Dynamic Gap Selector: A Smith Waterman Sequence Alignment Algorithm with Affine Gap Model Optimisation

Dynamic Gap Selector: A Smith Waterman Sequence Alignment Algorithm with Affine Gap Model Optimisation

Gianvito Urgese[1], Giulia Paciello[1], Andrea Acquaviva[1], Elisa Ficarra[1], Mariagrazia Graziano[2], and Maurizio Zamboni[2]

[1]Department of Control and Computer Engineering, Politecnico di Torino,
Corso Duca degli Abruzzi, 24, 10129 Turin, Italy
[2]Department of Electronics and Telecommunications, Politecnico di Torino,
Corso Duca degli Abruzzi 24, 10129 Turin, Italy
{gianvito.urgese,giulia.paciello,andrea.acquaviva,elisa.ficarra,
mariagrazia.graziano,maurizio.zamboni}@polito.it
http://www.polito.it

Abstract. Smith Waterman algorithm (S-W) is a widespread method to perform local alignments of biological sequences of proteins, DNA and RNA molecules. Indeed, S-W is able to ensure better accuracy levels with respect to the heuristic alignment algorithms by extensively exploring all the possible alignment configurations between the sequences under examination. It has been proven that the first amino acid (AA) or nucleotide (NT) inserted/deleted (that identify a *gap open*) found during the alignment operations performed on sequences is more significant, from a biological point of view, than the subsequent ones (called *gap extension*), making the so called *Affine Gap* model a viable solution for biomolecules alignment. However, this version of S-W algorithm is expensive both in terms of computation as well as in terms of memory requirements with respect to others less demanding solutions such as the ones using a *Linear Gap* model.

In order to overcome these drawbacks we have developed an optimised version of the S-W algorithm based on *Affine Gap* model called *Dynamic Gap Selector (DGS_S-W)*. Differently from the standard *S-W Affine Gap* method, the proposed *DGS_S-W* method reduces the memory requirements from 3*N*M to N*M, where N and M represent the size of the compared sequences. In terms of computational costs, the proposed algorithm reduces by a factor of 2 the number of operations required by the standard *Affine Gap* model.

DGS_S-W method has been tested on two protein and one RNA sequences datasets, showing mapping scores very similar to those reached thanks to the classical *S-W Affine Gap* method and, at the same time, reduced computational costs and memory usage.

Keywords: Smith-Waterman, Affine Gap model, Sequence Alignment Algorithm, Local Alignment Algorithm, Dynamic Programming, Exhaustive Sequence Alignment, Dynamic Gap Selector, DGS_S-W.

1 Introduction

Smith Waterman (S-W) algorithm [1] is a commonly alternative to perform local alignment analysis between two biological sequences. The wide usage is due to its capability to ensure higher accuracy levels with respect to the heuristic algorithms such as BLAST [2], thanks to the exploration of all the possible alignment configurations between the considered sequences. Furthermore the accuracy of this algorithm is ensured by a precise evaluation of the different events occurring when aligning biological sequences. These events can be summarized in: i) Match, that occur if two amino acids (AA) or nucleotides (NT) are equal in the compared sequences; ii) Mismatch, identified if two AA (or NT) are different in the two compared sequences; iii) Insertion, if one or more AA (or NT) can be observed in the compared sequence with respect to the reference one; iv) Deletion, if one or more AA (or NT) are missing in the compared sequence with respect to the reference.

Differently from global alignment methods, the local alignment ones are particularly suitable when making comparisons among very divergent sequences suspected to contain only small regions of similarity within their larger sequence context.

Being an exhaustive dynamic programming algorithm, S-W allows to find the optimal local alignment between the two compared sequences, respectively named *Query (Qry)* of length N and *Subject (Sbj)* of length M, by searching for a pair of segments, one from each of these two sequences, such that no other couple of segments with greater similarity could be detected. In particular *Qry* represents the selected sequence that has to be found into the database whereas the *Sbj* a certain sequence from the database that will be compared using S-W with the *Qry* sequence. The S-W algorithm takes advantage of a *Score Matrix F* of dimension (N+1, M+1) in which a score value, capable to account for matches, mismatches and gaps in the alignment, has been calculated for each cell (more details can be found in the Methods Section). The detected alignment with highest score is reconstructed in the last phase of the algorithm starting from the *Maximum Alignment Score* that has been identified in the score matrix thanks to a traceback procedure.

A variant of the S-W algorithm more suitable to sequences similarity analysis and called *S-W Affine Gap* method, has been proposed by Gotoh [3]. It assigns a higher penalty cost to the first insertion or deletion encountered (called gap open) rather than to the size of the same gap (defined as gap extension) [3]. On the other side, this version introduces remarkable computational costs and memory requirements with respect to the *S-W Linear Gap.*

In particular, for what is concerning biological sequences alignment score calculations, it has been proven [4] that the model proposed by Gotoh [3] is capable to better evaluate biological events.

In this paper we propose an optimised version of the *S-W Affine Gap* method called *Dynamic Gap Selector (DGS_S-W)*, designed to reduce the memory requirements from $3*N*M$ to $N*M$ and the operations performed by a factor of 2 with respect to the standard *S-W Affine Gap* method.

As depicted in *Fig. 1*, the purpose of the proposed algorithm is to approach the *Affine Gap* method in terms of performance, with a computational cost comparable with the *Linear Gap* solution.

The results obtained by applying *DGS_S-W* to protein and NT sequences show that the proposed methods succeeds in this objective. Indeed, it is able to provide biological sequence similarity evaluations very close to the ones produced by the *S-W Affine Gap* method with reduced computational costs and memory requirements, comparable to the *S-W Linear Gap* method.

As a consequence, *DGS_S-W* represents a viable choice when working with big amount of data such as those deriving from Next Generation Sequencing (NGS) technologies [5]. When dealing with raw NGS data, the alignment of produced sequences (called *reads*) over a known reference is a common starting point of several analysis pipelines [6], that can profit from *DGS_S-W* to improve their speed/accuracy trade-off.

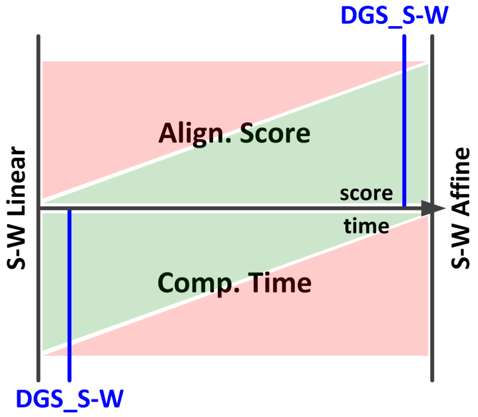


Fig. 1. Trade off between *DGS_S-W* vs *S-W Affine Gap* and *DGS_S-W* vs *S-W Linear Gap* methods, evaluated in terms of alignment score and computational time.

2 Materials and Methods

2.1 Scoring Models

In order to apply *S-W Linear Gap* [1], *S-W Affine Gap* [3] and *DGS_S-W* methods to perform the alignment of sequences, it is necessary to introduce the concept of *Scoring Model*. With refer to the alignment procedure a *Scoring Model* can be defined as a set of rules used to assess the possible situations that generally occur during alignment operations. The simplest example of such a model, usually used to score the similarity among NTs belonging to different sequences, is the *Match/Mismatch* model that assigns a score of +1 and -1 respectively if a match or a mismatch occurs, whereas a value equal to -d in case of gaps (insertions or deletions). In the evaluation of protein similarity however another scoring model is generally applied. It is based on the usage of a *Substitutional Matrix*, proven to better describe, from a biological point of view, events such as AA matches or mismatches. Moreover also this model makes use of gap penalties to consider insertions or deletions. *Substitutional Matrices* are built on the basis of the probability that a particular AA is replaced with another during the evolution process: They assign to each pair of AAs a value that indicates their degree of similarities, obtained thanks to statistical methods reflecting the frequency of a particular AAs substitution in homologous protein families [4]. So, a positive value in the *Substitutional Matrix*, means that the two AAs are similar or identical and that they are frequently exchanged each other without notable loss of function. *Blosum Matrix*, introduced in 1992 by Henikoff and Henikoff [7] is one of the most used *Substitutional Matrix* implemented by considering multiple alignments of evolutionarily divergent proteins. As said, instead, a common way to evaluate insertions or deletions is to use a *Gap Penalty* model. The most used schemes are: i) *Constant Gap* model, that assigns a penalty equal to a d value to each gap found during the alignment (*Eq.1*) and so capable to evaluate only the presence of a gap event but not its extension; ii) *Linear Gap* model that considers instead the gap length (g) to score the alignment (*Eq.2*), giving the possibility to evaluate with different scores gaps of different sizes; iii) *Affine Gap* model that, attributing different costs to the gap open (d parameter) and the gap extension (e parameter) events (*Eq.3*), is able to assign a higher penalty to the gap presence with respect to its relative extension size.

$$\gamma_{konst} = -d \tag{1}$$

$$\gamma_{lin}(g) = -g * d \tag{2}$$

$$\gamma_{aff}(g) = -d - (g-1) * e \tag{3}$$

In the following Subsections the *S-W Linear Gap* [1], *S-W Affine Gap* [3] and *DGS_S-W* methods will be presented. Even if different implementations of the first two algorithms are freely available, both have been reimplemented in C++ programming language using SeqAn Library [8] to make more meaningful comparisons with the novel *DGS_S-W* method that we developed, once again, in C++ taking advantage of SeqAn library [8].

2.2 Smith-Waterman Linear Gap Method

The *S-W Linear Gap* method [1] is a variation of the global alignment method called Needleman Wunsch (N-W) [10] that aims at identifying regions of similarity between a *Qry* and a *Sbj* sequence, through the calculation of a similarity score. S-W algorithm [1], as N-W [10], is a dynamic programming procedure characterized by three main steps: i) The initialization of the *Score Matrix F* with zeros in positions *F(i,0)* and *F(0,j)* that account for the beginning of a new alignment; ii) The calculation of the alignment score cell by cell thanks to *Eq.4* in which S represents the *Substitutional Matrix* if dealing with AAs, whereas the *Match/Mismatch* model if NTs are compared and d the linear gap penalty assigned to insertion or deletion events; iii) The traceback procedure beginning with the identification into the *Score Matrix F* of the Maximum Alignment

Score, followed by the reconstruction of the alignment path (coming back from child to father cell following the descendent pointers) and ending when a 0 is reached.

$$F(i,j) = max \begin{cases} 0 \\ F(i-1,j-1) + S(Qry(i), Sbj(j)) \\ F(i-1,j) - d \\ F(i,j-1) - d \end{cases} \tag{4}$$

2.3 Smith-Waterman Affine Gap Method

The *S-W Affine Gap* method, introduced by Gotoh in 1982 [3], with respect to the *S-W Linear Gap* one [1] is considered the model that better describe biological sequences similarity, assigning a higher penalty cost for the first gap encountered, called *gap open*, than the cost of the following gaps called *gap extension*. Even if the steps characterizing this method are identical to those proper of the simplest *S-W Linear Gap* method, the introduction of a more sophisticated model for gap evaluation account for highly divergent and more biologically significant alignment scores. Three different matrices are required to perform S-W alignments thanks to the *Affine Gap* model: The *Score Matrix F* of size (M+1,N+1) and two matrices of dimension (M,N), respectively called *Qry Matrix* X_{Qry} and *Sbj Matrix* Y_{Sbj}. By assuming that a deletion can't be followed by an insertion and viceversa, the alignment scores are calculated as described by *Eq.5*,

$$\begin{aligned} X_{Qry}(i,j) &= max \begin{cases} 0 \\ F(i-1,j) - d \\ X_{Qry}(i-1,j) - e \end{cases} \\ Y_{Sbj}(i,j) &= max \begin{cases} 0 \\ F(i,j-1) - d \\ Y_{Sbj}(i,j-1) - e \end{cases} \\ F(i,j) &= max \begin{cases} 0 \\ F(i-1,j-1) + S(Qry(i), Sbj(j)) \\ X_{Qry}(i-1,j-1) + S(Qry(i), Sbj(j)) \\ Y_{Sbj}(i-1,j-1) + S(Qry(i), Sbj(j)) \end{cases} \end{aligned} \tag{5}$$

where *F(i,j)* represents the matrix containing the alignment scores, $X_{Qry}(i,j)$ the insertion matrix storing all the gaps found in the *Qry* and $Y_{Sbj}(i,j)$ the deletion matrix that keeps trace of all the *Sbj* gaps. Moreover *S-W Affine Gap* method can be customized on the basis of specific requirements such as the kind of events that have to be considered in the alignment phase. For example another version of this algorithm, is built on the assumption that a deletion can be followed by an insertion and viceversa, allowing the model simplification. In particular the two matrices X_{Qry} and Y_{Sbj} are replaced, in this case, by a unique matrix K representing the possibility of being in a gapped region [9].

2.4 Dynamic Gap Selector S-W (DGS_S-W)

The *DGS_S-W* represents an optimisation of the S-W algorithm based on the *S-W Affine Gap* method [3], developed in order to reduce the computation complexity, the memory requirements and consequently the computational costs proper of the central phase of the S-W algorithm that consists in the Alignment Score calculations. In particular, aims of the proposed optimisation, can be essentially summarized in the calculation of local alignment scores between sequences with computational costs (both from a time and memory point of view) comparable with those proper of the *S-W Linear Gap* method [1] and accuracy levels in the sequences similarity evaluation typical of the *S-W Affine Gap* method [3] as depicted in *Fig. 1.* Main idea at the basis of *DGS_S-W* is to replace the usage of the two gap matrices proper of the *S-W Affine Gap* method [3], with a dynamic choice between the two kind of gap allowed, that are gap open and gap extension. In particular, the reduced memory requirements associated with this new method, are due to the substitution of the two integer gap matrices with two boolean variables capable to keep track of a gap open or a gap extension encountered. This feature allows to reduce memory requirements as said from $3{*}N{*}M$ to $N{*}M$ and, at the same time, to reduce by a factor of 2 the number of operations performed during the calculation. The reduction in the number of operations performed with respect to the *S-W Affine Gap* method can be directly observed by comparing *Eq.5* with *Eq.6*.
Furthermore two hardware implementations, developed in 2012 by Urgese et al. [11] and in 2014 by Causapruno et al. [12], confirmed both the reduction in memory usage and computational time required by the novel method.

Concordantly to these considerations *Eq.6* report the operations performed to retrieve the Maximum Alignment Score of interest,

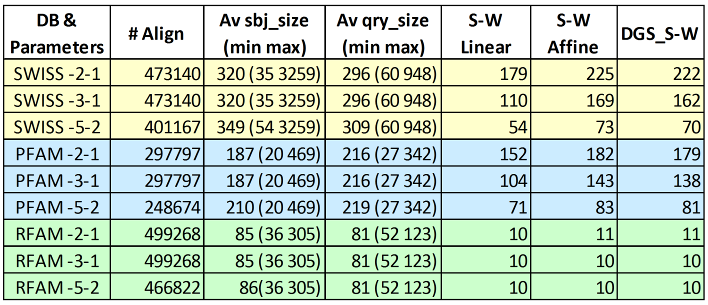
$$
\begin{aligned}
g_{Qry} &= \begin{cases} d & if\ \{sel_{Qry} = 0\} \\ e & if\ \{sel_{Qry} = 1\} \end{cases} \\
g_{Sbj} &= \begin{cases} d & if\ \{sel_{Sbj} = 0\} \\ e & if\ \{sel_{Sbj} = 1\} \end{cases} \\
F(i,j) &= max \begin{cases} 0 & \Rightarrow sel_{Qry} = 0, sel_{Sbj} = 0 \\ F(i-1,j-1) + S(Qry(i), Sbj(j)) & \Rightarrow sel_{Qry} = 0, sel_{Sbj} = 0 \\ F(i-1,j) - g_{Qry} & \Rightarrow sel_{Qry} = 1, sel_{Sbj} = 0 \\ F(i,j-1) - g_{Sbj} & \Rightarrow sel_{Qry} = 0, sel_{Sbj} = 1 \end{cases}
\end{aligned}
\tag{6}
$$

where g_{Qry} and g_{Sbj} are the two selected gap values for the *Qry* and the *Sbj* sequences, sel_{Qry} and sel_{Sbj} are the two boolean variables used to discriminate between the gap open or gap extension penalties usage for the *Qry* or the *Sbj* sequences, *F(i,j)* is the score value of the considered cell whereas the arrow symbols account for the boolean variable assignments.

3 Results

Data used to evaluate *DGS_S-W* method performances in aligning biological sequences were collected from three different publically available databases that are Pfam version 27.0 [13] and SWISS-PROT 2013_12 [14] for testing the algorithm with AA sequences and Rfam 11.0 [15] for NT sequences. In particular, in order to obtain a set of *Sbj* sequences of reasonable size (our search database), 5000 sequences have been casually extracted from each of the aforementioned database. Among them later, once again in a random way, 100 sequences have been isolated to generate instead the *Qry* dataset.
The results obtained by aligning the aforementioned subset of Pfam [13], SWISS-PROT [14] and Rfam [15] data using the *S-W Linear Gap* [1], *S-W Affine Gap* [3] and *DGS_S-W* methods will be presented in the following. In particular matches and mismatches were evaluated using *Blosum62 Matrix* [7] when dealing with protein sequences (i.e. SWISS-PROT [14] and Pfam [13] databases), whereas the simplest *Match/Mismatch* model, that assigns +1 in case of matches and -1 for mismatches, when aligning NT sequences (i.e. Rfam [15] database). Different configurations have been therefore tested for what is concerning the values assigned to the gap penalties, according to those widely selected by the most used local alignment tools.
A gap penalty equal to -3, -2 or -5 was imposed when applying *S-W Linear Gap* method. Furthermore gap open and gap extension penalties respectively of -3 and -1, -2 and -1, -5 and -2 were selected when testing *S-W Affine Gap* [3] and *DGS_S-W* methods.

In *Table 1* the data relative to the considered datasets and the results obtained thanks to the methods application, are summarized. A threshold score value of 80 for AA sequences and 5 for NT sequences was imposed in order to consider an alignment: If at least a value computed by an algorithm among the three evaluated overcome the chosen threshold, data are recorded for all the methods.



DB & Parameters	# Align	Av sbj_size (min max)	Av qry_size (min max)	S-W Linear	S-W Affine	DGS_S-W
SWISS -2-1	473140	320 (35 3259)	296 (60 948)	179	225	222
SWISS -3-1	473140	320 (35 3259)	296 (60 948)	110	169	162
SWISS -5-2	401167	349 (54 3259)	309 (60 948)	54	73	70
PFAM -2-1	297797	187 (20 469)	216 (27 342)	152	182	179
PFAM -3-1	297797	187 (20 469)	216 (27 342)	104	143	138
PFAM -5-2	248674	210 (20 469)	219 (27 342)	71	83	81
RFAM -2-1	499268	85 (36 305)	81 (52 123)	10	11	11
RFAM -3-1	499268	85 (36 305)	81 (52 123)	10	10	10
RFAM -5-2	466822	86(36 305)	81 (52 123)	10	10	10

Table 1. Average sequences sizes for the selected DBs subsets (Columns 3 and 4) that have been aligned (number of alignments reported in Column 2) and average alignment score values (Columns 5, 6 and 7) calculated applying the proposed three methods with the parameters reported in Column 1.

Column 2 reports on the number of significant detected alignments whereas the mean, minimum and maximum aligned *Qry* and *Sbj* sizes are shown respectively in Columns 3 and 4 for the different considered databases and configurations. Columns 5, 6 and 7 of *Table 1* shown instead the average alignment scores calculated by means of the three different considered methods.

In *Fig. 2* are depicted the differences, in terms of score values, calculated between *DGS_S-W* and *S-W Affine Gap* [3] methods for the AA sequences extracted by Pfam [13] and Swiss-Prot [14] databases and for the NT sequences coming from Rfam [15]. The comparisons have been made taking as reference the *S-W Affine Gap* method [3] since, as before highlighted, it can be considered the model that better describe sequences similarity. As it is possible to deduce from *Fig. 2* the score values deriving from *DGS_S-W* method application are lower or in the most of the cases equal to those calculated with *S-W Affine Gap* method. Differently from *S-W Affine Gap* method [3], *DGS_S-W* calculates indeed the score of a data cell by taking into account only the gaps open or extended during the immediately previous calculation (by means of ad hoc boolean variables). The *S-W Affine Gap* method [3] instead allows to keep trace of gap events that occurred many steps before in the calculation, ensuring that the score attributed to a data cell is effectively the maximum among those obtained when considering all the possible alignment paths.
None or small score differences can be however observed (x-axis of *Fig. 2.a*) for most of the alignments calculated (y-axis of *Fig. 2.a*), independently from the parameters settings imposed. Looking at the graphs relative to Pfam [13] and

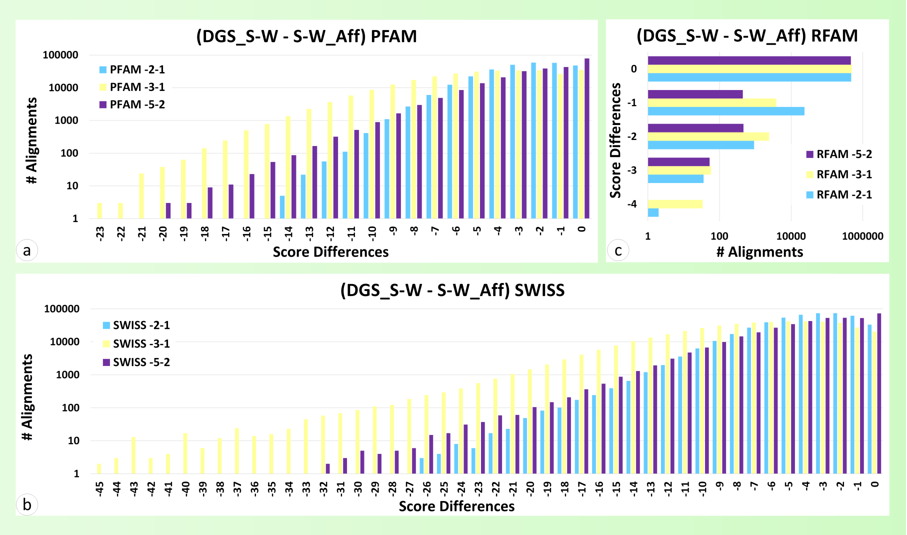


Fig. 2. Differences between *DGS_S-W* and *S-W Affine Gap* methods application computed for the three datasets (Swiss-Prot, Pfam and Rfam) imposing three settings of gap open and gap extension penalties (-2 -1, -3 -1, -5 -2).

SWISS-PROT [14] (*Fig. 2.a* and *Fig. 2.b*) databases, it seems furthermore clear, that an increasing ratio between gap open and gap extension penalty values is reflected in a bigger number of alignments characterized by considerable score differences. With the main objective of obtaining alignment scores as similar as possible to those provided by *S-W Affine Gap* method [3] application, is so recommended to select conveniently gap open and gap extension penalties. The same behaviour can not be observed in relation to the alignments performed on the NT sequences, as it is possible to deduce from the test performed on Rfam [15] database (*Fig. 2.c*). It is worth noting however that the two applied algorithms provide in the most of the cases very similar results.

An example of alignment performed using SWISS-PROT [14] database, for which the two algorithms reported the biggest amount of score differences (with gap open and gap extension penalties respectively equal to -3 and -1), is shown in *Fig. 3*. It can be appreciated that the bigger differences between *DGS_S-W* and *S-W Affine Gap* [3] alignment scores, represented with blue dots, are located only in the *B* quadrant of the plot, having on the x-axis the *Sbj* and on y-axis the *Qry* sizes. The *B* quadrant contains those Maximum Alignment Scores that probably correspond to couples of NTs or AAs located approximately at the end of the two compared sequences. The increasing score differences detected are due to the fact that at this positions, when both *Qry* and *Sbj* have high length sizes, a conspicuous number of different alignment paths can be potentially explored: Divergent predictions have been so accumulated essentially because of the different gap analysis performed by the two different approach, as before highlighted.

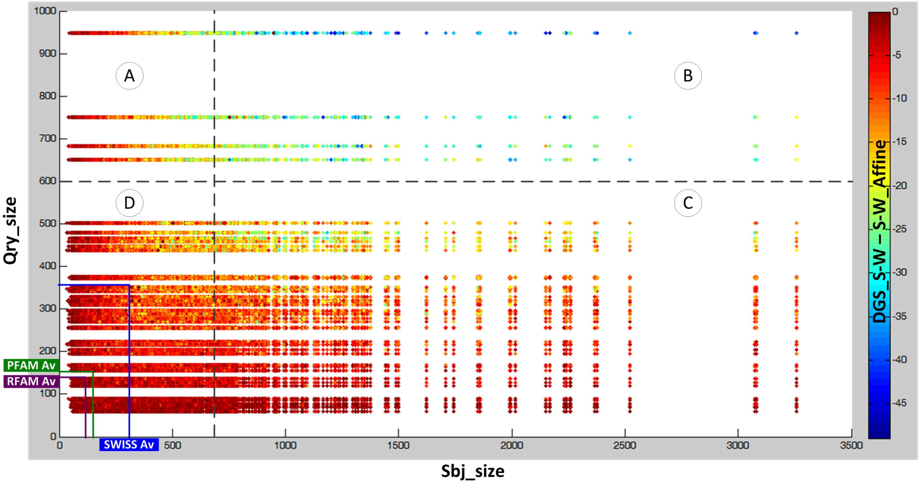


Fig. 3. Differences between *DGS_S-W* and *S-W Affine Gap* compared with the aligned sequences sizes. Test computed using Swiss-Prot data with settings of gap open and gap extension equal to -3 -1. On the y-axis are reported the average lengths of Pfam and Rfam sequences databases (green and violet labels) whereas on x-axis that relative to SWISS-PROT database (blue label). The quadrants A, B, C and D divide the matrix in four areas characterized by different sequences sizes.

It is worth noting however that in the other quadrants, corresponding to the differences calculated for *Qry* and *Sbj* sequences of smaller sizes, the alignment scores provided by the two different algorithms are very similar (red dots on the plot) proving once again the affidability of the alignment scores calculated by means of *DGS_S-W* method.

As previously discussed objective of the novel *DGS_S-W* method, apart from providing alignment scores comparable with those calculated applying the *S-W Affine Gap* method [3], has been identified in the reduction of the time required for the computation with respect to the *S-W Affine Gap* method [3]. In order to assess the reached computational performances, different tests have been performed on a Symmetric Multi-processing (SMP) architecture, namely a 4+4 Intel(R) Core(TM) i7 CPUs 920 2.67 GHz machine, on which no other user processes were running. The time spent to perform the alignment of the input files under examination using *S-W Linear Gap*, *S-W Affine Gap* [3] and *DGS_S-W* methods, was retrieved considering as starting point of the analysis the *Matrix Score F* initialization and as ending point the Maximum Alignment Score detection. Two different speed up percentages have been calculated: One to compare *DGS_S-W* with *S-W Linear Gap* method (*Eq.7*) and the other for *DGS_S-W* and *S-W Linear Gap* methods (*Eq.8*). Also for these analyses the before described configurations were selected.

$$100 * (S\text{–}W\_Linear\ -\ DGS\_S\text{–}W)\ /\ S\text{–}W\_Linear \tag{7}$$

$$100 * (S\text{–}W\_Affine\ -\ DGS\_S\text{–}W)\ /\ S\text{–}W\_Affine \tag{8}$$

In *Fig. 4* are reported the variation values calculated with respect to *S-W Linear Gap* method [1] (first nine bars) and those obtained against *S-W Affine Gap* method [3] (last nine bars). As expected the computational time required for *S-W Linear Gap* method application is always lower than that necessary for *DGS_S-W* alignment scores calculation, because of the high simplicity of the equations implemented by the first algorithm. The maximum percentage variation, obtained when aligning data from Rfam [15] database imposing a gap open

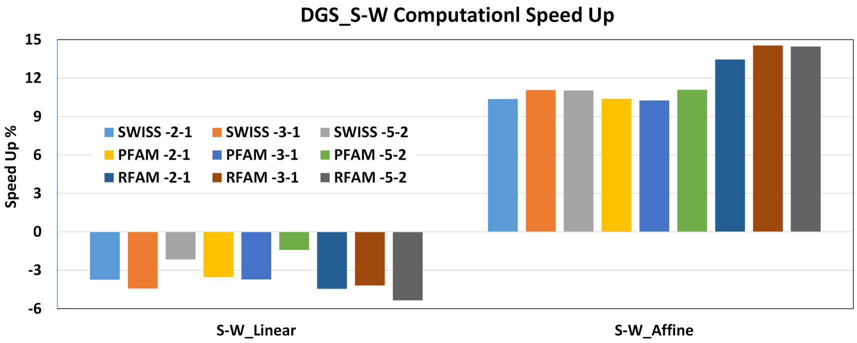


Fig. 4. Speed up reached by *DGS_S-W* method applications with respect to *S-W Linear Gap* and *S-W Affine Gap* methods.

penalty of -5 and a gap extension of -2, is however not so remarkable, being equal to 5.35%. Furthermore this value drops to 1.41% when tests are performed on Pfam [13] database fixing the two gap penalties to -5 and -2.
When comparing instead *DGS_S-W* with *S-W Affine Gap* [3] method the opposite behaviour can be observed. The time required by the novel method to perform the computations is always lower than the time need to apply *S-W Affine Gap* method [3] with a maximum speed-up reached on Rfam [15] database and equal to 14.54%.
In the light of these considerations it is finally possible to affirm that the proposed algorithm is capable to provide alignment scores comparable to those calculated using *S-W Affine Gap* method [3], as highlighted in *Fig. 2*, with a meaningful reduction of the computational costs (see *Fig. 4*) that become closer to those proper of *S-W Linear Gap* method usage.

4 Conclusions and Future Works

The method proposed in this paper, called *DGS_S-W*, represents an optimised version of the well known *S-W Affine Gap* one, at the days considered the most accurate method when looking for local sequences similarities, being capable to evaluate with different penalty scores gap open and gap extension events. Nevertheless the method application is characterized by remarkable computational costs and memory requirements that constitute the main drawbacks related to its usage. Starting from these considerations, the novel *DGS_S-W* method aims at: i) Providing local alignment scores with an accuracy level in the sequences similarity evaluation proper of *S-W Affine Gap* method usage; ii) Performing the calculation of the local alignment scores between sequences with computational costs (in terms of time and memory) close to those typical of *S-W Linear Gap* method application.
In order to achieve these objectives, the two integer gap matrices of *S-W Affine Gap* method have been replaced by two boolean variables capable to keep track of a gap open or a gap extension event leading to a reduction of the memory requirements (from $3*N*M$ to $N*M$) and to a decreasing in the number of operations performed (by a factor of 2) with respect to the *S-W Affine Gap* method.

Tests performed on input sequences extracted from three publically available databases (i.e. SWISS-PROT [14], Pfam [13] and Rfam [15]) proven *DGS_S-W* method potentiality in: i) Providing alignment scores in the most of the cases equal to those calculated thanks to *S-W Affine Gap* method application; ii) Performing the calculations with computational costs significantly lower than those proper of *S-W Affine Gap* method usage (with a maximum speed up reached on Rfam [15] database equal to 14,54%) and at the same time not so higher with respect to the *S-W Linear Gap* method application.
Furthermore the selected input sequences, being characterized by mean lengths similar to those of the real whole databases, allowed us to hypothesize *DGS_S-W* method to assume the same behaviour we highlighted during the tests performed

on reduced input files, even when aligning input files of bigger dimensions. It becomes evident, to the light of these findings, the great potential applications of the proposed method especially in the contest of NGS data analysis characterized, as said, by the need for very powerful and computationally cheap bioinformatics algorithms, capable to treat the big amount of data produced by NGS technologies.

References

1. Smith, T.F., Waterman, M.S.: Identification of common molecular subsequences. J Mol Biol., 147, 195–197 (1981)
2. Altschul, S.F., Gish, W., Miller, W., Myers, E.W., Lipman, D.J.: Basic local alignment search tool. J Mol Biol., 215, 403-410 (1990)
3. Gotoh, O.: An improved algorithm for matching biological sequences. J Mol Biol., 162, 705–708 (1982)
4. Durbin, R., Eddy, S.R., Krogh, A., Mitchison, G.: Biological sequence analysis: Probabilistic models of proteins and nucleic acids. Cambridge University Press, 147, 195-197 (1998)
5. Metzker, M.L.: Sequencing technologies the next generation. Nat Rev Genet., 11, 31–46 (2010)
6. Li, H., Homer, N.: A survey of sequence alignment algorithms for next-generation sequencing. Brief Bioinform. , 11, 473–483 (2010)
7. Henikoff, S., Henikoff, J.G.:Amino acid substitution matrices from protein blocks. Proceedings of the National Academy of Sciences of the United States of America, 89, 910-919 (1992)
8. Dring, A., Weese, D., Rausch, T., Reinert, K.: SeqAn an efficient, generic C++ library for sequence analysis. BMC Bioinformatics, 9-11, (2008)
9. Isaev, A.: Introduction to Mathematical Methods in Bioinformatics. Springer (2006)
10. Needleman, C., Wunsch, C.: A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of two Proteins. Journal of Molecular Biology, 48, 443–453, (1970)
11. Urgese, G., Graziano, M., Vacca, M., Awais, M., Frache, S., Zamboni, M.: Protein Alignment HW/SW optimisations, in IEEE International Conference on Electronics, Circuits, and Systems (ICECS), 145-148, 10.1109/ICECS.2012.6463779, (2012)
12. Causapruno, G., Urgese, G., Vacca, M., Graziano, M., Zamboni, M.: Protein Alignment Systolic Array Throughput Optimization, IEEE Transaction on Very Large Scale Integration Systems (TVLSI), 10.1109/TVLSI.2014.2302015, (In press)
13. Punta, M., Coggill, P.C., Eberhardt, R.Y., Mistry, J., Tate, J., Boursnell, C., Pang, N., Forslund, K., Ceric, G., Clements, J., Heger, A., Holm. L., Sonnhammer, E.L.L., Eddy, S.R., Bateman, A., Finn, R.D.: The Pfam protein families database. Nucleic Acids Research, 40, D290–D301, (2012)
14. Bairocha, A., Coggill, R.: The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. Nucleic Acids Research, 28, 45–58, (2000)
15. Burge, S.W., Daub, J., Eberhardt, R., Tate, J., Barquist, L., Nawrocki, E.P., Eddy, S.R., Gardner, P. P., Bateman, A.: Rfam 11.0: 10 years of RNA families. Nucleic Acids Research, 41, D226–D232, (2012)