Some new heuristic algorithms in analysis of the similarity of DNA-sequences

B.F. Melnikov[1], **E.A. Melnikova**[1], **S.V. Pivneva**[2]

[1]Russian State Social University, Wilhelm Pieck str., 4, Moscow, Russia, 129226
[2]Togliatti State University, Belorusskaya str., 14, Togliatti, Russia, 445020

Abstract. This paper describes algorithms, corresponding computer programs and the results of computations, supplementing results published earlier. We consider the multiple sequence alignment problem, which can be nominated by a central problem in computational biology. For it, we continue to consider some different versions of so-called "triangular norm" defined on the set of triangles formed by the different distance between genomes computed by different algorithms. Basically, the new results are associated with incorrect variants of obtaining the triangle inequality, which for matrices of the order of about 50×50 is violated in the two most successful metrics (including the Panin's metric earlier developed by us) in less than 1% of cases. Possible improvements are related to the use of neural networks that were not used by us in previous calculations. In this case, neural networks solve the inverse problem: we improve (reduce) the overall badness of the matrix of distances between genomes, forcibly changing the previously obtained distances; further, we try to reflect these forced changes in the original algorithms for calculating distances. In this paper, we present the results of concrete calculations obtained by us.

Keywords: DNA-sequences, heuristic algorithms, triangular metric, distance between genomes, neural networks.

1. Introduction and motivation

In this paper, we describe the continuation of research on the issues that we started in [1, 2, 3, 4].

Thus, this paper describes algorithms, corresponding computer programs and the results of computations, supplementing results published earlier. We consider the multiple sequence alignment problem, which can be nominated by a central problem in computational biology. For it, we continue to consider some different versions of so-called "triangular norm". (The name "triangular metric", also sometimes encountered in our previous papers, is also quite possible and does not contain errors, we will not give detailed comments on this thing. However, the "norm" in our case is some more correct, both when we speak about the whole matrix, and when we speak only about the only triangle.) Such norm defined on the set of triangles formed by the different distance between genomes computed by different algorithms. Thus, in this paper the word "metric" will be used only as the distance between the genomes, and the word "norm" as an indicator of the badness of a certain set of such distances.

Our new directions of work consist, first, in the further improvement of Panin's metric, and, secondly, in new variants of investigation of various variants of the triangular norm. Both these directions are interrelated. Namely, we improve the interpretation of the Panin's metric in the same way as it is done in some interpretations of genetic algorithms: we are trying to achieve a combination of parameters, for which the triangular norm reaches a minimum value (or is close to it).

And considering the second problem, i.e., the study of variants of the triangular norm, we proceed as follows. We consider incorrect variants of obtaining the triangle inequality, which for matrices of the order of about 50×50 is violated in the two most successful metrics (including the Panin's metric earlier developed by us) in less than 1% of cases. It is important to note that in order to improve the decrease in the quantitative index of the badness of the entire matrix of considerations, we also, first of all, consider triangles in which the triangle inequality is not violated, but in which the badness value is relatively large. Possible improvements are related to the use of neural networks that were not used by us in previous calculations. In this case, neural networks solve the inverse problem: we improve (reduce) the overall badness of the matrix of distances between genomes, forcibly changing the previously obtained distances; further, we try to reflect these forced changes in the original algorithms for calculating distances.

2. Preliminaries

Thus, like our previous papers, we consider the square symmetric matrix of distances between genomes. There it is necessary to note the following. First, the genomes we choose random enough and took them off the site [5]. Second, like previous works, we consider in fact *three* variants for each of the considered problems:

- for very distant species, including, for example, a mammal "Bison bison" and a reptilia "Apalone spinifera" (we use the official scientific Latin names), see detailed the species' list by the link for direct download [6];
- for a sufficiently close species (human and apes);
- and also for human races (in fact, they can be considered as subspecies of a biological species).

Let us note, looking ahead, we believe that our a theoretical construction is best applicable for more distant species, however, acceptable results are obtained also in two other cases.

Thus, we look at algorithms of comparing the quality of different algorithms that calculate the distance between two genomes. Apparently, all these algorithms are based on the use of various versions of the Levenshtein distance (or Levenshtein metric), see [7] and very many other following papers. It is very important to note, that for the simplest formulation of the problem (to make the strict calculating the value of Levenshtein metric for two given genomes), we unfortunately obtain a very long-running algorithm (program): it has to do with the actual length of the strings of genomes. Therefore, in each of the actual algorithms (see [8, 9, 10, 11] etc.), computation of distances between genomes in reality is a heuristic extension of the exact algorithm for calculating the Levenshtein distance. And apparently, the approach closest to our one is given by [12]; let us note, a little running forward, that it also uses a version of the branch-and-bound method.

Thus, the considered in our previous papers Panin's metric is no exclusion, It also is a heuristic algorithm for calculating the close version of such metric; it is an optimization problems, see [13] etc. However, we used in it a special approach (so called *multi-heuristic approach for discrete optimization problems*), and for it, we use the same heuristic as in very different problems. From many such problems, let us mention two ones only:

- the classical traveling salesman problem (however, we consider our own approach to this problem, and, most importantly, our original way of specifying the input data, different from the traditional geometric placement, see, e.g., [14]);
- and the problem of state-minimization for nondeterministic finite automata, see, e.g., [15, 16].

3. The triangular norms: their study and possible attempts of improvement

It was justified in our previous works cited above, there is desirable that in the matrix of distances between genomes, any of the resulting triangles be close to an acute angled isosceles one with two angles exceeding 60 degrees. Several various empirically selected numerical characteristics describing such differences are also given in our works, see [4] etc. However, in the previous articles we did not consider detailed examples, let us consider them in this section.

In [6], the results of calculations for several metrics and several norms are presented. In this section, we shall consider the Panin's metric only, and 3 norms ("badnesses" for the triangle under consideration). Thus, for each triangle with the sides $a \geq b \geq c$ and the angles $\alpha \geq \beta \geq \gamma$ we considered the following norms:

$$\mathrm{bad}_1 = (\alpha - \beta)/\pi\,, \quad \mathrm{bad}_2 = (\alpha - \beta)/\alpha\,, \quad \mathrm{bad}_3 = (a - b)/a\,.$$

In case if $\alpha \geq 90^o$, we have considered each norm by the maximum possible (1.0) or even usually exceeding this value. We assigned an even greater value to the value of badness in the case when the three considered sides do not form a triangle at all (that is, they do not satisfy the triangle inequality); let us note, running ahead, that similar situations is happened for any of metrics considered by us.

The resulting value of the norm *of the whole matrix* was considered as the arithmetic mean of the norms of all triangles. We note that for the matrices of distances between genomes (usually from 30×30 to 50×50), the number of triangles is $\frac{30 \cdot 29 \cdot 28}{2 \cdot 3} = 4060$ for dimensions 30, and 19600 for dimension 50; from these values, it is clear that the calculations we need are quite difficult.

Thus, let us consider the part of the table given on the page titled "Panin's metrics" of [6] (they are designed as an `xlsx`-file and are available for the direct download), see the table on Fig. 1. (The names of the considered species can be found there on the page titled "Types of animals".)

No. of genomes	1	2	3	4	5	6	7	8	9	10	...	15	...	25	...
1	0	2200	2904	5996	2580	4149	4206	6336	3057	3222	...	3182	...	2300	...
2	2200	0	2638	5998	2860	3922	4416	6000	3373	2982	...	2962	...	2150	...
3	2904	2638	0	6068	2890	4037	4639	6414	3647	3202	...	3201	...	2703	...
4	5996	5998	6068	0	5647	5849	5918	6066	5858	5508	...	5596	...	5618	...
5	2580	2860	2890	5647	0	4426	4145	6445	3274	3589	...	3533	...	3142	...
6	4149	3922	4037	5849	4426	0	4682	6492	4397	3996	...	4006	...	3919	...
7	4206	4416	4639	5918	4145	4682	0	6581	4230	4586	...	4577	...	4571	...
8	6336	6000	6414	6066	6445	6492	6581	0	5893	5731	...	5776	...	5950	...
9	3057	3373	3647	5858	3274	4397	4230	5893	0	3651	...	3579	...	3447	...
10	3222	2982	3202	5508	3589	3996	4586	5731	3651	0	...	1985	...	2953	...
...	...	...	...	...	...	...	...	...	...	...	0	...	...	...	...
15	3182	2962	3201	5596	3533	4006	4577	5776	3579	1985	...	0	...	2940	...
...	...	...	...	...	...	...	...	...	...	...	...	...	0	...	...
25	2300	2150	2703	5618	3142	3919	4571	5850	3447	2953	...	2940	...	0	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	0

Figure 1. The part of the table of Panin's metric.

Let us choose species with the numbers 5, 15, 25; while doing so, we specifically chose exactly the three of those considered, where the metric gives rather poor results (that means the following: the badnesses of Panin's metric give relatively worse results than other metrics comparing most other triangles of the matrix under consideration). For all 5 considered metrics, we obtain the following 5 triangles corresponding to the species with the numbers 5, 15, 25:

1) sides 3533, 3142, and 2940 (Panin's metric);
2) sides 1215, 1179, and 704 (van der Loo's 1st metric);
3) sides 4379, 4036, and 4029 (van der Loo's 2nd metric);
4) sides 2943, 2674, and 2492 (Pages' 1st metric);
5) sides 3444, 3046, and 2838 (Pages' 2nd metric)

(here, the numbers correspond to numbers of metrics of [4, 6]). Let us consider these 5 triangles and also 3 other ones, see Fig. 2:

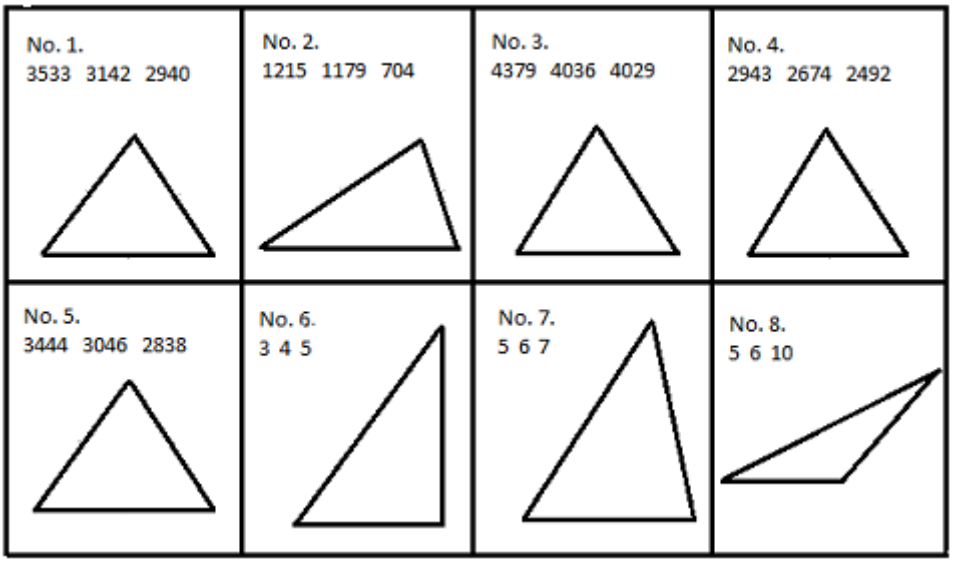

Figure 2. Examples of triangles of metrics (No.No. 1–5) and 3 other ones.

Let us note once again, that we only need the relative lengths of the sides of the triangle.

The further calculations yield the following auxiliary values and values of badnesses, see Table 1.

Table 1. The values of badnesses and the auxiliary calculations.

No.	a	b	c	$\cos(\alpha)$	$\cos(\beta)$	$\cos(\gamma)$	$\alpha, ^o$	$\beta, ^o$	$\gamma, ^o$	bad_1	bad_2	bad_3
1	3533	3142	2940	0.33	0.54	0.62	70.1	57.2	51.8	0.076	0.194	0.111
2	1215	1179	704	0.25	0.34	0.83	75.4	70.2	34.2	0.029	0.068	0.027
3	4379	4036	4029	0.41	0.541	0.544	65.8	57.19	57.05	0.048	0.130	0.078
4	2943	2674	2492	0.35	0.53	0.61	69.4	58.1	52.5	0.062	0.160	0.091
5	3444	3046	2838	0.32	0.54	0.62	71.6	57.0	51.4	0.081	0.203	0.116
6	5	4	3	0	0.6	0.8	90	53.1	36.9	–	–	–
7	7	6	5	0.2	0.54	0.71	78.4	57.1	44.4	0.119	0.272	0.143
8	10	6	5	−0.65	0.89	0.93	130.5	27.1	22.3	–	–	–

We can see, that three bad orderings for all five triangles give the same sequence of metrics. Once again, we mention (above, this thing has already been said, but in connection with other facts) that this ordering differs significantly from the ordering of the badness considered for complete matrices, see the results of the calculations below and, in more detail, in [4]. However, in all our calculations (in any case, with the exception of less than 1% of all triangles, i.e., including ordering for complete matrices), such ordering turns out to be the same for all three norms (badnesses).

Another option for investigating the comparative characteristics of norms is the following one (we will continue to consider 30 species, and the matrix of distance between genomes, given in [6]). We choose two metrics and for any one fixed norm we arrange 4060 triangles in order of increasing values of this norm. At the same time, when reading that both these norms are good (that is, they give acceptable results), we should ideally obtain *identical sequences* of triangles. Actually, one of these two sequences of triangles is obtained from the other by some sequence transpositions of neighboring elements. Since, as we have noted, the number of triangles in our case is 4060, then the maximum possible number of transpositions of neighboring elements is equal to $\frac{4060 \cdot 4059}{2} = 8\,239\,770$. Let us give concrete results of work (Table 2) for the first norm (value "bad_1") only.

Table 2. The joint study of pairs of metrics for the selected first norm (bad1).

Pairs	$\{1,2\}$	$\{1,3\}$	$\{1,4\}$	$\{1,5\}$	$\{2,3\}$	$\{2,4\}$	$\{2,5\}$	$\{3,4\}$	$\{3,5\}$	$\{4,5\}$
Transpositions	175199	163214	169720	175267	154301	159737	160614	180413	179384	181700
Percentages	2.13	1.98	2.06	2.13	1.87	1.94	1.95	2.19	2.18	2.21
Correlation	0.957	0.960	0.959	0.957	0.963	0.961	0.961	0.956	0.956	0.956

Let us give some comments to this table. The pair of metrics is selected in the first line. The number of transpositions of neighboring elements (for obtaining the monotone sequence) is given in the second line. The percent of the maximum possible number of transpositions of neighboring elements (8 239 770) is given in the third line. We do *not* use Spearman's rank correlation coefficient (and some others correlation coefficients used in similar problems); instead of them, we use the linear function having value 1 for 0 transpositions and value -1 for 8 239 770 transpositions.

Still, we note that the two other norms give somewhat worse results; but for them, the number of transpositions does not exceed 600 000 (i.e., less than 7.5%), and, therefore, calculated by our method rank correlation coefficient is more than 0.85.

Thus, all three norms almost always give similar results. Therefore we often use the singular for the word "norm": "some different versions of so-called triangular norm" etc. All this suggests that the task of improving the definition (algorithms for calculating) the norm is less important than improving the definition (calculation algorithms) of metrics, first of all our original Panin's metric. Let us consider these things in the next section.

4. The improvements of Panin's metric and some approaches to improvements of other ones

The object of each of the problems listed in previous section is programming anytime-algorithms; as we already said, Panin's metric algorithmized by the same approach. The methods for solving these problems are constructed on the basis of special combination of some different heuristics. For the implementation of Panin's metric, we use:

- some modifications of the truncated branch-and-bound method [17];
- and for the selecting immediate step, we apply dynamic risk functions [18];
- simultaneously for the selection of coefficients of the averaging-out, we use special version of genetic algorithm [13];
- and the reductive self-learning by the same genetic methods is used for the start of truncated branch-and-bound method [17].

(Thus, a peculiar "loop" of application of heuristics is obtained.) This combination of heuristics represents a special approach to construction of anytime-algorithms for the discrete optimization

problems, which can be considered as an alternative to the methods of linear programming, multi-agent optimization and neural networks.

As can be seen from the results of our work, cited above, when considering close species, several problems are obtained; although here the results are also acceptable, that is, according to any of the norms for the considered metric, the Panin's metric took at least the 2nd place among the 5 considered metrics. But all the same for the future work, apparently, it is necessary to significantly change the algorithm for calculating the distances between genomes. Thus, in this paper we mainly describe the results of calculations obtained *for long-range species*.

More precisely, let us briefly summarize the results of the calculations in the following table; apparently, it practically does not require comments. Let us say only, that "Bad triangles" is the number of triangles having violation of the triangle inequality; let us note, that this violation occurs (at least in 2 cases) in each of the metrics under consideration.

Table 3. Brief description of the obtained results.

No. of metric	Bad triangles	bad_1	bad_2	bad_3
1	9	0.039	0.094	0.017
2	185	0.071	0.161	0.030
3	2	0.025	0.064	0.012
4	33	0.037	0.090	0.016
5	207	0.057	0.131	0.023

However, apparently, much more important is the obtained ordering of the quality of metrics. This ordering is calculated for all 4060 triangles, it differs from the one given above for one triangle.

In some distance tables of [6], several (not more than 10) values are marked in red; it means the following. We slightly change the metric values obtained by means of algorithms, trying to reduce the whole badness of the matrix. In this case, of course, the goal function does not coincide with the badness, it contains:

- the badness (the main component);
- the number of elements of the matrix being changed (the value increases strongly with increasing this number);
- and the sum of the absolute values of these changes.

5. Conclusion

Thus, a very small change of the given matrix greatly reduces its badness. The values marked in red are chosen by us manually. However, at the present time we have a neural network that implements such an algorithm; we are going to describe this neural network in the following publication. But the following is much more important: these "red" changes give *input* information to another neural network, the one that computes the constants for the metric. Thus, along with one "loop" of algorithms already described in previous section, we get one more. Table 3 above already takes into account similar improvements in the metric.

As we said before, the received results of our computer programs are designed as an `xlsx`-file and are available for the direct download by [6]. The whole article is actually a comment to this file.

We also note that different "places" of different metrics for different cases (i.e., the case of distant animal species, the case of close animal species and the case of subspecies) talk about the need to continue research in this direction.

6. **Acknowledgements**
The authors of the article express their gratitude to Vladislav Dudnikov (Togliatti State University, Russia) for his help in preparing this paper.

The reported study was partially supported by RFBR according to the research project No. 16-47-630829.

7. References

[1] Melnikov, B. *On a parallel implementation of the multi-heuristic approach in the problem of comparison of* genetic sequences / B. Melnikov, A. Panin // Vektor Nauki of Togliatti State University. — 2012. — Vol. 4(22). — P. 83–86. (in Russian).

[2] Makarkin, S. *A parallel implementation of the multi-heuristic approach in the task of comparing genetic* sequences / S. Makarkin, B. Melnikov, A. Panin // Applied Mathematics. — 2013. — Vol. 4(10A).— P. 35–39. DOI: 10.4236/am.2013.410A1006.

[3] Melnikov, B. *Multiheuristic approach to compare the quality of defined metrics on the set of DNA sequences* / B. Melnikov, S. Pivneva, M. Trifonov // Modern Information Technologies and IT Education. — 2017.— Vol. 13(2). — P. 89–96. (in Russian). DOI: 10.25559/SITITO.2017.2.235

[4] Melnikov, B. *Various algorithms, calculating distances of DNA sequences, and some computational* recommendations for use such algorithms / B. Melnikov, S. Pivneva, M. Trifonov // CEUR Workshop Proceedings. — 2017. — Vol. 1902. — P. 43–50.

[5] *Nucleotide (The Nucleotide database)* [Electronic resource] / Access mode: http://www.ncbi.nlm.nih.gov/nuccore, free. – Title from the screen.

[6] Melnikov, B. *The processed results of the computer calculations* [Electronic resource] / Access mode: http://bormel.ru/BorMel-DNA.xlsx, free. – The link for direct download, xlsx-format.

[7] Levenshtein, V. *Binary codes capable of correcting deletions, insertions, and reversals* / V. Levenshtein // Soviet Physics Doklady. — 1966. — Vol. 10 (8). — P. 707–710.

[8] Winkler, W. *String comparator metrics and enhanced decision rules in the Fellegi-Sunter model of record linkage* / W. Winkler // Proceedings of the survey research methods sections, American Statistical Association. — 1990. — Vol. 4 (22). — P. 354–359.

[9] Pages, H. *Biostrings: String objects representing biological sequences, and matching algorithms. R package* version 2.44.2 [Electronic resource] / H. Pages, P. Aboyoun, R. Gentleman, S. DebRaoy. Access mode: https://rdrr.io/bioc/Biostrings/, free. – Title from the screen.

[10] Morgan, M. ShortRead: a bioconductor package for input, quality assessment and exploration of high-throughput sequence data [Electronic resource] / M. Morgan, S. Anders, M. Lawrence, et al. Access mode: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2752612/, free. – Title from the screen.

[11] Van der Loo, M. *The Stringdist Package for Approximate String Matching* / M. van der Loo // The R. Journal. — 2014. — Vol. 6. — P. 111–122.

[12] Althaus, E. *A branch-and-cut algorithm for multiple sequence alignment* / E. Althaus, A. Caprara, H.-P. Lenhof, K. Reinert // Mathematical Programming. — 2006. — Vol. 105. — P. 387–425.

[13] Melnikov, B. *Multiheuristic approach to discrete optimization problems* / B. Melnikov // Cybernetics and Systems Analysis. — 2006. — Vol. 42(3). — P. 335–341.

[14] Makarkin, S. *Geometrical methods for solving the pseudo-geometric version of the traveling salesman problem* / S. Makarkin, B. Melnikov // Stochastic optimization in informatics. — 2013. — Vol. 9(2). — P. 54–72. (in Russian).

[15] Melnikov, B. *Once more about the state-minimization of the nondeterministic finite automata* / B. Melnikov // Journal of Applied Mathematics and Computing. — 2000. — Vol. 7(3). — P. 655–662.

[16] Melnikov, B. *The state minimization problem for nondeterministic finite automata: the parallel* implementation of the truncated branch and bound method / B. Melnikov, A. Tsyganov // Proceedings. 5th International Symposium on Parallel Architectures, Algorithms and Programming (Taipei). — 2012. — P. 194–201.

[17] Melnikov, B. *Discrete optimization problems – some new heuristic approaches* / B. Melnikov // Proceedings. Eighth International Conference on High-Performance Computing in Asia-Pacific Region (Beijing, Chinese Academy of Sciences and China Computer Federation). — 2005. — P. 73–80.

[18] Melnikov, B. *Heuristics in programming of nondeterministic games* / B. Melnikov // Programming and Computer Software. — 2001. — Vol. 27(5). — P. 277–288.