A Program for Rapid Gap Closure Employing BLAST Organism Cross Comparison Against Multiple References

Lucas B. Chaney

Departments of Computer Science and Biology, Hiram College, Hiram, Ohio 44234 USA

Submitted August 13, 2004 in conclusion of the Bioinformatics & Bioengineering Summer Institute to Director Dr. J. Elhai, Virginia Commonwealth University

ABSTRACT

Motivation: The gap closure phase of prokaryotic genome sequencing projects is typically lengthy and labor intensive. Methods for gap closure that take advantage of DNA sequences in public databases are gaining in utility as the number of available sequences increases. We have developed a program called BOX-C (for Blast Organism (X)cross Comparison) to align contigs in the *Streptococcus sanguis* genome through comparison to the NCBI “nr” database. The program consists of a Perl application utilizing BioPerl, the NCBI Blast program and database, and the Primer3 program to automate the tasks of contig processing, query submission, blastn result analysis, alignment prediction, and primer design.

Results: This method identified a number of correct alignments in the *S. sanguis* genome not suggested by other methods, including comparisons conducted against single genomes of related species. This unique approach combines nucleotide comparison against multiple reference sequences, ease of use, user configurability, and multiple-tool integration, and can be easily adapted for use with other sequencing projects.

Availability: This program is available free of charge for academic or other non-profit use. Visit http://hgi.hiram.edu/ to download the latest version.

Contact: chaneylb@hiram.edu

Supplementary Materials: Primer3 for Microsoft Windows is available upon request.

INTRODUCTION

The falling costs of extracting DNA sequence has resulted in an increasing number of institutions involved in whole genome sequencing efforts. With this in mind it has been recognized that the number, quality, accessibility, and ease-of-use of fully automated programs available for assisting at each step in a sequencing effort will greatly affect how monetarily and chronologically expensive any specific step in the process is. Analysis of the genome sequencing and preparation process reveals the following steps as computationally intensive: assembly, gap closure, finishing, and annotation. It is important to note that increasing discrimination is being made between gap closure, or the determination of unsequenced portions of the genome remaining after a high shotgun coverage is obtained, and finishing, the minimization of errors in the consensus sequence. This separation is essential in discussing the tools which seek to deal with these problems as they each possess unique sources and require solutions which can vary greatly.

The futility of high shotgun coverage as a method for gap closure has been demonstrated in a recent mathematical categorization (Wendl and Yang, 2004), providing confirmation of previous observations that specific sections of any genome are likely to be difficult or impossible to clone (Herron-Olson *et al.*, 2003). The source of these gaps can be attributed to a number of problems, which may be biological, statistical, and/or computational in their origin. Host toxicity, probabilistic non-representation, fragment rearrangement, misassembly of repeat sequences, and secondary structure inhibition of sequencing reactions are recognized as the principal sources of such gaps. This long list of roadblocks helps explain why the gap closure phase is the most expensive and more importantly the most time consuming step in many sequencing projects, and highlights the need for a number of directed methods in gap closure.

While a number of biological methods, statistical tools, and computer programs have been made available over recent years, no single technique exists that can claim to be the most efficient in all conditions. We felt a further approach offering substantial benefit would be to develop a cross-platform compatible application requiring minimum configuration and minimum technical background to use. The tool should seek integration of the maximum number of steps in the process of going from assembly program output to ready-for-sequencing PCR product and should allow customization of as many aspects as possible. The tool should also integrate well with other methods such as genome walking and multiplex PCR.

Like previous work in the field, sequence comparison would be used as a tool for contig ordering. However, this application would generate ordered and directed pairs from the ends of two contigs not relying on comparison with the genome of a related species, but rather sufficient homology with any known sequence.

RELEVANT WORK

In examination of the currently available tools, it was noted that varying levels of automation can be applied to gap closure. For example, some techniques have been developed which are automated but contain no components of decision making. One such example is multiplex PCR, a process which was initially developed to work with genes of known sequence (Burgart *et al.*, 1992) and recently modified to serve as a tool in genome closure (Tettelin *et al.*, 1999). Another such method is read pair identification—the use of additional information stored by automatic sequencers about the origin of each read that in combination with assembly programs has been used to suggest contig alignment along gaps (Frohme *et al.*, 2001, Gordon *et al.*, 2001).

Some methods are also not highly suited to automation. One such method is the well known approach of physical mapping with restriction enzymes (Soulston *et al.*, 1988). Though still in use today (Weinel *et al.*, 2001), the method is time consuming, expensive, and may grow less useful as larger genomes are sequenced. Another older approach is that of PCR extension (Shymala and Ames, 1989) which remains in use today, with some modification (Carraro *et al.*, 2003). This method—often referred to as genome walking, primer walking, PCR walking, or simply “walking”—is so straightforward by itself that little computational optimization can occur.

Apart from the enhancement of established techniques, the bulk of recent development seeks to not only automate, to some extent, the gap closure process, but does so through methods that involve prediction of contig order and orientation. It should be noted that one characteristic shared by these tools, as seen in Table 1, is their reliance on the existence of some similar known sequence. This fact, coupled with the exponential growth in recent years of the number of known sequences, explains why they have only of late become significantly useful. This methodology of approaching the problem is in some ways an especially advantageous one, as it has long been observed that any tool implementing this strategy will only grow more helpful as the number of known sequences increases (Frangeul *et al.*, 1999).

The substantial limitations in the previously available tools can be classified into two categories. One group, consisting of NUCmer, MGView, Projector, and CAAT-Box, suffer from an inability to detect any pairs absent a single complete genome of a similar organism. The other tools, GMPTB and PGAAS, require similarity of a portion of the contig end, translated as a protein, to known protein sequence. A clear avenue for further research would be in optimizing searches for detecting meaningful hits in many different reference organisms, possibly occurring outside of protein coding regions. Additionally, the output of most of the programs requires some interpretation before primer design occurs, increasing the chance of human error.

ALGORITHM

The program that I have written to address the shortcomings mentioned above is called BOX-C for **B**LAST **O**rganism (**X**)cross – **C**omparison. Output of sequences from an assembly program is used as input to the BOX-C program. To increase compatibility, specific formatting of the input file is not required, so long as each contig is given a unique identifier and some widely recognized standard file format is used in conjunction with an appropriate file extension. Preprocessing of the contigs occurs through the removal of a user specified excluded region of nucleotides from the extreme contig ends (to allow for trimming of typically low quality end sequences if desired) and the generation of a set of left and right ends of user defined size. The program uses a well recognized alignment algorithm, BLASTN (Altschul *et al.* 1990), in order to find regions of homology between the contig ends and known sequences. Each

Table 1. Analysis of software intended for automation of Gap Closure

Application Name	Notes on methodology Notes on output format(s)	Publication
GMPTB	Largest ORF in each contig end BLASTP comparison to protein database Text output only	Frangeul *et al.*, 1999
NUCmer	Whole contigs suffix tree aligned to single reference genome Text output only	Delcher *et al.*, 2002
PGAAS	Contig ends each undergo BLASTX comparison to protein database Graphical output in form of sequence alignment shown with actual sequence	Zhou *et al.*, 2002
MGView	Whole contigs BLASTN aligned to single reference genome Graphical output in form of PDF files	Herron-Olson *et al.*, 2003
Projector	Trimmed contig ends BLASTN aligned on template genome, followed by contig centers BLASTN aligned between ends or alignment discarded Graphical output in form of scalable vector graphics; primer design files	van Hijum *et al.*, 2003
CAAT-Box	Contig ends BLASTN or BLASTX aligned to single genome HTML text tables	Frangeul *et al.*, 2004

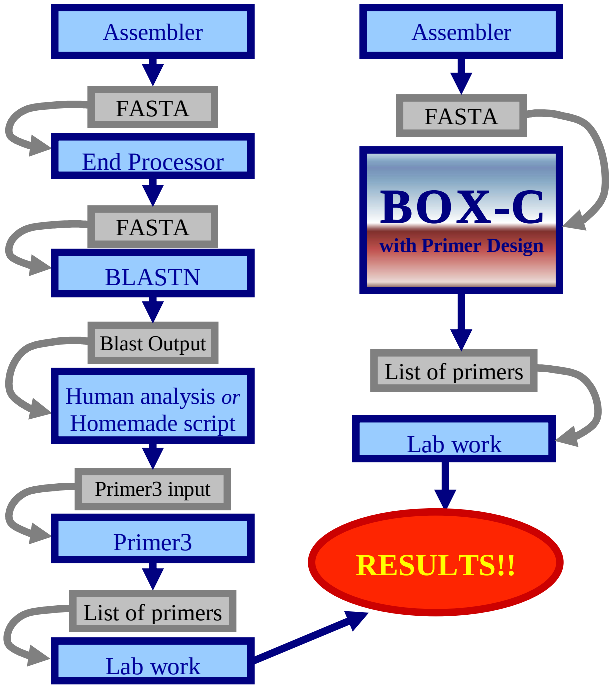

Fig. 1. The principle of BOX-C as demonstrated through comparison to the same method as performed before the program became available.

reference sequence that is hit more than once by end sequences has the hits parsed. This parsing ensures that pairs of matches meeting certain compatibility and user specified range criteria are stored for subsequent primer design. Primer3 is used to accomplish the generation of oligonucleotides targeting each of the specific gaps between the suggested alignments. Production of the primers, conduction of the PCR reaction, and sequencing of the reaction product—all activities which cannot be completed or accurately simulated *in silico*—are all that stand in the way of closed gaps. Figure 1 provides a graphical comparison of the individual steps to those preformed by the program. As can be clearly seen, the program has successfully integrated a number of tedious steps and much file processing behind a single interface.

IMPLEMENTATION

The program, written in Perl, makes extensive use of the freely available BioPerl modules (http://www.bioperl.org/). The primary factors in this decision were the portability of Perl code from Windows to UNIX, as well as the independent maintenance of BioPerl code to help insure future compatibility. Two versions are currently available which support either local BLAST or remote (NCBI) BLAST, and require a locally installed copy of Primer3. The possibility of remote Primer3 support is currently being investigated.

Running remote BLAST through NCBI ensures that the database against which the search occurs is up to date. However, the large volume of queries this method generates makes the use of local BLAST a powerful option for increasing overall speed. The inclusion of Primer3 is necessary for the maximum possible degree of integration.

A project to develop a web server for BOX-C is also currently underway. This server would perform BLAST and Primer3 primer design for submitted queries. This would allow users to upload their current assemblies, select a set of predetermined parameters, custom tailor these settings to suit their needs, and leave the program to generate suggested alignments in only a few hours, or overnight.

Until such a server can be made available, users have access to the currently available, stand-alone version of BOX-C which is highly user configurable and is designed to be nearly self explanatory upon initial execution. Future improvements, such as the creation of a separate function for configuration, will allow users to modify the default values of many parameters with a single command. (See Table 2.)

While BOX-C performs a similar function to currently available programs, it excels in many areas. BOX-C is most notably advanced in its comparison of contig sequences at a nucleotide level to most known sequences in one step. This technique has allowed local alignments of two contigs around a gap to be suggested in the *S. sanguis* genome by hits to sequences as distantly related as those from *Homo sapiens*. While the biological significance and ultimate accuracy of predictions based on such distant relationships has thus far been minimal, their utility in gap closure can still be found in their ability to highlight a contig whose end sequence is possibly a contaminant or vector sequence.

Another key feature is the high degree of configurability. This allows the user to determine the best settings for his particular organism and current degree of coverage, thus minimizing the number of missed or wrongly suggested gaps. As coverage increases and the number of predicted gaps decreases, users can relax restrictions on gap size to take into account large-scale rearrangement from known sequences; relax restrictions on e-values so that similar sequences outside of strong selection can be discovered; increase the allowed overlap, permitting BLAST to suggest contig alignments that may be missed because of misassembly upstream of the end; or perform almost any other optimization they have discovered.

Unlike most related programs, BOX-C also attempts to go beyond identification of contig pairs, taking the additional measure of suggesting primers that might be used in conjunction with PCR to cross the putative gap. This alleviates the possibility that the user might incorrectly interpret the results and allows for further automation of the process.

The program allows intermediate files to be output at many stages in the process, allowing the user to observe, at any convenient time, what data have been

Table 2. A list of BOX-C parameters with explanation of the function of each

Parameter identifier	Default value (if present)	Type of argument expected	Description of function
-h		None	If present, a help message is displayed
-v	*true*	"0" or none	Verbose output of debugging information
-i[1]		Path[2]/filename	Location of the contig file; *optional*- may be included a second time with location of intermediate BLAST results file
-f	ps	ebmps	Outputs files corresponding to each given letter: e: ends file generated in preprocessing b: BLAST results as individual files m: matches, or paired contigs, found by the program p: primer pairs for each match s: summary of where the matches were found
-o	box-c.fasta	Path[2]/filename	Basic filename given to output files, will be slightly modified for output of each specific file
-r	*false*	None	If passed as an unvalued argument, ends generated by contigs where the ends are not entirely unique will not be considered
-e	500	Integer[3]	Size in base pairs of ends taken for comparison
-y	0	Integer[3]	Size in base pairs of contig end-regions excluded before ends of the specified length are taken
-l	0	Integer[3]	Number of base pairs which can overlap on a BLAST-matched sequence before match is excluded as something which should have been identified by the assembler
-g	3000	Integer[3]	Maximum number of base pairs which separate two possible contig ends in order for ends to be considered a likely pair
-t	1e-20	BLAST-style e-val	Threshold of BLAST hits to consider for alignment
-d	nr	BLAST database	BLAST database against which to search (see http://www.ncbi.nlm.nih.gov/BLAST/ for a list)
-w	15	Integer	Time in seconds to wait between requests to NCBI BLAST server (minimum of 5 seconds)

[1]-must be given and be the location of the file of contigs to be processed
[2]-optional
[3]-must be non-negative

used, and grants the opportunity to verify the results the program generates. However, none of the output is mandatory, avoiding the waste of space that occurs with the generation of unnecessary files.

DISCUSSION

The demonstrated ability of BOX-C to suggest meaningful alignments helps categorize the program as a success. Despite some issues with configuration of BioPerl and Primer3, the program's general ease of use and cross-platform portability should greatly increase the accessibility to groups in need of such a product.

In tests conducted with the *S. sanguis* genome, the program was found to be most useful if applied to a genome assembly for which genome walking primers are being designed. The BOX-C method relies on only one specific primer per contig end, allowing users to reduce the overall number of primers by trying BOX-C suggested pairs before designing a second primer for genome walking. Since the primer locations and sizes can be specified by the user, the primers for any contig ends not successfully closed can be used in the genome walking reaction, resulting in reduced primer costs to the user with the added benefits of generally much lower PCR and sequencing reaction costs.

The ability of BOX-C to suggest alignments with a moderate success rate, while integrating with traditional approaches such as genome walking, makes it a valuable tool suitable for immediate public use. The continued integration of currently available methodologies in future software applications will be essential for the long term success of these programs.

REFERENCES

Altschul,S.F., Gish,W., Miller,W., Myers,E.W. and Lipman,D.J. (1990) A basic local alignment search tool. *J. Mol. Biol.*, **215**, 403-410.

Andrade,M.A., Brown,N.P., Leroy,C., Hoersh,S., Daruvar,A., Reich,C., Franchini,A., Tamames,J., Valencia,A., Ouzounis,C. and Sander,C. (1999) Automated genome sequence analysis and annotation. *Bioinformatics*, **15**, 391-412.

Bonfield,J.K., Smith,K.F. and Staden,R. (1995) A new DNA sequence assembly program. *Nucleic Acids Res.*, **23**, 4992-4999.

Burgart,L.J., Robinson,R.A., Heller,M.J., Wilke,W.W., Iakoubova,O.K. and Cheville,J.C. (1992) Multiplex polymerase chain reaction. *Mod. Pathol.*, **5**, 320-323.

Carraro,D.M., Camargo,A.A., Salim,A.C., Grivet,M., Vasconcelos,A.T. and Simpson,A.J. (2003) PCR-assisted contig extension: stepwise strategy for bacterial genome closure. *Biotechniques*, **34**, 626-632.

Couronne,O., Poliakov,A., Bray,N., Ishkhanov,T., Ryaboy,D., Rubin,E., Pachter,L. and Dubchak,I. (2003) Strategies and tools for whole-genome alignments. *Genome Res.*, **13**, 73-80.

Dear,S. and Staden,R. (1991) A sequence assembly and editing program for efficient management of large projects. *Nucleic Acids Res.*, **19**, 3907-3911.

Delcher, A. L., Phillippy, A., Carlton, J. and Salzberg, S. L. (2002) Fast algorithms for large-scale genome alignment and comparison. *Nucleic Acids Res.*, **30**, 2478-2483.

Frangeul,L., Glaser,P., Rusniok,C., Buchrieser,C., Duchaud,E., Dehoux,P. and Kunst,F. (2004) CAAT-Box, contigs-assembly and annotation tool-box for genome sequencing projects. *Bioinformatics*, **20**, 790-797.

Frangeul,L., Nelson,K.E., Buchrieser,C., Danchin,A., Glaser,P. and Kunst,F. (1999) Cloning and assembly strategies in microbial genome projects. *Microbiology*, **145**, 2625-2634.

Frohme,M., Camargo,A.A., Czink,C., Matsukuma,A.Y., Simpson, A.J.G., Hoheisel,J.D. and Verjovski-Almedia,S. (2001) Directed gap closure in large-scale sequencing projects. *Genome Res.*, **11**, 901-903.

Gordon,D., Abajian,C. and Green,P. (1998) Consed: a graphical tool for sequence finishing. *Genome Res.*, **8**, 195-202.

Gordon,D., Desmarais,C. and Green,P. (2001) Automated finishing with Autofinish. *Genome Res.*, **11**, 614-625.

Havlak,P., Chen,R., Durbin,J., Egan,A., Ren,Y. Song,X., Weinstock,G.M. and Gibbs,R.A. (2004) The Atlas genome assembly system. *Genome Res.*, **14**, 721-732.

Herron-Olson,L., Freeman,J., Zhang,Q., Retzel,E.F. and Kapur,V. (2003) MGView: an alignment and visualization tool to enhance gap closure of microbial genomes. *Nucleic Acids Res.*, **31**, e106.

Médigue,C., Reschenmann,F., Danchin,A. and Virai,A. (1999) Imagene: an integrated computer environment for sequence annotation and analysis. *Bioinformatics*, **15**, 2-15.

Rutherford,K., Parkhill,J., Crook,J., Horsnell,T., Rice,P., Rajandream,M. and Barrell,B. (2000) Artemis: sequence visualization and annotation. *Bioinformatics*, **16**, 944-945.

Shymala,V. and Ames,G.F. (1989) Genome walking by single-specific-primer polymerase chain reaction: SSP-PCR. *Gene*. **84**, 1-8.

Sulston,J., Mallett,F., Staden,R., Durbin,R., Horsnell,T. and Coulson,A. (1988) Software for genome mapping by fingerprinting techniques. *Comput. Appl. Biosci.*, **4**, 125-132.

Tettelin,H., Radune,D., Kasif,S., Khouri,H. and Salzberg,S.L. (1999) Optimized multiplex PCR: efficiently closing a whole-genome shotgun sequencing project. *Genomics*, **62**, 500-507.

van Hijum,S.A.F.T., Zomer,A.L., Kuipers,O.P. and Kok,J. (2003) Projector: automatic contig mapping for gap closure purposes. *Nucleic Acids Res.*, **31**, e144.

Weinel,C., Tummler,B., Hilbert,H., Nelson,K.E. and Kiewitz,C. (2001) General method of rapid Smith/Birnstiel mapping adds for gap closure in shotgun microbial genome sequencing projects: application to *Pseudomonas putida* KT2440. *Nucleic Acids Res.*, **29**, e110.

Wendl,M.C. and Yang,S.P. (2004) Gap statistics for whole genome shotgun DNA sequencing projects. *Bioinformatics*, submitted manuscript.

Zhou,Y. Li,T., Zhao,J. and Luo,J. (2002) PGAAS: a prokaryotic genome assembly assistant system. *Bioinformatics*, **18**, 661-665.