Efficiency of Method for Biological Sequence Alignment

Hristo Stoev

***Abstract* — Bioinformatics is one of the most rapidly developing and promising sciences in recent years, which makes it possible to carry out scientific experiments using computer models and simulations based on effective methods, algorithms and means of storage, management, analysis and interpretation of a huge amount of biological data. A challenge in data analysis in bioinformatics is to offer integrated and modern access to the progressively increasing volume of data, as well as efficient algorithms for their processing. Considering the vast databases of biological data available, it is extremely important to develop efficient methods for processing those data.**

***Index Terms* — bioinformatics, biological data sequences, sequences alignment, trilateration.**

I. INTRODUCTION

A major problem in biological data processing is the search for similar sequences in a database. Algorithms such as Needleman-Wunsch [1] and Smith-Waterman [2], which accurately determine the degree of similarity between two sequences, take a long time to process all entries in large datasets. For faster searches in large databases, scientists use heuristic methods and algorithms that significantly speed up the search time, but reduce the quality of the results obtained. FASTA is a software package for DNA and protein sequence alignment that introduces heuristic methods for sequence alignment - querying the entire database. BLAST is one of the most widely used sequence search tools [3, 4]. The heuristic algorithm it uses is much faster than other approaches, such as computing an optimal arrangement. The BLAST algorithm is more time-efficient than FASTA, searching only the most significant sequences, but with comparable sensitivity. Even the parallel execution of the above algorithms is limited by the hardware systems [5-13]. The metaheuristic method for multiple sequence ordering adopts the idea of generating a favorite sequence, after which all other sequences from the database are compared with the favorite sequence [14]. In this way, the favorite sequence becomes a benchmark for the rest of the sequences in the database. Some problems arise when using this approach, such as insertion of new records or removing any of the existing one.

Since the favorite sequence is generated based on the existing records: (1) Change of the data set requires the favorite sequence to be recalculated. (2) Each of the sequences in the database must be compared again with the newly generated favorite sequence to obtain a new result, which consumes computational time and resources. (3) Each database computes own favorite sequence, and this can lead to problems when merging different databases, especially in big data, where there is a collection of many different database structures and access methods.

Evaluation of biological sequence alignment algorithms mainly considers algorithm efficiency and sensitivity to obtain the best alignment results. The Smith-Waterman algorithm for sequence pair alignment is highly sensitive, but its complexity is very high. The FASTA and BLAST methods decrease the predicted sensitivity in exchange for an increase in speed. The ClustalW algorithm is the most common and efficient among multiple sequence alignment algorithms. The main issue in sequence alignment is whether the sensitivity of the alignment and the efficiency of the algorithm are improved for sequences with large differences.

To improve the idea of the existing heuristic algorithms, an attempt will be made to propose improvements in the directions:

1. Constant favorite sequence – i.e. independent of the data set in the database and remaining the same when data set is changed;
2. Avoiding comparisons or reducing the number of comparisons with the favorite sequence in searching of the database (for each sequence a complex comparison algorithm is applied against the favorite sequence)
3. Unification/standardization of sequence favorites for all databases.

The purpose of the research in this chapter is to propose a new efficient and unified method for arranging DNA sequences based on the trilateration method. This method offers a solution for the three main problems in biological sequence alignment: (1) creating a constant favorite sequence, (2) reducing the number of comparisons with the favorite sequence, and (3) unifying / standardizing the favorite sequence by defining benchmark sequences.

II. METHOD FOR DNA SEQUENCE ALIGNMENT BASED ON TRILATERATION

At the heart of the idea of a favorite sequence is to find a starting point - a benchmark against which the rest of the data in the database can be analyzed. Or, looking at it mathematically, one could represent the sequence favorite as a function of N unknowns (speaking of DNA the unknowns are the 4 bases: adenine, thymine, guanine, and cytosine), then represent the remaining base entries again as functions of the same variables. In such a case, the similarity comparison would represent the distance of the individual sequence to the favorite sequence. In other words, find the location of

https://doi.org/10.47978/TUS.2022.72.03.007

a point described by the sequence function relative to another point defined by the favorite sequence function. When comparing to a sequence favorite, there is a set of points (the database entries) and since there is no coordinate system, a point is generated somewhere around the center of the cloud of points that is used as a reference (sequence favorite). But if some kind of coordinate system is introduced, or three or more reference points are found, then it would be possible, by means of elementary analytical geometry, or in particular trilateration, to determine the positions of the points relative to each other, which will reflect the degree of similarity between the records in the database. Also, to eliminate the need of calculation of the sequence favorite.

A new method for aligning DNA sequences, called CAT, based on the trilateration method, was proposed [15, 16]. Three constant benchmarks have been established for the application of trilateration, which creates a constant favorite sequence - i.e. it does not depend on data set in the database and remains the same when it changes.

Since the reference sequences established are constant (i.e. they do not depend on either the data set or their number), this allows calculations to be made at the very beginning - when the sequences are entered into the database and this is useful information accompanying each sequence. This way, sequences will not have to be compared to favorite sequence during lookup (which is the slowest operation), but instead only the utility information generated during the data entering will be compared.

By establishing the benchmark sequences, problem (3) unification/standardization of favorite sequences for all databases is also solved. There are now unified sequences that are standardized for all databases using the described alignment algorithm.

The calculations proposed in the presented method are relatively simple and fast to implement, which makes it suitable for application as a first step in biological sequence alignment algorithms such as FASTA, as well as for multiple alignments such as ClustalW.

III. Experimental Results

After further analysis of the results, it was found that the greatest deviation in the accuracy of the CAT method occurs when comparing sequences with a relatively large difference in lengths. For example, a sequence of length 20 to be searched in a sequence of length 150. The length of the shorter sequence is several times shorter than the length of the sequence in which it is searched.

After analyzing the results and the main idea of the method (to represent an entire sequence as a point of a coordinate system, based on statistical information about the bases, more abstractly expressed - to represent a polynomial with n number of terms as a point in space), we can infer why this difference occurs.

When calculating the statistical information for the two sequences, since they are of different lengths (n and k in number of members, where n is many times greater than k), for the longer one there is an accumulation of very redundant statistical information, which results on the final result. That is, in the longer sequence there are many bases that, when compared with a precise algorithm, will not be relevant to the alignment of the sequences.

An example

AGDTDDTTGAG and DTG would be aligned AGDTDTTGAG, after alignment it is not AGTD relevant to the result, but it is involved in representation of the sequence as a point of the coordinate system and leads to deviations when applying the CAT method.

We could improve the accuracy of the method in such situation, if we could reduce the length of the longer sequences so as to isolate the bases that would not be involved in the final result anyway, we would get more accurate results. An improvement direction is to find the error accumulation factor as the ratio of the lengths of the two sequences and apply it to the calculation of the similarity value of the sequences.

delat_x = p1.Length / p2.Length

p1,p2 profiles of the compared sequences

We can apply this coefficient directly in the formula:

$$S_1S_2 = \sqrt{|AD_1.delta_x - AD_2|^2 + |h_1.delta\_x - h_2|^2}$$

To further research of the CAT method, the following experiments were performed, generating sequences of different lengths:

1. A comparison of the generated sequence with itself was made – WI (Table IV).
2. It is taken sub sequence from the beginning of the generated and compared with the generated – FH (Table I).
3. It is taken sub sequence from the end of the generated and compared with the generated – SH (Table II).
4. It is taken sub sequence from the environment of the generated and compared to the generated – M (Table III)
5. A second sequence of shorter length than the generated is generated – R (Table V).

When the second sequence is many times shorter (over 6 times and more) than the first, the Needleman-Wunsch algorithm makes an optimal alignment so that after the alignment the bases of the shorter sequence correspond to positions in the longer one. In other words, we get 1 when we count the matches in the ordered sequences. Which is totally expected for this algorithm. While for the CAT method we get values from 0.99-0.63, and with the improved CAT method values 0.5 - 0.1. This is due to multiple length overruns in both sequences, resulting in the accumulation of a lot of redundant statistical information in the CAT method. But from a speed point of view in Needleman-Wunsch as the length of either of the two compared sequences increases, the execution time increases. It starts from 0.06 milliseconds for the shortest sequences to 7.89 milliseconds for the longest. In the CAT method, it oscillates around 0.002 milliseconds.

TABLE I

COMPARISON RESULTS OF SUB SEQUENCE TAKEN FROM THE BEGINNING OF THE GENERATED AND COMPARED WITH THE GENERATED. BOLDED ROWS ARE AVERAGE OF THE BELOW ROWS SECTION. BOLDED SUB / SEQUENCE LENGTH IS LENGTH OF THE ORIGINAL GENERATED SEQUENCE

Sub / Sequence length	FirstHalf Average of CAT	Average of NW + Gaps	Average of NW	Average of CAT Elapsed Time	Average of NW Elapsed Time
100	**0,9290**	**1**	**1**	**0,00517**	**0,81**
10	0,7923	1	1	0,00064	0,17
30	0,9096	1	1	0,00050	0,44
50	0,9406	1	1	0,01379	0,67
70	0,9643	1	1	0,00817	0,91
90	0,9782	1	1	0,00062	1,24
97	0,9893	1	1	0,00727	1,43
1000	**0,9812**	**1**	**1**	**0,00621**	**137,56**
100	0,9488	1	1	0,00124	23,74
300	0,9740	1	1	0,00134	71,78
500	0,9841	1	1	0,00137	118,06
700	0,9893	1	1	0,01542	166,43
900	0,9941	1	1	0,01641	214,88
970	0,9969	1	1	0,00146	230,45
10000	**0,9936**	**1**	**1**	**0,01046**	**15351,73**
1000	0,9822	1	1	0,00673	2528,73
3000	0,9912	1	1	0,01494	8123,24
5000	0,9945	1	1	0,00875	13633,87
7000	0,9964	1	1	0,01492	18597,60
9000	0,9982	1	1	0,00199	23914,08
9700	0,9991	1	1	0,01541	25501,87
50000	**0,9972**	**1**	**1**	**0,00126**	**67806,26**
5000	0,9931	1	1	0,00111	11708,00
15000	0,9961	1	1	0,00097	35798,11
25000	0,9975	1	1	0,00128	58130,19
35000	0,9983	1	1	0,00173	84807,04
45000	0,9992	1	1	0,00127	105537,36
48500	0,9995	1	1	0,00123	116363,51

TABLE II

COMPARISON RESULTS OF SUB SEQUENCE TAKEN FROM THE END OF THE GENERATED AND COMPARED WITH THE GENERATED. BOLDED ROWS ARE AVERAGE OF THE BELOW ROWS SECTION. BOLDED SUB / SEQUENCE LENGTH IS LENGTH OF THE ORIGINAL GENERATED SEQUENCE

Sub / Sequence length	SecondHalf Average of CAT	Average of NW + Gaps	Average of NW	Average of CAT Elapsed Time	Average of NW Elapsed Time
100	**0,8783**	**1**	**1**	**0,000570**	**0,81**
10	0,7844	1	1	0,000497	0,09
30	0,8773	1	1	0,000533	0,36
50	0,8926	1	1	0,000543	0,80
70	0,9011	1	1	0,000586	0,98
90	0,9070	1	1	0,000616	1,25
97	0,9061	1	1	0,000642	1,37
1000	**0,9766**	**1**	**1**	**0,006232**	**136,81**
100	0,9468	1	1	0,001273	23,05
300	0,9749	1	1	0,001711	71,48
500	0,9841	1	1	0,001356	115,41
700	0,9888	1	1	0,017584	165,59
900	0,9943	1	1	0,007782	212,20
970	0,9705	1	1	0,007685	233,12
10000	**0,9938**	**1**	**1**	**0,009993**	**15416,21**
1000	0,9837	1	1	0,029912	2576,16
3000	0,9916	1	1	0,015799	8217,27
5000	0,9945	1	1	0,001861	14180,53
7000	0,9962	1	1	0,001881	18392,94
9000	0,9980	1	1	0,008521	23705,93
9700	0,9989	1	1	0,001890	25611,14
50000	**0,9972**	**1**	**1**	**0,001156**	**68162,06**
5000	0,9927	1	1	0,001200	11773,05
15000	0,9959	1	1	0,001088	35236,93
25000	0,9975	1	1	0,001000	60064,42
35000	0,9983	1	1	0,001181	84267,90
45000	0,9993	1	1	0,001381	108520,13
48500	0,9996	1	1	0,001088	114692,08

TABLE III

COMPARISON RESULTS OF SUB SEQUENCE TAKEN FROM THE MIDDLE OF THE GENERATED AND COMPARED WITH THE GENERATED. BOLDED ROWS ARE AVERAGE OF THE BELOW ROWS SECTION. BOLDED SUB / SEQUENCE LENGTH IS LENGTH OF THE ORIGINAL GENERATED SEQUENCE

Row Labels	Middle Average of CAT	Average of NW + Gaps	Average of NW	Average of CAT Elapsed Time	Average of NW Elapsed Time
100	**0,8743**	**1**	**1**	**0,000726**	**0,82**
10	0,7748	1	1	0,000488	0,13
30	0,8652	1	1	0,000538	0,58
50	0,8922	1	1	0,000624	0,69
70	0,9001	1	1	0,001486	0,99
90	0,9052	1	1	0,000614	1,23
97	0,9080	1	1	0,000605	1,32
1000	**0,9621**	**1**	**1**	**0,004918**	**137,93**
100	0,9392	1	1	0,001307	23,93
300	0,9598	1	1	0,007767	70,20
500	0,9647	1	1	0,007967	118,39
700	0,9682	1	1	0,001635	170,74
900	0,9700	1	1	0,009309	214,29
970	0,9706	1	1	0,001521	230,02
10000	**0,9925**	**1**	**1**	**0,009768**	**15130,27**
1000	0,9847	1	1	0,007668	2627,08
3000	0,9911	1	1	0,018686	7942,96
5000	0,9943	1	1	0,001840	13419,71
7000	0,9963	1	1	0,015638	18145,65

9000	0,9981	1	1	0,001936	23498,57
9700	0,9908	1	1	0,012790	25335,28
50000	**0,9963**	**1**	**1**	**0,001285**	**68099,07**
5000	0,9916	1	1	0,001465	11569,67
15000	0,9958	1	1	0,001294	34961,92
25000	0,9973	1	1	0,001300	61926,51
35000	0,9984	1	1	0,001269	83043,45
45000	0,9992	1	1	0,001125	107200,75
48500	0,9959	1	1	0,001244	115496,29

TABLE IV
COMPARISON RESULTS OF GENERATED SEQUENCE COMPARED WITH ITSELF. BOLDED ROWS ARE AVERAGE OF THE BELOW ROWS SECTION. BOLDED SUB / SEQUENCE LENGTH IS LENGTH OF THE ORIGINAL GENERATED SEQUENCE

Sub / Sequence length	WithItself Average of CAT	Average of NW + Gaps	Average of NW	Average of CAT Elapsed Time	Average of NW Elapsed Time
100	1	1	1	0.023699	1.59419
100	1	1	1	0.023699	1.59419
1000	1	1	1	0.000981	238.639932
1000	1	1	1	0.000981	238.639932
10000	1	1	1	0.001189	26981.1444
10000	1	1	1	0.001189	26981.1444
50000	1	1	1	0.0174	116611.309
50000	1	1	1	0.0174	116611.309

When the ratio of the length of the sequences is above 0.8, we observe how values calculated by CAT approach 1, but no collisions are observed, i.e. we do not have values equal to 1. Approaching 1 is justified by the accumulation of many statistics, and the fact that no collisions are observed - that the method is sensitive to the different sequences and their arrangement.

TABLE V
COMPARISON RESULTS OF RANDOM GENERATED SEQUENCE WITH ANOTHER RANDOM GENERATED SEQUENCE WITH DIFFERENT LENGTH. BOLDED ROWS ARE AVERAGE OF THE BELOW ROWS SECTION. BOLDED SUB / SEQUENCE LENGTH IS LENGTH OF THE ORIGINAL GENERATED SEQUENCE

Sub / Sequence length	Random Average of CAT	Average of NW + Gaps	Average of NW	Average of CAT Elapsed Time	Average of NW Elapsed Time
100	**0,8903**	**0,7877**	**0,7270**	**0,008364**	**0,88**
10	0,7783	1,0000	1,0000	0,000669	0,18
30	0,8737	0,9845	0,9543	0,005990	0,52
50	0,9096	0,9001	0,8086	0,010066	0,66
70	0,9119	0,7686	0,6717	0,009359	1,02
90	0,9169	0,6434	0,5712	0,026861	1,25
97	0,9228	0,6126	0,5454	0,004950	1,25
100	0,9187	0,6049	0,5380	0,000656	1,25
1000	**0,9661**	**0,8084**	**0,7530**	**0,009034**	**148,23**
100	0,9427	1,0000	1,0000	0,014773	23,81
300	0,9617	0,9937	0,9795	0,001457	71,17
500	0,9688	0,9212	0,8431	0,014828	117,70
700	0,9698	0,7978	0,7073	0,008580	161,56
900	0,9736	0,6708	0,6002	0,001563	206,55
970	0,9720	0,6430	0,5753	0,013244	222,55
1000	0,9742	0,6325	0,5655	0,008790	234,28
10000	**0,9898**	**0,8137**	**0,7603**	**0,008312**	**16907,70**
1000	0,9813	1,0000	1,0000	0,009414	2557,00
3000	0,9892	0,9957	0,9856	0,007863	7915,84
5000	0,9902	0,9271	0,8541	0,001912	13119,75
7000	0,9912	0,8039	0,7158	0,015851	18260,72
9000	0,9925	0,6787	0,6090	0,012952	23842,70
9700	0,9922	0,6496	0,5832	0,001897	26408,65
10000	0,9920	0,6409	0,5743	0,008292	26249,21
50000	**0,9953**	**0,8146**	**0,7617**	**0,001344**	**78652,58**
5000	0,9911	1,0000	1,0000	0,001063	11904,96
15000	0,9943	0,9961	0,9871	0,001169	38402,49
25000	0,9956	0,9282	0,8564	0,001369	61006,88
35000	0,9966	0,8055	0,7181	0,001400	85139,63
45000	0,9965	0,6792	0,6099	0,001600	111722,62
48500	0,9964	0,6511	0,5848	0,001444	118340,87
50000	0,9965	0,6422	0,5758	0,001363	124050,62

TABLE VI
COMPARISON RESULTS OF RANDOM GENERATED SEQUENCE WITH ANOTHER RANDOM GENERATED SEQUENCE WITH DIFFERENT LENGTH WITH CALCULATED STANDARD DEVIATION. BOLDED ROWS ARE AVERAGE OF THE BELOW ROWS SECTION. BOLDED SUB / SEQUENCE LENGTH IS LENGTH OF THE ORIGINAL GENERATED SEQUENCE

Sub / Sequence length	Random Average of CAT	Average of NW + Gaps	Average of NW	StdDevp of Delta CAT/Needleman Wunsch	StdDevp of Delta CAT/Needleman Wunsch + Gaps
100	0,7877	0,7270	0,7458	0,1298	0,1154
10	1,0000	1,0000	0,5607	0,0912	0,0913
30	0,9845	0,9543	0,7657	0,0529	0,0519
50	0,9001	0,8086	0,8153	0,0512	0,0264
70	0,7686	0,6717	0,7932	0,0545	0,0480
90	0,6434	0,5712	0,7781	0,0540	0,0573
97	0,6126	0,5454	0,7793	0,0541	0,0596
100	0,6049	0,5380	0,7283	0,0535	0,0581
1000	**0,8084**	**0,7530**	**0,8756**	**0,1549**	**0,1310**
100	1,0000	1,0000	0,8871	0,0244	0,0244
300	0,9937	0,9795	0,9337	0,0150	0,0170
500	0,9212	0,8431	0,9206	0,0156	0,0139
700	0,7978	0,7073	0,8907	0,0159	0,0146
900	0,6708	0,6002	0,8699	0,0166	0,0168
970	0,6430	0,5753	0,8574	0,0170	0,0189
1000	0,6325	0,5655	0,7698	0,0148	0,0162
10000	**0,8137**	**0,7603**	**0,9121**	**0,1677**	**0,1432**

1000	1,0000	1,0000	0,9643	0,0072	0,0072
3000	0,9957	0,9856	0,9814	0,0034	0,0047
5000	0,9271	0,8541	0,9522	0,0046	0,0043
7000	0,8039	0,7158	0,9201	0,0042	0,0041
9000	0,6787	0,6090	0,8952	0,0046	0,0050
9700	0,6496	0,5832	0,8871	0,0041	0,0045
10000	0,6409	0,5743	0,7842	0,0048	0,0056
50000	**0,8146**	**0,7617**	**0,9193**	**0,1702**	**0,1464**
5000	1,0000	1,0000	0,9833	0,0044	0,0044
15000	0,9961	0,9871	0,9895	0,0028	0,0020
25000	0,9282	0,8564	0,9589	0,0018	0,0017
35000	0,8055	0,7181	0,9257	0,0017	0,0016
45000	0,6792	0,6099	0,8993	0,0021	0,0024
48500	0,6511	0,5848	0,8923	0,0027	0,0030
50000	0,6422	0,5758	0,7861	0,0021	0,0023

From the Table VI it is clear that the standard deviation is in the range of 0.1, which shows that the results obtained by CAT are actually in a different dimension from those obtained by Needleman-Wunsch.

To illustrate the trend of the CAT score curve versus the Needleman-Wunsch ranking, we will need to compare the dimensions of the two types of scores. To do this, we calculated the average of the two deltas CAT / Needleman-Wunsch and CAT / Needleman-Wunsch + Gaps (gaps resulting from Needleman-Wunsch sequencing are also taken into consideration here) and used this value to transfer all in one dimension. We subtracted this delta from the resulting CAT values (Table VII).

TABLE VII
COMPARISON RESULTS OF RANDOM GENERATED SEQUENCE WITH ANOTHER RANDOM GENERATED SEQUENCE WITH DIFFERENT LENGTH WITH CORRECTION - AVERAGE OF THE DELTA PER GROUP. BOLDED ROWS ARE AVERAGE OF THE BELOW ROWS SECTION. BOLDED SUB / SEQUENCE LENGTH IS LENGTH OF THE ORIGINAL GENERATED SEQUENCE

	Random				
Sub / Sequence length	**Average of CAT**	**Average of NW + Gaps**	**Average of NW**	**Average of CAT - Average Delat**	**Average of CAT - Average Delta + Gaps**
100	**0,8903**	**0,7877**	**0,7270**	**0,74581**	**0,76046**
10	0,7783	1,0000	1,0000	0,56072	0,56072
30	0,8737	0,9845	0,9543	0,76572	0,75905
50	0,9096	0,9001	0,8086	0,81532	0,83196
70	0,9119	0,7686	0,6717	0,79321	0,81742
90	0,9169	0,6434	0,5712	0,77811	0,79616
97	0,9228	0,6126	0,5454	0,77929	0,79609
100	0,9187	0,6049	0,5380	0,72833	0,76179
1000	**0,9661**	**0,8084**	**0,7530**	**0,87560**	**0,89093**
100	0,9427	1,0000	1,0000	0,88708	0,88708
300	0,9617	0,9937	0,9795	0,93375	0,93082
500	0,9688	0,9212	0,8431	0,92060	0,94012
700	0,9698	0,7978	0,7073	0,89075	0,91337
900	0,9736	0,6708	0,6002	0,86986	0,88750
970	0,9720	0,6430	0,5753	0,85738	0,87431
1000	0,9742	0,6325	0,5655	0,76981	0,80333
10000	**0,9898**	**0,8137**	**0,7603**	**0,91208**	**0,92733**
1000	0,9813	1,0000	1,0000	0,96425	0,96425
3000	0,9892	0,9957	0,9856	0,98139	0,98106
5000	0,9902	0,9271	0,8541	0,95222	0,97033
7000	0,9912	0,8039	0,7158	0,92013	0,94200
9000	0,9925	0,6787	0,6090	0,89520	0,91263
9700	0,9922	0,6496	0,5832	0,88713	0,90373
10000	0,9920	0,6409	0,5743	0,78422	0,81732
50000	**0,9953**	**0,8146**	**0,7617**	**0,91930**	**0,93473**
5000	0,9911	1,0000	1,0000	0,98329	0,98329
15000	0,9943	0,9961	0,9871	0,98946	0,99050
25000	0,9956	0,9282	0,8564	0,95890	0,97687
35000	0,9966	0,8055	0,7181	0,92573	0,94761
45000	0,9965	0,6792	0,6099	0,89931	0,91662
48500	0,9964	0,6511	0,5848	0,89230	0,90888
50000	0,9965	0,6422	0,5758	0,78612	0,81932

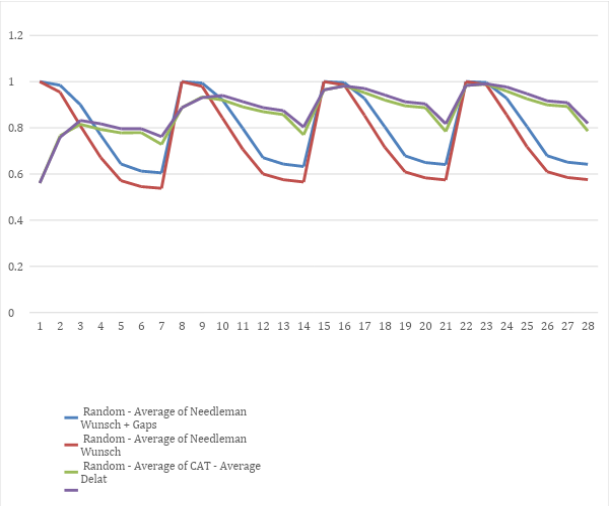

Fig. 1. Graphical representation of Table VII results

The graph in Fig. 1 shows how the profiles calculated by CAT start to follow the trend of the curve obtained after the Needleman-Wunsch alignment.

IV. CONCLUSION

The paper presents the results of the developed program implementation of the proposed method CAT for biological sequences alignment. Experiments have been carried out with different datasets for DNA sequence alignment using the triplet-based CAT method. An analysis of the experimental results was made.

ACKNOWLEDGMENT

This research was funded by National Science Fund, Bulgarian Ministry of Education and Science, grant number KP-06-N37/24, project “Innovative Platform for Intelligent Management and Analysis of Big Data Streams Supporting Biomedical Scientific Research”.

REFERENCES

1. Needleman, S., 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.
2. Smith, T., Waterman, M.: Identification of Common Molecular Subsequences, Journal Molecularly Biology, 147, 195-197, 1981.
3. Altschul, S., et al: Basic local alignment search tool, Journal of Molecular Biology, 215(3), 1990.
4. Altschul, S., et al: Gapped BLAST and PSIBLAST: a new generation of protein database search programs, Nucleic Acids Research, 1997, 25:3389–3402.
5. D. Hoksza and D. Svozil, "Multiple 3D RNA structure superposition using neighbor joining", IEEE/ACM Trans. Comput. Biol. Bioinf., vol. 12, no. 3, pp. 520-530, May/Jun. 2015.
6. Ebedes J., and Datta A., Multiple sequence alignment in parallel on a workstation cluster, Bioinformatics, Vol. 20 no. 7, 2004, pp, 1193–1195.
7. Mikhailov D., et al, 2001. Performance Optimization of ClustalW: Parallel ClustalW, HT Clustal, and MULTICLUSTAL, White paper, Silicon Graphics, Mountain View, CA.
8. Thompson J., Higgins D., Gibson T., ClustalW: Improving the Sensitivity of Progressive Multiple Sequence Alignment through Sequence Weighting, Position-Specific Gap Penalties and Weight Matrix Choice, Nucleic Acids Research, Vol. 22, No. 22, 1994, pp. 4673-4680.
9. Cheetham J., et al, Parallel ClustalW for PC Clusters, Proceedings of International Conference on Computational Science and its Applications, Montreal, Canada, 2003.
10. Zhang, F., Qiao, X. Z., Liu, Z. Y.: A Parallel Smith-Waterman Algorithm Based on Divide and Conquer, Proceedings of the Fifth International Conference on Algorithms and Architectures for Parallel Processing ICA3PP'02, 2002.
11. Farrar, M.: Striped Smith-Waterman Speeds Database Searches Six Times over other SIMD Implementations, Bioinformatics, 2007 Jan 15; 23(2), pp.156-61.
12. Sharma C., Agrawal P. and Gupta P., "Article: Multiple sequence alignments with parallel computing", Proc. IJCA Int. Conf. Adv. Comput. Eng. Appl. ICACEA, no. 5, pp. 16-21, Mar. 2014.
13. Sathe S. R. and Shrimankar D. D., "Parallelizing and analyzing the behavior of sequence alignment algorithm on a cluster of workstations for large datasets", Int. J. Comput. Appl., vol. 74, no. 21, pp. 1-13, Jul. 2013.
14. Borovska P., Gancheva V., Landzhev N. Massively parallel algorithm for multiple biological sequences alignment, Proceeding of 36th IEEE International Conference on Telecommunications and Signal Processing (TSP), 2013 pp. 638-642, DOI: 10.1109/ TSP.2013.6614014.
15. Stoev H., An Effective and Unified Method for DNA Sequence Alignment Based on Trilateration, Challenges in Higher Education & Research, vol. 14, pp. 100-105.
16. Gancheva V., Stoev H., DNA sequence alignment method based on trilateration, Bioinformatics and Biomedical Engineering, Lecture Notes in Computer Science, 2019, vol. 11466, Springer, Cham, pp. 271-283, https://doi.org/10.1007/978-3-030-17935-9_25.