Base Qualities Help Sequencing Software

Richard Durbin[1] and Simon Dear[1]

Sanger Centre, Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SA, UK

With the complete sequencing of the human genome under way and the sequencing of complete microorganism genomes becoming commonplace, we have truly entered the era of large-scale DNA sequencing. Why now? As in some other data-rich areas of modern biology, for example, protein structure determination, it can be argued that the rate-limiting factors in increasing efficiency and throughput have been computer power and software. We could have run thousands of sequencing gels 20 years ago, but without image-processing software and fragment assembly packages it would not have been feasible to put together all of the individual sequence fragments from the gels to give megabases of continuous, accurate sequence. At any rate, the development of powerful computational tools is central to large-scale sequencing.

This special informatics issue contains several papers on the software used in genome sequencing centers, and in particular three papers on the set of programs from Phil Green's group at the University of Washington in Seattle (Ewing and Green 1998; Ewing et al. 1998; Gordon et al. 1998). These programs have played a key role in the progress of the largest-scale projects under way. They have been used extensively in the 100-Mb *Caenorhabditis elegans* project being completed this year and predominate among groups sequencing the human genome.

Such sequencing groups start with large clones such as BACs or PACs of 100 kb or more, or small genomes of up to a few megabases, for which the goal is to obtain complete accurate sequence. However, the raw sequences, or ''reads,'' obtained from the gels run on automated machines such as ABI 377s are only on the order of 500–1000 bp long and contain errors, particularly at the start and end of the read. To build up the longer sequence, many large-scale projects use a shotgun strategy, in which the first step is to collect thousands of primary reads from random subclones. These are pieced together by assembly software based on overlaps detected by sequence comparison. Following assembly, the sequence is made contiguous and accurate by adding extra ''finishing'' reads selected from the subclones to fill gaps and cover ambiguous regions where the primary data did not give sufficiently reliable information.

The goals of computer software in this process are to (1) make the most of the available data, so as to minimize costly data collection, and (2) reduce and simplify human interaction by a combination of clever algorithms and good ergonomics. Currently no system works in a completely automated fashion; there are some pattern recognition and analysis tasks that humans still perform much better than our software does. We support the view expressed by Churchill and Waterman (1992), that it will continue to be important to involve human input, targeted at progressively more specific cases, and via progressively better interfaces. This will both improve overall accuracy, and, importantly, provide the source of new ideas for increasing automation.

Simplistically, sequencing software is involved in three stages: (1) obtaining the primary read data from the gel images; (2) assembling the reads into the correct map to derive a consensus; and (3) supporting the finishing process. The first two are essentially automatic, but for now the last is interactive, involving human input to make those remaining decisions that cannot yet be left reliably to computers.

A number of different software packages have been developed to handle these tasks over the years, in both academic and commercial settings. Until recently, these dealt exclusively with base sequences determined from the reads. Where bases disagreed because of errors, either sufficient reads had to be present for a clear consensus to be obtained (which might still be wrong) or a user had to examine the original trace data manually. To minimize editing, the reads were conservatively clipped to avoid the lower accuracy regions at the ends. Programs such as GAP (Dear and Staden 1991; Bonfield et al. 1995), followed by many others, made this manual editing process much easier by presenting aligned trace data graphically, but editing continued to be a significant bottleneck.

The major innovation of the software from Phil Green's group has been to always keep an error probability measure, known as a ''quality,'' attached to each base prediction, either in a read or in the consensus. The initial quality values are obtained by the program phred (Ewing and Green 1998; Ewing et al. 1998), which makes base and quality calls for each read from the raw trace data. The assembly program phrap (P. Green, pers. comm.) uses the qualities both to significantly improve assembly and also to give a more accurate consensus sequence. Finally, the interactive program consed (Gordon et al. 1998) works in tight conjunction with phrap to provide a finishing environment, with an emphasis on editing the quality values and reassembly using these together with new finishing reads, so as to minimize editing the base calls themselves in the traditional fashion. Using estimates of confidence per base is not a new idea, for example, see Lawrence and Solovyev (1994) and Bonfield and Staden (1995), but the phred/phrap/consed package is perhaps the first to use it in such a central and ubiquitous fashion.

One of the most important gains coming from systematic use of qualities is that clipping is no longer needed before sequence assembly: The entire read length can be used. This has made an enormous difference for assembling human genomic sequence, ~35% of

[1]E-MAIL rd@sanger.ac.uk; sd@sanger.ac.uk; FAX 1223-494919.

which consists of *Alu* and other repeats (Smit 1996). Many of these are several hundred base pairs long; use of full read lengths allows them to be bridged, where the clipped, good sequence might not. Not only is the bridging important, but also having quality values allows more stringent matching, as more weight can be attached to cases where two aligned high-quality bases disagree than when two bases of unknown reliability disagree.

The quality values obtained in phred have been calibrated extensively, so they can be used to give reliable estimates of error rates; this calibration is the subject of one of the papers in this issue (Ewing and Green 1998), and it has been verified across data from a variety of sites by Richterich (1998). The resulting objectivity has had important consequences, both in terms of establishing standards that can be used meaningfully by others and also in allowing the quality values to be used for many other purposes. For example, they have been used for polymorphism detection (Nickerson et al. 1997), during oligonucleotide primer selection to avoid potentially inaccurate regions (Li et al. 1997), and to clip reads when the reliable part of single reads is needed, such as in EST projects.

As illustrated by these multiple ancillary uses, a modular approach to sequencing software is important. It allows insertion of extra checks and components, and replacement of components from one group with those of another when new features become available. By adopting such standards as SCF files (Dear and Staden 1992) and tag value–based text files like .ace files, *phred*, *phrap*, and *consed*, easy integration has been allowed into the inevitably complex software environments in place in large genome centers. One example of how this can be done is also illustrated in a paper in this issue, on the CAF package from the Sanger Centre and Genome Sequencing Center, St Louis (Dear et al. 1998).

Despite the evident successes of using base quality values, it is worth noting that they are not inevitable. Clearly one wants to make use of the qualitative information in the raw data (the traces), but other approaches to doing that can be imagined besides extracting a single number per base. After all, human editors use all of the trace information when editing interactively. The autoeditor module of the CAF package refers back to the original traces in the context of a complete assembly to make edits, and there have been explorations of use of the complete trace data during assembly (R. Jones, pers. comm.). For now, however, the complexities of using traces directly have not been robustly overcome, and the single well-defined probabilistic quality measure extracted by phred and used by phrap and consed clearly defines the state of the art.

REFERENCES

Bonfield, J.K. and R. Staden. 1995. *Nucleic Acids Res.* 23: 1406–1410.

Bonfield, J.K., K.F. Smith, and R. Staden. 1995. *Nucleic Acids Res.* 24: 4992–4999.

Churchill, G.A. and M.S. Waterman. 1992. *Genomics* 14: 89–98.

Dear, S. and R. Staden. 1991. *Nucleic Acids Res.* 19: 3907–3911.

———. 1992. *DNA Sequence* 3: 107–110.

Dear, S., R. Durbin, L. Hillier, G. Marth, J. Thierry-Mieg, and R. Mott. 1998. *Genome Res.* (this issue).

Ewing, B. and P. Green. 1998. *Genome Res.* (this issue).

Ewing, B., L. Hillier, M.C. Wendl, and P. Green. 1998. *Genome Res.* (this issue).

Gordon, D., C. Anajian, and P. Green. 1998. *Genome Res.* (this issue).

Lawrence, C.B. and V.V. Solovyev. 1994. *Nucleic Acids Res.* 22: 1272–1280.

Li, P., K.C. Kupfer, C.J. Davies, D. Burbee, G.A. Evans, and H.R. Garner. 1997. *Genomics* 40: 476–485.

Nickerson, D.A., V.O. Tobe, and S.L. Taylor. 1997. *Nucleic Acids Res.* 25: 2745–2751.

Richterich, P. 1998. *Genome Res.* (this issue).

Smit, A.F.A. 1996. *Curr. Opin. Genet. Dev.* 6: 743–748.