PAIRWISE END SEQUENCING

"Let the praises of God be in their mouth: and a two-edged sword in their hands."

The Book of Common Prayers

Random subcloning is a simple tool for mapping and sequencing DNA. In the previous chapter I provided a detailed analysis of mathematical models for random subcloning. More sophisticated strategies exist. With the maturation of the science of genomics, a wide variety of clever and innovative strategies have been developed (e.g., Evans, 1991; Burland et al., 1993; Li and Tucker, 1993; Kasai et al., 1992; Siemieniak et al., 1991). This plethora of strategies is a welcome delight to the researcher, for it adds to the armamentarium of tools available for genetic analysis. However, it brings with it the sometimes difficult decision of choosing which strategy is best.

Mathematical modeling, simulations, and experience provide the data upon which a strategy decision is made. In the previous chapter, I concentrated on a mathematical model supported by simulations as a tool for evaluating random subcloning strategies. However, it is not always possible to develop a sufficiently accurate mathematical model for a strategy. This is particularly true for the more complex strategies. In such cases, however, it is often possible to use simulations to acquire the necessary data. In this chapter I will describe a promising strategy for which a useful mathematical model has not been obtained and illustrate the use of simulations to provide data necessary for strategic decision making.

2.1 THE DOUBLE-BARREL SHOTGUN

Large-scale genomic projects are typically divided into two phases: first mapping, and then sequencing. A common strategy is to produce a rough map of approximately 40 kb completeness (terminology of Olson and Green, 1993), which is the level of cosmids. Cosmids provided by such a mapping effort can then be employed in sequencing strategies, often using a shotgun approach.[1]

[1] More recently, BACs have become a preferred sequencing template, necessitating maps of only about 100 kb completeness.

After a genome has been sequenced, a map becomes useless. This is because the information in the map is redundant with information in the genomic sequence.[2] Mapping information is a subset of sequencing information. A map is only valuable insofar as it can reduce the cost of subsequent sequencing.[3] For this reason, some have sought to combine mapping and sequencing. Because sequencing provides data that can be useful mapping information, it can make sense to begin sequencing before mapping is completed, so that the early sequence information can be used to lower the final cost of mapping. Taking this principle to its extreme, one can imagine a strategy that uses sequence data exclusively to build a map, bypassing completely the need for a separate mapping phase. Pairwise end-sequencing provides data that is particularly useful for this sort of approach. A variation of this strategy has recently been proposed as the method of choice for sequencing the human genome (Venter et al., 1996). The strategy in its pure form, dubbed “double-barrel shotgun sequencing,” is described here.

The double-barrel shotgun is a complete integration of mapping and sequencing, with a fine-scale map arising automatically from sequence data as a project proceeds. The strategy I describe retains the simplicity of random shotgun approaches, but, due to the fine-scale map produced, eliminates the need for more than minimal overdetermination of target sequence. It can half the final sequencing redundancy necessary to complete a project compared with a pure shotgun strategy. Its primary process of “scaffold building,” described below, is highly automatable and requires neither iterative steps nor intervention from highly trained individuals. At a low sequence redundancy this strategy can achieve target-spanning maps. However, since sequence accuracy is largely a function of its redundancy, after the initial scaffold-building phase, a directed sequence-finishing phase will be necessary to complete the sequence. The scaffold constructed in the first phase is ideal for choosing templates for sequencing in the final phase.

Recall that DNA is double stranded. Typically a sequence is read from only one strand of a clone. A sequence read is currently 600 to 1000 bp long; a clone can be as long as 10 kb.

[2] One exception is that the map can be used as a partial check on the accuracy of the sequence. Information used to construct maps can also be used as a partial check on the integrity of a clone library. Mapping data can be used to detect chimeric and deleted clones. Thus, the actual economics of how much mapping data (and what type) to acquire is more complex than I can cover in detail here.

[3] In the interim period (perhaps indefinite) between mapping and sequencing, a map can have considerable value. This interim is becoming shorter with the worldwide increase in sequencing capacity and speed.

Thus much of a clone in a pure random sequencing project remains unsequenced. Some technical and economic reasons for this were discussed in Section 1.2 and Section 1.3. It is advantageous to use the same primer for every sequencing reaction, so this primer must be derived from the end of the vector sequence. However, the vector attaches to both ends of the unknown cloned DNA, forming a circle. As a result, there are two ends of vector sequence from which two separate standard primers can be derived.[4] Therefore a clone can be sequenced at both ends. For a double barrel strategy, it is best to choose clones that are at least as long as twice the length of a sequence read. Therefore, from a sequencing perspective, one will obtain two fragments from the same clone.[5]

From the most simplistic viewpoint, double-barrel shotgunning is merely a way to obtain twice the number of sequence reads from a set clones, thereby halving the cost of clone isolation over the course of a project. This is indeed a major advantage, but the true beauty of this approach lies in the use of the knowledge of the pairwise correlation of fragments. For each pair of fragments, not only is their separation distance known, but also their relative orientation. Together, these data enable map construction, empowered tremendously by the orientation data, and aided to some extent by the distance data.

A depiction of an executed pairwise strategy is shown in Figure 2.1.

2.2 FORMULATION

A project begins with a target of length G. The length of unknown cloned inserts is designated I. The Monte Carlo simulations presented here, except where indicated, assume a constant insert length. Practically, insert lengths seldom are less than 1 kb or exceed 10 kb; it is this range that I will focus my attention on.

For these simulations I assume the sequence read length L to be a constant 400 bp.[6] The number of inserts successfully sequenced in a project is denoted n. Since inserts are sequenced at both ends, the total number of sequence reads will be $2n$ and the total amount of

[4] In most cases, these primers would be the m13-forward and m13-reverse primers.

[5] The term “fragment” was defined in Section 1.3.

[6] This choice of sequence read length is already antiquated. The simulations presented here were initially run in early 1993. A more typical sequence read length today would be 600 bp. A motivating factor behind the choice of a short sequence read length was to present a “worst case scenario” to demonstrate the power of the double-barrel shotgun even under adverse conditions. The desirability of the strategy improves as sequence read length improves.

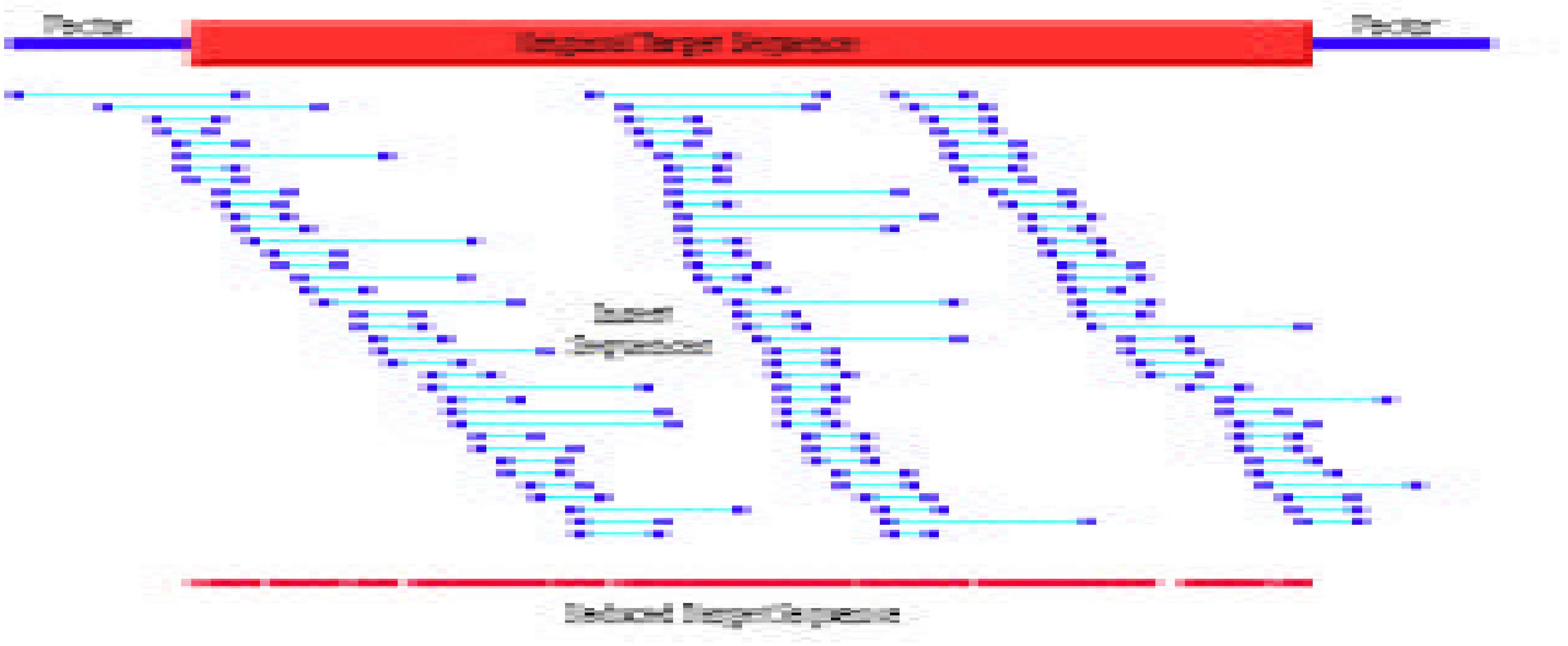

Figure 2.1. A model "double-barrel shotgun" assembly. A 2.25 sequence redundancy produces eighteen contigs which span ninety percent of an original target cosmid at 99.9% accuracy. Contig orientation and order are determined as shown. All but one gap are less than 400 bp; the remaining is 751 bp. More statistics are presented in Table 2.1.

sequence determined will be $2nL$. The redundancy of sequence data, denoted R_s, is defined to be $2nL/G$. Most of my results depend primarily on redundancy and only secondarily on sequence read length or quantity. Thus many short sequence reads are roughly equivalent to proportionally fewer long reads, and my choice of 400 bp for L is not critical. Note also that in mapping projects, redundancy is usually defined as the total length of all subcloned inserts analyzed. In the formulation presented here this quantity is denoted R_m and defined to be nI/G. The use of R_m permits comparison of double-barrel mapping results with other mapping techniques, such as restriction mapping.

After all insert end sequences have been determined, data can be analyzed and sequences can be assembled into islands and contigs. With a pairwise sequencing strategy, assembly of contigs is facilitated by knowledge of the pairwise orientation of sequences derived from the same insert. At low redundancies, it will not necessarily be possible to determine a single non-degenerate map for a project, as there may be sequence islands for which order or orientation is not determined. For a map to be finished, there must exist a path of bridging inserts between any two sequence islands, either directly or indirectly through other islands. Until enough redundancy is present to overcome this potential problem, there may be multiple coexisting and possibly overlapping contigs of clones. In order to address and discuss this issue, I define an ordered and oriented list of sequence islands to be a "scaffold." Since a scaffold consists of one or more overlapping subcloned inserts, it could also be legitimately called an island, and would be if our discussion centered solely on mapping issues. Here, I reserve the term "island" to denote a set of overlapping sequence reads.[7]

Beyond a certain point, as redundancy increases, the number of both islands and scaffolds will decrease, ultimately resulting in a single scaffold. Such scaffolds usually contain the entire target and as such are termed "complete." A complete scaffold usually contains vector sequence as well, but for statistical purposes is considered to be equal to the length of the target. The longest scaffold resulting from a project is termed the "maximum" scaffold. Gaps in sequence data internal to a scaffold have previously been termed "sequence-mapped gaps," or SMGs (Edwards and Caskey, 1991). For a complete scaffold, the size of the SMGs determines the completeness of the physical map, in the sense of Olson and Green (1993).

[7] Port et al. (1995) refer to scaffolds as "gapped islands," and sequence islands as "block islands."

For my computer simulations, I assume that a mutual overlap of length *T* is necessary and sufficient to detect overlap between two sequence reads. This overlap *T* was set at 30 bp, but the effect of choosing a different *T* would be slight, particularly because $T \ll L$. Assigning *T* any value between 1 and 50 bp does not noticeably alter my results (data not shown).

For most projects, a target sequence will have been fragmented along with its vector (i.e. YAC, BAC, cosmid, phage). To minimize the sequencing of vector, one might employ target sequence as a probe to pick positive inserts, or vector sequence as a probe to screen out vector. I present here only simulations of the first strategy, which I also find to be representative of other strategies (data not shown).[8] To this end, I assume that any insert that contains at least 40 bp of target sequence is a candidate for inclusion in a project. One advantage of this "positive screening" approach is that a few inserts will overlap vector sequence, and can be used to anchor the ends of some scaffolds to the vector.[9] However, this effect is slight, especially with longer target lengths.

My analysis centers on two target lengths: 35 kb and 200 kb. I chose 35 kb as a representative length for cosmids. I chose 200 kb as a representative length for a BAC or YAC, to demonstrate the feasibility of using pairwise data to facilitate sequencing targets of that size or larger. All computer simulation data points represent the average of 100 determinations.

2.3 COMPUTER SIMULATIONS

Complete scaffolds are often an ideal project endpoint, so I focused on determining optimal methods for their derivation. I have also characterized the expected values for certain parameters, including average SMG size and total scaffold length. To these ends, I employed computer simulations which I in turn supplemented with a raw data simulation based on a highly redundant random shotgun project. I will discuss the computer simulations first.

[8] Screening is a labor intensive process. Therefore, as sequencing cost continue to drop, while screening costs might actually rise, it is more likely that projects of the future will not bother to screen, but rather accept the slight increased cost in redundancy due to sequencing the vector as well as the target.

[9] If a screening approach were to be used, a "negative" screen would be more likely, as it is easier to implement. Also, "positive" screens generally have higher false positive and false negative rates.

The results of the computer simulations are presented in Figure 2.2 for 35 kb targets and in Figure 2.3 for 200 kb targets. In general, the number of scaffolds rises sharply at low redundancies and then declines at higher redundancies. The sharp rise occurs because each pair of insert sequence data added to a project at low redundancy has a high probability of forming a new scaffold. At higher redundancies, inserts begin to merge scaffolds and the number of scaffolds drops. For most projects, a single scaffold will form at a sequence redundancy between twofold and threefold. Slightly greater sequence redundancies were necessary to achieve single scaffolds of 200 kb targets than of 35 kb targets. Nonetheless, when 10 kb inserts were used, a single scaffold was always obtained at a redundancy less than twofold. In general, fewer scaffolds resulted when longer insert lengths were used. This is a result of longer inserts having a higher probability of spanning greater distances between sequence islands, and emphasizes the value of using as long an insert length as possible, which maximizes R_m.[10]

At high redundancies complete scaffolds are always obtained, as seen from the graphs of average maximum scaffold length (Figure 2.2 and Figure 2.3). For example, when 1.2 kb inserts are used for a 200 kb target, complete scaffolds are obtained around sevenfold redundancy. However, to obtain an improvement over traditional random shotgun sequencing strategies, complete scaffolds should be obtained at lower redundancies. This was clearly possible when longer insert lengths were employed. For example, redundancies of twofold were sufficient to ensure complete scaffolds when 10 kb inserts were simulated. When a project resulted in a single scaffold, this scaffold was also complete, or nearly so (data not shown).

I did not notice significant differences in redundancies necessary to achieve analogous results for either 35 kb (Figure 2.2), 200 kb (Figure 2.3), or even 1 Mb targets (data not shown). This suggests that sequencing effort scales roughly linearly to results, and not exponentially, even with relatively large targets. This rough linearity stems from the use of pairwise data, and indirectly from the high mapping redundancy R_m.[11]

The number of SMGs in maximum scaffolds increases, then decreases, as sequence redundancy increases. The initial increase is due to both the increasing length of the maximum

[10] This result is intuitive, at least in retrospect. One of the more useful contributions of the work presented in this chapter was the demonstration that "longer is better," at least with respect to pairwise insert length.

scaffold, enabling it to contain more gaps, and to the division of large gaps into smaller gaps as sequence islands bisect them. The subsequent decrease in SMGs is due to additional sequence data closing gaps. Roughly speaking, the largest number of SMGs tends to occur in complete scaffolds that have been obtained with a minimum of sequence redundancy. Thus, at the redundancies between twofold and fourfold that we envision as reasonable for pairwise projects, a significant number of SMGs are likely to result.

For many projects a complete target sequence is desired, with no gaps fragmenting continuity. For other projects, such as gene finding, complete sequence is not a priority, but gap characterization may be of interest. In general, project design should aim for gaps no longer than a single sequence read, or at most two reads. A gap that is a sequence read long can be closed by one directed sequence form either end of the gap using the spanning insert as a template. Double stranded coverage can be obtained by a single read from each direction. A gap that is two reads long can be covered by sequence walking with one walking iteration. If gaps longer than two read lengths occur in a project, it is likely that a cost-benefit analysis will dictate continuing the random phase.

The simulations demonstrate that large gaps occur as expected at very low redundancies, but at redundancies above 1.5 average gap length tends to be less than a single sequence read length. More importantly, for all projects with sequence redundancies above twofold, the *maximum* observed gap length tended to be less than 800 bp, requiring at most two sequence read lengths to close. Occasionally longer gaps occur. For example, at a redundancy of 2.5 with a 35 kb target, 100 simulations of a project employing 2 kb inserts contained one gap greater than 800 bp in 17 cases, and two such gaps in a single case. Above twofold redundancies, there were no significant differences in SMG length resulting from alternative choices of insert size. In consequence, an occasional project will require continued random sequencing after a complete scaffold is obtained in order to eliminate long gaps.

Long insert lengths are not always convenient sequencing templates.[12] For this reason,

[11] Note that $R_{\mathrm{m}} = R_{\mathrm{s}}\frac{I}{2L}$. Therefore, for 10 kb inserts and 400 bp read lengths, R_m will be 25 when R_s is 2. Plugging a redundancy of 25 into the equations presented in Chapter 1 will give the reader an intuitive feel for exactly how powerful the double-barrel shotgun can be as a mapping tool. This back-of-the-envelope approach also brings home the value of using large insert lengths.

[12] Discussed further in Section 1.2.

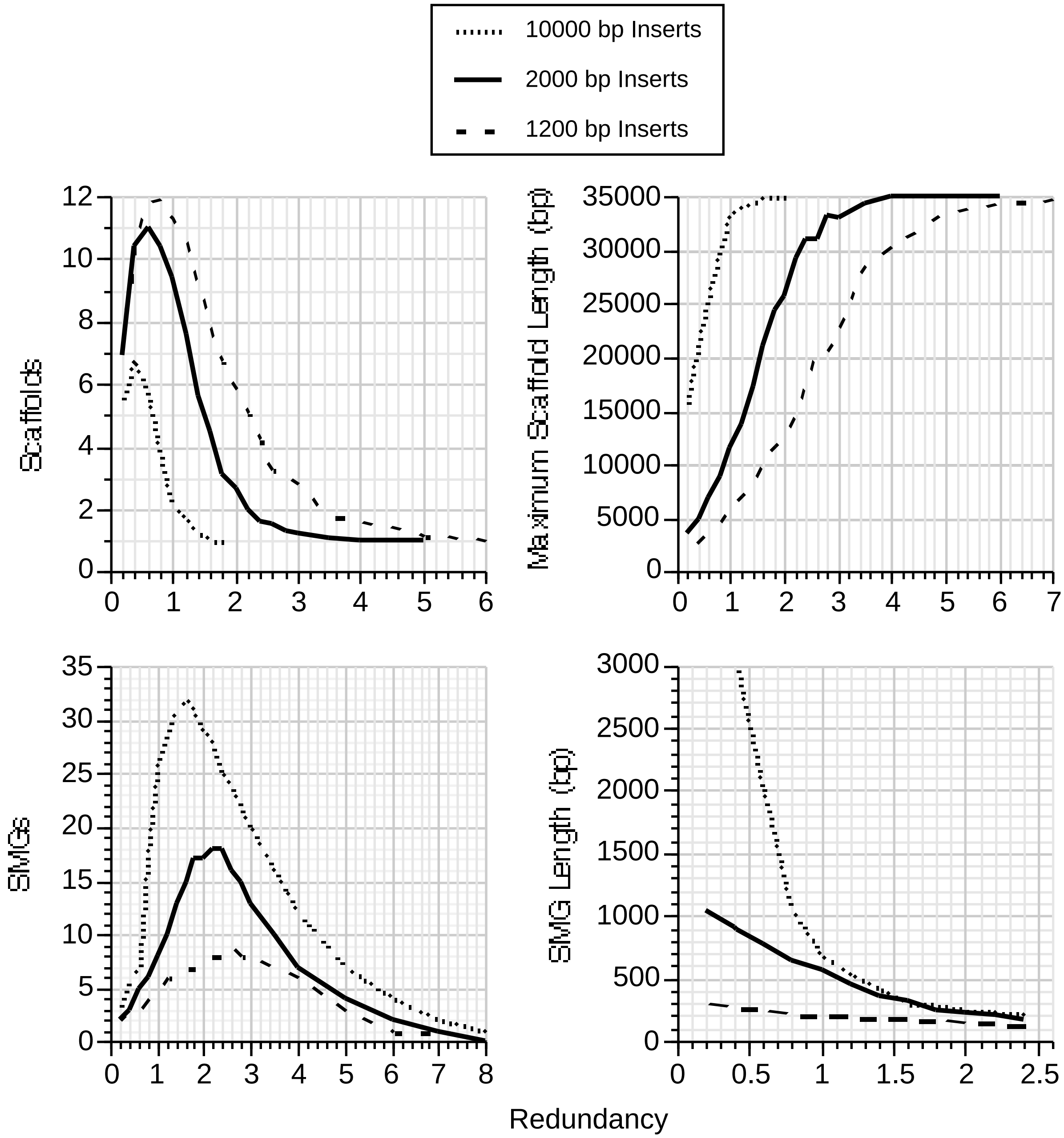

Figure 2.2. Parameters from a 35 kb pairwise project evaluated as a function of sequence redundancy. (*L*=400; *T*=30)

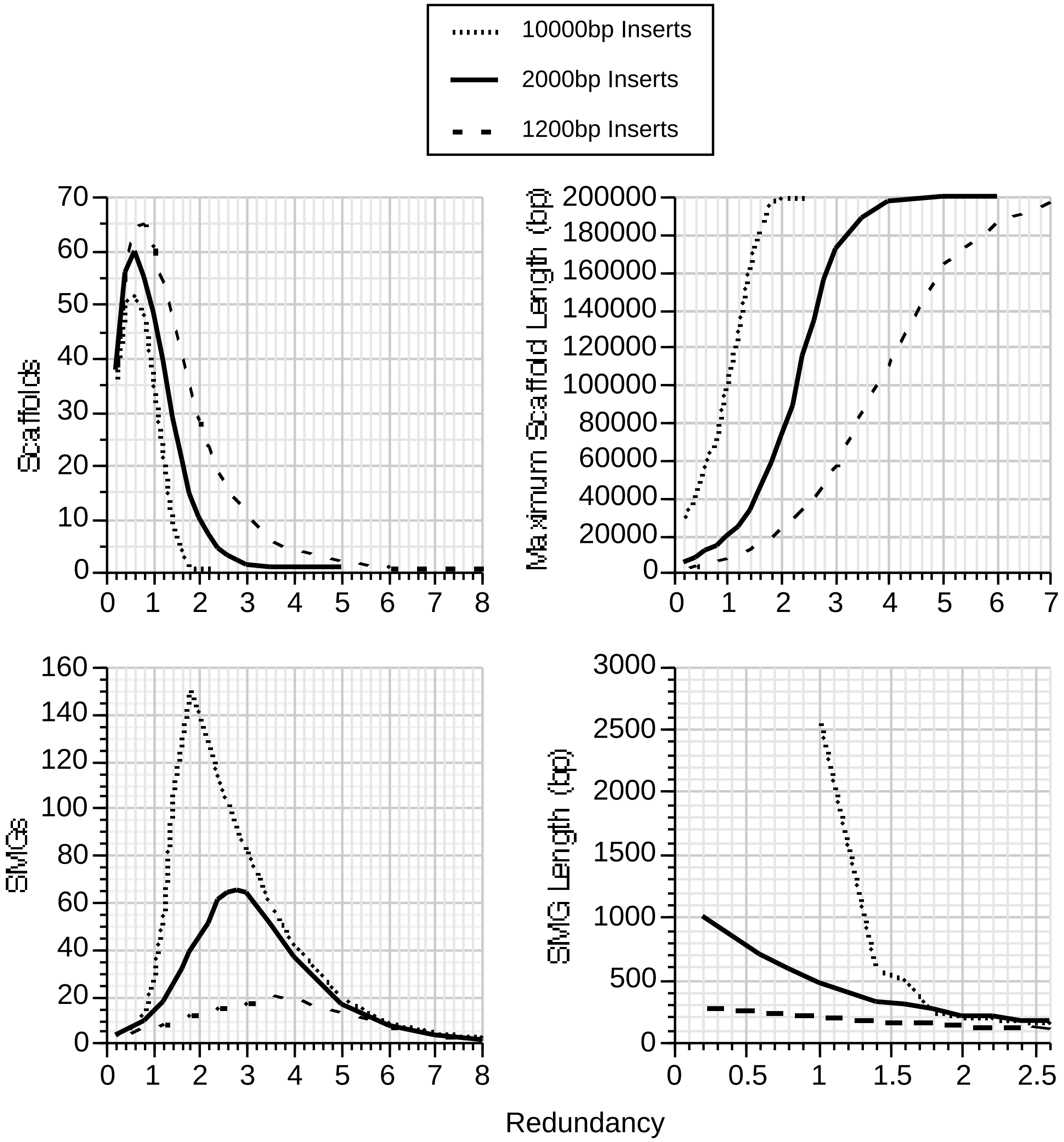

Figure 2.3. Parameters from a 200 kb pairwise project evaluated as a function of sequence redundancy. (*L*=400; *T*=30)

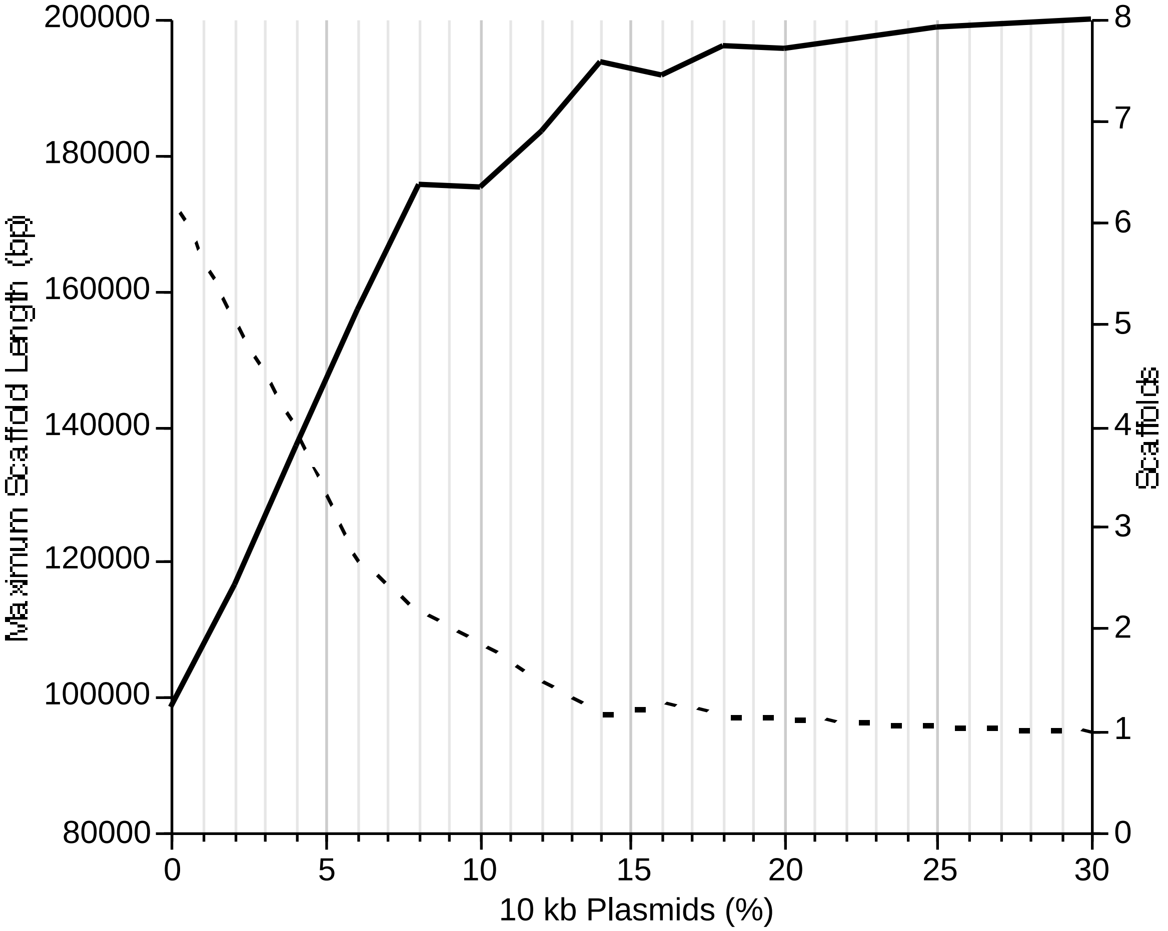

Figure 2.4. Pairwise strategies employing a mix of insert sizes were simulated. Here, a mix of 2,000 bp and 10,000 bp inserts were simulated at a constant sequence redundancy (2.25). As seen, a small proportion of larger inserts produces results comparable to those achieved when only large inserts are used. At 2.25 redundancy, complete scaffolds can be obtained with only a 15% mix of longer insert lengths. A 200 kb target was assumed (*L*=400; *T*=30). Maximum scaffold length, solid line; Scaffolds, dashed line.

I sought a strategy that minimized the need for longer inserts, and explored strategies that employed mixtures of insert sizes. In general, I found that benefits derived from large inserts could be obtained even when they represented a small fraction of the total number of inserts sequenced. In particular, I simulated strategies that employed a mixture of 2 kb and 10 kb inserts (Figure 2.4). For these simulations I held redundancy constant at 2.25 and assumed a 200 kb target. I found no significant differences between projects utilizing entirely 10 kb inserts and those that used only 15% 10 kb inserts.

I also envisioned strategies that mix pairwise data with data derived from a single strand only, such as might be obtained with m13 templates. A relatively small fraction of

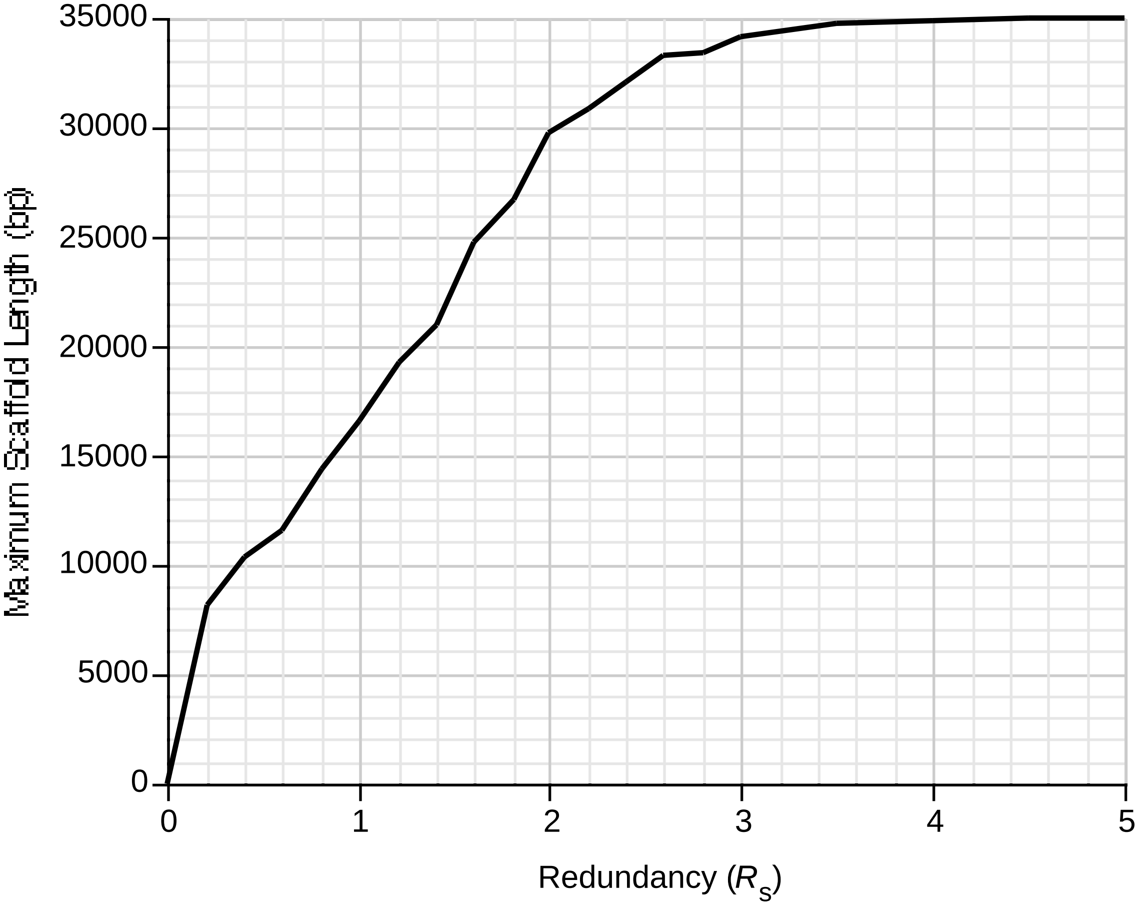

Figure 2.5. A hybrid strategy employing a combination of single strand and pairwise data was simulated. A mix of 60% single strand, 30% 2,000 bp, and 10% 10,000 bp data was employed. With this approach a complete scaffold can be obtained at less than threefold redundancy. A 35 kb target was assumed. (*L*=400; *T*=30)

pairwise data suffices for the formation of complete scaffolds which are largely composed of single strand data (Figure 2.5). With a mixture of 60% single strand data, 30% 2 kb pairwise insert data, and 10% 10 kb pairwise insert data, a maximum scaffold was reached before threefold redundancy for a 35 kb target. This simulation addresses a practical question, for sequencing reactions will occasionally fail, which implies that most pairwise projects will be supplemented with a cohort of widowed sequences.

My simulations met the 1-e^{-R} expectation of the Clarke-Carbon formula (data not shown). Therefore, at any given redundancy R_s, target coverage will be the same for either a traditional shotgun or a pairwise sequencing strategy. I emphasize that increased target coverage is not an advantage of pairwise strategies. The advantage of pairwise strategies lies in their ability to map and not in more efficient placement of random sequences. At the redundancies of about 2.5 necessary to build complete scaffolds, target coverage will be about 92%.

The computer simulations presented here hold both sequence read length *L* and insert lengths *I* constant. In actual projects, such as that represented by the raw data simulation presented below, these parameters will vary. I have incorporated variations into several additional computer simulations (data not shown), particularly by allowing *I* to vary as a squarewave centered on a target value. No significant differences in predicted results were noticed when *I* was allowed to vary. Variations in *L* also have no significant effect, as long as redundancy remains constant (data not shown).

Another assumption of the computer simulations was that all target fragments are equiprobable. The accuracy of this approximation is dependent on the fragmentation method (see, for an out-of-date example, Deininger, 1983). For most cases this approximation is quite valid, for the regions of fluctuation in fragmentation probability tend to be smaller than the length of the inserts. See a more detailed discussion in Section 1.3.

2.4 RAW DATA SIMULATION

I wished to verify that results from my computer simulations accurately modeled real projects. Such projects utilize raw sequence data and might employ templates with significant repeat elements. In addition, I was interested in determining the ease of assembling scaffolds by hand.[13] To this end, I designed a simulation built around a cosmid from the human T-cell receptor β locus that had previously been sequenced to a high redundancy using traditional shotgun sequencing.

This cosmid, designated A1-4, had been sequenced using a random shotgun strategy to a final redundancy of 8.4 (Koop et al., 1993). This cosmid consists of a 35343 bp target cloned into a 8213 bp vector. The target is notable in that it contains several repeats, including two 8.4 kb homologous elements. Their identity ranges from 85% to over 99% when 400 bp sliding windows are used for analysis. For this reason, the cosmid A1-4 was judged to represent a significant challenge for assembly (Lee Rowen, personal communication). The sequences used for the original assembly of A1-4 were derived primarily from single-stranded M13 templates and were sequenced with either Sequenase® or *Taq* cycle sequencing protocols.[14]

[13] Computer programs are yet to be developed that can maximally utilize pairwise data. However, the proprietary assembler in use at Celera as of late 1999 is a major advance.

For my pairwise assembly simulation I chose a subset of these 678 sequences that might represent typical data from a pairwise project. To this end I planned for a 2.25 final redundancy R_s. I wished to pursue a strategy that employed a mix of long and short templates, so simulated 88 2.5 kb inserts and 22 7 kb inserts. I determined the start locations of these fragments with a random number generator. The length of the fragments was modified randomly with a squarewave to simulate uncertainty in fragment length, as might occur if such fragments were size selected by banding on an agarose gel. Sequence reads of the proper orientation were then chosen from the A1-4 data set to represent the pairwise end sequences of my hypothetical fragments. The closest raw sequences to my randomly generated fragment endpoints were used, although no sequence was used twice. The final range of short fragment lengths was 1738-3418 bp (2375 +/- 287 s.d.), while the range of long fragments was 5312-8245 bp (6819 +/- 781 s.d.). For my initial assembly, all sequences longer than 400 bp were clipped to 400 bp in order to demonstrate that long sequence reads are not necessary for the success of pairwise assemblies. In addition, a few sequences were shorter, although no sequence was less than 250 bp. My final redundancy R_s was thus slightly less than 2.25. I judged my protocol for sequence selection to be a reasonable approximation of what might be likely to result from an actual pairwise project.

I then assembled these pairwise sequences into a single scaffold. Before and during this assembly I was blind to the nature of the repeats in A1-4 other than that it was a “difficult” cosmid. Additionally I was blind to the exact length of the fragments, other than that they were either “long” or “short.” Sequence contigs were assembled with the software package DNA∗® (Madison, WI). Scaffolds were assembled by sliding pieces of paper on a large table and were ultimately merged into a single scaffold (Figure 2.1). This assembly took about a day, illustrating that it would be a task best relegated to software implementation. Following the generation of this scaffold, each sequence contig was edited by hand for maximum accuracy. At this point, for editing purposes, the ends of sequences extending beyond 400 bp were used. Some of these sequences, although often low-accuracy, served to verify the ordering and orienting of the contigs within the scaffold. Additionally, they helped improve the overall accuracy of the sequence data.

The results of this raw data simulation compared favorably with the averages predicted by my computer simulations (Table 2.1). 89% of the target sequence was

[14] The Sequenase® protocol is now obsolete.

Table 2.1. Results from a raw data simulation of a pairwise strategy employed on a 35,343 bp cosmid which had previously been shotgun sequenced to ninefold redundancy. For this simulation, 2.25 sequence redundancy was derived from the end sequences of a mixture of hypothetical subclone inserts, 20% approximately 7,000 bp in length, and 80% approximately 2,500 bp in length. These results agree with results from my computer simulations and suggest the practicality of obtaining complete scaffolds in the range of twofold redundancy. The computer simulation column depicts average values from 100 independent determinations.

	Computer Simulation	Raw Data Simulation
Number of Scaffolds	1.02	1
Scaffold Length (bp)	35,267	35,343
Number of SMGs	21	17
Average SMG Length (bp)	169	223
% Target Covered	90.1	89.2

represented in this scaffold. The remaining unknown sequence was contained in 17 SMGs. Sequence accuracy was 99.9%. All but one of the 44 errors were present in regions covered by only a single strand. This suggests that double-strand coverage is capable of obtaining extremely high accuracy, which could be obtained for these regions by sequencing opposite strands. The exact lengths of the 17 SMGs were unknown, but could be estimated. Ten of these SMGs were spanned by the low quality ends of sequence reads present in my data set. This data was insufficient for base calling, but allowed the estimation of gap lengths to within a few base pairs. The lengths of the remaining SMGs could be estimated based on the lengths of the fragments that spanned them. As subsequently verified, all 17 SMGs were less than 800 bp and all but two were less than 300 bp.

2.5 PERSPECTIVE ON PAIRWISE STRATEGIES

Pairwise knowledge was first used extensively during the sequencing of the HPRT locus (Edwards et al., 1990).[15] The strategy itself was elucidated by Edwards and Caskey (1991). Smith et al. (1994) describe an approach in which the sequence islands in a scaffold can be employed as landmarks in a physical mapping project. Such landmarks were termed

[15] Sources within the genomics community report that Barrell, at the Sanger Centre, may have been the first to use pairwise sequencing, circa 1985. Edwards also used pairwise sequencing at an early stage. Civitello may have been the first to consider the implications of pairwise sequencing for fine-scale mapping.

“mapped and sequenced tags,” or MASTs.

One notable example of a pairwise strategy has been designated “ordered shotgun sequencing” (OSS). OSS was proposed by Chen et al. (1993). OSS is characterized by a low-redundancy pairwise approach that produces multiple unlinked scaffolds which form the basis for further directed sequencing. The genome-wide strategy described in Venter et al. (1996) bears many similarities to OSS. A few preliminary simulations and a review of pairwise strategies was provided by Richards et al. (1994).

Notable recent implementations of pairwise projects include scaffold construction from the 115 kb sigL locus of *Bacillus subtilis* (Fabret et al., 1996), the identification of a MHC class I-like gene linked to hereditary haemochromatosis (Feder et al., 1996), the identification of a candidate gene for Branchio-Oto-Renal syndrome (Abdelhak et al., 1997), and the complete sequencing by OSS of a 135 kb YAC (Chen et al., 1996).

It would seem that the advantages of pairwise strategies are overpowering. There appear to be no drawbacks. By executing a random strategy in a pairwise manner, one gains all the data of a traditional shotgun strategy, plus additional mapping data. This mapping data is acquired at no additional cost. In fact, the cost is slightly less, as fewer templates need to be prepared in a pairwise project. There is, however, one consideration which favors a pure shotgun approach.

The template best suited for sequencing is derived from the virus m13. This template is single stranded (ss), and can only be sequenced from one direction, preventing a pairwise strategy. Thus, for a pairwise strategy to be executed, one of two solutions must be employed. The usual approach is to use double-stranded (ds) plasmids as templates in place of ss m13. The problem with this approach is that with current methodology there is usually a small compromise in sequence read length. This creates a difficult cost-benefit decision. Should one pay a small cost in decreased sequence read length in order to benefit from the advantages of the double-barrel shotgun? In order to answer this question an exact dollar amount needs to be assigned to the components of the calculation, but once this is done the equations presented in this and the previous chapter enable the decision to be made. In most cases it is likely that the double-barrel benefits will outweigh any cost associated with a slight decrease in sequence read length.[16]

The second pairwise solution is the Janus strategy described by Burland et al. (1993). In this solution, after a single read is obtained from a ss m13 clone, the clone is transformed into a ds m13 plasmid, and the opposite strand is read. This permits acquisition of higher quality data, but at a considerable increase in clone handling and isolation costs. Therefore this strategy is not economically viable unless few clones are chosen for pairwise sequencing, leaving the majority of the reads as orphaned data. It is conceivable that such a strategy might be economically competitive with a plasmid-based pairwise strategy, as Figure 2.5 demonstrates that only a small amount of pairwise data is necessary for complete scaffold building. However, for now the high cost of converting ss m13 clones has sidelined this strategy.

It is unclear with current sequencing methodologies exactly how much is lost from the sequence read length when double-stranded in place of single-stranded templates are used. The effect appears to be rather slight, however, which will tend to bias a decision towards a plasmid-based pairwise project.

2.6 MATHEMATICAL MODELS

Little progress has been made towards the development of a mathematical model for the double-barrel shotgun strategy. An attempt was made by Port et al. (1995) to extend the approach of Lander and Waterman (1988) to pairwise data, but in addition to suffering from the drawbacks discussed in Section 1.11, these authors were limited to considerations of algorithmically “greedy” definitions of scaffolds.[17] It is unlikely that further progress can be made with this approach.

A very simple attempt to predict the expected number of SMGs and scaffolds was offered by Edwards and Caskey (1991). Their approach was to apply the Lander-Waterman equations independently to the inserts and the sequence reads, and to assume that the number of SMGs was equal to the gaps in the sequence islands, with the number of scaffolds equaling the number of gaps in the insert islands. This approach fails to take into account any of the

[16] In practice, it is extremely difficult to discover the actual dollar cost involved for any of the various costs, such as isolating a clone or sequencing it, and even more difficult to estimate the value of benefits such as a decrease in the difficulty of assembly due to the presence of pairwise data. Nevertheless, reasonable attempts can be made to estimate costs, particularly by standardizing on “laboratory operations” in place of dollars (see Siegel et al. 1998a, 1998b, and 1999). Genome Systems quotes the cost of subcloning a cosmid into m13 clones as $1500 and the cost of generating a single sequence read as approximately $25 (price quote, 12/97).

[17] In short, the greedy algorithm blinds itself to certain intricate topological interconnections of scaffolds that consist of three or more clones. These interconnections are more likely with longer inserts, so one drawback of the greedy approach is a failure to predict the advantages of using longer inserts.

topological intricacies of scaffolds. It can be recommended only for its simplicity, which gives some rough insight into the number and kind of gaps likely to be present in a pairwise project.

It is my intuition that a potentially useful model for pairwise projects may eventually be developed by treating the pairwise-characterized inserts as analogs to polymer building blocks in solution and applying mathematical analyses originally designed for gelling reactions. The scaffold-building process can be thought of as a gelling process.

As an alternative to the possible elegance of a gelling analogy, brute mathematical force may be utilized. For each number of inserts *n*, all possible topologies can be explored. The probability of each topology can be determined together with the characteristics of that topology, such as average scaffold length. The enumeration of topologies quickly becomes difficult. In Appendix A, I present an analysis for *n*={1,2,3}. I have not had the patience to work out the *n*=4 case. In actual projects, *n* will be on the order of hundreds to thousands. The limitations of the brute force approach should be obvious.[18] One result described in Appendix A is applicable to all *n* — increasing the insert length results in an increased scaffold length and an increased probability of obtaining a single contig. This is consistent with the results of the simulations.

2.7 DISCUSSION

The work on pairwise end sequencing presented here focuses on utility. I was primarily interested in determining the minimum amount of sequence redundancy necessary to reach satisfactory endpoints for projects that might actually be implemented in a laboratory. A scaffold that equals or exceeds target length is an ideal endpoint for a random strategy. Such a “maximum scaffold” is an ideal starting point for a directed strategy. I determined that it is possible to achieve such scaffolds at sequence redundancies around twofold.

A key factor in producing scaffolds at twofold redundancy is the choice of insert lengths. I found that the longer an insert is, the more useful it is. This is in considerable contrast with a misconception that the ideal insert length is three times the sequence read length.[19] Nevertheless, there is a practical upper limit to useful insert size. This limit depends

[18] A computer algorithm could be designed to enumerate and evaluate pairwise topologies. This could surely work for cases above *n*=3. However, the problem appears to be quite “hard” in the computer sense of the word. This would imply that even a computer would have a hard time “brute-forcing” calculations at practical choices of *n*.

[19] This misconception was common prior to 1995.

on three factors. First, it is difficult to routinely clone large fragments. Secondly, longer inserts have correspondingly more sequence complexity, which tends to degrade the quality of the raw data. Thirdly, assembly becomes more difficult with longer fragments, as the absolute uncertainty of the length between pairwise ends tends to increase. These limitations vary in stringency depending on available technology and resources. Thus the optimal choice of fragment size may vary from one laboratory to another. However, given the option, fragment sizes should be chosen as large as possible. It should be noted that in addition to their advantages in scaffold-building, large fragments are also extremely useful in detecting and resolving repetitive elements in target sequence.[20]

The use of a mixture of small and large inserts gains most of the advantages that would occur with the sole use of large inserts (Figure 2.2 and Figure 2.3). This is true even when the large inserts represent a relatively small fraction of the total. Generally speaking, the total length of all the inserts should be chosen to maximize the mapping redundancy R_m. If for technical reasons an insert library is constructed of a single intermediate size, a slightly higher sequence redundancy can be used to ensure completeness. The exact balance between redundancy and insert lengths will depend on the laboratory and should be determined on a case-by-case basis with the aid of computer simulations.

Double-barrel shotgun sequencing has many advantages over traditional shotgun sequencing. Notably the mapping redundancy R_m for single-barrel sequencing is $2nL=R_s$. The mapping redundancy for double-barrel sequencing is nI, which should be several times greater than R_s. This creates a high-redundancy mapping situation which permits efficient low-redundancy sequencing. Pairwise strategies are not confined to low-pass sequencing and are equally valuable at high redundancies, particularly for sequence assembly. For these reasons, I feel that all random strategies should employ pairwise data, at least with the goal of generating complete scaffolds as a basis for further sequencing. Such sequencing can either continue to be random or switch to directed approaches.

I expect that most projects will move to directed sequencing after a complete scaffold is obtained. This “gap closure” phase will entail obtaining sequence for SMGs as well as

[20] A detailed discussion is beyond the scope of the present work. As a general rule, a repeat cannot be properly analyzed if it is longer than the size of the mapping fragment, or smaller than the uncertainty in the position of the markers in the map. The positional uncertainty of finished sequence is very close to zero, so repeats can be efficiently detected if only if they are shorter than the mapping fragment size. For traditional shotgun sequencing this is L, but for double-barrel sequencing it is I.

reverse sequence from regions of single-strand coverage. The templates localized during scaffold construction are ideal substrates for such directed sequencing. One gap closure methodology is to sequence a PCR product spanning the gap. If the entire target sequence is not needed, only gaps of interest need be filled. For example, the entire target may not be needed once a gene is localized in a gene-finding effort. Likewise, if a gap of known size is clearly bounded by the 5' and 3' ends of a known element, such as an *Alu* repeat, the gap need not necessarily be sequenced.

For any reasonably large project, computational tools will be necessary to assemble and analyze scaffolds. I expect that such software will evolve in the future, as the advantages of pairwise assembly drive the market for assembly software. On the other hand, without assembly software many of the advantages of pairwise sequencing are tempered. Thus pairwise sequencing will not become a universal tool until such a time as good software becomes available to the community at large. Currently no available software tools use pairwise data to aid the assembly algorithm, although several are capable of displaying pairwise data or using it to verify accuracy.

Building a first generation software tool should be straightforward. One simple assembly algorithm is a four step process. First, assemble individual sequences into islands, blind to their pairwise nature. Second, order the resulting sequence islands by linking together sequences with their mates from opposite ends of the inserts. Third, check for inconsistencies, remove suspect pairs of sequences, and iterate the process. Finally, make rough estimates of gap distances based on insert lengths and on low quality ends of sequence reads. This algorithm was successfully employed by hand to assemble the cosmid A1-4, which I believe to have been a robust test of its efficiency. Improvements on such an algorithm can be made, but even an implementation as straightforward as this would be tremendously useful to workers in the field. The finished sequence from pairwise algorithms is more robust and accurate than that of traditional algorithms. Each paired sequence offers a positional check on its mate, allowing a majority of misplaced sequences to be immediately located following an assembly. For example, without this check I would have misplaced several sequences during my raw data simulation of the A1-4 assembly.

One concern occasionally raised to pairwise sequencing is that the exact lengths of the pairwise inserts is uncertain. Such uncertainty arises because fragments are typically band purified on an agarose gel and not subsequently characterized.[21] In extreme cases, particularly at low redundancies, such uncertainty might result in indeterminate island order within a scaffold.[22] However, in my raw data simulation of cosmid A1-4, I found that a redundancy R_s

of 2.25 was more than enough to avoid any such problems. Thus, a knowledge of exact insert lengths would have contributed little to this project. This will be generally true in practice, for all SMGs are highly likely to be less than a sequence read in length. Therefore, rather than measuring their size, one should simply sequence across them. In the rare event that the gap was not closed after one or two iterations of sequence walking, the lengths of the fragments could at that point be determined.

Many modifications to the basic pairwise strategy have been proposed. For example, Burland et al. (1993) suggest sequencing only one end of inserts initially, and then only sequencing opposite of ends of clones that are likely to produce new information, such as those ends from inserts that extend beyond contig edges. This strategy makes sense if the cost of sequencing an opposite clone end is considerably greater than the cost of sequencing an initial end, as in the Janus strategy. It can also, however, alter the independent and random accumulation of sequence information. For example, one can intentionally avoid sequencing a fragment that lies in region already covered to a high depth in previous sequence reads. However, this comes at a high cost in clone isolation and clone-tracking administration. Additionally, an increased use of orphan sequences will prolong the achievement of a complete scaffold. I recommend sequencing both ends of all inserts, at least until complete scaffolds are obtained.

Another modification to the basic pairwise strategy is to halt a project before it reaches the complete scaffold stage. This is the approach of OSS.[23] An extreme example is presented by Smith et al. (1994), in which cosmid clones are entirely mapped before their ends are sequenced. I see no advantage in halting pairwise projects before complete scaffolds are achieved. The extra sequence redundancy necessary to achieve complete scaffolds is relatively small compared with the labor that is otherwise necessary to assemble unlinked scaffolds into a complete map.

[21] It is possible to more accurately measure the lengths of fragments, but this involves extra effort. Conveniently, such effort is not needed as the length information obtained is redundant with information which will be gained during the sequence finishing phase. In specific cases, if deemed worthwhile, one could measure the length of a fragment retrospectively.

[22] Such a scaffold would fail to meet the strict definition of a scaffold, which requires that all islands of a scaffold be ordered and oriented.

[23] It is also the approach of Venter et al. (1996), but in the Venter case, the BAC end sequences are not used for sequence finishing, so there is a driving incentive to halt the pairwise sequencing early.

Pairwise strategies can effectively handle megabase targets. My simulations demonstrate that sequence redundancies between twofold and threefold are more than adequate to span such targets with complete scaffolds (data not shown). By permitting direct shotgun sequencing, double-barrel strategies eliminate the need to use intermediate subclones of large mapping vectors such as BACs or YACs. This elimination of cosmid subcloning and mapping can represent a significant increase in the efficiency of genomic sequencing efforts. I particularly recommend double-barrel shotgun sequencing for small bacterial and viral genomes. Pairwise strategies were used extensively during the sequencing of the *Haemophilus influenza* genome, the first genome ever to be completely sequenced (Fleischmann et al., 1995). Pairwise strategies were again used for the *Mycoplasma genitalium* genome (Fraser et al., 1995). The speed with which these genomes were sequenced stunned the genomics community.[24]

Low redundancy pairwise strategies are particularly useful for gene finding, as they provide most of the sequence data from a target region, which can then be utilized in similarity or feature identification searches. High accuracy sequence is not needed, nor is complete coverage. Regions of interest can be singled out for subsequent special attention facilitated by the structured nature of the scaffold.

To summarize: pairwise end sequencing can be characterized as mapping at high redundancy, but sequencing at low redundancy. It generates complete scaffolds more economically and more quickly than traditional shotgun sequencing. The advantages of this strategy include its simplicity and the absence of any need for clone mapping other than that which results as an incidental by-product of sequencing. It is capable of handling relatively large repeats or complex templates. Its utility includes STS generation, gene finding, low- and high-pass sequencing, and ultra-fine-scale mapping.

[24] By contrast, the *E. coli* genome project did not use pairwise data and took years (rather than months) to complete (Blattner et al., 1997). To be fair, other factors also influenced the slow rate of *E. coli* sequencing.