIM: Fragments Assembly Simulation using Biased-Sampling Model and Assembly Simulation for Microbial Genome Shotgun Sequencing

HUR, CHEOL-GOO[1,3], SUNNY KIM[1], CHANG HOON KIM[1], SUNG HO YOON[1], YONG-HO IN[2], CHEOLMIN KIM[3], AND HWAN GUE CHO[3]*

[1]*Division of Genomics and Proteomics, Korea Research Institute of Bioscience and Biotechnology (KRIBB), Daejon 305-333, Korea*
[2]*Bioinfomatix, Inc., Nam Chang Bldg, 748-16, Yeoksam-dong, Gangnam-gu, Seoul 135-925, Korea*
[3]*Bioinformatics Cooperative Course, Pusan National University, Pusan 609-735, Korea*

Received: March 29, 2005
Accepted: June 12, 2005

Abstract We have developed a program for generating shotgun data sets from known genome sequences. Generation of synthetic data sets by computer program is a useful alternative to real data to which students and researchers have limited access. Uniformly-distributed-sampling clones that were adopted by previous programs cannot account for the real situation where sampled reads tend to come from particular regions of the target genome. To reflect such situation, a probabilistic model for biased sampling distribution was developed by using an experimental data set derived from a microbial genome project. Among the experimental parameters tested (varied fragment or read lengths, chimerism, and sequencing error), the extent of sequencing error was the most critical factor that hampered sequence assembly. We propose that an optimum sequencing strategy employing different insert lengths and redundancy can be established by performing a variety of simulations.

Key words: Fragments assembly simulation, sampling model, genome, shotgun sequencing

With the advent of the shotgun sequencing approach, genome sequence data are being rapidly accumulated. As for microorganisms, 157 complete genomes have been published as of March 2004. Whole genome shotgun sequencing (WGSS) is to sequence randomly generated fragments of an entire genome, and then to assemble the sequences by computer program [13]. In addition the study of function based on microbial genomes is dynamic in progress [6, 8, 10, 11]. Availability of a sequence assembly program and well-designed shotgun approach can considerably reduce the cost and time for genome projects. *In silico* simulation of entire sequencing steps such as base calling from chromatograms, vector masking, and sequence assembly can be extremely helpful in understanding each step and in designing the experiments with its predictive power. However, this "virtual experience of genome project" is hampered by the limited availability of real experimental data sets.

Generation of random short fragment sequences by computer program can be a useful alternative to the real data set. To the best of our knowledge, only two programs have been reported to date: GenFrag 2.1 [2] and celsim [12]. GenFrag was developed to fragment and mutate a DNA sequence for testing assembly algorithms. Celsim can generate synthetic shotgun data sets, allowing repeat structures and polymorphic variants from the real DNA sequences. However, in developing assembling tools and simulating WGSS, there still remains the need for generating various data sets by considering many experimental factors such as chimerism, sequencing error, distribution of fragment, or read length. Specifically, when generating random fragments from known sequences, assumption of uniformly sampled clones can lead to unrealistic results. This is caused by the observation that biased clone distribution always happens because of various reasons: incomplete fracturing of genome, biased insertion of fragments into vectors, and improper cloning of some insert/vector combinations [13].

In this study, we developed a program suite for generating random fragments in microbial genome sequencing. To obtain a more realistic program, a probabilistic model of biased sampling distribution was applied to generate synthetic shotgun data sets from known genome sequences. We also report the effects of various experimental factors on the sequence assembly.

*Corresponding author
Phone: 82-51-510-2283; Fax: 82-51-515-2208;
E-mail: hgcho@pusan.ac.kr

MATERIALS AND METHODS

Generation of a Probabilistic Model for Biased Random Sampling

A probabilistic model for biased sampling distribution was developed by using an experimentally derived data set that was constructed from the genome project of *Mannheimia succiniciproducents* MBEL55E (accession no. AE016827, unpublished). This strain is a novel succinic acid-producing bacterium, which was isolated from bovine rumen [7]. The chromosome consists of 2,314,078 bp, which were cloned into 18,958 of 2-kb clones and 294 of 40-kb clones. As shown in Fig. 1, the bacterial chromosome was scanned for determining the start position of each read by the BLASTN program [1]. The cumulative distribution function for biased sampling, F(x), was defined as

$$F(x)=\frac{1}{N}\sum_{i=0}^{x} f(i) \qquad (0\le x\le \text{genome size}) \quad (1)$$

where f(i) is the number of reads positioned at the ith position, and N is the total number of reads. Once a random number between zero and one is selected, the read position corresponding to that number is traced from F(x). Starting from a generated position, inserts having various lengths such as 2 kb, 40 kb, or 150 kb can be taken out from the published genome sequence.

Development of a Data Set Generation Program and Sequence Assembly

The program was written in C and runs as a UNIX or LINUX command line tool. It was designed to generate sequence files of a shotgun data set from known genome sequences, and the procedures are outlined in Fig. 2. Among the options allowing users to control the read sequences are the parameterization of normal distribution of fragments and/or read lengths, and degree of chimerism and sequencing error rate. Two file names, genome sequence file and vector sequence file, are arguments with several parameter settings (Fig. 2). The positions of every fragment are generated from a derived probabilistic model (see Materials and Methods). The information file containing the inserts distribution of length and position of inserts is also provided. After reads were generated from the ends of inserts, a vector sequence was added to the 5' and 3' ends of each read sequence. At this step, chimeric reads can be generated by combining two randomly selected reads.

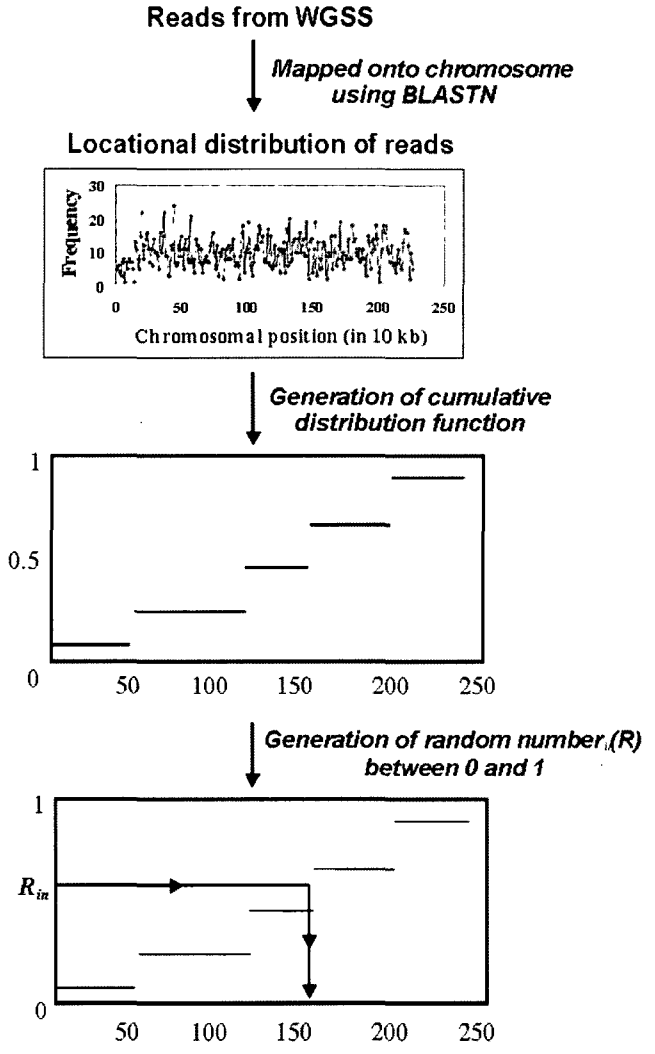

Fig. 1. Generation of biased-distributed data from known a genome sequence.

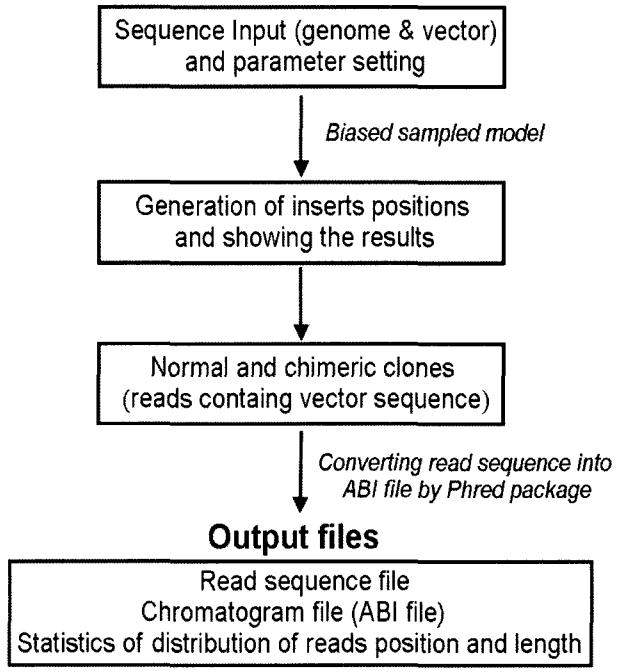

Fig. 2. Flow diagram of the data set generation program.

pGEM3Zf was used as a vector sequence for the 2-kb clone and pEpiFOS-5 for the 40- and 150-kb clones.

By using the Phred package [3] (http://www.phrap.org), the resulting read sequences were converted into ABI files, and then, base calling from these files was done. After the vector sequence was masked, read sequences were assembled into contigs (overlapped fragments in a contiguous region) and scaffolds (maximally linked and ordered set of contigs) by CAP3 [4].

RESULTS

Distribution of Fragments Generated by the Program

To test the performance of the program, we generated sequencing reads uniformly from a published genome sequence, *Pasteurella multocida* [15] (accession no. NC_002663), which is a circular genome having 2,257,487 bp (G). The simulation condition was given as insert length of 2 kb and read length of 600 bp (L). No experimental errors were assumed. The contigs were generated by assembling the reads with the setting of minimal overlapped length between reads (T) as 30 bp. Theoretically, if all the reads of the same size are uniformly sampled from the genome, the expected number of contigs can be calculated from

$$Ne^{-N\frac{(L-T)}{G}} \quad (2)\ [5]$$

where N is the number of sequenced reads.

For the clarity of explanation, we defined the term "redundancy" as the ratio of the summed length of total sequence reads to the total genome size, and "coverage" as the percentile of genome that can be covered by sequencing effort at some redundancy. As shown in Fig. 3A, the summed number of contigs and singlets, which were reads left from contig assembly, was perfectly matched to Eq. (2) as redundancy went. This simulation showed that the developed program performs well in generating shotgun data sets. In theory, uniformly sampled reads with no experimental errors can guarantee 99% coverage with eight-fold redundancy

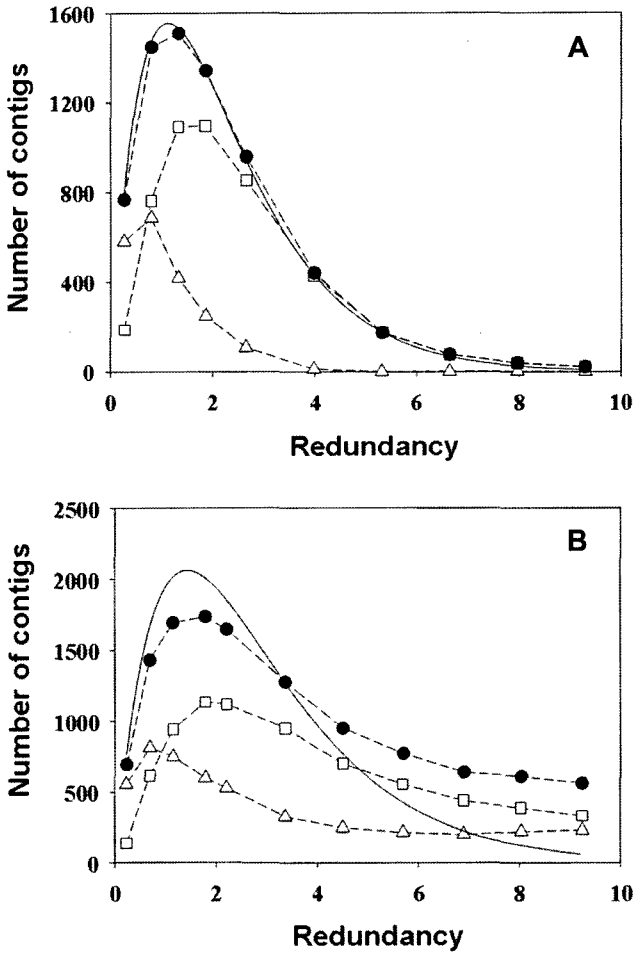

Fig. 3. Redundancy profiles of contig number generated by the program (**A**) and from the *Manheimia* genome sequencing project (**B**).
--□-- contigs; --△-- singlets; --●-- contigs+singlets; solid line for Eq. (2).

[14]. However, such a situation hardly occurred in real WGSS, especially for a bacterial genome having a high G+C content. Figure 3B implies the large discrepancy in contig numbers from an ideal sequencing project and the

Table 1. Combinations of inserts size with various redundancies and their effects on sequence assembly. Data sets were derived from a biased-sampled model.

Total redundancy	Redundancies of different sized inserts (2 kb/40 kb/150 kb)	No. contigs	No. scaffolds	Genome coverage (%)
3X	3X/0X/0X	994	250	74.82
5X	5X/0X/0X	735	165	85.40
	4X/1X/0X	653	126	90.72
	3X/1X/1X	642	120	91.34
	3X/0X/2X	536	109	93.62
7X	7X/0X/0X	584	121	89.54
	6X/1X/0X	454	96	93.35
	5X/1X/1X	447	83	94.57
	5X/0X/2X	360	71	95.93

real *Manheimia* WGSS, and addresses the requirement of a more realistic biased clone distribution for the simulation of sequence assembly.

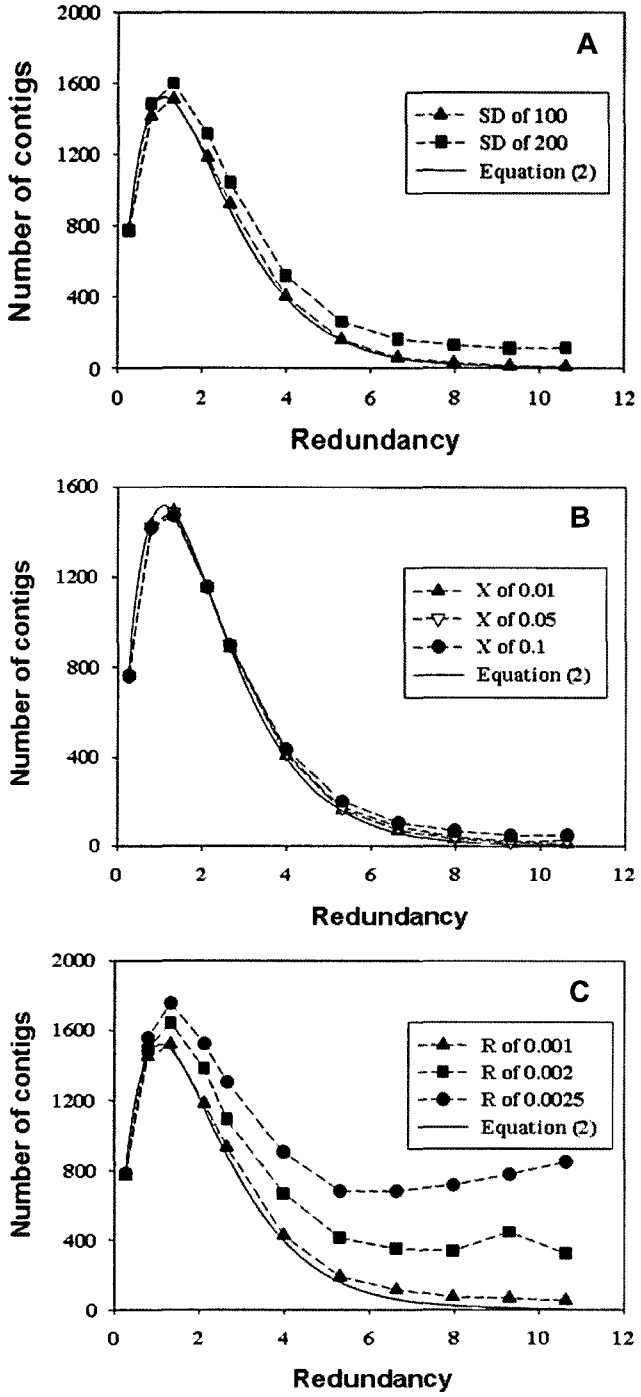

Fig. 4. Effects on sequence assembly by read length deviation (**A**), chimerism (**B**), and sequencing error rate (**C**).
Mean read length was set to 600 bp, and unless otherwise mentioned, parameters were set to zero. Solid line is for Eq. (2). Abbreviations: SD, standard deviation of read length; X, rate of chimerism; R, sequencing error rate.

Effect of Sequencing Strategy on Sequence Assembly

Redundancy of shotgun sequencing and the choice of insert length are key factors in reducing scaffolds [14]. To determine the effect of sequencing strategy on assembly, combinations of these two factors were simulated and the results are compared in Table 1. The data sets consisting of 2-, 40-, and 150-kb inserts were generated based on a biased-sampled model, for the purpose of simulating a more realistic situation. Different redundancy combinations of each insert length largely affected the number of contigs and scaffolds, and genome coverage. Use of longer inserts resulted in fewer scaffolds and larger coverage when the total redundancy was fixed. At 5× total redundancy, 2-kb inserts with 5× redundancy were sequenced to produce 165 scaffolds corresponding to an 85.4% coverage. The number of scaffolds rapidly decreased as the proportion of longer inserts increased, and the combination of 150-kb inserts with 2× redundancy and 2-kb inserts with 3× redundancy gave the most efficient assembly of 109 scaffolds and 93.62% coverage. The same observation was found at sequence assembly with 7× total redundancy.

Effect of Experimental Conditions on Sequence Assembly

Simulations were carried out to find out the effects of experimental conditions on sequence assembly (Fig. 4). By randomly sampling from the *Pasteurella multocida* genome [15] sequence, we generated 2-kb inserts allowing various experimental conditions. The simulation results were compared with Eq. (2), which assumed uniformly sampled reads with no experimental errors. Distribution of read lengths with a standard deviation (SD) of 100 bp gave no difference (Fig. 4A). However, an SD of 200 bp had an effect on sequence assembly, and 115 contigs were remained even at the 9.3× redundancy. This implies that it is critical to prepare libraries with a narrow deviated insert length. In contrast to varied read length, the variation in fragment length resulted in no difference (data not shown). Chimerism made little difference and 10% of chimerism resulted in slight deviation. Considering that 10% of chimerism is hard to achieve in real experiments, chimerism seems to be a minor factor in WGSS. Among the experimental errors tested, the degree of sequencing error was the most critical factor in sequence assembly (Fig. 4C). Large deviation from Eq. (2) was found when the sequencing error was assumed to be above 0.1%. Interestingly, the contig number was increased even at the higher redundancy, when sequencing error was above 0.25%.

DISCUSSION

Although the advent of sequencing technology makes WGSS easy to do, a genome project still costs a huge amount of money and time. Therefore, it is essential to establish

an optimal sequencing strategy before getting into the sequencing. In this study, we have developed a data set generation program for microbial WGSS. In contrast to uniformly-sampled clones that were adopted by previous programs, we applied a biased distributed sampling model to our program. Sampled reads biased to come from a particular region of the target sequences frequently occur in WGSS [13]. This can happen by incomplete fracturing of the genome, biased insertion of fragments into vectors, and improper cloning of some insert/vector combinations. To date, the data set generators [2, 12] were designed based on uniformly sampled model. However, as shown in Fig. 3B, this assumption did not hold in real experiments and a more realistic simulation is possible when sequence reads are generated from a biased distributed probabilistic model. This perspective was supported by the simulation result that the number of scaffolds was not reduced when the proportion of larger inserts were contained in the data set that were generated from a uniformly sampled clone model (data not shown).

Based on the biased sampling model, we generated various data sets consisting of different redundancies of different sized inserts and carried out sequence assemblies (Table 1). When the total redundancy was 7× and 8×, the smallest number of scaffolds and largest coverage were achieved when 150-kb inserts with 2× redundancy were contained in the data set. This result is consistent with a previous finding that among one type of inserts, fewer scaffolds resulted when longer insert lengths were used [15]. This is due to the fact that the possibility of spanning a larger region between contigs can be enhanced by using longer inserts. However, it should be noted that there is practical limitation to using a large insert size. This limitation is caused by the experimental difficulty of sequencing the ends of long inserts and preparing a large sized library such as cosmid and BAC (bacterial artificial chromosome) clones, and can bring a sharp rise in sequencing cost. Therefore, sequence redundancy, insert length, and their combination should be determined by considering the economic balance. For example, in Table 1, although its total redundancy is high, 5X/1X/1X of 2 kb-, 40 kb-, and 150-kb inserts can cost less than 3X/0X/2X because of the reduced preparation of BAC libraries for 150-kb inserts.

It has been widely accepted and an open secret that the quality of sequencing ultimately determines the success of WGSS. However, there was no systematic study to support this idea. Among the experimental parameters tested (varied insert and read length, chimerism, and sequencing error), sequencing error was the most critical factor in hampering sequence assembly. When the read sequences contained a sequencing error rate of 0.1%, the contig number started to deviate from that of the Eric-Lander Eq. (2). Specifically, sequence assembly with a data set containing 0.25% sequence error gave a divergent pattern of contig number as redundancy went (Fig. 4C). The type of error (substitution, insertion, or deletion) possibly affects the assembling result. However, the increased proportion of insertion and deletion rate led to a slightly increased contig number (data not shown).

Development of sequence assembly related tools can be benefited from a systematically controlled data set by testing its algorithm and capability. Allowing users to vary parameters such as distribution of fragment length and read length, sequence error rate, and chimerism rate facilitates WGSS by establishing an optimal strategy. We believe that students and researchers in microbiology, who have limited access to WGSS, can take advantage of our program for enhancing their understanding in genomics. The detailed information of the software is accessible at http://plant.pdrc.re.kr:7777/fasim/, and a current release of the software can be available upon request to hurlee@kribb.re.kr.

Acknowledgments

We thank the *Manheimia succiniciproducents* MBEL55E genome sequencing team - Professor Sang Yup Lee at KAIST, Bioinfomatix Inc., and GenoTech Corporation for providing the genome sequences. We appreciate Dr. X. Huang the at Michigan Univ. and Dr. P. Green for providing CAP3 and Phred/Phrap package, respectively. This work was supported by the 21C Frontier Microbial Genomics and Applications Center Program (grant MG02-0402-001-1-0-0) from the Ministry of Science and Technology (MOST) of the Republic of Korea.

REFERENCES

1. Altschul, S. F., T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. 1997. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* **25:** 3389–3402.
2. Engle, M. L. and C. Burks. 1994. Artificially generated data sets for testing DNA sequence assembly algorithms. *Genomics* **16:** 286–288.
3. Ewing, B., L. Hillier, M. Wendl, and P. Green. 1998. Base-calling of automated sequencer traces using Phred. I. Accuracy assessment. *Genome Res.* **8:** 175–185.
4. Huang, X. and A. Madan. 1999. CAP3: A DNA sequence assembly program. *Genome Res.* **9:** 868–877.
5. Lander, E. S. and M. S. Waterman. 1988. Genomic mapping by fingerprinting random clones: A mathematical analysis. *Genomics* **2:** 231–239.
6. Lee, D.-H., W. J. Jun, J. W. Yoon, H. Y. Cho, and B. S. Hong. 2004. Process strategies to enhance the production of 5-aminolevulinic acid with recombinant *E. coli*. *J. Microbiol. Biotechnol.* **16:** 1310–1317.
7. Lee, P. C., S. Y. Lee, S. H. Hong, and H. N. Chang. 2002. Isolation and characterization of a new succinic acid-producing

bacterium, *Mannheimia succiniciproducens* MBEL55E, from bovine rumen. *Appl. Microbiol. Biotechnol.* **58:** 663–668.
8. Lim, S. Y., K. H. Yong, and S. Y. Ry. 2005. Analysis of *Salmonella* pathogenicity island 1 expression in response to the changes of osmolarity. *J. Microbiol. Biotechnol.* **15:** 175–182.
9. Kim, H. W., K. M. Kim, E. J. Ko, S. K. Lee, S. D. Ha, K. B. Song, S. K. Park, K. S. Kwon, and D. H. Bae. 2004. Development of antimicrobial edible film from defatted soybean meal fermented by *Bacillus subtilis*. *J. Microbiol. Biotechnol.* **14:** 1303–1309.
10. Kang, S. A., J. C. Lee, Y. M. Park, C. Lee, S. H. Kim, B. I. Chang, C. H. Kim, J. W. Seo, S. K. Rhee, S. J. Jung, S. M. Kim, S. K. Park, and K. I. Jang. 2004. Secretory production of *Rahnella aquatilis* ATCC 33071 levansucrase expressed in *Escherichia coli*. *J. Microbiol. Biotechnol.* **14:** 1232–1238.
11. May, B. J., Q. Zhang, L. L. Li, M. L. Paustian, T. S. Whittam, and V. Kapur. 2001. Complete genomic sequence of *Pasteurella multocida*, Pm70. *Proc. Natl. Acad. Sci. USA* **98:** 3460–3465.
12. Myers, G. 1999. A dataset generator for whole genome shotgun sequencing. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* pp. 202–210.
13. Myers, G. 1999. Whole-genome DNA sequencing. *Comput. Sci. Eng.* **1:** 33–43.
14. Pop, M., S. Salzberg, and M. Shumway. 2002. Genome sequence assembly: Algorithms and issues. *IEEE Computer* **35:** 47–54.
15. Roach, J. C., C. Boysen, K. Wang, and L. Hood. 1995. Pairwise end sequencing: A unified approach to genomic mapping and sequencing. *Genomics* **26:** 345–353.