TURNING REPEATS TO ADVANTAGE: SCAFFOLDING GENOMIC CONTIGS USING LTR RETROTRANSPOSONS

A. Kalyanaraman*[1], S. Aluru[1] and P.S. Schnable[2]
[1] *Department of Electrical and Computer Engineering*
[2] *Departments of Agronomy, and Genetics, Development and Cell Biology*
Iowa State University,
Ames, IA 50011, USA
Email: {ananthk,aluru,schnable}@iastate.edu

The abundance of repeat elements in the maize genome complicates its assembly. Retrotransposons alone are estimated to constitute at least 50% of the genome. In this paper, we introduce a problem called *retroscaffolding*, which is a new variant of the well known problem of *scaffolding* that orders and orients a set of assembled contigs in a genome assembly project. The key feature of this new formulation is that it takes advantage of the structural characteristics and abundance of a particular type of retrotransposons called the *Long Terminal Repeat (LTR) retrotransposons.* This approach is not meant to supplant but rather to complement other scaffolding approaches. The advantages of retroscaffolding are two fold: (i) it allows detection of regions containing LTR retrotransposons within the unfinished portions of a genome and can therefore guide the process of finishing, and (ii) it provides a mechanism to lower sequencing coverage without impacting the quality of the final assembled genic portions. Sequencing and finishing costs dominate the expenditures in whole genome projects, and it is often desired in the interest of saving cost to reduce such efforts spent on repetitive regions of a genome. The retroscaffolding technique provides a viable mechanism to this effect. Results of preliminary studies on maize genomic data validate the utility of our approach. We also report on the on-going development of an algorithmic framework to perform retroscaffolding.

1. INTRODUCTION

Hierarchical sequencing[4] is being used to sequence the maize genome[18]. In this approach, a genome is first broken into numerous smaller clones of size up to 200 *kbp* each called a *Bacterial Artificial Chromosome* (or *BAC*). Next, a combination of these BACs that provide a *minimum tiling path* based on their locations along the genome is determined. Each selected BAC is then individually sequenced using a shotgun approach that generates numerous short (~500-1,000 *bp* long) *fragments.* The problem of assembling the target genome is thereby reduced to the problem of computationally assembling each BAC from its fragments.

The fragments generated by a shotgun experiment approximately represent a collection of sequences originating from positions distributed uniformly at random over each BAC. As with a jigsaw puzzle, the idea is to generate fragments such that each genomic position is expected to be *covered* (or sampled) by at least one fragment — and also ensuring that there is sufficient computable evidence in the form of "overlaps" between fragments to carry out the assembly. Regardless of the *coverage* specified, however, *gaps* invariably occur during sequencing, i.e., it cannot be guaranteed that every position is covered by at least one fragment. Coverage affects the nature of gaps — a low coverage typically results in several long gaps, while a high coverage results in fewer and shorter gaps. Because of gaps, assembling a set of fragments sequenced from a BAC typically results in not one but many assembled sequences called *contigs* that represent the set of all contiguous genomic stretches sampled. The next step, scaffolding, aims at determining the order and orientation of the contigs relative to one another. Once scaffolded, the identified gaps between contigs can be filled through targeted experimental procedures called *pre-finishing* and *finishing.* For simplicity, we use the term "finishing" to collectively refer to both these procedures.

The main focus of this paper is the scaffolding step. The need for scaffolding arises from the fact that there could be gaps in sequencing. To be able to identify a pair of contigs corresponding to adjacent genomic stretches, current methods generate shotgun fragments in "pairs" — each BAC is first bro-

*Corresponding author.

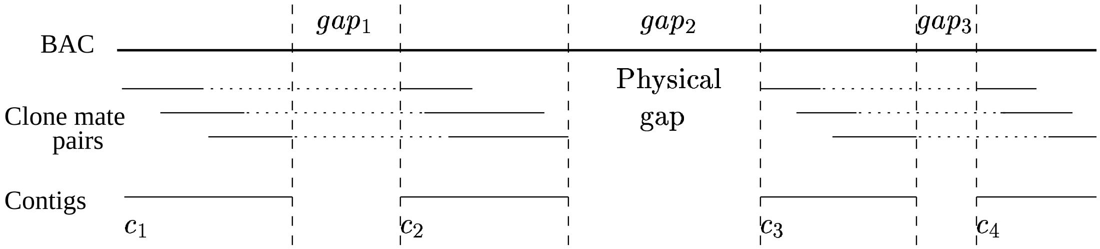

Fig. 1. An example showing 6 pairs of clone mate fragments (shown connected in dotted lines) sequenced from a given BAC. The relative order and orientation between contigs c_1 and c_2 (also, between c_3 and c_4) can be inferred from the clone mates. The supplied clone mate information is , however, not sufficient to determine the scaffolding information between all pairs of contigs in this example.

ken into smaller clones of length ~5 *kbp*, and each such clone is sequenced from both ends thereby producing two fragments which are referred to as *clone mates* (or a *clone pair*). During scaffolding, the fact that a pair of clone mates originated from the same ~5 *kbp* clone can be used to impose distance and orientation constraints for linking contigs that span the corresponding fragments[1, 9, 10, 17, 19]. Figure 1 illustrates an example of scaffolding contigs based on clone mate information. This technique is not, however, sufficient to link contigs surrounding gaps without a flanking pair of clone mates (gap_2 in Figure 1). Such gaps, called *physical gaps*, are typically harder to "close", and involve costly finishing efforts. Performing a higher coverage sequencing is an effective but expensive approach to reduce the occurrences of gaps. The approach proposed in this paper provides an alternative mechanism to scaffold around physical gaps as well, subject to their repeat content.

In this paper, we introduce a new variant of the scaffolding problem called the *retroscaffolding* problem. The problem is to order and orient contigs based on their span of LTR retrotransposon-rich regions of the genome. This approach has the following advantages:

- It does not require clone mate information. Thus, our approach complements existing scaffolding approaches for genomes with significant LTR retrotransposon content. Also, with the advent of newer sequencing technologies[13] that do not generate clone mate information, the importance of our approach is further emphasized.
- It can be used to identify LTR retrotransposon-rich portions within the unfinished genomic regions. Such information can be useful if it is decided to not finish repetitive regions in the interest of saving costs, as is the case with the maize genome project[18].
- In genome projects of highly repetitive genomes, most of the sequencing and finishing efforts are expected to be spent on repeat rich regions. This is one of the main concerns in the on-going efforts to sequence the maize genome, at least 50% of which is expected to be retrotransposons. The retroscaffolding technique provides a mechanism to reduce sequencing coverage without affecting the quality of the genic portion of the final assembly, thereby providing a means to reduce the sequencing costs.

In Section 2, we describe the retroscaffolding idea, formulate it as a problem, and discuss the various factors that affect the ability to retroscaffold. For obtaining a proof of concept, we conducted experiments on previously sequenced maize BAC data. The results show that (i) 3X/4X coverage sequencing is suited for exploiting the data's repeat content towards retroscaffolding, (ii) retroscaffolding can yield over 30% savings in finishing costs, and (iii) with retroscaffolding it is possible to opt for a lower sequencing coverage. These and other experimental results assessing the effects of various factors on retroscaffolding are presented in Section 3. As part of the NSF/DOE/USDA maize genome project[18], we are working on applying the retroscaffolding technique to the maize data as it becomes available. To this effect, we are developing an algorithmic framework to perform retroscaffolding as described in Section 4.

In Section 5, we present the results of our experiments to assess the effect of applying both clone mate based scaffolding and retroscaffolding on maize genomic data. Various strengths and limitations of the retroscaffolding technique are discussed in Section 6. Given that retrotransposons are abundant in genomes of numerous plant crops yet to be sequenced (e.g., wheat, barley, sorghum, etc.), the capability of retroscaffolding to exploit this repeat content can provide a significant means to reduce sequencing and finishing costs.

2. RETROSCAFFOLDING

Retrotransposons are DNA repeat elements abundant in several eukaryotic genomes — occupying at least 45% of the human genome[6], >50% in maize[15, 20], and up to 90% in wheat[7]. Long Terminal Repeat (LTR) retrotransposons constitute one of the most abundant classes of retrotransposons, and have been studied in relation to genome evolution, genomic rearrangements and retroviral transposition mechanisms[2, 3]. As their name suggests, LTR retrotransposons are distinctly characterized in their structure by two terminal repeat sequences — one each at the $5'$ and $3'$ ends of a retrotransposon inserted in a host genome. Given that these retrotransposons are typically 10-15 *kbp* long, their flanking LTRs can also be expected to be separated by as many *bp*s along the genome[a]. Moreover, the LTR sequences are identical at the time a retrotransposon inserts itself into a host genome, and gradually diverge over time due to mutations. Yet, the LTRs flanking most retrotransposons are similar enough for detection. These properties form the basis of our retroscaffolding idea, as explained below.

Low coverage sequencing of a genome with significant LTR retrotransposon content is likely to result in a proportionately large number of gaps that span these repetitive regions. If it so happens that the sequencing covers only the two LTRs of a given retrotransposon, a subsequent assembly can be expected to have two contigs each spanning one of the LTRs. Therefore, the detection of two identical or highly similar LTR-like sequences in two contigs is a necessary (but not sufficient) indication that the contigs sample the flanking regions of an inserted retrotransposon. If this indication can be further validated to sufficiency by searching for other structural signals of an LTR retrotransposon (described below), then the contigs can be relatively ordered and oriented (because LTRs are directed repeats). In addition, this implies that the intervening region between two consecutively ordered contigs contains retrotransposon related sequences — an information that can be used to prioritize the gaps for finishing, and potentially reduce efforts spent on finishing repetitive regions, if so desired.

The structure of a full-length LTR retrotransposon (illustrated in Figure 2a) is characterized by the following key attributes:

- **L1** The $5'$ and $3'$ LTRs share a "high" sequence identity.
- **L2** The starting positions of the $5'$ and $3'$ LTRs are at least D_{min} *bp* and at most D_{max} *bp* apart along the genome.
- **L3** Typically, LTRs start with TG and end in CA.
- **L4** The 5 (or 6) *bp* immediately to the left of the $5'$ LTR are "highly similar" (if not identical) to the 5 (or 6) *bp* immediately to the right of the $3'$ LTR. This repeat is referred to as a *Target Site Duplication* (*TSD*) because it corresponds to the 5 (or 6) *bp* duplicated in the host genome at the time and site of the retrotransposon's insertion.
- **L5** The intervening region between the $5'$ and $3'$ LTRs contains several signals that correspond to an inserted retrotransposon. These include a primer binding site (*PBS*), retrotransposon genes (*gag*, *pol*, and *env*), and a poly-purine tract (*PPT*).

For a sequence s, let $s^f = s$, and s^r denote its reverse complement. A sequence c is said to *contain* a sequence l if there exists between c and either l^f or l^r, a "good quality" alignment that spans a sufficiently "long" suffix or prefix of the latter sequence. Let an *LTR pair* $(l_{5'}, l_{3'})$ denote the two LTRs of a given LTR retrotransposon.

[a] Sometimes, LTR retrotransposons can be nested within one another, accordingly affecting the distances between the $5'$ and $3'$ LTRs.

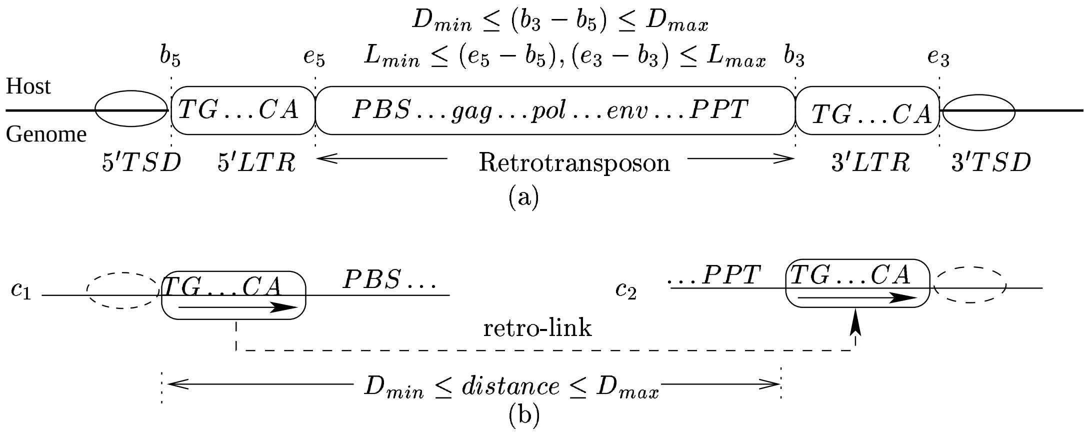

Fig. 2. (a) Structure of a full-length LTR retrotransposon. (b) An example showing two contigs c_1 and c_2 with a retro-link between them.

Definition of a Retro-link: Given a set L of n LTR pairs, two contigs c_i and c_j are said to be *retro-linked* if $\exists\ (l_{5'}, l_{3'}) \in L$ such that both c_i and c_j contain $l_{5'}$ or $l_{3'}$ or both.

An example of a retro-link between two contigs is shown in Figure 2b. As shown, the above definition is extended to account for additional structural attributes such as L3, L4 and L5, to ensure that a retro-link indeed spans the same full-length LTR retrotransposon. Details are omitted for brevity.

The Retroscaffolding Problem: Given a set C of m contigs and a set L of n LTR pairs, partition C such that:

- each subset is an ordered set of contigs, and
- every pair of consecutive contigs in each subset is retro-linked and there is no contig that participates in two retro-links in opposite orientations.

The retroscaffolding problem can be viewed as a variant of the standard scaffolding problem, called the *Contig Scaffolding Problem* that is NP-complete[9]. In the latter, the input is a set of contigs and a set of clone mates, where each clone mate pair is a pair of fragments sequenced from the same clone of a known approximate length. This is similar to the distance constraint imposed by a retro-link between the two contigs containing two LTRs of the same retrotransposon. In addition to the LTRs, a retro-link accounts for other structural attributes of an LTR retrotransposon. Also, like in the original scaffolding problem, not all retro-links may be used in the final ordering and orientation. Similar to the contig scaffolding problem, the retroscaffolding problem can be formulated as on optimization problem.

The effectiveness of retroscaffolding on a genome is dictated by the following factors:

LTR retrotransposon abundance: The ability to retroscaffold depends on the number of retro-links that can be established, which is limited by the number of detectable LTR retrotransposons in the genome. Note that this approach of exploiting the abundance in retrotransposons offers a respite from the traditional view that these are a source of complication in genome projects.

Presence of distinguishable LTRs: LTRs from different retrotransposons but from the same "family" may share substantial sequence similarity. Therefore, it is essential to take into account other structural evidence specific to an insertion before establishing a retro-link between two contigs. Even if the same LTR retrotransposon is present in two different locations of a genome, it can be expected that the TSDs are different because they correspond to the host genomic sequence at the site of insertion. It may still happen that a target genome contains the same family retrotransposons in abundant quantities, and other structural attributes become less distinguishable as well. If BAC-by-BAC sequencing is used, the above situation can be alleviated by applying retroscaffolding to contigs corresponding to the same BAC (instead of across BACs). This is because the likelihood of the same family occurring multiple times at a BAC level is much smaller than

Table 1. Summary of the LTR retrotransposons identified in 4 maize BACs using *LTR_par*.

	GenBank	BAC Length	Number of LTR	Retrotransposons in BAC	
	Accession	(in *bp*)	retrotransposons	Length in *bp*	% *bp*
BAC_1	AC157977	107,631	3	29,578	27%
BAC_2	AC160211	132,549	6	60,391	46%
BAC_3	AC157776	147,470	8	73,099	50%
BAC_4	AC157487	136,932	6	57,783	42%

Table 2. *LTR_par* parameter settings.

Parameter Name	Default Value	Description
D_{min}/D_{max}	600/15,000 *bp*	Distance constraints between 5′ and 3′ LTRs (L2)
τ	70%	% identity cutoff between 5′ and 3′ LTRs (L1)
L_{min}/L_{max}	100/2,000 *bp*	Minimum/maximum allowed length of an LTR
Match/mismatch	2/-5	Match and mismatch scores
Gap penalties	6/1	Gap opening and continuation penalties

at a genome level.

Sequencing coverage: Retroscaffolding targets each sequencing gap that spans an inserted retrotransposon such that its flanking LTRs are represented in two different contigs. Henceforth, we will refer to such gaps as *retro-gaps*. Given the length of such an insert ranges from 10-15 *kbp* (greater, if it is a nested retrotransposon), the coverage at which the genome is sequenced is a key factor affecting the ability to retroscaffold. If the sequencing coverage is too high (e.g., 10X), then there are likely be so few (short) sequencing gaps that the need for any scaffolding technique diminishes. Whereas at very low coverage (e.g., 1X) long sequencing gaps that may span entire LTR retrotransposons are likely to prevail.

3. PROOF OF CONCEPT OF RETROSCAFFOLDING ON MAIZE GENOMIC DATA

In this section, we provide a proof of concept for retroscaffolding. For this purpose, four finished maize BACs (listed in Table 1) were acquired from Cold Spring Harbor Laboratory[14]. The first step was to determine the LTR retrotransposon content of these BACs. *LTR_par*[11], which is a program for the *de novo* identification of LTR retrotransposons, was used to analyze each BAC with the parameters specified in Table 2. Table 1 summarizes the findings. As can be observed, the fraction of LTR retrotransposons in these BACs averages 42%, consistent with the latter's estimated abundance in the genome.

The effect of sequencing at different coverages was assessed as follows. A program that "simulates" a random shotgun sequencing over an arbitrary input sequence at a user-specified coverage was provided by Scott Emrich at Iowa State University[5]. Each run of the program produces a set (or *sample*) of fragments, along with the information of their originating positions. We ran this program on each BAC for coverages 1X through 10X, and for each coverage 10 samples were collected to simulate sequencing 10 such BACs. For each sample, using the knowledge of the fragments' originating positions, the set of all contiguous genomic stretches covered (and thereby the set of sequencing gaps) was determined. Ideally, assembling the sample would produce a contig for each contiguous stretch. Based on the placement information of the contigs on the BAC and that of the LTR pairs (Table 1) on the BAC, each LTR pair was classified into one of these three classes (see Figure 3):

- **CgC:** both LTRs are contained in two different contigs,
- **C_C:** both LTRs are contained in the same contig, and
- **GgX:** at least one LTR is not contained by any contig (i.e., it is located in a gap).

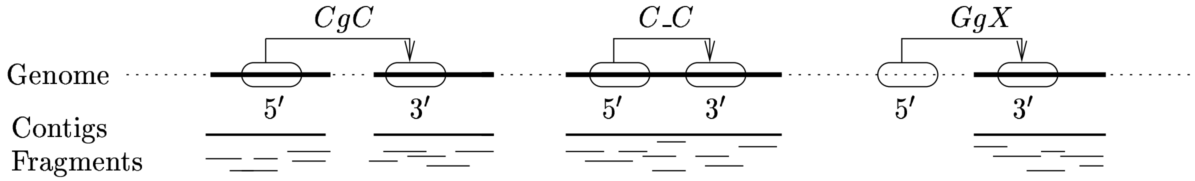

Fig. 3. Classification of LTR pairs based on the location of sequencing gaps, LTRs, and contigs. Dotted lines denote sequencing gaps. Retro-links correspond to the class CgC.

Table 3. Classification of the LTR pairs in 4 BACs, with respect to a set of 10 shotgun samples obtained from each BAC at different coverages.

Coverage	BAC_1				BAC_2	BAC_3	BAC_4
	CgC	C_C	GgX	$CgC\%$	$CgC\%$	$CgC\%$	$CgC\%$
1X	16	1	13	53	83	63	63
2X	26	0	4	87	95	77	92
3X	25	3	2	83	**100**	**97**	**100**
4X	27	3	0	**90**	**100**	88	**100**
5X	24	6	0	80	95	93	95
6X	22	8	0	73	83	76	98
7X	19	11	0	63	83	61	**100**
8X	18	12	0	60	77	64	67
9X	16	14	0	53	48	50	60
10X	7	23	0	23	37	31	43

In this classification scheme, it is easy to see that retro-links can be expected to be established only for CgC LTR pairs. Therefore, the ratio of the number of CgC LTR pairs to the total number of LTR pairs is indicative of the maximal value of retroscaffolding at a given coverage. We computed this ratio for each of the 4 BACs used in our experiments, by considering one coverage at a time, and counting the LTR pairs in each of the three classes over all 10 samples. From Table 3, we observe that the ratio is maximum for a 3X coverage for 3 out of the 4 BACs, and 4X for the other BAC. This implies that a 3X/4X coverage project is expected to best benefit from the retroscaffolding approach. To understand the above results intuitively, observe that a very high coverage has a high likelihood of sequencing an LTR retrotransposon region to entirety, making retroscaffolding unnecessary. While a very low coverage results in a high likelihood of LTRs falling in gaps, making retroscaffolding ineffective. Both these expectations are corroborated in our experiments — in Table 3, the gradual increase in C_C and the decrease in GgX with increasing coverage. The C_C increase with coverage also indicates the amount of efforts spent in sequencing retrotransposon-rich regions.

In our next experiment, we assess the potential savings that can be achieved at the finishing step through the information provided by retroscaffolding on gap content. Table 4 shows the number of gaps generated at various sequencing coverages, and the number of which can be detected using retroscaffolding (i.e., retro-gaps). While the results are shown only for two BACs (due to lack of space), we observed a similar pattern in all four BACs. As each retro-gap corresponds to a potential region of the genome that may not necessitate finishing, the ratio of the number of retro-gaps to the total number of sequencing gaps indicates the potential savings achievable at the finishing step because of retroscaffolding. From the table we observe this ratio ranges from 23%-40% for BAC_2, and 24%-49% for BAC_4; averaging over 34% savings for both BACs.

Table 4 also shows that sequencing BAC_2 at a 6X coverage is expected to result in ~37 sequencing gaps; while sequencing at a 4X coverage and subsequently applying retroscaffolding is expected to result in an effective 39 gaps ($\approx 65.7 - 26.6$). This implies that through retroscaffolding it is possible to reduce the coverage from 6X to 4X on BAC_2 without much loss of scaffolding information. As retroscaf-

Table 4. Number of retro-gaps vs. all sequencing gaps. Measurements are averaged over all 10 samples of each of the two BACs.

Coverage	BAC_2			BAC_4		
	All gaps	Retro-gaps	%Retro-gaps	All gaps	Retro-gaps	%Retro-gaps
1X	70.5	26.4	37.4	78.0	24.8	31.8
2X	88.7	33.6	37.9	93.5	33.4	35.7
3X	84.6	32.2	38.1	84.0	31.0	36.9
4X	65.7	26.6	40.5	64.5	19.5	30.2
5X	50.6	19.3	38.1	46.4	16.7	36.0
6X	37.4	13.7	36.6	39.5	13.2	33.4
7X	28.3	9.5	33.6	26.6	9.1	34.2
8X	18.7	6.5	34.8	19.1	6.3	33.0
9X	13.0	3.0	23.1	11.9	5.9	49.6
10X	9.3	2.7	29.0	9.5	2.3	24.2

folding can be used independent of clone mate information, we are working on evaluating the collective effectiveness of both clone mate-based scaffolding and retroscaffolding approaches. If similar results can be shown at a much larger scale of experimental data for a target genome, then retroscaffolding can be used to advocate for a low coverage sequencing, directly impacting the sequencing costs of repetitive genomes.

4. A FRAMEWORK FOR RETRO-LINKING

We developed the following two-phase approach to retroscaffolding. In the first phase, retro-links are established between contigs that show "sufficient" evidence of spanning two ends of the same LTR retrotransposon. Once retro-links are established, the process of scaffolding the contigs is the same as scaffolding them based on clone mate information, i.e., each retro-link can be treated equivalent to a clone mate pair that imposes distance and orientation constraints appropriate for LTR retrotransposon inserts. Therefore, in principle, any of the programs developed for the conventional contig scaffolding problem[1, 9, 10, 17, 19] can be used to achieve retroscaffolding from the retro-linked contigs[b]. In what follows, we describe our approach to establish retro-links.

There are two types of retro-links that can be established among contig data: (i) those that correspond to LTR retrotransposons that are already known to exist in the genome of the target organism or closely related species, and (ii) those that are *de novo* found in the contig data. The first class of retro-links can be established by building a database of known LTR retrotransposons and detecting contigs that overlap with LTR sequences of the same retrotransposon. However, such a database of already known LTR sequences of a target genome may hardly be complete in practice. For this reason, the second class of retro-links that are based on a *de novo* detection of LTR sequences in the contig data is preferable. However, additional validation will be necessary to ensure the correctness of such retro-links.

In what follows, we describe the algorithmic framework we developed to establish retro-links based on already known LTR retrotransposons, and the results of applying it on maize genomic data.

4.1. Building a Database of LTR Pairs

Given that the entire genome of maize has not yet been assembled, the first step in our approach is to build a database of maize LTR pairs from previously sequenced maize genomic data. A set of 560 known full-length LTR retrotransposons and 149 solo LTRs[c] was acquired from San Miguel[16]. In addition, a set of 470 maize BACs were downloaded from GenBank[5]. Because the information about the LTR sequences within the full-length retrotransposons and BACs

[b] For our experiments, we used the *Bambus*[19] program.

[c] Solo LTRs are typically the result of a deletion/recombination event at a site of an inserted LTR retrotransposon, in which only either a 5′ or a 3′ LTR (or a part of it) survives.

Table 5. Summary of LTR pairs predicted by *LTR_par*.

Input	Number of sequences	Number of full-length predictions	Number of LTR pairs
LTR retrotransposons[16]	560	556	556
Solo-LTRs[16]	149		149
Maize BACs[5]	470	1,234	1,234
		Total	1,939

was not available, we used the *LTR_par* program to identify LTR retrotransposons and their location information. We did not include the LTRs identified in the four maize BACs listed in Table 1, so that they can be used as benchmark data for validating retroscaffolding.

Given a set of sequences, *LTR_par* identifies subsequences within each sequence that bear structural semblance to full-length LTR retrotransposons. Desired values for structural attributes can be input as parameters. We used the values shown in Table 2. As part of each prediction, the locations of both the $5'$ and $3'$ LTRs are output. A prediction is made only if the identified region satisfies LTR sequence similarity (L1) and LTR distance (L2) conditions. Based on the presence of other signals such as the $TG..CA$ motif (L3) and TSDs (L4), each prediction is also associated with a "confidence level". A confidence level of 1 implies presence of both L3 and L4, 0.5 implies either L3 or L4 but not both, and 0 implies only L1 and L2. In this paper, we use level 1 predictions, although we are currently evaluating other combinations of LTR pairs from across confidence levels. Table 5 shows the statistics over the resultant total of 1,939 LTR pairs.

4.2. An Algorithm to Establish Retro-links

Let C denote a set of m contigs generated through an assembly of maize fragments corresponding to one BAC, and let L denote the set of n LTR pairs (n =1,939 in Table 1). Our algorithmic framework performs the following steps:

- **S1** Compute $P = \{(c,(l_{5'},l_{3'}))|c \in C,(l_{5'},l_{3'}) \in L, c$ contains $l_{5'}$ or $l_{3'}$ or both$\}$.
- **S2** Construct a set $G = \{G_1, G_2, \ldots, G_n\}$, such that $\forall G_i \subseteq C$, $\forall c \in G_i$, $(c,(l^i_{5'},l^i_{3'})) \in P$. Note that G need not be a partition of C. We call each G_i a *contig group*.
- **S3** $\forall G_i \in G$, compute $R_i = \{(c_i,c_j)|c_i,c_j \in G_i, c_i$ and c_j are retro-linked by $(l^i_{5'},l^i_{3'})\}$.

A naive way to perform step S1 is by evaluating each of the $m \times n$ pairs of the form $(contig, LTR\,pair)$, to check if a contig contains one of the LTRs. The check can be performed through standard dynamic programming techniques for computing semi-global alignments that take time proportional to the product of the lengths of the sequences being aligned. As reverse complemented forms also need to be considered, this approach involves $4 \times m \times n$ alignments in the worst case. We developed a run-time efficient method based on the observation that if two sequences that align significantly, then they also have a "long" exact match between them (although the converse need not hold). Thus it is sufficient to evaluate only pairs of the form $(contig, LTR)$ that have an exact match of a minimum cutoff length. For this purpose, we adapted a parallel algorithm for detecting maximal matches across DNA sequences that we had originally developed for a clustering problem[12]. The algorithm runs in linear space and run-time proportional to the number of the output pairs. For each generated pair, an optimal semi-global alignment is computed. Only pairs that have alignments satisfying a specified criteria are output. As pairs are output, the set G is computed as well in constant time per pair (step S2).

Steps S1 and S2 ensure that two contigs are paired if and only if they contain LTRs from the same LTR pair. To perform S3, it is therefore necessary only to establish additional structural evidence such as the presence of TSDs, PPT, PBS, and/or retrotransposon genes. The attributes to look for, however, depends on the location of the subsequences corresponding to the LTRs within the contigs — for e.g., it may not be possible to look for retrotrans-

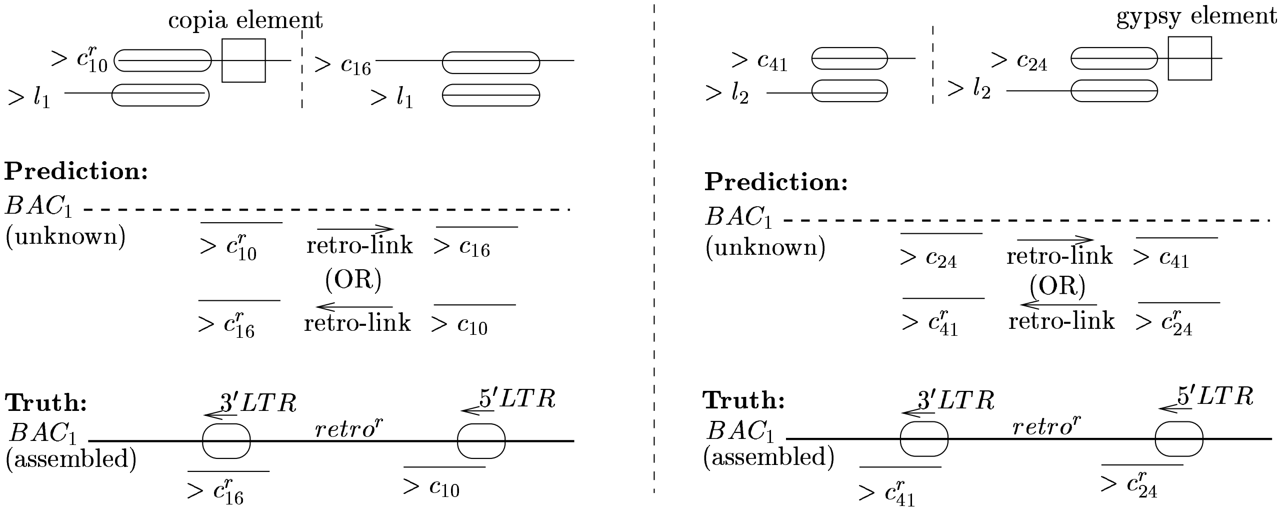

Fig. 4. Validation of two retro-links — between contigs c_{10} and c_{16}, and contigs c_{41} and c_{24}. Vertically aligned ovals denote overlapping regions, and squares denote retrotransposon hit through *tblastx* against the GenBank *nr* database.

poson genic sequences if the LTR regions within the contigs are a suffix of one contig and a prefix of another (see Figure 2b). We perform S3 as follows: we concatenate each pair of contigs under consideration in each of the 4 possible orientation combinations, and run *LTR_par* on the concatenated sequence. A retro-link is established between a pair only if sufficient structural evidence is detected.

Preliminary Validations:

We validated the retro-linking algorithm on BAC_1 of Table 1 as follows. Shotgun fragments were experimentally sequenced at a 3X coverage of the BAC[14], and were assembled[5] using the CAP3 assembler[8]. The resulting 45 contigs were input along with the 1,939 LTR pairs (in Table 5) to our retro-linking program. Note that the 1,939 LTR pairs do not include the 3 LTR pairs in BAC_1 as identified by *LTR_par* (Table 1) — that way, the validation reflects an assessment of retro-linking under practical settings in which a target BAC sequence and its LTR pairs are unknown prior to the retroscaffolding step. The experiment resulted in 44 contig groups ($= |G|$), and upon investigation we found that most of the groups were "equivalent", i.e., the corresponding LTR pairs share a significant sequence identity ($> 95\%$). The equivalent groups were merged.

The subsequent step was to evaluate each contig pair of a merged group for a valid retro-link. For detecting retrotransposon genic sequences in contigs, we queried the contigs against the GenBank *nr* database using the *tblastx* program. Other structural attributes were detected using *LTR_par*. This step resulted in only two retro-linked pairs: ($c^r_{10} \to c_{16}$), and ($c_{24} \to c_{41}$) with the arrows implying the order in which the contigs can be expected to occur along the "unknown" BAC sequence (BAC_1) in the specified orientations. We verified the predictions by aligning each of these 4 contigs directly against the known sequence of BAC_1 and found that the retroscaffolding prediction is correct (see Figure 4).

5. SCAFFOLDING WITH CLONE MATES AND RETRO-LINKS

Retroscaffolding differs from conventional contig scaffolding as it relies on the presence of LTR retrotransposons instead of the clone mate information. While this suggests that either of the techniques can be applied independent of one another, the output may themselves be not mutually exclusive — i.e., it is possible that the relative ordering and orientation between the same two contigs are implied by both the techniques. While such redundancies in output can be used as additional supporting evidence for bolstering the validity of scaffolding, the actual value added by either of these two techniques is dictated by its respective unique share in output scaffolding. Ideally, we would hope that these two outputs to

Table 6. Results of (i) scaffolding contig data for BAC_4 (136,932 *bp*) using clone mate information, (ii) retroscaffolding, and (iii) combined scaffolding using both clone mate and retro-link information.

	Clone mate scaffolding	Retroscaffolding	Combined scaffolding
Number of scaffolds	32	5	27
Total span of scaffolds (*bp*)	120,350	65,605	138,356
Average span of scaffold (*bp*)	3,760	6,246	4,457
Number of contig pairs scaffolded	42	10	71
Number of assembly gaps covered	22	17	28

complement one another.

We assessed the effect of a combined application of retroscaffolding and clone mate based scaffolding on real maize genomic contig data as follows: 62 contigs were generated by performing a CAP3 assembly over a 3X coverage set of fragments sequenced from BAC_4. Ideally, all 62 contigs would be part of just one "scaffold" if the contigs were all to be ordered along the target BAC.

The scaffolding achievable from just the clone mate information was first assessed by running the Bambus[19] program on the contigs. This resulted in 32 scaffolds spanning an estimated total of 120,350 *bp* and each with an average span of 3,760 *bp*. (Note that the "span" of a scaffold output by Bambus is only an estimate, because it includes the size estimated for sequencing gaps between the scaffolded contigs.) We then assessed the scaffolding achieved by retroscaffolding the contig data — retro-links were first established using the framework described in Section 4 and the output was transformed as input to Bambus. While retroscaffolding resulted in many fewer scaffolds (5), the total span was smaller (65,605 *bp*) when compared to clone mate scaffolding. However, the average span of each scaffold was almost twice as large in retroscaffolding. This is as expected because the distance constraint used for each retro-link was longer ($[5000, 15000]$) than that of clone mate links ($[2200, 3800]$).

In the next step, we input both the retro-link and clone mate information with their respective distance and orientation constraints to Bambus. This combination resulted in fewer scaffolds (27) and a longer total span (138,356 *bp*) than was achieved by just clone mate scaffolding — implying that retroscaffolding provides added information that is not provided by clone mate information. The above results are summarized in Table 6. The table also shows the number of contig pairs scaffolded as a result of the respective scaffolding strategies; the higher this number is, the more inclusive scaffolding is on the contigs — ideally, we would expect all contigs to be in one scaffold thereby implying $\binom{62}{2}$ contigs pairs.

We also assessed the individual effect of these scaffolding techniques on "assembly gaps": Each of the 62 contigs was individually aligned to the assembled BAC_4 sequence and the stretch along which each has a maximum alignment score was selected to be its locus on the BAC. A maximal stretch along the BAC not covered by any of the 62 contigs was considered an "assembly gap". There were a total of 42 such gaps. For each of the three scaffolding strategies (i.e., clone mate based, retroscaffolding and combined), an assembly gap is said to be "covered" (alternatively, "not covered") if there exists a (alternatively, does not exist any) pair of scaffolded contigs spanning the gap. Based on this definition, the number of covered assembly gaps was 22 for clone mate scaffolding, 17 for retroscaffolding, and 28 for the combined scaffolding. This further demonstrates the value added by retroscaffolding.

6. DISCUSSION

Our preliminary studies on maize genomic (Section 3) and the experimental results on maize contig data (Section 5) demonstrate a proof of concept and the value added by retroscaffolding in genome assembly projects. For retroscaffolding to be effective in a genome project, it is necessary that the LTR retrotransposons in the genome are both abundant and distinguishable. LTR sequences within the same family of LTR retrotransposons are harder to distinguish, and repeat-rich genomes (e.g., maize) could have numerous copies of the same family scattered across the genome. Therefore, applying retroscaffolding at a genome level may cause several spurious retro-links to be established, thereby confounding the process of scaffolding. It is for this reason that retroscaffolding is more suited for genome projects involving hierarchical (e.g., BAC-by-BAC) sequencing. Retroscaffolding can also be used to order and orient BACs, if the overlapping ends of two consecutive BACs along a tiling path span an LTR retrotransposon.

In genome projects which generate clone mate information, the scaffolding information derived from retroscaffolding may in part be already provided by clone mates. In the worst case, even if no new scaffolding information is provided by retroscaffolding, we can benefit from the scaffolding information provided by retroscaffolding in two ways: (i) we will have information about not only the genomic loci but also the composition of the assembly gaps covered by retroscaffolding, as they are expected to contain sequences corresponding to a retrotransposon insert. Therefore, we can prioritize the gaps to finish based on this information, and (ii) the scaffolding output by retroscaffolding can be used to as supporting evidence to validate the output of clone mate information.

Retroscaffolding will be useful in projects which do not generate clone mate information. New sequencing technologies such as the 454 sequencing[13] that do not generate clone mate information are increasingly becoming popular due to their high throughput and cost-effectiveness. Such sequencing technologies may be an appropriate choice for low-budget sequencing projects, and retroscaffolding could make the task of carrying out the assembly in such projects practically feasible.

Retroscaffolding also provides a mechanism to explore the feasibility of a lower coverage sequencing in genome projects. While reducing the sequencing coverage as low as 3X may expose more gaps to span LTR retrotransposons in a target genome, it also implies that there is less redundancy in fragment data. This might affect the quality of the output assembly, especially of those contigs corresponding to the non-repetitive regions of the genome. To circumvent this issue in a hierarchical sequencing project, we propose the following iterative approach to sequencing and assembly: first, sequence all the BACs at a low coverage and assemble them. If a subsequent retroscaffolding reveals the low repeat content in a subset of the input BACs, then perform additional coverage sequencing selectively on these BACs, and reassemble them with the fragments sequenced from all sequencing phases. In practice, this procedure can be repeated until sufficient information is gathered to completely assemble and scaffold each BAC. This approach provides a cost-effective mechanism to sequence repeat-rich genomes without compromising on the quality of the output assembly.

7. CONCLUSIONS

Genome projects of several economically important plant crops such as maize, barley, sorghum, wheat, etc., are either already underway or are likely to be initiated over the next few years. Most of these plant genomes contain an enormous number of retrotransposons that are not only expected to confound the assembly process, but are also expected to consume the bulk of the sequencing and finishing budget. In contrast to this perspective, the retroscaffolding approach proposed in this paper offers the possibility of exploiting the abundance of LTR retrotransposons, thus serving three main purposes: (i) to scaffold contigs that are output by an assembler, (ii) to guide the process of finishing by providing information on the unfinished regions of the genome, and (iii) to introduce the possibility of reducing sequencing coverage without loss of information regarding the sequenced genes and their relative ordering. Given that sequencing and finishing account for most of the expenditures in genome projects, continued research in developing this new methodology further could have a high impact.

Several developments have been planned as future work on this research. Specifically, we plan to evaluate the collective effectiveness of retroscaffolding and clone mate based scaffolding at a larger scale. The algorithmic framework for retroscaffolding is still at an early stage of development. Further validation of the framework on sequenced genomes and at much larger scales are essential to ensure an effective and high-quality application of our methodology in forthcoming complex genome projects. To this effect, the application of retroscaffolding on the on-going maize genome project will provide a good starting point.

ACKNOWLEDGMENTS

We thank the reviewers of an earlier version of this manuscript for providing several useful insights. We are grateful to Scott Emrich for the scripts to simulate shotgun sequencing and for discussions. We thank Richard McCombie for the maize BAC data, and Phillip San Miguel for the maize retrotransposon data. This research was supported by the NSF award DBI #0527192 and an IBM Ph.D. Fellowship to A. Kalyanaraman.

References

1. S. Batzoglou, D.B. Jaffe, K. Stanley, J. Butler *et al.* ARACHNE a whole-genome shotgun assembler. *Genome Research*, 12(1):177–189, 2002.
2. J.L. Bennetzen. The contributions of retroelements to plant genome organization, function and evolution. *Trends in Microbiology*, 4(9):347–353, 1996.
3. J.M. Coffin, S.H. Hughes, and H.E. Varmus. Retroviruses. *Plantview*, 1997.
4. International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409:860–921, 2001.
5. S.J. Emrich. *Personal Communication*, 2005.
6. E.S. Lander, L.M. Linton, B. Birren, C. Nusbaum, *et al.* Initial sequencing and analysis of the human genome. *Nature*, 409:860–921, 2001.
7. R.B. Flavell. Repetitive DNA and chromosome evolution in plants. *Philosophical Transactions of the Royal Society of London. B.*, 312:227–242, 1986.
8. X. Huang and A. Madan. CAP3: A DNA sequence assembly program. *Genome Research*, 9(9):868–877, 1999.
9. D.H. Huson, K. Reinert, and E. Myers. The greedy path–merging algorithm for sequence assembly. *Proc. International Conference on Research in Computational Biology (RECOMB)*, pages 157–163, 2001.
10. D.B. Jaffe, J. Butler, S. Gnerre, E. Mauceli *et al.* Whole-genome sequence assembly for mammalian genomes: Arachne 2. *Genome Research*, 13:91–96, 2003.
11. A. Kalyanaraman and S. Aluru. Efficient Algorithms and Software for Detection of Full-Length LTR Retrotransposons. In *Proc. IEEE Computational Systems Bioinformatics Conference*, pages 56–64, 2005.
12. A. Kalyanaraman, S. Aluru, V. Brendel, and S. Kothari. Space and time efficient parallel algorithms and software for EST clustering. *IEEE Transactions on Parallel and Distributed Systems*, 14(12):1209–1221, 2003.
13. M. Margulies, M. Egholm, W.E. Altman, S. Attiya, *et al.* Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437:376–380, 2005.
14. R. McCombie. *Personal Communication*, 2005.
15. B.C. Meyers, S.V. Tingey, and M. Morgante. Abundance, distribution, and transcriptional activity of repetitive elements in the maize genome. *Science*, 274:765–768, 1998.
16. P.S. Miguel. *Personal Communication*, 2005.
17. J.C. Mullikin and Z. Ning. The phusion assembler. *Genome Research*, 13:81–90, 2003.
18. NSF. NSF, USDA and DOE Award $32 Million to Sequence Corn Genome. `http://www.nsf.gov/news/news_summ.jsp?cntn_id=104608\&org=BIO\&from=news`, Press Release 05-197, 2005.
19. M. Pop, D.S. Kosack, and S.L. Salzberg. Hierarchical scaffolding with Bambus. *Genome Research*, 14:149–159, 2004.
20. P. SanMiguel, B.S. Gaut, A. Tikhonov, Y. Nakajima, and J.L. Bennetzen. The paleontology of intergene retrotransposons of maize. *Nature Genetics*, 20:43–45, 1998.