Identification of novel chromosomal rearrangements via massively parallel sequencing and primer-directed *in silico* assembly

Joseph Fass[1], Vince Buffalo[1], Shyh-Jen Shih[2], Manuel Diaz[3], Andrew Vaughan[2], Dawei Lin[1]

[1] UC Davis Genome Center Bioinformatics Core, Davis, CA 95616

[2] UC Davis Medical Center Department of Radiation Oncology, Sacramento, CA 95817

[3] Loyola University Medical Center, Maywood, IL 60153

UC DAVIS Genome Center

Bioinformatics Core

This work was supported by NIH Grant 3 P01 CA105049-05S1

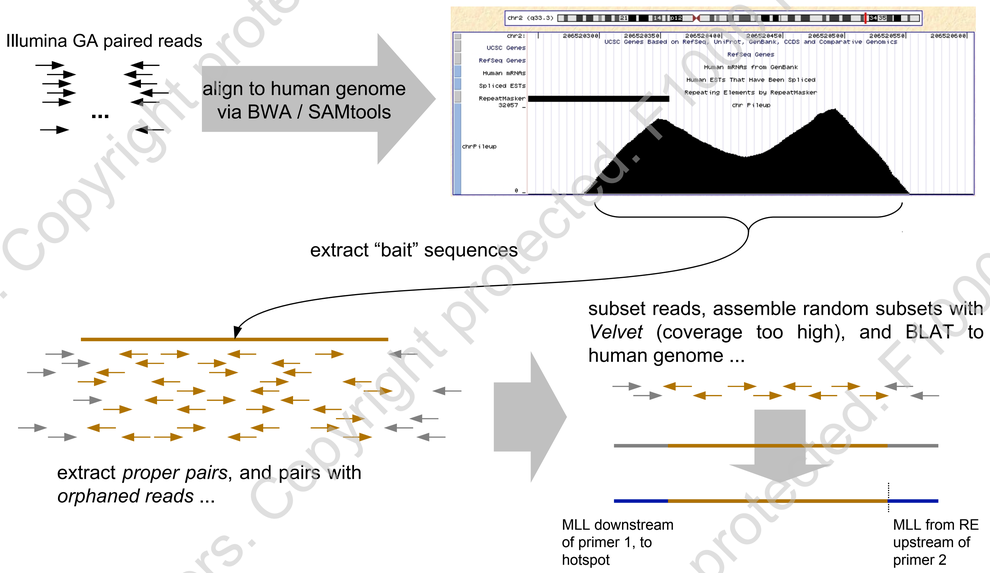
ABTRACT

Human *MLL* contains a site of frequent chromosomal rearrangement in instances of therapy-related luekemia. Previously, inverse PCR (IPCR) followed by gel extraction and Sanger sequencing has been used to interrogate this translocation hotspot and determine chromosomal rearrangement partners and breakpoints. To scale up this rearrangement detection procedure, we performed high-throughput sequencing and tested two short read assemblers, as well as a custom semi-greedy heuristic assembler, for correct assembly of IPCR rearrangement products. We found that currently available short read assemblers fail to correctly and completely assemble IPCR product sequences from a test lane of single-ended Illumina reads, while our novel, Semi-Greedy Assembler (SEGRASS) assembled four IPCR products that have been cross-validated by PCR and Sanger sequencing. Our assembler is designed for sequenced molecules that have regions of high or full identity among them, and which may have been sequenced at different, ultra-high depths within a set of short reads.

Assembly attempts with *Velvet*, *SSAKE*

While the use of paired-end sequences allowed successful assembly of IPCR products (see box to right), we wished to avoid both the cost of paired-end sequencing and the ambiguity in selecting candidate genomic regions. Therefore, we attempted *de novo* assembly of single-ended Illumina reads using Velvet (Zerbino 2009 *PLoS One* 4:e8407; Zerbino 2008 *Genome Research* 28:821). Velvet, run on a full lane, or ~8M reads, and on various subsamples down to ~1 / 3,300 of the lane's reads, assembled contigs that were, at most, less than half the length of the validated IPCR products. In many cases, subsequent read mapping and sequence alignment demonstrated that these contigs were short sections of the real products. Partial assembly may have been due to the partial sequence identity among IPCR products, combined with varying high depths of sequencing. In addition to Velvet, we also attempted to assemble the IPCR products using SSAKE (Warren 2007 *Bioinformatics* 23:500) on quality-trimmed Illumina reads, seeded with the sequences of primer 1 or 2. SSAKE successfully assembled the most common sequence in the data set – in this case, unrearranged MLL – but not the desired rearrangement products. SSAKE is designed to call a single consensus sequence during extension, which disallows multiple sequences that branch from identical sequence. ISSAKE (Warren 2009 *Bioinformatics* 25:458) addresses this possibility, but since the branch points are unknown *a priori*, their identification for use in ISSAKE would further complicate the assembly procedure.

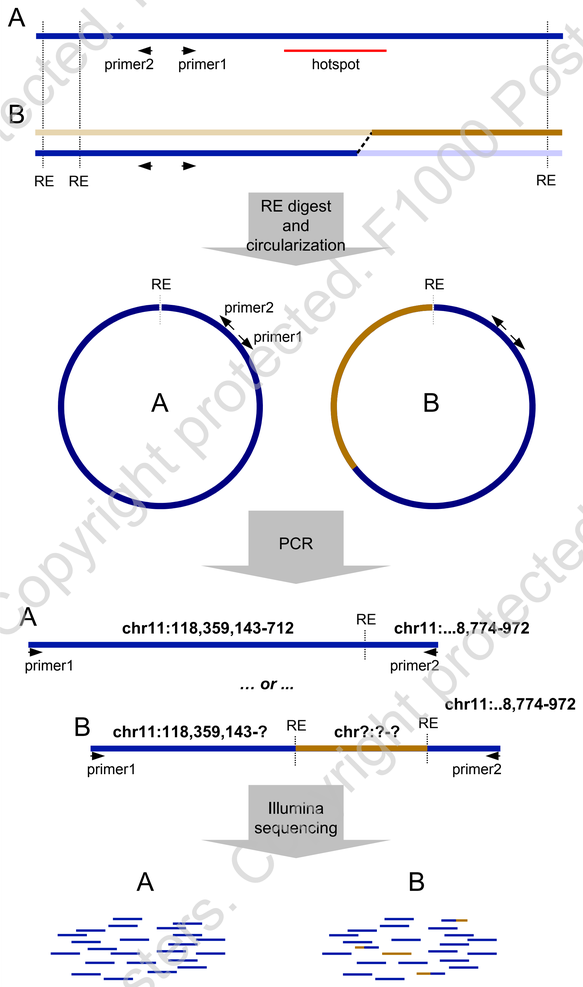
Genome alignment, read pair extraction, and assembly



Our first approach involved aligning all paired-end reads to the human genome using BWA (Li 2009 *Bioinformatics* 25:1754) and SAMtools (Li 2009 *Bioinformatics* 25:2078), then extracting regions outside of MLL with extremely high coverage. By selecting read pairs that either mapped as proper pairs to these regions, or with one read "orphaned," and then assembling (a subset; very high coverage defeats assembly) with Velvet, we obtained four candidate IPCR products to be cross-validated by gel extraction and Sanger sequencing.

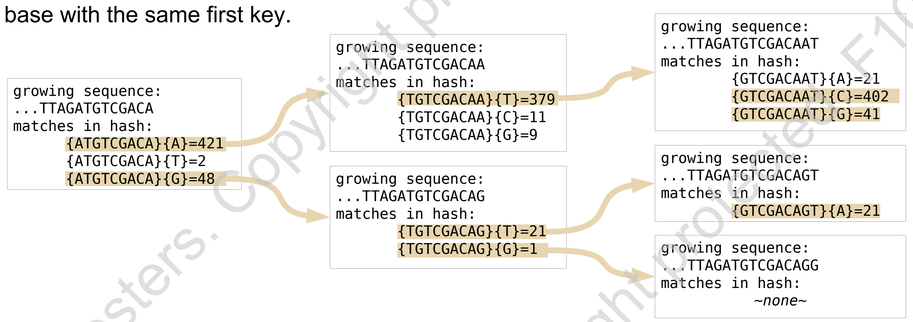
Targeted Sequencing of Rearrangement Hotspot

Primers were chosen to amplify restriction enzyme fragments containing either unrearranged MLL (A), or a fusion between MLL on chromosome 11 and another chromosome (B).

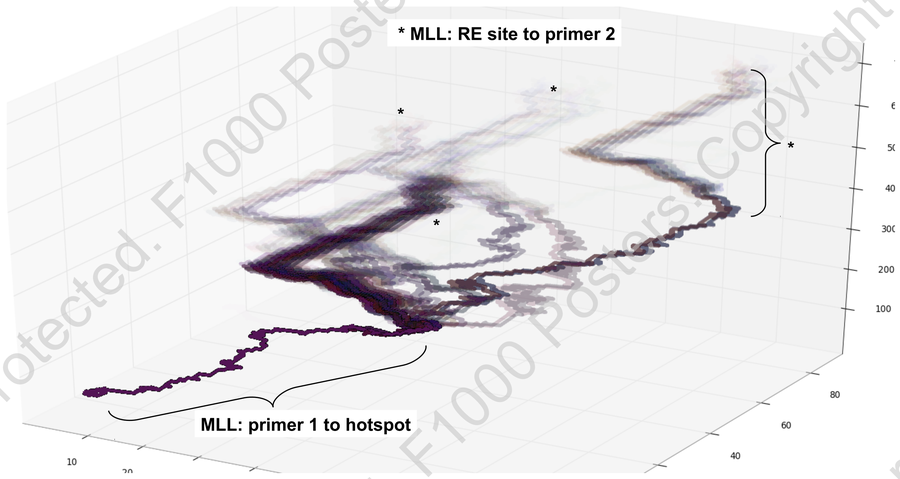


SEGRASS (Semi-Greedy Assembler)

Our assembler hashes single-ended Illumina reads into a double-keyed hash, where the two keys: {first *k* bases of read} and {*k+1*[th] base} correspond to a value equal to the number of times those (*k+1*) bases are seen at the 5'-end of a read. A sequence is extended with the *k+1*[th] base if the frequency (hash value) is either a minimum value *m*, or at least a fraction *1/r* of the most frequent *k+1*[th] base with the same first key.



Sequences assembled as above, starting with primer 1, are visualized below using a technique described in Vlachos M, et al. (2007. Visual Exploration of Genomic Data, *in* Lecture Notes in Computer Science: Knowledge Discovery in Databases: PKDD 2007, pp. 613-20). Briefly, each successive base in a sequence adds a unit vector according to the base type; thus, sequence paths remain collinear as long as they are identical, but diverge at polymorphic sequence. Subsequent identical sequence will be parallel, but not collinear. Sequence paths have low alpha (transparency) values, so dark lines indicate many overlapping sequences.



Results / Conclusions

In this application, where a small number of short sequences with full identity along part of their length were sequenced at extremely high read depth, several current assemblers – Velvet and SSAKE – were unable to correctly assemble the full original sequences. We wrote SEGRASS to address the specific challenges of our data: extremely high depth (with concordant high counts of error-containing reads), and branching assembly paths due to partial identity in the target sequences. SEGRASS effectively assembles this type of sequence data by extending sequences in a semi-greedy fashion. This ensures that low count, likely erroneous reads lead to the termination of growing sequences, while branch points with frequent reads supporting multiple candidate bases for extension allow sequence duplication and further extension. After termination of all sequences, clustering at high identity removes the short sequences that partially duplicate true target sequences, but were terminated due to incorporation of errors. In the data shown (see box to left), this procedure removed the complexity in the assembled sequence set, and resulted in four sequences that were ~99% identical to the four sequences generated using paired read information (see box above), that were cross-validated using PCR on genomic DNA, followed by Sanger sequencing.

Performance: SEGRASS assembled one lane of untrimmed 63 bp Illumina reads (~8M reads) in ~5 minutes; however, most of this time was spent in the clustering step. Future developments will include read flagging to avoid repeat loops, and the creation of multiple hashes to accommodate discontinuous coverage.