

INSERTION_PENALTY: -40

OPENING_PENALTY: -10

DELETION_PENALTY: -5

The relationship between the gap penalties (INSERTION_PENALTY, OPENING_PENALTY) above help limit the number of gap openings, i.e., favor grouping gaps together, by setting the gap insertion penalty higher than the gap opening cost. Of course, alternative relationships between MISMATCH_PENALTY, MATCH_BONUS, INSERTION_PENALTY, OPENING_PENALTY and DELETION_PENALTY are possible.

Once the alignment is complete, the aligned sequences can be assembled to produce a sequence that can be compared to a reference (i.e., a genetic standard) to identify variants. The variants can provide insight regarding diseases, stages of disease, recurrence and the like. In the case of amino acid alignments, the assembled amino acid sequences can be compared to a standard to determine evolutionary information about the protein, or functional information about the protein. This standard method of disease comparison is time consuming, however, because many of the variants are not necessarily correlated with a disease. For example, when the genetic standard is from a population having an ancestry different from the sample, many of the called variants are due to differences in things like hair color, skin color, etc.

SUMMARY

The ~~invention provides~~disclosure describes algorithms and methods for their implementation that transform linear, local sequence alignment processes such as, for example, Smith-Waterman-Gotoh, into multi-dimensional alignment algorithms that provide increased parallelization, increased speed, increased accuracy, and the ability to align reads through an entire genome. Algorithms ~~of the invention~~disclosed herein provide for a “look-back” type analysis of sequence information (as in Smith-Waterman), however, in contrast to known linear methods, the look back ~~of the invention~~disclosed herein is conducted through a multi-dimensional space that includes multiple pathways and multiple nodes in order to provide more accurate alignment of complex and lengthy sequence reads, while achieving lower overall rates of mismatches, deletions, and insertions. In one aspect, the present invention refers to a system for aligning a plurality of sequence reads comprising a processor and non-transitory memory, wherein the memory comprises instructions that, when executed, cause the processor to obtain a plurality of

sequence reads as strings of symbols; compare each string of symbols corresponding to a sequence read to a plurality of positions in a reference sequence construct, wherein the construct comprises at least two different strings of symbols at multiple positions in the construct, score overlaps between each string of symbols corresponding to a sequence read and each of the plurality of positions in the reference sequence construct, wherein a higher score corresponds to a greater amount of overlap; identify the overlap corresponding to the highest score for each sequence read; assign each sequence read to a location on the construct corresponding to the highest score; and write a file to memory corresponding to the location of each aligned sequence read. In another aspect, the present invention refers to a method of aligning a plurality of sequence reads, comprising obtaining a plurality of sequence reads; scoring sequence overlaps for each sequence read against a reference sequence construct, the construct comprising at least two alternative sequences per position at multiple positions in the construct, wherein greater overlap results in a higher score; and aligning each sequence read to a location in the construct such that the score for each sequence read is maximized. In yet another aspect, the present invention refers to a method of aligning a plurality of nucleic acid reads, comprising obtaining a plurality of nucleic acid reads; aligning the nucleic acid reads against a reference sequence construct using an algorithm that scores overlaps between a base in the read and the construct, the construct comprising at least two alternative sequences per position at multiple positions in the construct. In a further aspect, the present invention refers to a method of aligning a plurality of amino acid reads, comprising obtaining a plurality of amino acid reads aligning the amino acid reads against a reference sequence construct using an algorithm that scores overlaps between an amino acid in the read and the construct, the construct comprising at least two alternative sequences per position at multiple positions in the construct. In yet another aspect, the present invention refers to a computer-implemented method of aligning a plurality of reads, comprising obtaining a plurality of reads as strings of symbols; comparing each string of symbols corresponding to a read to a plurality of positions in a reference sequence construct using a computer processor, wherein the construct comprises at least two different strings of symbols at multiple positions in the construct, scoring overlaps between the string of symbols and each of the plurality of positions in the reference sequence construct with a processor, wherein a higher score corresponds to a greater amount of overlap; identifying the overlap corresponding to the highest score for each read; and aligning the reads to a location on the construct corresponding to

the highest score. In a further aspect, the present invention refers to a computer implemented method of aligning a plurality of reads, comprising obtaining a plurality of reads as strings of symbols; aligning the reads against a reference sequence construct using a processor executing an algorithm that scores overlaps between the read and the construct, the construct comprising at least two alternative sequences per position at multiple positions in the construct. In yet another aspect, the present invention refers to a system for aligning a plurality of reads comprising a processor and memory, wherein the memory comprises instructions that, when executed, cause the processor to obtain a plurality of reads as strings of symbols; align the reads against a reference sequence construct using an algorithm that scores overlaps between the read and the construct, the construct comprising at least two alternative sequences per position at multiple positions in the construct; and write a file to memory corresponding to the aligned sequence. In one aspect, the present invention refers to a method of aligning a plurality of sequence reads, comprising obtaining a plurality of sequence reads; scoring sequence overlaps for each sequence read against a reference sequence construct, the construct comprising at least two alternative sequences per position at multiple positions in the construct, wherein greater overlap results in a lower score; and aligning each sequence read to a location in the construct such that the score for each sequence read is minimized. In another aspect, the present invention refers to a computer-implemented method of aligning a plurality of reads, comprising obtaining a plurality of reads as strings of symbols; comparing each string of symbols corresponding to a read to a plurality of positions in a reference sequence construct using a computer processor, wherein the construct comprises at least two different strings of symbols at multiple positions in the construct, scoring overlaps between the string of symbols and each of the plurality of positions in the reference sequence construct with a processor, wherein a lower score corresponds to a greater amount of overlap; identifying the overlap corresponding to the lowest score for each read; and aligning the reads to a location on the construct corresponding to the lowest score. In a further aspect, the present invention refers to a system for aligning a plurality of reads comprising a processor and memory, wherein the memory comprises instructions that, when executed, cause the processor to obtain a plurality of reads as strings of symbols; compare each string of symbols corresponding to a read to a plurality of positions in a reference sequence construct, wherein the construct comprises at least two different strings of symbols at multiple positions in the construct, score overlaps between the string of symbols and each of the plurality of positions in the reference

sequence construct, wherein a lower score corresponds to a greater amount of overlap; identify the overlap corresponding to the lowest score for each nucleic acid read; assign the nucleic acid reads to a location on the construct corresponding to the lowest score; and write a file to memory corresponding to the aligned sequence.

In practice, the ~~invention~~-disclosure is implemented by aligning sequence reads to a series of directed, acyclic sequences spanning branch points that account for all, or nearly-all, of the possible sequence variation in the alignment, including insertions, deletions, and substitutions.

Such constructs, often represented as directed acyclic graphs (DAGs) can be easily assembled from available sequence databases, including “accepted” reference sequences and variant call format (VCF) entries. When combined with DAGs, or other directional construct, the disclosed algorithm thus provides a multi-dimensional approach to sequence alignment that greatly improves alignment accuracy and allows for sequence resolution not possible with conventional algorithms. The techniques can be used with any sequence information, in fact, however they are most useful for aligning nucleic acid sequences and amino acid sequences, as discussed herein.

The ~~invention~~ disclosure additionally ~~provides~~ describes methods to make specific base calls at specific loci using a reference sequence construct, e.g., a DAG that represents known variants at each locus of the genome. Because the sequence reads are aligned to the DAG during alignment, the subsequent step of comparing a mutation, vis-à-vis the reference genome, to a table of known mutations can be eliminated. Using the disclosed methods, it is merely a matter of identifying a nucleic acid read as being located at a known mutation represented on the DAG and calling that mutation. Alternatively, when a mutation is not known (i.e., not represented in the reference sequence construct), an alignment will be found and the variant identified as a new mutation. The method also makes it possible to associate additional information, such as specific disease risk or disease progression, with known mutations that are incorporated into the reference sequence construct. Furthermore, in addition to having the potential to find all genetically relevant results during alignment, the disclosed methods reduce the computational resources required to make the alignments while allowing for simultaneous comparison to multiple reference sequences.

The ~~invention~~ disclosure additionally includes methods for constructing a directed acyclic graph data structure (DAG) that represents known variants at positions within the sequence of an organism. The DAG may include multiple sequences at thousands of positions, and may include multiple variants at each position, including deletions, insertions, translations, inversions, and single-nucleotide polymorphisms (SNPs). It is also possible to tag each variant in the DAG with correlated diagnostic information, such as “breast cancer,” thereby reducing the steps needed to identify the risks to a patient providing a sample. In some ~~embodiments~~ examples, the variants will be scored, weighted, or correlated with other variants to reflect the prevalence of that variant as a marker for disease.

The ~~invention disclosure~~ additionally includes systems for executing the methods of the ~~invention as disclosed herein~~. In one ~~embodiment example~~, a system comprises a distributed network of processors and storage capable of comparing a plurality of sequences (i.e., nucleic acid sequences, amino acid sequences) to a reference sequence construct (e.g., a DAG) representing observed variation in a genome or a region of a genome. The system is additionally capable of aligning the nucleic acid reads to produce a continuous sequence using an efficient alignment algorithm. Because the reference sequence construct compresses a great deal of redundant information, and because the alignment algorithm is so efficient, the reads can be tagged and assembled on an entire genome using commercially-available resources. The system comprises a plurality of processors that simultaneously execute a plurality of comparisons between a plurality of reads and the reference sequence construct. The comparison data may be accumulated and provided to a health care provider. Because the comparisons are computationally tractable, analyzing sequence reads will no longer represent a bottleneck between NGS sequencing and a meaningful discussion of a patient's genetic risks.

BRIEF DESCRIPTION OF THE FIGURES

FIG. 1 depicts the construction of a directed acyclic graph (DAG) representing genetic variation in a reference sequence. FIG. 1(A) shows the starting reference sequence and the addition of a deletion. FIG. 1(B) shows the addition of an insertion and a SNP, thus arriving at the Final DAG used for alignment;

FIG. 2 depicts three variant call format (VCF) entries represented as directed acyclic graphs;

FIG. 3(A) shows a pictorial representation of aligning a nucleic acid sequence read against a construct that accounts for an insertion event as well as the reference sequence;

FIG. 3(B) shows the matrices and the backtrack used to identify the proper location of the nucleic acid sequence read "ATCGAA";

FIG. 4 depicts an associative computing model for parallel processing;

FIG. 5 depicts an architecture for parallel computation.

47. The method of claim 29, wherein each read comprises a string of symbols representing a measured sequence of amino acids.

48. The method of claim 47, wherein the measured sequence of amino acids is determined with mass spectrometry.

49. The method of claim 29, wherein the reference sequence construct comprises greater than 1,000 bases.

50. The method of claim 49, wherein the reference sequence construct comprises greater than 1,000,000 bases.

51. The method of claim 29, wherein the plurality of reads comprises greater than 1000 reads.

52. The method of claim 29, wherein the majority of the plurality of reads are greater than 100 base pairs in length.

53. The method of claim 29, wherein the majority of the plurality of reads are greater than 10 amino acids in length.

54. The method of claim 29, wherein the reference sequence construct comprises a database of positions and sequences stored on a computer-readable medium.

55. The method of claim 54, wherein the reads comprise strings of symbols stored on a computer-readable medium.

56. The method of claim 29, further comprising sequencing a plurality of nucleic acids from a sample.

57. The method of claim 56, further comprising ~~obtaining~~ the sample obtained from a patient.