Int. J. Appl. Math. Comput. Sci., 2009, Vol. 19, No. 4, 675–678
DOI: 10.2478/v10006-009-0054-y

SOME REMARKS ON EVALUATING THE QUALITY OF THE MULTIPLE SEQUENCE ALIGNMENT BASED ON THE BALIBASE BENCHMARK

JACEK BŁAŻEWICZ [*,**], PIOTR FORMANOWICZ [*,**], PAWEŁ WOJCIECHOWSKI [*]

[*] Institute of Computing Science
Poznań University of Technology, Piotrowo 2, 60–965 Poznań, Poland
e-mail: {Jacek.Blazewicz,Piotr.Formanowicz,Pawel.Wojciechowski}@cs.put.poznan.pl

[**] Institute of Bioorganic Chemistry
Polish Academy of Sciences, Noskowskiego 12/14, 61–714, Poznań, Poland

BAliBASE is one of the most widely used benchmarks for multiple sequence alignment programs. The accuracy of alignment methods is measured by *bali_score*—an application provided together with the database. The standard accuracy measures are the Sum of Pairs (SP) and the Total Column (TC). We have found that, for non-core block columns, results calculated by *bali_score* are different from those obtained on the basis of the formal definitions of the measures. We do not claim that one of these measures is better than the other, but they are definitely different. Such a situation can be the source of confusion when alignments obtained using various methods are compared. Therefore, we propose a new nomenclature for the measures of the quality of multiple sequence alignments to distinguish which one was actually calculated. Moreover, we have found that the occurrence of a gap in some column in the first sequence of the reference alignment causes column discarding.

Keywords: multiple sequence alignment, reference alignment, alignment accuracy.

1. Introduction

Multiple sequence alignment methods are widely used in biological research. Many algorithms for the construction of such alignments are known and a lot of their implementations are available in commercial and non-commercial packages. Since different criteria are used in order to construct good alignments, it is important to know what these criteria are and whether they correspond to the biological context of the research. A probably even more important question is how close the alignments obtained as the optimal ones (according to various criteria) are to the "real"alignments resulting from the comparison of 3D structures of the tested proteins.

Looking for the answer to the latter question, we have discovered that the way in which *bali_score* calculates alignment scores for non-core blocks does not agree with the definitions known from the literature. The *bali_score* is part of the widely used BAliBASE database system and the way in which it calculates the alignment measures is the main topic of this work.

2. Inaccurate definitions of measures

BAliBASE database containing reference multiple sequence alignments is available with an application, called *bali_score*, which evaluates the quality of a test alignment in comparison to a reference alignment (cf. `http://www-bio3d-igbmc.u-strasbg.fr/balibase/`). The measures calculated by *bali_score* are SP (Sum of Pairs) and the TC (Total Column), originally defined for the multiple sequence alignment in (Thompson *et al.*, 1999). (They were the same in the previous versions of BAliBASE.) Both measures are also given for the core blocks (Thompson *et al.*, 2005), but their calculation requires an additional file with an annotation of the core blocks columns. Our studies of the quality of several multiple sequence alignments have led us to the following conclusions:

- the SP and TC values given by *bali_score* are different from those exactly following their definitions;
- the occurrence of a gap in some column of the first sequence of the reference alignment leads to discarding this column from calculations.

The SP and TC are defined as follows (Thompson *et al.*, 2005).

SP score. Let us consider a test alignment of *N* sequences consisting of *M* columns. The i-th column in the alignment can be denoted by $A_{i1}, A_{i2}, \ldots, A_{iN}$. For each pair of residues A_{ij} and A_{ik}, let us define p_{ijk} such that $p_{ijk} = 1$ if the residues A_{ij} and A_{ik} are aligned with each other in the reference alignment, and $p_{ijk} = 0$ otherwise. The S_i score for the i-th column is defined as

$$S_i = \sum_{j=1}^{N} \sum_{k=1, j\neq k}^{N} p_{ijk}. \tag{1}$$

The SP score for the alignment is equal to

$$SP = \frac{\sum_{i=1}^{M} S_i}{\sum_{i=1}^{M_r} S_{ri}}, \tag{2}$$

where M_r is the number of columns in the reference alignment and S_{ri} is the score S_i for the i-th column in the reference alignment.

TC score. Let us define a C_i score for the i-th column in the alignment: $C_i = 1$ if all the residues in this column are aligned in the reference alignment, otherwise $C_i = 0$. The TC for the alignment is then given by

$$TC = \sum_{i=1}^{M} \frac{C_i}{M}. \tag{3}$$

Let us denote the SP and TC for the core blocks as $SP_{\text{core\_blocks}}$ and $TC_{\text{core\_blocks}}$ respectively. This annotation should avoid confusion about which measure is used.

Let us consider the following simple example.

Example 1.

Reference alignment		Tested alignment	
s01	ABCDE	s01	ABCDE.
s02	A...B	s02	A....B
s03	AB..C	s03	AB..C.
s04	ABC.D	s04	ABC.D.
s05	ABCDE	s05	ABCDE.
s06	ABCDE	s06	ABCDE.
s07	ABCDE	s07	ABCDE.
s08	ABCDE	s08	ABCDE.
s09	ABCDE	s09	ABCDE.
s10	....A	s10	....A.

♦

According to the theoretical definitions, the correct calculations of the SP and TC measures for Example 1 are as follows.

SP. There are the following pairs in columns in the reference alignment:
1st column: $9 \cdot 8/2 = 36$,
2nd column: $8 \cdot 7/2 = 28$,
3rd column: $7 \cdot 6/2 = 21$,
4th column: $6 \cdot 5/2 = 15$,
5th column: $10 \cdot 9/2 = 45$,
which results in the total number of pairs in the reference alignment, equal to 145. In the tested alignment the first four columns are the same as in the reference alignment and in the fifth column there are $9 \cdot 8/2 = 36$ correct pairs. So, the total number of correct pairs in the tested alignment is equal to 136, and thus $SP = 136/145 = 0.938$.

TC. Since four columns are correct, $TC = 4/5 = 0.8$.

The values of the SP and TC calculated by the *bali_score* program (v. 3.01) differ significantly from their formal definitions and are equal to $SP_{\text{bali\_score}} = 0.889$ and $TC_{\text{bali\_score}} = 0.5$, respectively. The explanation of this fact is as follows.

$TC_{\text{bali\_score}}$. The difference between TC and $TC_{\text{bali\_score}}$ is that in the latter only columns from the reference alignment with fewer than *gaps threshold* (gt) gaps are taken into account, where gt is defined as

$$gt = \left\lfloor \frac{N \times 20}{100} \right\rfloor. \tag{4}$$

The remaining columns are omitted.

$SP_{\text{bali\_score}}$. Similarly, in the case of $SP_{\text{bali\_score}}$ calculations, only those residue pairs of the reference alignment which are in columns with fewer than gt gaps are taken into account. The other pairs are omitted.

In Example 1, $SP_{\text{bali\_score}}$ and $TC_{\text{bali\_score}}$ are calculated as follows. In the case of $SP_{\text{bali\_score}}$, only the first and fifth columns of the reference alignment satisfy the requirement concerning the percentage of gaps in a column. So, the remaining columns are not taken into account. In the first column of the reference alignment there are $9 \cdot 8/2 = 36$ pairs and in the fifth column there are $10 \cdot 9/2 = 45$ pairs, which gives the total number of pairs equal to 81. In the tested alignment, the first column is identical to the reference alignment (36 pairs) and in the fifth column there are $9 \cdot 8/2 = 36$ correct pairs. The total number of correct pairs in the tested alignment is equal to 72. Thus, $SP_{\text{bali\_score}} = 72/81 = 0.889$.

In Example 1, $TC_{\text{bali\_score}}$ is equal to 0.5 because there are two columns taken into account and only the first one is correct.

The information about the way of calculating the SP and TC score by *bali_score* follows from the analysis of the program source code. Unfortunately, we have not found any other information explaining the calculations. The above problem does not concern the core

block columns, because according to the definition of such columns they cannot contain any gaps.

The second problem concerning *bali_score* regards gaps in the first row of the reference alignment. Let us analyze the following example: If the order of sequences in the reference and the tested alignment files is identical to the one presented in Example 1, then $SP_{\text{bali\_score}}$ and $TC_{\text{bali\_score}}$ are correct. But if the order is changed by moving the sequence $s10$ to the first row in the reference alignment (in fact the order for the tested alignment is not important), then the measures from *bali_score* are $SP_{\text{bali\_score}} = 0.889$ and $TC_{\text{bali\_score}} = 0.000$.

Following the analysis of the source code of *bali_score*, we have discovered that columns for which in the first sequence of the reference alignment there is a gap are not taken into account even if they have fewer than gt gaps. So, in the example only the fifth column is taken into account and since this column is different from the corresponding one in the tested alignment, the $TC_{\text{bali\_score}}$ score given by *bali_score* is equal to 0. Such cases are quite easy to find by comparing the reference alignment with itself in the verbose mode ("-v" option of *bali_score*). Example 2 shows the columns 132 to 144 of the BB11002 reference alignment. The last row shows whether a column is taken into account for SP and TC calculation (marked as "1") or not (marked as "."). In the case of BB11002, the columns 132–134 are not taken into account because of a gap in the first sequence, despite the fact that there is only one gap in these columns and the total number of sequences is equal to 8.

The gap in the first column will not occur in the case of core blocks because, according to the definition of such blocks, they cannot contain gaps.

Example 2.
Result of the evaluation of the BB11002 reference alignment

```
---NLFVALYDFV
MNKGVIYALWDYE
AEGYQYRALYDYK
MIQNFRVYYRDSR
GFMFKVQAQHDYT
HPISMAVALQDYM
YAGEPYVAIKAYT
SSGEIAQVTSAYV

...1111111111
```

♦

3. Conclusions

We realize that the most important feature of BAliBASE is the reliability of alignments constructed within core block columns, and we would like to warn users against unaware employment of the *bali_score* application for other purposes than core block evaluation. We showed that measures calculated by *bali_score* for non-core blocks can be incorrect even for the reference alignments included in BAliBASE. Thus, the errors made at the stage of quality evaluation of the methods for the multiple sequence alignment can lead to a choice of the method which is not the most reliable one. These errors are then propagated to further steps of biological analysis. For example, as a result of a faulty alignment, a protein can be classified as a member of an improper protein family. An incorrectly created alignment can also lead to building an inappropriate phylogenetic tree, because phylogenetic analysis is based on alignments. What is more, the sequence which is not a promoter can be assigned the promoter function in the case of improper alignment creation.

The goal of this paper was to show differences between the formal definitions of some measures used to evaluate alignments and the way in which the values of these measures are calculated by the *bali_score* program for non-core blocks columns. It seems a good idea to use separate symbols for denoting the values of SP and TC scores calculated by the BAliBASE system and for those directly following from the formal definitions, e.g., the denotations proposed in this paper. It will clarify how the values should be interpreted.

We hope that this work will help the users of BAliBASE avoid potential errors. On the other hand, the authors of BAliBASE could easily improve their tool by taking into account the described differences between the measures.

Acknowledgment

We thank the authors of the BAliBASE database for making the source codes of the *bali_score* application accessible. This allowed us to discuss doubts about the measures considered.

This research has been partially supported by the Polish Ministry of Science and Higher Education under Grant No. N N519 314635.

References

Thompson, J. D., Koehl, P., Ripp, R. and Poch, O. (2005). Balibase 3.0: Latest developments of the multiple sequence alignment benchmark, *PROTEINS: Structure, Function, and Bioinformatics* **61**(1): 127–136.

Thompson, J. D., Plewniak, F. and Poch, O. (1999). A comprehensive comparison of multiple sequence alignment programs, *Nucleic Acids Research* **27**(13): 2682–2690.

Jacek Błażewicz is a professor of computer science at the Poznań University of Technology. His research interests include algorithm design and complexity analysis of algorithms, especially in bioinformatics as

well as in scheduling theory. He has published widely in the above fields (over 320 papers) in many outstanding journals. He is also the author and co-author of 14 monographs. His science citation index exceeds 2300. In 1991 he was awarded the EURO Gold Medal for his scientific achievements in the area of operations research. In 2002 he was elected a corresponding member of the Polish Academy of Sciences. In 2006 he was received an honorary doctorate from the University of Siegen.

Piotr Formanowicz was born in Poznań, Poland, in 1969. He received the M.Sc. degree from the Poznań University of Technology in 1994, the Ph.D. degree in 2000 and the habilitation qualification in 2005 from the same university. He is an assistant professor at the Institute of Computing Science, Poznań University of Technology, and at the Institute of Bioorganic Chemistry, Polish Academy of Sciences. His research interests concern computational biology, scheduling theory, and computational complexity theory.

Paweł Wojciechowski was born in 1977 in Poznań. He received the M.Sc. degree in computer science from the Poznań University of Technology in 2002. He is a Ph.D. student at the Institute of Computing Science at the same university. His research interests include computational biology, especially multiple sequence alignment problems.

Received: 5 January 2009
Revised: 28 May 2009