A Novel High Accuracy Algorithm for Reference Assembly in Colour Space

Balazs Gor *balazs.gor@astridbio.com*
Astrid Research Inc.
Debrecen, Hungary

Anett Balla *anett.balla@astridbio.com*
Astrid Research Inc.
Debrecen, Hungary

Edit Tukacs *edit.tukacs@astridbio.com*
Astrid Research Inc.
Debrecen, Hungary

Faculty of Informatics
University of Debrecen
Debrecen, Hungary

Istvan Nagy *nagy.istvan@brc.mta.hu*
Institute of Biochemistry,
Biological Research Center of the
Hungarian Academy of Sciences
Szeged, Hungary

Zsolt Torok *zsolt.torok@astridbio.com*
Astrid Research Inc.
Debrecen, Hungary
Medical and Health Science Centre
University of Debrecen
Debrecen, Hungary

Abstract

Although numerous algorithms exist for genome alignment using Next Generation Sequencing tags, assembly of colour coded reads remains a challenge. We present a novel pairwise sequence aligner algorithm derived from Smith-Waterman method. Original feature of the algorithm is that it translates the reference sequence into colour code and performs the alignment in colour space. While operating on this base it can prevent most read error-derived assembly errors. Based on dynamic programming it gives the optimal alignment in colour space. Further, validation on empirical dataset with capillary sequencing proved high mapping accuracy. We can also conclude that the novel algorithm provides performance comparable with the currently available solutions. The algorithm can be implemented into any reference assembly software thereby improving mapping accuracy while maintaining high speed mapping.

Keywords: Smith-Waterman, Reference Assembly, Colour Space Code, Algorithm, Next Generation Sequencing

1. INTRODUCTION

Next Generation Sequencing (NGS), also known as high-throughput or massively parallel sequencing, is currently the leading genomic technology, enabling researchers to perform analysis at a whole-genome level. [1] Rapid evolution of the NGS technology has led to the fact that producing millions of sequence tags is not an issue anymore. [2, 3] Handling such a huge

amount of sequence data yielded a challenge on its own, but it is the complexity of the data interpretation that is the primary problem. From the sequencing point of view, currently two approaches dominate the market: sequencing by synthesis for example Genome Analyzer (Illumina Inc., San Diego, CA, USA) [4] and 454 FLX (F. Hoffmann-La Roche AG, Basel, Switzerland) [5, 6] and sequencing by ligation which is utilized by the SOLiD System [6] (Applied Biosystems, now part of Life Technologies Corporation, Carlsbad, CA, USA). [7, 8]

Nucleotide space outputs characterize both Genome Analyzer and 454 FLX while the SOLiD System has a unique output format consisting of colour coded short read tags. In a given tag two adjacent nucleotides define a single colour, subsequently the nucleotide and the colour coded sequence can be converted into each other with the help of an adaptor nucleotide [9, 10].(Figure 1)

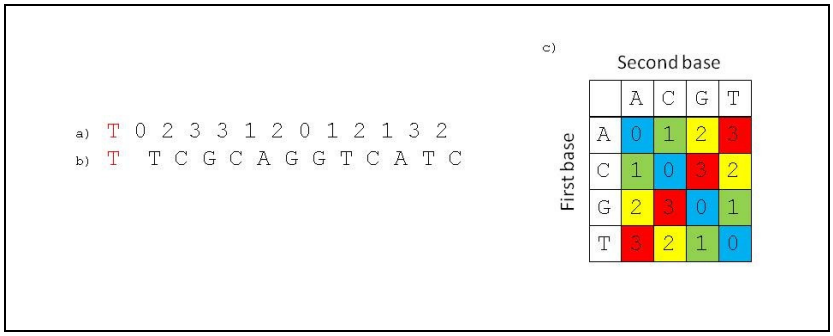

FIGURE 1: *Colour coded sequence*. This figure represents the a) colour coded tag and b) its translation to a nucleotide sequence. The c) conversion matrix for the translation is the same for every SOLiD sequencer. Note that a starter adaptor nucleotide is needed to define the correct translation; in our example it is the first T nucleotide in red.

Nowdays many bioinformatics research groups have been working on specialized algorithms that are able to assemble the colour coded tags, and a couple of these algorithms have already been developed and gained popularity. [11, 12] These algorithms utilizing different indexing and alignment methods have been published and are publicly available, for example SHRiMP [13], Bowtie [14], BFAST [15, 16] and BWA [17]. The most important features of SHRiMP, an algorithm based on the Smith-Waterman method [18], are read indexing with the help of a hash table, q-gram filter approach and spaced seeds. [19] Bowtie indexes the genome with a process based on Burrows-Wheeler transformation and subsequently uses a backtracking algorithm for finding the alignments with significant quality. [20] To accelerate the process, a 'double indexing' strategy is applied, which eliminates unnecessary backtracking. By using a hash table BFAST indexes the whole genome instead of the reads. It identifies candidate local alignments which are used to speed up the mapping of the reads. Mismatches, mutations, including small insertions and deletions are also considered during alignment made with arbitrary multiple independent indexes. The Smith Waterman method is implemented as well to make a precise read alignment. BWA is another package that uses Burrows-Wheeler transformation. It performs backward search, during which gaps and mismatches are also considered. It is very efficient and can handle large reference genomes, working either in colour or base space. [21]

Our goal was to develop a novel, easily tuneable mapping algorithm that works in colour space and enables the optimal read alignment with the highest possible accuracy without much focus on speed and run time. During the alignment, first we turn the reference genome into a colour coded sequence to realize mapping in colour space. Utilizing this novel approach, we are able to make use of tags containing read errors which are otherwise excluded at assembles performed using nucleotide-based algorithms. This way coverage increases, and a more accurate sequence alignment results; consequently variants, such as single and multi nucleotide polymorphisms (SNPs and MNPs, respectively), insertions and deletions are identified with more certainty.

Here we present key features, underlying methods and implementation of the newly developed algorithm. The performance was tested on empirical data obtained by the NGS sequencing of a *Propionibacterium acnes* (*P. acnes*) strain and the accuracy was validated by capillary sequencing as well as a performance comparison was done on a dataset available in the NCBI SRA database.

2. METHODOLOGY AND RESULTS

In the frame of our work we have developed a highly accurate local sequence aligner for short read assembly in colour space. The algorithm is implemented in the modular BFAST for testing, and its accuracy was validated in one of our biological projects.

2.1 Description of the Algorithm

The following sections describe the mathematical principles which the algorithm is based on. It is demonstrated on general and on specific examples how the algorithm is able to find the optimal local alignment recursively in colour space.

2.1.1 Mapping in Colour Space

At mapping output reads to the reference sequence identity or variation of colour codes need to be determined. To this purpose colour space algorithms define a $\oplus$ operation. The operation is a special summation that maps the Cartesian product of a given set with itself to the original set. The operation with the given set yields a so called Klein group, which is a kind of Abel group. Its most important characteristics: the set is closed under this operation; the operation is associative; the set has an identity element; each element can be inverted; the operation is commutative.

Consider the $\oplus$ operation as $\oplus$: $\{0,1,2,3\} \times \{0,1,2,3\} \rightarrow \{0,1,2,3\}$ defined by the operation table (Table1).

$\oplus$	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

TABLE 1: *The $\oplus$ operation table.* The cells of the first row and the first column contain the possible colour codes included in the operation. Cells in the intersection show the colour codes assigned by the operation.

The composition of the$\oplus$ operation is used to detect SNPs, MNPs, insertions, deletions and read errors [1]. SNPs and indels can also occur next to each other, and that case can also be described by the $\oplus$ operation.

To get a clear idea of the operation and mutation detection, consider the partial sequences that originate from given positions of two colour coded sequences and are alignable. The partial sequences with the length of p and q started from the i-th position of the $A = a_0,...,a_i,...,a_{i+p},...,a_n$ sequence and the j-th position of the $B = b_0,...,b_j,...,b_{j+q},...,b_m$ sequence, respectively, can be aligned if $a_i \oplus ... \oplus a_{i+p} = b_j \oplus ... \oplus b_{j+q}$ holds, where

$$a_0,\ldots,a_n,b_0,\ldots,b_m \in \{0,1,2,3\}, \quad p,q,i,j \in \mathrm{N} \cup \{0\}, i+p \le n,\ j+q \le m.$$

We can infer the following cases from the length of the alignable partial sequences:
In the i-th position of sequence A and the j-th position of sequence B,

I. p=q=0 indicates a Match,
II. a, p=q=1 indicates an SNP,
b, p=q>1 indicates an MNP,
III. p=0 and q>0, or q=0 and p>0 indicates a Gap,
IV. p≠q and p,q>0 indicates a Mix.

The Mix function consists of an adjacent SNP and an insertion or a deletion. Figure 2 shows the above mentioned four alignment possibilities.

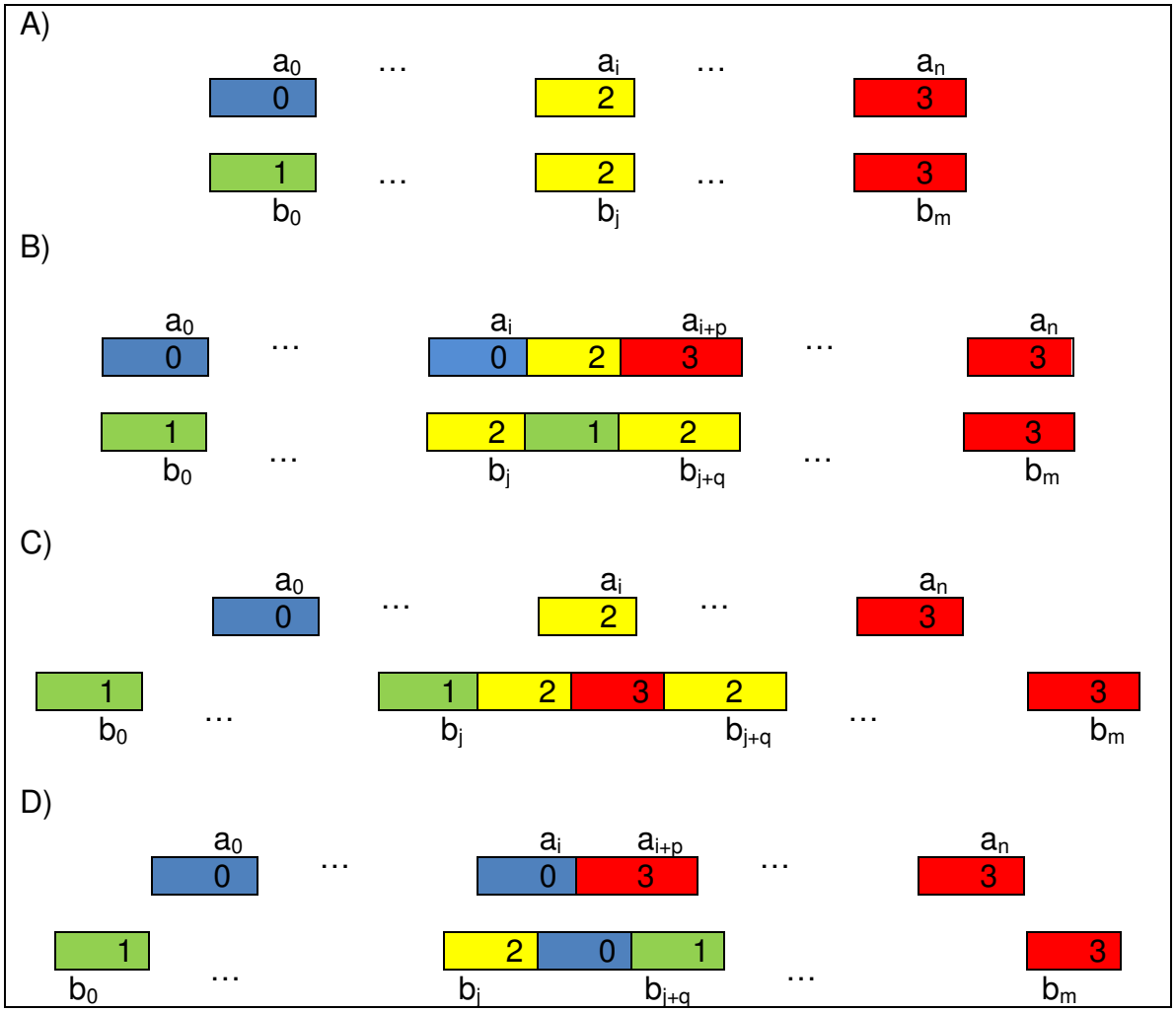

FIGURE 2: *Alignment in colour space.* a) a Match is shown, where p=q=0; b) example of a SNP, where p=q=2; a Gap is shown in c), where p=0, q=3; d) example of a Mix, where p=1, q=2.

The difference of the i-th colour code in sequence A and the difference of the j-th colour code in sequence B can also occur due to read errors if none of the mentioned four cases are true. The majority of the reads contains a read error. Handling read error in colour space might prevent false translation to nucleotide sequence as it occurs at nucleotide space assemblies. (Figure 3)

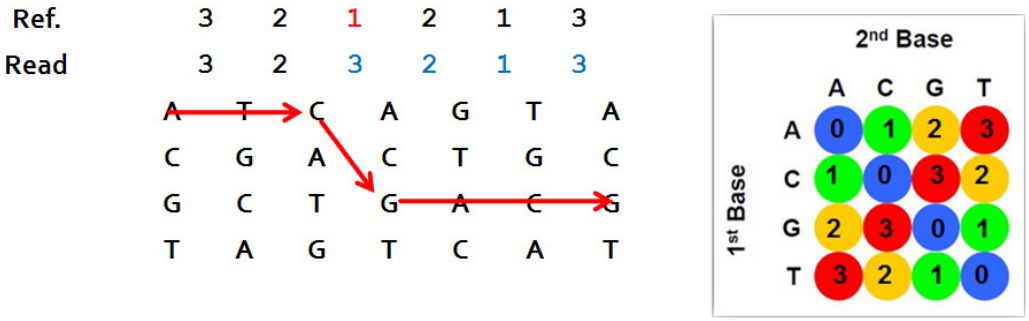

FIGURE 3:*Read error.* A3 refers to a T, T2 refers to a C, but thereafter there is a read error. C1 refers to an A, but as a consequence of the read error C3 refers to a G, and downstream part of the read will be totally incorrect.

In the algorithm developed by our team the preliminary task before aligning reads is the transformation of the reference sequence into colour codes. Main advantage of this approach is the possibility of mapping reads containing read error as well, whereas alignments performed on a nucleotide space base must skip these false reads. By increasing the proportion of mapped reads, deeper coverage can be achieved. Further, if the coverage increases, SNPs, MNPs, indels and SVs might be identified with higher confidence and the algorithm can be applied in analyses where maximum accuracy is a prerequisite.

2.1.2 Finding the Optimal Alignment Routes

To find the optimal alignment we are creating a scoring matrix, which is filled in with the score of the Match in case of a sequence match, or with penalty scores in case of mutations. To this end we have to define the following penalty functions:

$S, G_{open}, G_{extend} \in \mathbb{R}^-$

$Gap(x) = G_{open} + (x-1)G_{extend}, (x > 0, x \in \mathbb{N})$

$Snp(x) = Sx, (x > 0, x \in \mathbb{N})$

Gap and Snp functions define penalty scores belonging to gaps and SNPs with the length of x. Note that we use an affine gap penalty for Gaps.

The scoring matrix of variable-length partial sequences including the match values and penalty scores defined by the penalty functions has the form:

$$A = \begin{pmatrix} \alpha_{0,0} & \cdots & \alpha_{0,m} \\ \vdots & \ddots & \vdots \\ \alpha_{n,0} & \cdots & \alpha_{n,m} \end{pmatrix},$$

where the elements are defined as follows:

$M \in \mathbb{R}^+$ is a constant representing the value of Match,

$i \in \{0,1,2,\dots,n\}$, where $n+1 = |a|, a = a_0 \dots a_n$

$j \in \{0,1,2,\dots,m\}$, where $m+1 = |b|, b = b_0 \dots b_m$

$$\alpha_{i,j} = \begin{cases} M, i = 0 \wedge j = 0 & (1) \\ Gap(i) + M, i > 0 \wedge j = 0 & (2) \\ Gap(j) + M, j > 0 \wedge i = 0 & (3) \\ Snp(i) + M, i = j \wedge i > 0 \wedge j > 0 & (4) \\ Snp(min(i,j)) + Gap(abs(i-j)) + M, otherwise & (5) \end{cases}$$

Case (1) defines the value of a match, case (2) defines the value of an insertion or deletion, case (3) defines the value of a deletion or insertion, case (4) defines the value of an SNP, case (5)

defines the value of a Mix. Every mutation identified in colour space is followed by a match in nucleotide space, therefore the score of the match is added to each mutation.

In order to find the best local alignment we use the colour space correspondent of the Smith-Waterman algorithm, which fills in a table based on the given recursive formula, utilizing the scoring matrix:

$M \in \mathbb{R}^+$ is a constant representing the value of the Match,
$R \in \mathbb{R}^-$ is a constant representing the value of the Read Error,
$i \in \{0,1,2,\dots,n-1\}$, where $n = |a|, a = a_0 \dots a_{n-1}$,
$j \in \{0,1,2,\dots,m-1\}$, where $= |b|, b = b_0 \dots b_{m-1}$,
$a_0, a_1, \dots, a_{n-1} \in \{0,1,2,3\}$,
$b_0, b_1, \dots, b_{m-1} \in \{0,1,2,3\}$.

The matrix can be filled in column- or row-wise, as well as diagonally. If we start out from the upper left corner, we can compute the elements according to the following recursive formula:

$$t_{i,j} = max \begin{cases} 0, \\ t_{i-1,j-1} + R, a_i \neq b_j \wedge i > 0 \wedge j > 0, \\ M, a_i = b_j \wedge (i = 0 \vee j = 0), \\ \{t_{i-k-1,j-l-1} + \{a_{k,l} | a_i \oplus \dots \oplus a_{i-k} = b_j \oplus \dots \oplus b_{j-l} \wedge i - 1 \geq k \geq 0 \wedge j - 1 \geqq l \geqq 0\}\}, \end{cases}$$

where $t_{i,j}$ is the (i,j)-th element of the scoring matrix (Table 2).

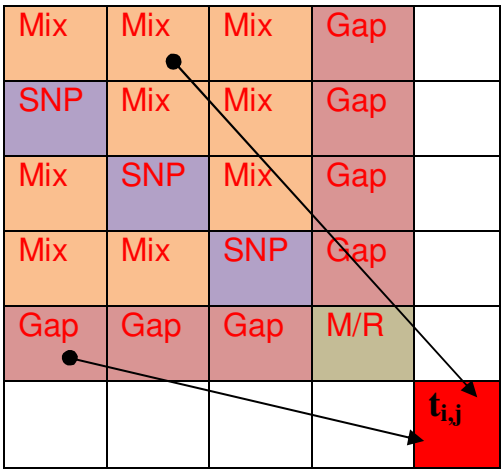

TABLE 2: *Dynamic programming table.* The value of the cell $t_{i,j}$ can be derived from the previous values of the table in such a way that the value of the cell $t_{i,j}$ must be the maximum. SNP: single nucleotide polymorphism; M/R: Match/Read error.

Similarly to the Smith-Waterman algorithm here we examine every possible alignment during filling in the table. This results in optimal local alignment by selecting one of the global maxima and backtracking along the parent cells, similarly to the Smith-Waterman algorithm.

2.1.3 Implementation and Adaptation of the Algorithm

We implemented our algorithm as a part of BFAST, a widely used assembler software. However, adaptable to any reference assembler software, the reason why we chose the BFAST assembler software tool for the practical implementation is its easily modifiable modular structure. It also utilizes an efficient indexing method that increases the number of mappable reads. BFAST consists of the following modules:

1. Preprocessing
2. Indexing
3. Finding the Candidate Alignment Locations

4. Local alignment
5. Postprocessing

Our local aligner algorithm was adapted and implemented as Module 4 and has the capability of running in parallel. While bearing paralellity, the number of threads is parameterable and the total complexity of the algorithm won't exceed $O(n^4)$. We used Java programming language for the implementation of the algorithm, since its platform independent and object- oriented characteristics. It also promotes further improvements.

2.2 Findings and Proofs: Testing the Performance and Validation on Empirical and Publicly Available Data

For testing the performance of the newly developed algorithm we resequenced a *Propionibacterium acnes* strain ATCC11828 [22] belonging to Type II cluster and aligned it to the genome of strain KPA 171202 [23] belonging to Type IB phylogenetic cluster.

By sequencing the whole ATCC11828 genome using SOLiD System we obtained more than 14 million sequence tags of 50 nucleotide length [24]. We performed the alignment using the BFAST integrated new algorithm. In order to test the accuracy of the variant call (SNPs, MNPs, short and long indels) we capillary sequenced 12, highly variant genomic regions (Table 3) and cross-checked the identified variants with those sequenced by SOLiD and assembled by our algorithm. Reference alignment identified single nucleotide and large variations as well. Here, we detail examples of the above. A putative anti-sigma factor was sequenced and reference aligned; Figure 4 shows the full sequence and flanking regions. All the SNPs and short variations were validated by capillary sequencing.

gene	note	SNP	MNP	deletion	insertion	GenBank accession No.
recA	full *recA* and flanking regions	15	1 (2nt)			JF449972
dapF	partial	14	1 (2nt)			JF449974
miaA	partial	11	2 (2nt)			JF449975
nadE	partial	30				JF449976
pfkA	partial	17				JF449985
tig	partial (3')	17				JF449981
	partial (5')	17	1 (2nt)	1 (1nt) 1 (45nt)	1 (1nt)	JF449982
uvrA	partial	9	1 (2nt)	1 (3nt)		JF449977
PPA0026	partial	21	1 (2nt)	1 (1nt)		JF449980
PPA1355 PPA1356	partial PPA1355 partial PPA1356	11		1 (1nt) 1 (6nt)		JF449978
PPA2382	full PPA2382 partial PPA1266	17	1 (3nt)		1 (1nt) 1 (2nt)	JF449979
PPA0497	partial	29	1		1 (1nt) 2 (2nt) 1 (4nt)	JF449973
	total	208	9	6	7	

TABLE 3: *Validation of the mapping accuracy using capillary sequencing.* 12 regions were chosen for the validation as these regions in the genome of strain ATCC11828 contains numerous

variations as identified by the novel algorithm. The sequence of recA was previously published [DQ059328], we here show data for the full recA sequence and 104 nucleotides flanking the gene (96 and 58 nucleotides up- and downstream, respectively) [JF449972]. Sequencing by traditional capillary electrophoresis confirmed each of the variant identified by mapping of the short sequence tags. SNP, single nucleotide polymorphism; MNP, multi nucleotide polymorphism (adjacent SNPs).

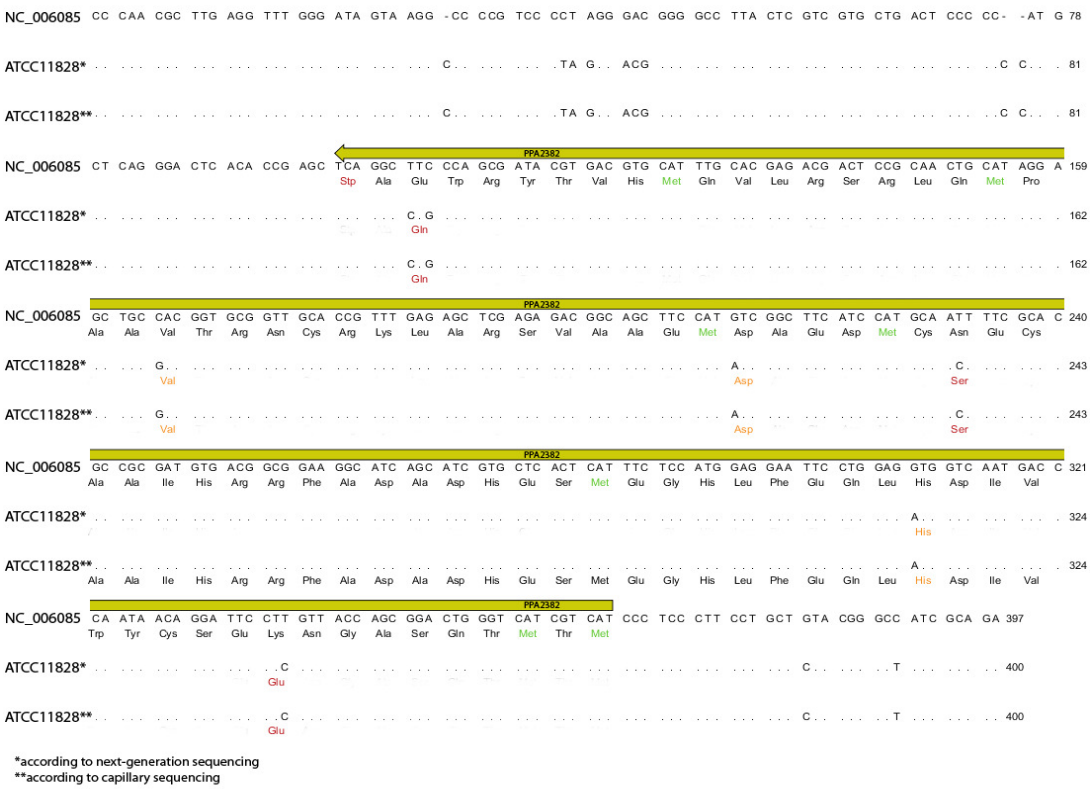

FIGURE 4: *Validation of the mapping accuracy using capillary sequencing.* Coding sequence (yellow arrow) and flanking sequence of the putative anti-sigma factor (PPA2382) is shown (sequenced with SOLiD). A region spanning 397 nucleotides (positions 1.372.929 - 1.373.326) was chosen for the validation: we have identified with our algorithm 9 SNPs of which 4 SNPs are causing altogether 3 amino acid changes (amino acids shown in red), 2 MNPs and 2 insertions. All of the variations were found valid(identical positions and lengths) by capillary sequencing.

The alignment also identified large (>1kb) deletions in the genome of ATCC11828. We have chosen 2 deletions for validation: a deletion of ~3.6kb resulting in partial loss of genes PPA0199 and PPA0200 (Figure 5. A.) and a deletion of ~19kb resulting in complete loss of 14 genes (Figure 6. A.). Capillary sequencing confirmed both of the deletions (Figures 5. B,C. and 6. B,C.). While the ~3.6kb deletion was perfectly matched, capillary sequencing identified 70 additional nucleotides within the ~19kb deletion (Figure 6. C.).

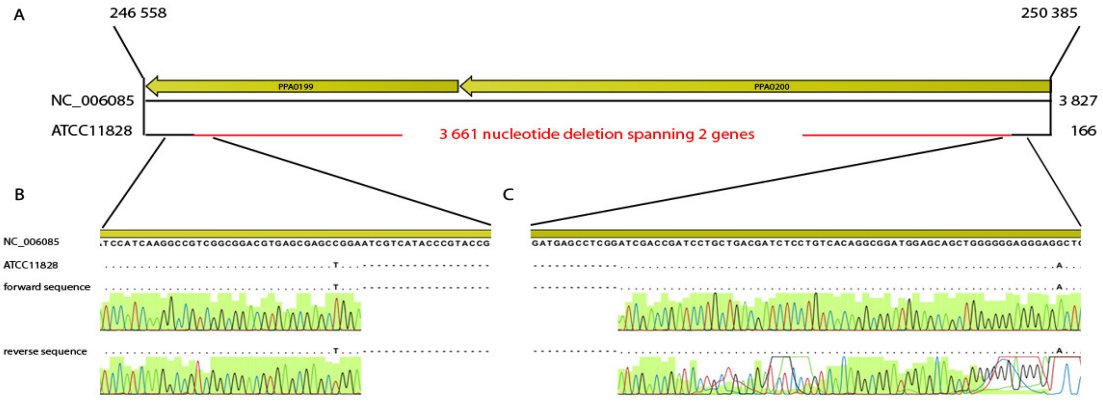

FIGURE 5: *Validation of a ~3.6kb deletion.* A region spanning 3.827 nucleotides of the reference genome (positions 246.558 - 250.385) was chosen for the validation as the genome of strain ATCC11828 contains a deletion of 3.661 nucleotides (A) when aligned to the reference genome; in addition 2 SNPs are shown. We have validated the deletion and SNPs by capillary sequencing (B and C).

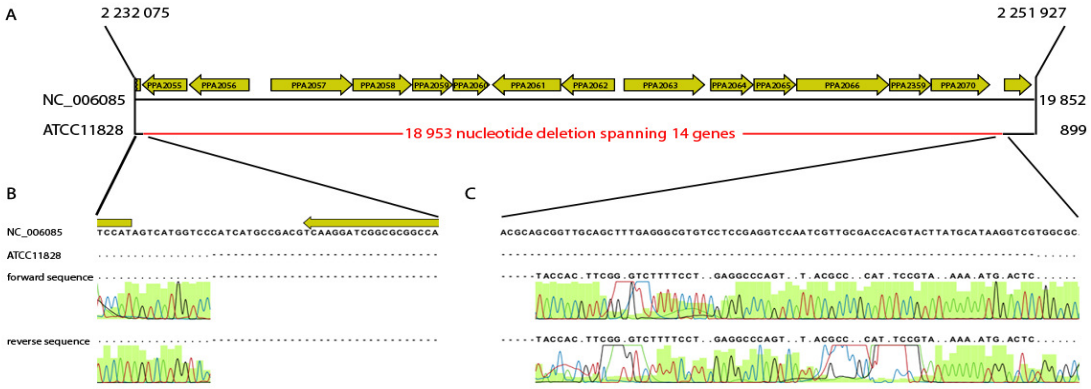

FIGURE 6: *A deletion of ~19Kb in the genome of ATCC11828 causes the loss of 14 genes.* A region spanning 19.852 nucleotides of the reference genome (positions 2.232.075 - 2.251.927) was chosen for the validation as the genome of strain ATCC11828 contains a deletion of 18.953 nucleotides (A) when aligned to the reference genome. We have validated the deletion by capillary sequencing (B and C). Note that we identified a region of 70 nucleotides by capillary sequencing that was not recognized by the alignment of short sequence tags (C).

Summarizing the results:

- Reference strain: *Propionibacterium acnes* KPA171202
- Resequenced strain: *Propionibacterium acnes* ATCC11828
- Reference genome size: 2,560,265 bp
- Number of sequenced reads: 14 473 881
- Number of aligned reads: 7 487 891

As data show the algorithm mapped to the reference roughly the half of the reads. Reasonable explanation to this phenomenon is that we set very strict alignment conditions so as to achieve maximum accuracy. Obviously it is possible to map more reads by lowering the alignment standards, but it will likely result in more false alignments and wrongly predicted read errors and mutations.

Table 4 shows validation results after the examination of twelve gene regions. Numbers indicate the ratio of coincidence between our alignment and the results validated by traditional capillary sequencing. An SNP/insertion/deletion is considered valid if SOLiD alignment and capillary sequencing results entirely correspond.

SNP	False SNP	Insertion	False insertion	Deletion	False deletion
98,94%	0,54%	100%	0%	96,36%	0%

TABLE 4: *SNPs/Insertions/Deletions.* The percentage of the validated SNPs/insertions/deletions that our algorithm found in relation to those detected by capillary sequencing; False SNP/False insertion/False deletion: the SNPs/insertions/deletions found by our algorithm but not proven by capillary sequencing.

We also evaluated the performance of our algorithm using the dataset available in the NCBI SRA database (ID number: SRR007448). The data is generated using SOLiD System 2.0, fragment library with 36bp reads. We have used the first 100,000 reads of the sequence data for the evaluation.

Algorithms	indexing time	read processing time	reads Per Hour	memory	aligned%	failed%
Bowtie	8m31s	0m3.5s	102857143	243MB	9,68	90,32
Shrimp	8m58s	2m33s	2352941	5.3GB	26,32	73,68
BFAST	167m15s	3m10s	1894737	14GB	18,2	81,8
Our algorithm	14m10s	4m2s	1487603	13GB	31,6	68,4

TABLE 5: *Performance results of alignment algorithms*. This table shows the timing and memory details of the algorithms as well as the mapping percentages.

From Table 5 we can see that the highest performing algorithm is Bowtie, both in terms of read processing time and memory footprint. Bowtie however does not perform gapped local alignment. Recently there is a new version called Bowtie2 which does both global and local alignment and gapped alignment. [25] We can also see that the indexing time is very high in case of BFAST, it seems to be optimized for alignment speed at the cost of one-time indexing. This is appropriate if there are a lot of samples to align to a single reference.

In general we can conclude that our algorithm provides performance comparable with the currently available solutions. However the detailed parameter settings make it possible to achieve highly sensitive alignments on the data generated by the SOLiD platform.

2.3 Materials and Methods

Hardware configuration for the test assembly: in order to achieve fast mapping we used a high performance computer with the following configuration: Processor: 2x2x4 Intel CPU cores with 8M Cache, 5.86 GT/s QPI speed; Memory: 96 GB DDR3 1333 MHz; Disk capacity: 10 TB RE Hard Drive.

Bacterial data sets used in the training and validation of the algorithm: the complete genome sequence of *P. acnes* was previously determined [23] and is publicly available [NC_006085.1

GI:50841496]; we have used it as a reference genome. Genome sequencing of *P. acnes*strain ATCC11828 was performed on the SOLiD 3 System (Applied Biosystems, now part of Life Technologies) following the manufacturer's instructions. For this, DNA was extracted using the AquaGenomic kit (MultiTarget Pharmaceutical) and the preparation of the libraries and sequencings were performed using cycled ligation sequencing on a SOLiD 3 System [24].

Confirmation of structural variations: SNPs, MNPs, insertions and deletions identified by our algorithm were validated by conventional capillary DNA sequencing (3500 Series Genetic Analyzer, Life Technologies) using oligonucleotide primers (Integrated DNA Technologies) listed in Table 5. which were designed with Clone Manager 9.0 software (Sci-Ed Software).

gene	forward primer sequence (5'-3')	reverse primer sequence (5'-3')	forward primer binding site*	reverse primer binding site*
recA	AGCTCGGTGGGGTTCTCTCATC	GCTTCCTCATACCACTGGTCATC	1095184-1095205	1096362-1096384
dapF	CCGGCAATGACTTCGTCATC	GGTTGACAAGAGCGTGTTCG	1104255-1104274	1105054-1105073
miaA	AGTCAATCTTGCCCGGCAACG	GTCCTCAACATCTCCGGCTC	1103744-1103764	1104369-1104388
nadE	GGCCGTCGTGGAGGTAAATC	TGGTTGGCTATCCGGAAGAG	2451528-2451547	2452342-2452361
pfkA	AGCGGGTTATCTGACCGAGG	TCCGTGGCAATCTCCAGTGC	102707-102726	103344-103362
tig	GGTCGAGGCGAGCCATATTC	CCAGCCTTGACAAGGCCTAC	1703041-1703070	1703777-1703796
uvrA	ATCAGCCGGGTCGGTTCTCC	CTACCTCCGGTCTGGGTAAG	888500-888519	889335-889354
PPA0026	TGCGATCAGTTGCTGGTTGG	AAGAACAGTGTGGGATCGAG	26672-26691	27429-27448
PPA1355	TCGACTTCGGCTTGGCCATC	TAAGGCGGGCCGACTTATGG	1478573-1478592	1479298-1479317
PPA2382	GTCGTGCGCGTCGTAAGAAG	GGCTTGGCTGTATCGCATTC	1372888-1372908	1373725-1373744
PPA0497	GGTGAAACGCCGCTGACAAG	TCATCTCCACCGCGAACCTG	547385-547404	548073-547093
region9	CTGGTAGTGCGTCTCTACCG	TAAGCGACGCCAACAGGTTC	246558-246577**	250369-250388
region30	CAGGTCCAAGCGTGACATTC	CTCCTGGTGACGGGTTATTC	2232076-2232095	2251234-2251253

TABLE 5:*Oligonucleotide primers used in this study and their bindingsites.**the positions are given taking into account the reference genome [NC_006085], **contains 3 mismatches with respect to the reference sequence

3. CONCLUSIONS

We have developed a novel algorithm, derived from Smith-Waterman method, for pairwise sequence alignment in colour space. During the assembly the algorithm is proven to select the optimal alignment for any given read utilizing the scoring conditions which are adjusted by the user. The performance of the algorithm was tested on empirical dataset obtained by sequencing the genome of *Propionibacterium acnes*.

Validation by capillary sequencing confirmed that the algorithm called true sequence variants, including SNPs, MNPs, insertions and deletions when compared to the available reference genome. Our algorithm showed high accuracy even in the presence of sequencing errors, thus it greatly facilitates the downstream analysis of colour coded NGS data. We have also shown that in spite of the high accuracy achievable by our algorithm the performance does not suffer much compared to other applications.

Taking together, the outstanding test results and the biologically highly significant findings show the true importance of an accurate aligner algorithm that is able to select a compact and correct set of variants even at low diversities and in the presence of sequence errors.

4. ACKNOWLEDGEMENTS AND FUNDING

We thank Dr. Zoltán Hegedűs (BRC, Szeged, Hungary) and Dr. András Hajdu (Faculty of Informatics, Debrecen, Hungary) for critical reading of the manuscript and to Judit Hunyadkürti, Zsuzsanna Maros-Szabó, Bálint Domokos, Attila Horváth, László Steiner, Miklós Laczik, Zoltán Pistár, Péter Szilágyi and János Szunai for their valuable contribution in sequencing and implementation of the algorithm. The work in the lab of IN was supported by the Hungarian National Office for Research and Technology Teller program (OMFB-00441/2007) and by the French-Hungarian Associated European Laboratory (LEA) SkinChroma (OMFB-00272/2009). The IT work was supported by the János Bolyai grant of the Hungarian Academy of Sciences, the Gabor Baross Program (OMFB-00609/2009) and Hungarian Economic Development Operational Programmes (GOP-1.1.1-08/1-2008-0047).

5. REFERENCES

[1] Z. Su, B. Ning, H. Fang, H. Hong, R. Perkins , W. Tong, L. Shi. “Next-generation sequencing and its applications in molecular diagnostics”. *Expert Review of Molecular Diagnostics,* vol. 11, pp. 333-343, Apr. 2011.

[2] R. M. Durbin, D. L. Altshuler, R. M. Durbin, G. A. R. Abecasis, D. R. Bentley, A. Chakravarti, A. G. Clark, F. S. Collins et al. "A map of human genome variation from population-scale sequencing". *Nature*, vol. 467, pp.1061–1073, Oct. 2010.

[3] R. A. Gibbs, J. W. Belmont, P. Hardenbol, T. D. Willis et al. “The International HapMap Project”. *Nature*, vol. 426, pp. 789–796, Dec. 2003.

[4] D. R. Bentley. “Whole-genome re-sequencing”. *Current Opinion in Genetics & Development*, vol. 16, pp. 545-552, Oct. 2006.

[5] M. Margulies, M. Egholm, W. E. Altman, S. Attiya, J. S. Bader, L. A. Bemben et al. “Genome sequencing in microfabricated high-density picolitre reactors”. *Nature*, vol. 437, pp. 376-380, Sep. 2005.

[6] D. R. Smith, A. R. Quinlan, H. E. Peckham, K. Makowsky, W. Tao, B. Woolf et al. “Rapid whole-genome mutational profiling using next-generation sequencing technologies”. *Genome Research*, vol. 18, pp. 1638–1642, Oct. 2009.

[7] J. Shendure, H. Ji. “Next-generation DNA sequencing”. *Nature Biotechnology*, vol. 26, pp. 1135-1145, Oct. 2008.

[8] M. L. Metzker. “*Sequencing technologies – the next generation*”. *Nature Reviews Genetics*, *vol.* 11, pp. 31-46, Jan. 2010.

[9] Applied Biosystems Incorporated. “Principles of Di-Base Sequencing and the Advantages of Color Space Analysis in the SOLiD System”. 2008.

[10] H. Breu."A Theoretical Understanding of 2 Base Color Codes and Its Application to Annotation, Error Detection, and Error Correction", 2010.

[11] A. Magi, M. Benelli, A. Gozzini, F. Girolami, F. Torricelli, M. L. Brandi."Bioinformatics for Next Generation Sequencing Data". *Genes*, vol. 1, pp. 294-307, Sep. 2010.

[12] P. Flicek, E. Birney. “Sense from sequence reads: methods for alignment and assembly”. *Nature Methods*, vol. 6, pp. S6–S12, Nov. 2009.

[13] S. M. Rumble, P. Lacroute, A. V. Dalca, M. Fiume, A. Sidow, M. Brudno. (2009, May). "SHRiMP: accurate mapping of short color-space reads". *PLoS Computational Biology,*5(5), Available:http://www.ploscompbiol.org/article/info:doi/10.1371/journal.pcbi.1000386

[14] B. Langmead, C. Trapnell, M. Pop, S.L. Salzberg. (2009, March)."Ultrafast and memory-efficient alignment of short DNA sequences to the human genome". *Genome Biology*, vol.10, 10:R25, Available:http://genomebiology.com/2009/10/3/R25

[15] N. Homer, B. Merriman, S.F. Nelson."BFAST: an alignment tool for large scale genome resequencing". *PLoS One*,4(11) e7767., 2009.

[16] N. Homer, B. Merriman, S.F. Nelson. (2009, June). "Local alignment of two-base encoded DNA sequence". *BMC Bioinformatics*,10:175, Available:http://www.biomedcentral.com/1471-2105/10/175

[17] H. Li, R. Durbin."Fast and accurate short read alignment with Burrows-Wheeler transform". *Bioinformatics,* vol. 25, pp 1754-1760, 2009.

[18] T. F. Smith, M. S. Waterman. "Identification of common molecular subsequences". *Journal of Molecular Biology,* vol.147, pp 195-197, 1981.

[19] K. R. Rasmussen, J. Stoye, E. W. Myers. “Efficient q-gram filters for finding all epsilon-matches over a given length”. *Journal of Computational Biology*, vol. 13, pp. 296–308, Mar. 2006.

[20] R. A. Lippert. “Space-efficient whole genome comparisons with Burrows-Wheeler transforms”. *Journal of Computational Biology*, vol. 12, pp. 407-415, May 2005.

[21] S. Bao, R. Jiang, W. Kwan, B. Wang, X. Ma, Y. Q. Song. “Evaluation of next-generation sequencing software in mapping and assembly”. *Journal of Human Genetics*, vol. 56, pp. 406-414, Jun. 2011.

[22] I. Nagy, A. Pivarcsi, K. Kis, A. Koreck, L. Bodai, A. McDowell, H. Seltmann, S. Patrick, C.C. Zouboulis, L. Kemeny. "Propionibacterium acnes and lipopolysaccharide induce the expression of antimicrobial peptides and proinflammatory cytokines/chemokines in human sebocytes". *Microbes Infect.,* vol.8 ,pp 2195-2205, 2006.

[23] H. Bruggeman, A. Henne, F. Hoster, H. Liesegang, A. Wiezer, A. Strittmatter, S. Hujer, P. Durre, G. Gottschalk. "The complete genome sequence of Propionibacterium acnes, a commensal of human skin". *Science,* vol. 305, pp. 671-673, 2004.

[24] B. Horvath, J. Hunyadkurti, A. Voros, Cs. Fekete, E. Urban, L. Kemeny, I. Nagy. “Genome sequence of Propionibacteriumacnes type II strain ATCC 11828”. *Journal of Bacteriology,* vol. 194, pp 202-203, 2012.

[25] B. Langmead, S. L. Salzberg. “Fast gapped-read alignment with Bowtie 2”. *Nature Methods*, vol. 9, pp. 357–359, Mar. 2012.