BACCardI—a tool for the validation of genomic assemblies, assisting genome finishing and intergenome comparison

Daniela Bartels[1], Sebastian Kespohl[1], Stefan Albaum[1], Tanja Drüke[1], Alexander Goesmann[1], Julia Herold[1], Olaf Kaiser[2], Alfred Pühler[2], Friedhelm Pfeiffer[3], Günter Raddatz[4], Jens Stoye[5], Folker Meyer[1,*] and Stephan C. Schuster[4,*]

[1]Universität Bielefeld, Center for Biotechnology (CeBiTec), D-33594 Bielefeld, Germany, [2]Universität Bielefeld, Lehrstuhl für Genetik, Fakultät für Biologie, D-33594 Bielefeld, Germany, [3]Max-Planck-Institut für Biochemie, Abt. Membranbiochemie, Am Klopferspitz 18a, D-82152 Martinsried, Germany, [4]Max-Planck-Institut für Entwicklungsbiologie, Spemannstr. 35, D-72076 Tübingen, Germany and [5]Universität Bielefeld, Technische Fakultät, D-33594 Bielefeld, Germany

ABSTRACT

Summary: We provide the graphical tool BACCardI for the construction of virtual clone maps from standard assembler output files or BLAST based sequence comparisons. This new tool has been applied to numerous genome projects to solve various problems including (a) validation of whole genome shotgun assemblies, (b) support for contig ordering in the finishing phase of a genome project, and (c) intergenome comparison between related strains when only one of the strains has been sequenced and a large insert library is available for the other. The BACCardI software can seamlessly interact with various sequence assembly packages.

Motivation: Genomic assemblies generated from sequence information need to be validated by independent methods such as physical maps. The time-consuming task of building physical maps can be circumvented by virtual clone maps derived from read pair information of large insert libraries.

Availability: The BACCardI tool is freely available at http://www.cebitec.uni-bielefeld.de/groups/brf/software/baccardi/

Contact: fm@Cebitec.Uni-Bielefeld.DE

INTRODUCTION

Complete genomic sequences are assembled from numerous short sequences in a process known as the whole genome shotgun (WGS) technique (Fleischmann *et al*., 1995). By summing up the unique information from redundant, partial sequences, the complete genetic information of an organism becomes available. To cope with this task, several computer programs (assemblers) have been devised. Assemblers in general use the sequence information generated by the end sequencing of clones and piece them together into larger fragments of contiguous sequence information (contigs). The complete information of the contig sequence, the sequence of the original reads and their position in the contig, together with potential conflicts are stored in a standardized output format. This format called *.ace* is being used by most of today's assemblers and a description can be found at http://www.phrap.org. Since the assemblies are built on the basis of sequencing overlaps, repetitive elements can constitute a major problem in the form of assembly ambiguities. Many of these repetitive sequences cannot be resolved by the assembly technique using only sequencing reads from short or middle-sized insert libraries. To overcome these problems, large insert libraries such as cosmids, fosmids or BACs are needed. In order to anchor these large insert sized libraries onto the WGS data, endsequencing of these large clones is performed and the sequencing reads are assembled together with the sequence information from the WGS phase. The positional information of each large insert clone end, as has resulted from the assembly process, can therefore be used to compute a virtual 'physical map' or 'clone map' that allows to verifiy the assembly.

Assembly programs such as PHRAP (P. Green, unpublished) or Cap3 (Huang and Madan, 1999) use the overlap of sequencing reads to generate contigs by using Smith-Waterman-like algorithms. As these assembly algorithms are solely based on sequence information, the programs discard relevant information such as clone size, read orientation, and read pair information.

Several of the modern assemblers such as ARACHNE (Batzoglou *et al*., 2002; Jaffe *et al*., 2003), Euler assembler (Pevzner *et al*., 2001), CELERA assembler (Huson *et al*., 2001), PHUSION (Mullikin and Ning, 2003), and PCAP (Huang *et al*., 2003) make use of the read pairing information, but PHRAP is still the most widespread in use for bacterial genome projects. While PHRAP has demonstrated its robustness and its ability to assemble data in the least number of contigs (Pevzner *et al*., 2001), it is particularly susceptible to assembly mistakes that arise from repetitive regions in a genome. Therefore, a manual inspection of misplaced reads, which originate from artificial algorithmic effects of these assemblers, is necessary. Several

*To whom correspondence should be addressed.

software packages are available that allow the user to visualize the assemblies represented in *.ace* files. Most of these packages are limited to the display of sequence read positions and do not allow taking advantage of the information arising from read pairs. The two most commonly used packages that do display read pair information are Gap4 (Staden *et al*., 2000) and Consed (Gordon *et al*., 1998), with Consed being the only package that allows to display read pair information on a genomic level and to interface directly with the PHRAP assembler. In addition to Consed and its tool Autofinish (Gordon *et al*., 2001), that are indispensable in the finishing phase of a project, several other tools have recently been published that address the need for post-processing of sequence assemblies (Herron-Olson *et al*., 2003; Havlak *et al*., 2004; Tammi *et al*., 2004; Pop *et al*., 2004). However, none of these packages allow the visualization of a clone map from the assembly data. These virtual clone maps have been demonstrated to be versatile, not only for the validation of an assembly and the comparison of related strains, but also for the integration of other genomic data such as physical maps.

Here, we present the BACCardI tool for generating such virtual clone maps. The software was devised in a way that allows its integration into most assembly pipelines, as it uses a generic assembler output format.

SYSTEM AND METHODS

Basic design

In working with the BACCardI tool, three work steps are performed. The first step is a data pre-processing including the parsing of the input data and a *classification* of the clones. Secondly, the obtained data is *visualized* in two different ways, a circular genome view and a linear contig view. The third work step contains functions, e.g., for computing contig scaffolding and detection of misassemblies. In the following, some of these methods are described in more detail.

Classification of clones

BACCardI allows the projection of read pair information as obtained from positioning of end sequences onto the genome assembly.

Upper and lower boundaries for the clone sizes are specified by the user for each individual project. For the current BACCardI version, we distinguish between cosmid/fosmid libraries and BAC libraries for large insert size clone types. The user can state different clone size intervals for each of these types. In later versions, we plan to hold clone size intervals for each different library used in a project. This can be important when more than one BAC library with different average clone sizes and different standard deviations are used in the same project. Using BAC libraries for validation also does make sense only if the standard deviation of the clone sizes is quite smaller than the N50 contig size and thus also depends on the state of the assembly. As cosmids or fosmids have a defined size interval predetermined by their construction method, they are often better suited for validating an assembly.

The classification of clones is done using the direction and the distance of the end sequences of the same clone. Clones are rated ‘ok’ when the end sequences point towards each other and when the clone size is within user specified limits. For contig-spanning clones, the exact clone size cannot be computed. Instead, the size is given as the sum of the distances of the read positions to the end of the contigs. Clones are ‘ok’ and considered to ‘bridge’ two contigs when their size is below the upper limit. Clones below the lower size limit are still considered ‘ok’ and are used to indicate a relative order of the contigs. However, the size constraint means these contigs should not abut. Clones are rated ‘problematic’ when the size is out of bounds or when the end sequences do not point towards each other. Accordingly, we distinguish between clones that are ‘too short’, ‘too long’, or ‘incorrectly oriented’.

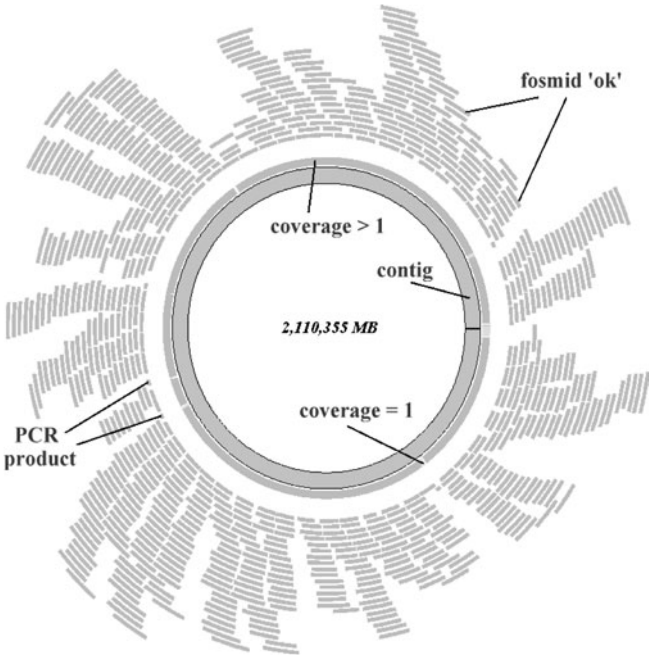


Fig. 1. A virtual large insert clone map built with BACCardI. The circle displays a fosmid map of the 2.1 MB Wolinella succinogenes genome. The inner circle of a large insert size clone map represents the contigs of a given genome assembly. In this case, as the genome is already finished, there is only one contig. For the complete validation of the genome sequence, a number of PCR products were done at parts with no fosmid coverage. The second circle shows the coverage of the contigs with fosmids (or PCR products). The green parts are covered with more than one, the yellow ones with exactly one fosmid. Regions that are not covered by any fosmid would be shown in red. The third layer represents each of the large insert clones classified to be ‘ok’ as a green arc. The outer layers representing clones classified as ‘problematic’ are not shown in this figure.

Visualization

BACCardI has two principal display modes: (a) the circular mode, representing a whole genome assembly of a prokaryotic genome, and (b) the linear mode, allowing a detailed analysis of a specific genomic region.

In the circular mode, BACCardI displays an overview of the genome and the virtual clone map in six layers (as illustrated in Figs 1, 2, and 4). From center to outside, layer 1 contains the contigs ordered as resulting from BACCardI’s analysis. Within this layer, repeats are marked in red, thus highlighting potential misassemblies. Layer two indicates the coverage with the large insert clones rated ‘ok’. This layer is a summary of the detailed data from layer 4. Regions covered by at least two clones are considered reliable and are therefore colored green. Regions covered by only one clone are colored yellow. Regions that are not covered by any clone are colored red and indicate potential misassembly points. Layer 4 shows the large insert clones rated ‘ok’ (green) and the large insert clones that are ‘ok’ but bridging gaps between contigs (red). In between, there are clones where only one end sequence is known. Such end sequences are represented by triangles pointing to the direction where the other clone end should be located. Layer 5 shows large insert clones rated ‘problematic’, e.g., large insert clones that are too short (dark green). There are also large insert clones which are too long (light green) and large insert clones where the end sequences do not point towards each other (blue).

In the linear mode (Fig. 3), the same type of information is represented in a similar way. The blue rectangle represents the contigs with repeats, directly below is a line representing coverage by large insert clones rated ‘ok’. This

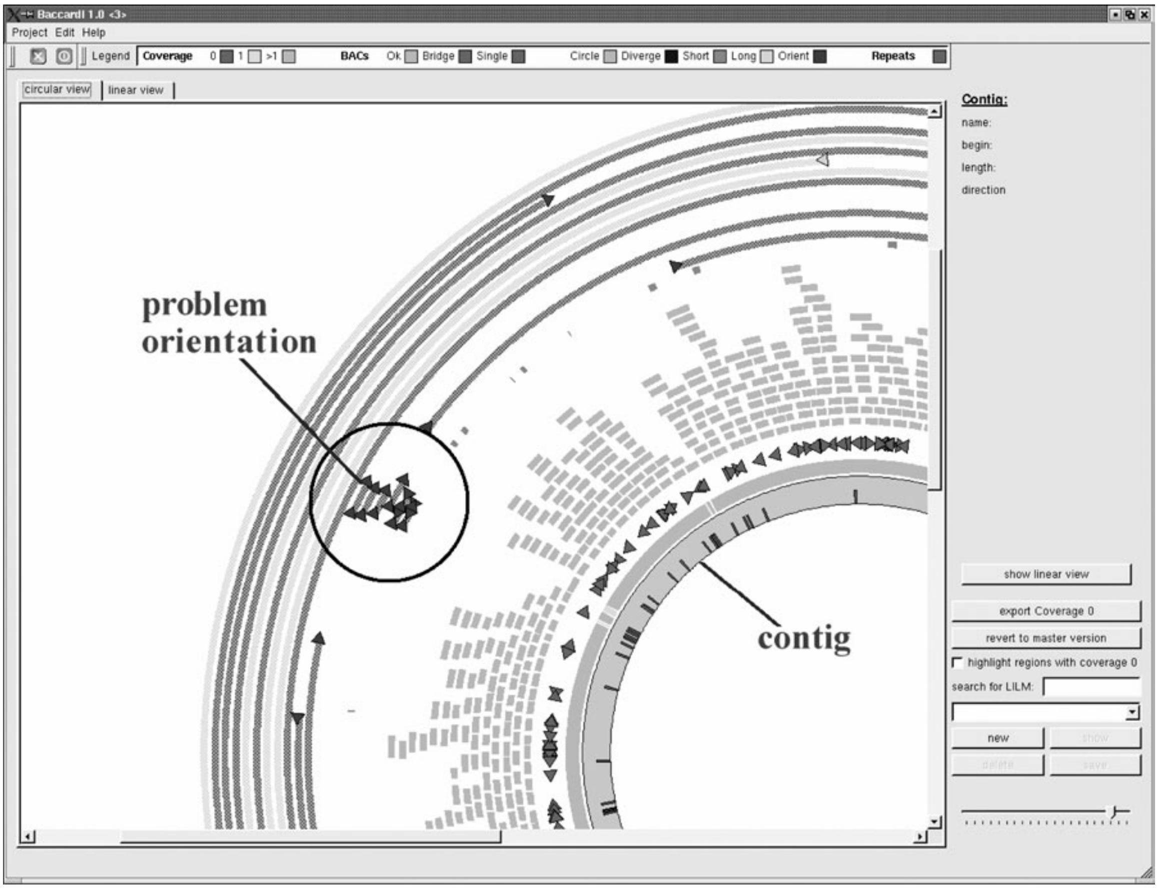


Fig. 2. A misassembled region identified with BACCardI using a fosmid map during the finishing phase of the *X.campestris* pv. vesicatoria genome (ongoing project). Coloring of the four inner layers corresponds to that in Figure 1. The fifth layer shows clones which are 'too short' (dark green). The sixth layer shows other clones classified as 'problematic' (light green: clones are 'too long'; blue: the end sequences do not point towards each other). The black circle marks eight fosmid clones colored in blue. As indicated by the triangles, the end sequences do not point towards each other.

is followed by a series of lines representing large insert clones rated 'ok', followed by clones where only one end has been sequenced. Problematic clones rated 'too long' are indicated above the contig by black triangles that point towards the other end. Thus, the arrows point to the direction of a potential misassembly, facilitating its detection. This provides a very user-friendly interface allowing a quick overview of the current state of the assembly.

Interaction with other software tools (interfaces)

BACCardI supports different kinds of data *input formats*. For constructing a clone map from a given assembly, the *.ace* file obtained from the assembly software is required. Contigs and clone ends, as well as tagged repetitive elements, are parsed from the *.ace* file. If the clone ends are not included in the *.ace* file, a mapping of the clone ends onto the contigs can be performed. Both the contigs and the clone ends must therefore be supplied to the software in multiple *fasta* format.

Another *interface* to BACCardI is a self-defined flat file containing the already classified clones in reference to the contigs they are mapped onto (documented on our website). This file format can be loaded and saved by the software. This allows programmers to connect their own tools to the visualization component of BACCardI.

Connecting the BACCardI tool with other tools, e.g., in an assembly pipeline (Kaiser *et al*., 2003), we have defined a number of *output formats*. For example, two kinds of Consed custom navigation (*.nav*) files can be exported, one for large insert sized clone ends and another for contig consensus positions. This enables the user to find misassemblies with BACCardI and view them in the Consed editor.

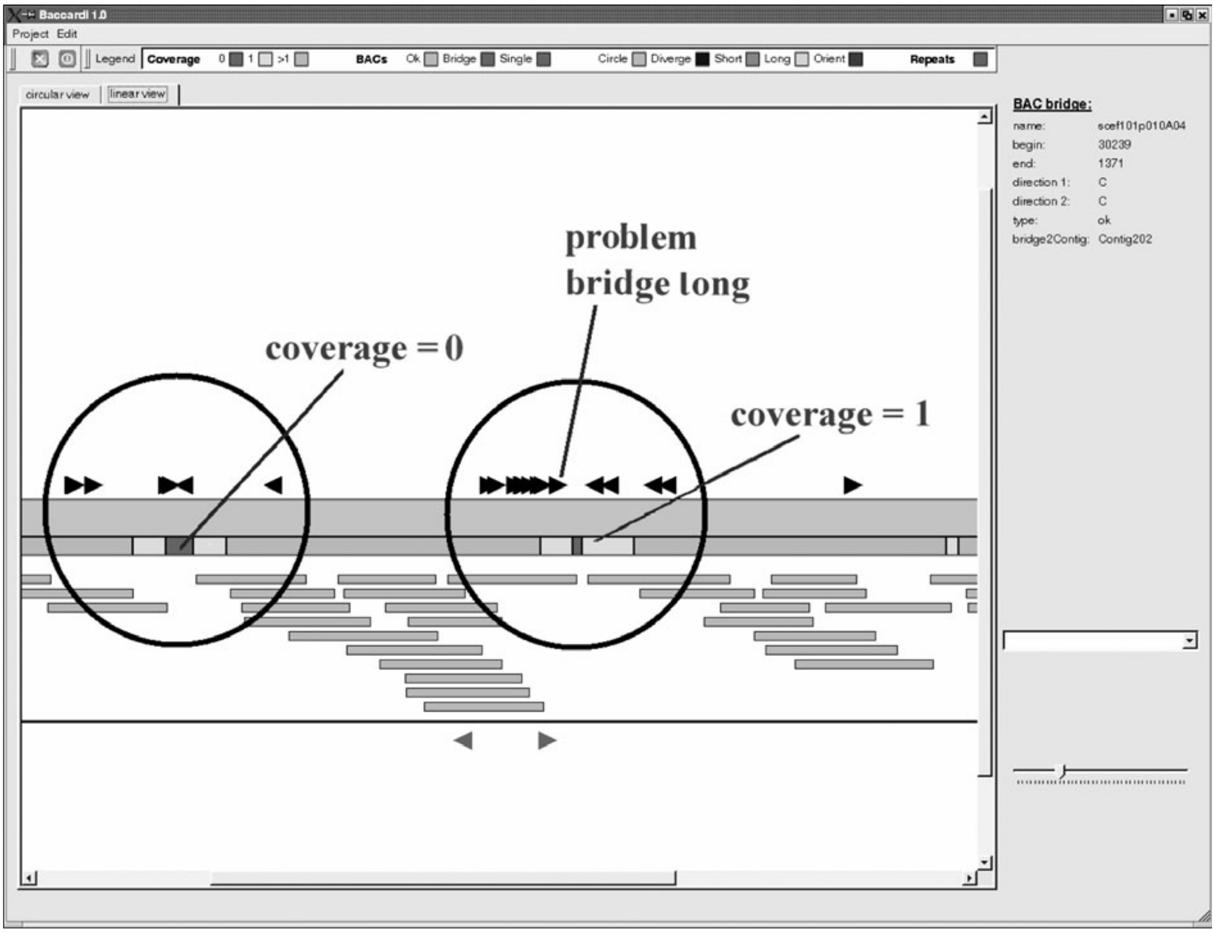
Contig scaffolding

The clones classified as 'bridging' are used for ordering the contigs obtained from an assembly. The algorithm BACCardI uses for this scaffolding step is related to the greedy path merging algorithm described in Huson *et al*. (2002). It was slightly modified and simplified for our purpose. One major difference is due to using the classification step described above, which implies that instead of the mean and the standard deviation of the members of the large insert clone library, exact values defining a size interval are used.

As the scaffolding software Bambus provides more sophisticated algorithms for contig scaffolding, we plan to make future versions of BACCardI compatible with the Bambus file format.

Finding misassemblies

Misassemblies are commonly linked to clones classified as 'problematic'. There are three categories: clones with a correct orientation of end sequences may be 'too short' or 'too long', but clones may also have an incorrect orientation ('orient') of end sequences. Such problems are quite frequent for individual clones, but a high density of problematic clone ends of the same or complementary types within a small region indicates a problematic region that needs to be further analyzed (Huson *et al*., 2003).



BAC bridge:
name: scef101p010A04
begin: 30239
end: 1371
direction 1: C
direction 2: C
type: ok
bridge2Contig: Contig202

Fig. 3. A contig in the early finishing phase of the ongoing *S.cellulosum* genome project. There are two misassembled regions indicated by lack of covering large insert clones (red in lane 3) and fosmids which are too long. In the BACCardI linear contig view, fosmids that are too long are displayed as black triangles that point towards where the other end is located. As a consequence, these triangles point from both sides to the misassembled region. Displaying them in the top line facilitates their detection and highlights potential misassemlies.

For each genome region we summarize the number of large insert clones classified ‘ok’. Regions with misassembly problems usually stand out by completely lacking clones classified ‘ok’.

In BACCardI we define a set of problematic clones as a cluster if their number is higher than a user defined threshold. Using these clusters in combination with the clone coverage of the clones stated as ‘ok’ in the region they occur, potential misassemblies can be identified. We call an often occurring scenario the early state misassembly (ESM, Fig. 5), as it occurs in the early stages of assemblies. It consists of two clusters whose (too long) reads face each other and that flank a region of no coverage with ‘ok’ clones. The interpretation of an ESM is that two unrelated genome regions have been misassembled into one contig. As these regions do not relate to each other, no clone covers this region. Clusters and ESMs can be highlighted in the BACCardI visualization. Contigs including ESM regions can be cut to get a better contig scaffolding (see also Jaffe *et al*., 2003).

Requirements

The BACCardI software runs on Unix and Linux systems. The CPU and memory requirements depend on the size of the genome. For the 13 MB *Sorangium cellulosum* genome, BACCardI used about 100 MB of memory on a Solaris system. The pre-processing and classification for an *.ace* file was done in about 10 min for this genome.

RESULTS

We have used the BACCardI tool in our genome assembly pipeline for several ongoing genome projects. Here, we describe four applications for the software in different phases of genome projects.

Automatic generation of large insert size clone maps

During the finishing phase of WGS projects the validation of the latest assembly is an important task (Kaiser *et al*., 2003). Figure 1 shows the large insert clone map for the *Wolinella succinogenes* (Baar *et al*., 2003) genome project at the end of the finishing phase.

The map shows a good verification of the genomic sequence since every part is covered by at least one fosmid or PCR product.

Finding misassemblies in genome projects

In almost any project known to the authors usually a small number of large insert clones exist which show contradictory information. This can be due to a number of reasons: (a) incorrect clone end naming, (b) incorrect positioning of an end due to bad sequence quality or because the end is located within a repeat, (c) a chimeric clone, or (d) a misassembled region. As usually some clones with such problems

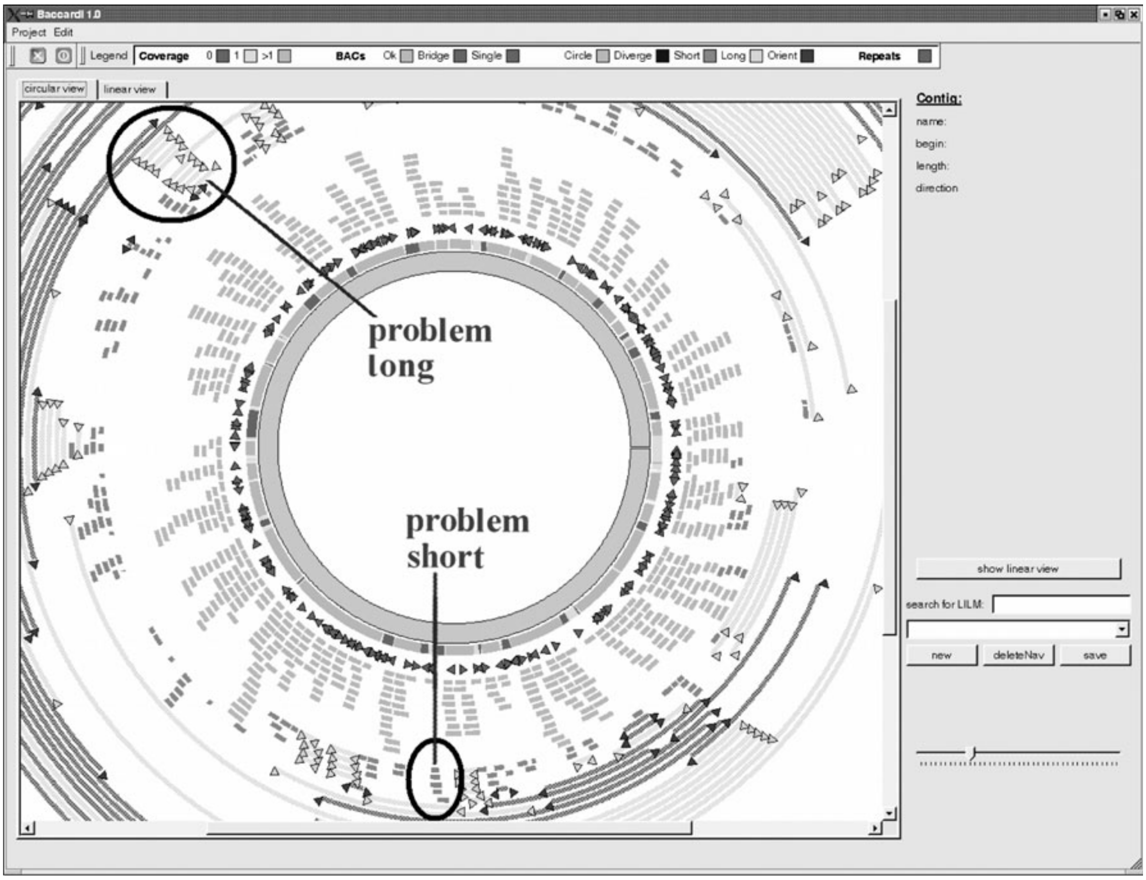


Fig. 4. Fosmids of a Xanthomonas genome mapped onto a closely related genome of a different strain of the same species. Layer 4 contains a large number of green fosmids that are classified as ‘ok’ and indicate synteny between the two genomes. In the layer with problematic clones, fosmids that are too long are shown in light green and those that are too short are shown in dark green. The regions marked by black circles indicate differences between the two genomes. In the top left region, there is an insertion in the reference genome resulting in fosmids that seem ‘too long’. In the bottom region there is a deletion in the reference genome resulting in fosmids that seem ‘too short’. Both regions are highlighted by red sections in layer 2, indicating a complete lack of clones that are ‘ok’.

are encountered, a single problematic clone alone may not be relevant when validating an assembly. If more than one clone indicates that a contig is problematic, this suggests a misassembly (Fig. 2).

In the ongoing *Xanthomonas campestris* pv. vesicatoria genome project with many repeats, a number of misassemblies were identified during the finishing phase. In the example seen in Figure 2, the genome has a high number of repetitive sequences (≥90 IS elements). Two of them are 24 kb apart and inverted with respect to each other. The region between these IS elements is inverted in the assembly as indicated by the eight problematic (wrongly oriented) clones (colored in blue). A comparison of the genome sequence at this stage with the finished genome sequence has proved this indication to be true.

In the early finishing phase, there are often a number of problematic regions in the genome. Figure 3 shows a BACCardI linear contig view of a part of the 12 MB *S.cellulosum* genome (ongoing project) with two problematic regions. The two misassemblies shown were derived from sequencing problems in parts of the genome where secondary structures (e.g., Rho-independent terminators) occur in combination with small repetitive sequences.

Using the display of BACCardI to guide the finishing process, these regions were readily identified and the problems were resolved. BACCardI generated a navigation file for Consed to allow an in-depth analysis at read level.

Mapping of large insert size clone libraries onto related genomes

Comparison of genomic sequences is a very useful albeit costly method to identify commonalities and differences between two organisms. We have devised a method that allows the identification of dissimilar regions in two genomes based on a large insert sized clone library.

If the genome sequence of an interesting organism is not available, information can be gained using a large insert size library if the sequence of a close relative, i.e., a reference genome, is already known. Figure 4 shows the clones of a fosmid library from a 5 MB *X.campestris* pv. vesicatoria genome mapped onto the reference genome from a different strain of the same species. For this application, BACCardI does not rely on a co-assembly strategy using the assembly programs, but rather uses BLAST (Altschul *et al*., 1997) as a mapping tool.

As is readily visible from the figure, large parts of the two genomes appear to be completely syntenic, resulting in a high coverage with

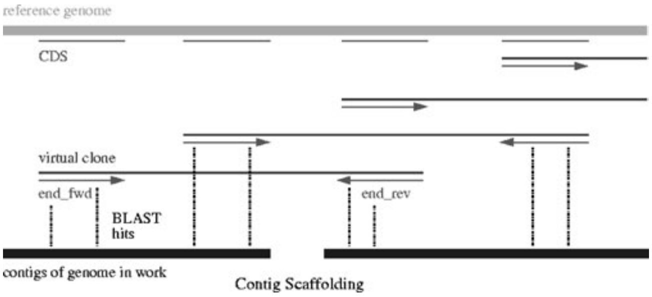


Fig. 5. Contig scaffolding using a related reference genome. Gene pairs of the reference genome located close to each other are transformed into virtual clones. The virtual clone consists of two terminal plus n (a user-specified number) internal CDS sequences (in the example given, $n = 1$). The CDS sequences derived from the reference genome are mapped to the analyzed genome using BLAST at the protein level. When the terminal CDS sequences map onto the terminal regions of distinct contigs and when the computed distance of the matching regions complies with the length of the virtual clones within certain limits, this is taken to indicate that the contigs are neighbors.

fosmids that are classified ‘ok’. Regions lacking coverage of ‘ok’ fosmids indicate genomic differences (red colored areas in the second layer in Fig. 4). A researcher interested only in differences between the strains can ignore the green colored regions with some degree of certainty. Consequently, the regions not covered contain novel genetic information not present at this side in the already known strain. In the upper left of Figure 4 one finds a region lacking fosmids classified ‘ok’ but having a number of fosmids which seem ‘too long’. This is consistent with a sequence insertion with respect to the reference genome. A probable deletion with respect to the reference genome is indicated at the bottom where correct fosmids are lacking and where all fosmids seem ‘too short’.

Using genomes of related organisms for finishing purposes

While the technique described above works for very similar genomes, there is a clear drive to use genomes of related organisms for finishing purposes. The idea of using protein matches to the ends of contigs to order and orient the contigs was first described in Fleischmann *et al*. (1995). We therefore decided to extend the reach of this approach by ‘virtual clone maps’.

For a 4 MB genome project, there exist more than one hundred genes that are lethal in the *Escherichia coli* host strain used for the shotgun and cosmid libraries. This prevents scaffolding of the contigs which result from the assembly using spanning clone information.

This problem is tackled by using the coding sequences (CDSs) from a related reference genome. We simulated a large insert sized (virtual) clone library as follows: The end sequences of a virtual clone are two CDSs of the reference genome that have a certain distance. This distance is defined as the number of CDSs that lie in between the two CDSs. Performing this for every CDS in the reference genome, we obtain a large number of virtual clones for the project.

These clones are mapped to the contigs of the analyzed genome using BLAST at the protein level. When the terminal coding sequences map onto the terminal regions of distinct contigs and the distance between the matching regions is within certain limits, the gene pair indicates that the two contigs are neighbours. This permits the selection of PCR primers and to close gaps experimentally. In the actual 4 MB genome project, more than 60 gaps could be closed by the application of this method.

DISCUSSION

Despite the fact that more than 200 complete genomes are already publicly available (see GOLD, Bernal *et al*., 2001), assembling genomes from WGS data can still be a tedious and demanding task. Several pathogenic bacteria with a highly flexible genome structure have been shown to be particularly difficult to assemble from just WGS reads. Therefore, it has become a commonly accepted strategy to assemble sequencing reads from DNA clone libraries of different insert sizes. The insert sizes have to be adjusted to the size of the repetitive elements found in a particular genome, but as a general rule, fosmid libraries have proven to be particularly useful due to their narrow size range and their large enough clone insert size. While several software packages are available that visualize contig information from the output files of assembly programs, none of them is capable of generating virtual clone maps.

While the most obvious use of such a clone map lies in the validation and verification of the WGS assembly, it is advantageous at the same time for the identification of bridging clones, a representation of the clone coverage in particular areas of the genome, and the resolution of misassemblies as well as scaffolding the contigs. The development of the BACCardI tool has become necessary due to the shortcoming of assembly programs like PHRAP in incorporating read pair and clone size information and can such be considered as a tool for the post-processing of the automated assembly step. Although some of the more recent assemblers like Arachne or the Celera Assembler make good use of mate pair data, validation of the results is still very useful. Seamless integration of BACCardI into existing pipelines was achieved by adhering to well-established standard formats.

Additional uses of BACCardI comprise genome comparison via mapping of large insert clone libraries onto related genomes and finishing support via the use of virtual clone libraries based on related genome sequences. By isolating large insert size clones that cannot be readily anchored or change their size and/or read pair orientation when mapped onto a related genome sequence, clones of interest for genomic comparison can be identified. These clones represent macroscopic changes in the genetic information of the respective genomes. Using related genomes to guide finishing via scaffolding contigs based on a virtual clone map generated by using in-silico clone information can help reducing the number of PCR reactions that need to be performed to close the remaining gaps in a sequencing project.

REFERENCES

Altschul,S.F., Madden,T.L., Schaffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Baar,C., Eppinger,M., Raddatz,G., Simon,J., Lanz,C., Klimmek,O., Nandakumar,R., Gross,R., Rosinus,A., Keller,H., Jagtap,P., Linke,B., Meyer,F., Lederer,H. and Schuster,S.C. (2003) Complete genome sequence and analysis of Wolinella succinogenes. *Proc. Natl Acad. Sci. USA*, **100**, 11690–11695.

Batzoglou,S., Jaffe,D.B., Stanley,K., Butler,J., Gnerre,S., Mauceli,E., Berger,B., Mesirov,J.P. and Lander,E.S. (2002) ARACHNE: a whole-genome shotgun assembler. *Genome Res.*, **12**, 177–189.

Bernal,A., Ear,U. and Kyrpides,N. (2001) Genomes OnLine Database (GOLD): a monitor of genome projects world-wide. *Nucleic Acids Res.*, **29**, 126–127.

Fleischmann,R.D., Adams,M.D., White,O., Clayton,R.A., Kirkness,E.F., Kerlavage,A.R., Bult,C.J., Tomb,J.F., Dougherty,B.A. and Merrick,J.M. (1995) Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. *Science*, **269**, 496–512.

Gordon,D., Abajian,C. and Green,P. (1998) Consed: a graphical tool for sequence finishing. *Genome Res.*, **11**, 614–625.

Gordon,D., Desmarais,C. and Green,P. (2001) Automated finishing with Autofinish. *Genome Res.*, **8**, 195–202.

Havlak,P., Chen,R., Durbin,K.J., Egan,A., Ren,Y., Song,X.-Z., 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.

Huang,X. and Madan,A. (1999) CAP3: a DNA sequence assembly program. *Genome Res.*, **9**, 868–877.

Huang,X., Wang,J., Aluru,S., Yang,S.-P. and Hillier,L. (2003) PCAP: a whole-genome assembly program. *Genome Res.*, **13**, 81–90.

Huson,D.H., Reinert,K., Kravitz,S.A., Remington,K.A., Delcher,A.L., Dew,I.M., Flanigan,M., Halpern,A.L., Lai,Z., Mobarry,C.M., Sutton,G.G. andMyers,E.W. (2001) Design of a compartmentalized shotgun assembler for the human genome. *Bioinformatics*, **17**(Suppl 1), S132–S139.

Huson,D.H., Reinert,K. and Myers,E.W. (2002) The greedy path-merging algorithm for contig scaffolding. *J. ACM*, **49**, 603–615.

Huson,D.H., Halpern,A.L., Lai,Z., Myers,E.W., Reinert,K. and Sutton,G.G. (2003) Comparing assemblies using fragments and mate-pairs. In *Algorithms in Bioinformatics: First International Workshop, WABI 2001*. pp. 294–306.

Jaffe,D.B., Butler,J., Gnerre,S., Mauceli,E., Mesirov,J.P., Zody,M.C. and Lander,E.S. (2003) Whole-genome sequence assembly for mammalian genomes: Arachne 2. *Genome Res.*, **13**, 91–96.

Kaiser,O., Bartels,D., Bekel,T., Goesmann,A., Kespohl,S., Pühler,A. and Meyer,F. (2003) Whole genome shotgun sequencing guided by bioinformatics pipelines—an optimized approach for an established technique. *J. Biotechnol.*, **106**, 121–133.

Mullikin,J.C. and Ning,Z. (2003) The phusion assembler. *Genome Res.*, **13**, 81–90.

Pevzner,P.A. and Tang,H. (2001) Fragment assembly with double-barreled data. *Bioinformatics*, **17**, S225–S233.

Pevzner,P.A., Tang,H. and Waterman,M.S. (2001) An Eulerian path approach to DNA fragment assembly. *PNAS*, **98**, 9748–9753.

Pop,M., Kosack,D.S. and Salzberg,S.L. (2004) Hierarchical scaffolding with Bambus. *Genome Res.*, **14**, 149–159.

Staden,R., Beal,K.F. and Bonfield,J.K. (2000) The Staden package, 1998. *Methods Mol. Biol.*, **132**, 115–130.

Tammi,M.T., Arner,E., Kindlund,E. and Andersson,B. (2004) ReDiT: Repeat Discrepancy Tagger—a shotgun assembly finishing aid. *Bioinformatics*, **20**, 803–804.