

An Ant-Based Model for Multiple Sequence Alignment

Frédéric Guinand, Yoann Pigné

▶ **To cite this version:**

Frédéric Guinand, Yoann Pigné. An Ant-Based Model for Multiple Sequence Alignment. Large-Scale Scientific Computing, 6th International Conference, LSSC 2007, Jun 2007, Sozopol, Bulgaria. pp.553-560, 10.1007/978-3-540-78827-0_63 . hal-00264040

HAL Id: hal-00264040
https://hal.science/hal-00264040

Submitted on 13 Mar 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

hal-00264040, version 1 - 13 Mar 2008

An Ant-Based Model for Multiple Sequence Alignment

Frédéric Guinand and Yoann Pigné*

LITIS laboratory, Le Havre University, France
www.litislab.eu

Abstract. Multiple sequence alignment is a key process in today's biology, and finding a relevant alignment of several sequences is much more challenging than just optimizing some improbable evaluation functions. Our approach for addressing multiple sequence alignment focuses on the building of structures in a new graph model: the factor graph model. This model relies on block-based formulation of the original problem, formulation that seems to be one of the most suitable ways for capturing evolutionary aspects of alignment. The structures are implicitly built by a colony of ants laying down pheromones in the factor graphs, according to relations between blocks belonging to the different sequences.

1 Introduction

For years, manipulation and study of biological sequences have been added to the set of common tasks performed by biologists in their daily activities. Among the numerous analysis methods, multiple sequence alignment (MSA) is probably one of the most used. Biological sequences come from actual living beings, and the role of MSA consists in exhibiting the similarities and differences between them. Considering sets of homologous sequences, differences may be used to assess the evolutionary distance between species in the context of phylogeny. The results of this analysis may also be used to determine conservation of protein domains or structures. While most of the time the process is performed for aligning a limited number of thousands bp-long sequences, it can also be used at the genome level allowing biologists to discover new features that could not be exhibited at a lower level of study [5]. In all cases, one of the major difficulties is the determination of a biologically relevant alignment, performed without relying explicitly on evolutionary information like a phylogenetic tree.

Among existing approaches for determining such relevant alignments, one of them rests on the notion of block. A block is a set of factors present in several sequences. Each factor belonging to one block is an almost identical substring. It may correspond to a highly conserved zone from an evolutionary point of view. Starting from the set of factors for each sequence, the problem we address is the building of blocks. It consists in choosing and gathering almost identical factors

* Authors are alphabetically sorted. The work of Y. Pigné is partially supported by French Ministry of Research and Higher Education.

common to several sequences in the most appropriate way, given that one block cannot contain more than one factor per sequence, that each factor can belong to only one block and that two blocks cannot cross each other. For building such blocks, we propose an approach based on ant colonies. This problem is very close to some classical optimization issues except that the process does not use any evaluation function since it seems unlikely to find a biologically relevant one. As such, it also differs notably from other works in the domain setting up ant colonies for computing alignments for a set of biological sequences [6,2].

The following section details the proposed graph model. Sect. 3 goes deeper into the ant algorithm details. Finally, Sect. 4 studies the behavior of the algorithm with examples.

2 Model

There exist many different families of algorithms for determining multiple sequence alignments, dynamic programming, progressive or iterative methods, motif-based approaches... However, if the number of methods is important, the number of models on which these methods operate is much more limited. Indeed, most algorithms use to consider nucleotide sequences either as strings or as graphs. In any case however, the problem is formulated as an optimization problem and an evaluation function is given. Within this paper, we propose another approach based on a graph of factors, where the factors are sub-sequences present in, at least, two sequences. Instead of considering these factors individually, the formulation considers that they interact with each other when they are neighbors in different sequences, such that our *factor graph* may be understood as a factor/pattern interaction network. Considering such a graph, a multiple sequence alignment corresponds to a set of structures representing highly interacting sets of factors. The original goal may be now expressed as the detection of such structures and we propose to perform such a task with the help of artificial ants.

2.1 Graph Model

An alignment is usually displayed sequence by sequence, with the nucleotides or amino acids that compose it. Here the interest is given to the factors that compose each sequence. So each sequence of the alignment is displayed as a list of the factors it is composed of. Fig. 1 illustrates such a representation, where sequences are displayed as series of factors.

There exists a relation between factors (named 1, 2 and 3 in Fig. 1) as soon as there are almost identical. Indeed, two identical factors on different sequences may be aligned. Such an alignment aims at creating blocks. Together with factors, these relations can be represented by a graph $G = (V, E)$. The set V of nodes represents all the factors appearing in the sequences, and edges of E link factors that may be aligned. These graphs are called *factor graphs*. A *factor graph* is a complete graph where edges linking factors attending on the same sequence are removed. Indeed, a given factor f may align with any other

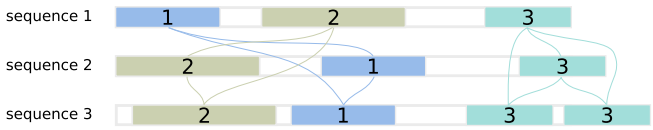


Fig. 1. This is a set of three sequences. Common subsequences of these sequences which are repeated are labeled. After the conversion, each sequence of the alignment is displayed as a list of factors. Here sequence 1 = [1,2,3], sequence 2 = [2,1,3] and sequence 3 = [2,1,3,3]. Thin lines link factors that may be aligned together.

identical factor f' provided f' does not belong to the same sequence. In Fig. 1, thin links between factors illustrate the possible alignments between them.

From a graph point of view the sequential order "sequence by sequence" has no sense. The alignment problem is modeled as a set of *factor graphs*. So as to differentiate the different factors, they are given a unique identifier. Each factor is assigned a triplet $[x, y, z]$ where x is an identifier for the pattern, y is the identifier of the sequence the factor is located on and z is the occurrence of this pattern on the given sequence. For instance, on Fig. 1, the bottom right factor of the third sequence is identified by the triplet $[3, 3, 2]$. Namely, it is the pattern "3", located on the sequence 3 and it occurs for the second time on this sequence, given the sequences are read from left to right.

Using that model the sequences are not ordered as it is the case when considering progressive alignment methods. Fig. 2 illustrates such graphs according to the representation of Fig. 1.

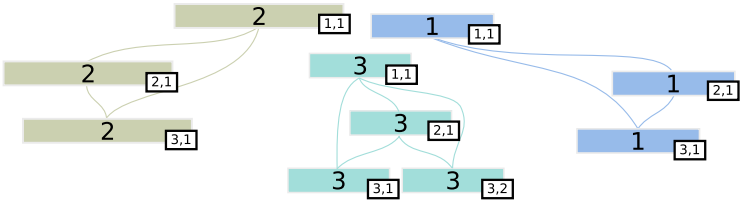


Fig. 2. The alignment seen in Fig. 1 displayed as a set of *factor graphs*.

From each *factor graph* a subset of factors may be selected to create a block. If the block is composed of one factor per sequence, it is a complete block, but if one sequence is missing the block is said partial. Not all block constructions are possible since blocks crossing is not allowed and the selection of one block may prevent the construction of another one. For instance, from Fig. 1 one can observe that a block made of factors "1" may be created. Another block with factor "2" may also be created. However, both blocks cannot be present together in the alignment. These blocks are said *incompatible*. A group of blocks is said to be *compatible* if all couples of blocks are compatible.

Another relation between potential blocks can be observed in their neighborhood. Indeed, a strong relevance has to be accorded to potential blocks that are closed to one another and that do not cross. If two factors are neighbors in many sequences, there is a high probability for these factors to be part of a bigger factor with little differences. Such relations are taken into account within our approach and are called friendly relations. An example of friendly relation in the sample alignment can be observed sequences 2 and 3 between factors "1" and "2".

The *factor graphs* are intended to represent the search space of blocks according to the set of considered factors. However, they do not capture compatibility constraints and friendly relations between blocks. For that purpose, we first consider $G' = (V', E')$ dual graph of $G = (V, E)$, G being the graph composed of the entire set of *factor graphs*. The set V' of G' corresponds to the set E of G. Each couple of adjacent edges $e_1, e_2 \in E$ corresponds to one edge in E'. Moreover, in order to represent compatibility constraints and friendly relations two new kinds of edges have to be added to this graph: namely E_c for compatibility constraints and E_f for friendly relations. We call *relation graph* (Fig. 3) the graph $G_r = (V', E' \cup E_f \cup E_c)$.

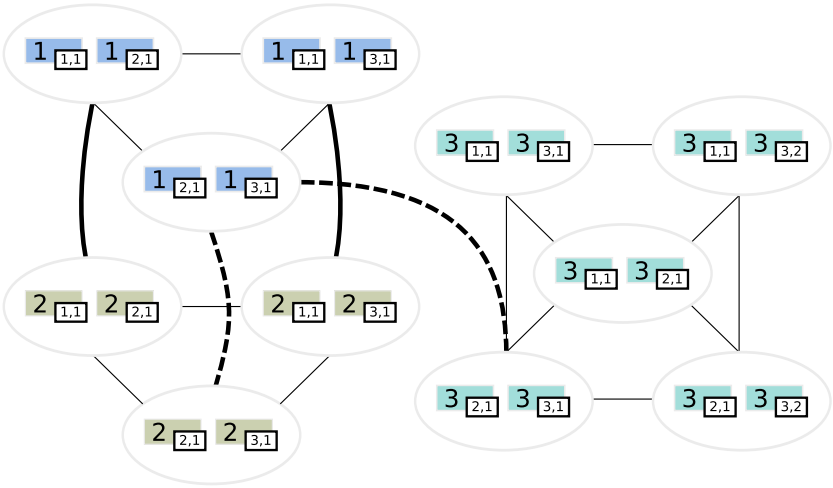


Fig. 3. The relation graph G_r based on the dual graph of G with additional edges corresponding to compatibility constraints, represented by thick plain edges, and friendly relations, represented by dashed links.

Our approach makes use of both graphs. Ants move within the factor graphs, but their actions may also produce some effects in remote parts of the factor graphs, according to relation graph topology as explained in Sect. 3.

2.2 Ants for Multiple Sequence Alignment

Ant based approaches have shown their efficiency for single or multiple criteria optimization problems. The central methodology being known as Ant Colony Optimization [3]. Ants in these algorithms usually evolve in a discrete space modeled by a graph. This graph represents the search space of the considered problem. Ants collectively build and maintain a solution. Actually, this construction takes place thanks to pheromone trails laid down the graph. The search is led both by local information in the graph and by the global evaluation of the produced solutions.

The model issued in the previous section proposed a graph model that raises local conflicts and attractions that may exist in the neighborhood of the factors in the alignment. For this, the model may handle the local search needed by classical ACOs. However, providing a global evaluation function for this problem is unlikely. Indeed, defining a relevant evaluation function for MSA is in itself a problem since the evaluation should be aware of the evolutionary history of the underlying species, which is part of the MSA problem. Molecular biologists themselves do not all agree whether or not one given alignment is a good one. Popular evaluation functions like the classical *sum of pairs* [1] are still debated. As a consequence, instead of focusing on such a function, our approach concentrates on building structures in the factor graphs according to the relation graph. These structures correspond to compatible blocks. The building of blocks is made by ants which behavior is directly constrained by the pheromones they laid down in the graph and indirectly by the relation graph since this graph has a crucial impact on pheromone deposit location.

The global process can be further refined by taking into account the size of the selected factors, as well as the number of nucleotides or amino acids located between the factors of two neighbor blocks. These numbers are called *relative distances* between factors in the sequel. Thus, the factor graphs carry local information necessary to the ant system and acts like the environment for ants. Communication via the environment also known as stigmergic communication [4] takes place in that graph; pheromones trails are laid down on the edges according to ants move, but also according to the relation graph. A solution of the original problem is obtained by listing the set of compatible blocks that have been bring to the fore by ants and revealed by pheromones.

3 Algorithm

The proposed ant-based system does not evaluate the produced solutions, in this way it is not an ACO. However, the local search process remains widely inspired from ACOs. The general scheme of the behavior of the ant based system follows these rules:

- Ants perform walks into the factor graphs.
- During these walks each ant lay constant quantities of pheromones down on the edges they cross.

– This deposit entails a change in pheromone quantities of some remote edges according to the relation graph G_r as described in Sect. 2.
– Ants are attracted by the pheromone trails already laid down in the environment.

Finally a solution to the problem is a set of the most pheromone loaded edges of the *factor graph* that are free from conflicts. In the following section pheromone management is more formally detailled.

3.1 Pheromone Trails

Let τ_{ij} be the quantity of pheromone present on edge (i, j) of graph G which links nodes i and j. If $\tau_{ij}(t)$ is the quantity of pheromone present on the edge (i, j) at time t, then $\Delta\tau_{ij}$ is the quantity of pheromone to be added to the total quantity on the edge at the current step (time $t + 1$). So:

$$\tau_{ij}(t + 1) = (1 - \rho).\tau_{ij}(t) + \Delta\tau_{ij} \tag{1}$$

Note that the initial amount of pheromone in the graph is close to zero and that ρ represents the evaporation rate of the pheromones. Indeed, the modeling of the evaporation (like natural pheromones) is useful because it makes it possible to control the importance of the produced effect. In practice, the control of this evaporation makes it possible to limit the risks of premature convergence of the process.

The quantity of pheromone $\Delta\tau_{ij}$ added on the edge (i, j) is the sum of the pheromone deposited by all the ants crossing the edge (i, j) with the new step of time. The volume of pheromone deposited on each passage of an ant is a constant value Q. If m ants use the edge (i, j) during the current step, then:

$$\Delta\tau_{ij} = mQ \tag{2}$$

3.2 Constraints and Feedback Loops

Generally speaking, feedback loops rule self-organized systems. Positive feedback loops increase the system tendencies while negative feedback loops prevent the system from continually increasing or decreasing to critical limits. In this case, pheromone trails play the role of positive feedback loop attracting ants that will deposit pheromones on the same paths getting them more desirable. Friendly relationship found in the G_r graph may also play a positive feedback role. Indeed, more pheromones are laid down around friendly linked blocks. On the other side, conflicts between blocks act as negative feedback loops laying down 'negative' quantities of pheromone.

Let consider an edge (i, j) that has conflict links with some other edges. During the current step, the c ants that cross edges in conflict with edge (i, j) define the amount of negative pheromones to assign to (i, j): $\Delta\tau_{ij}^{conflict} = cQ$.

Besides, an edge (i, j) with some friendly relations will be assigned positive pheromones according to the f ants that cross edges in friendly relation with (i, j) during the current step : $\Delta\tau_{ij}^{friendly} = fQ$.

Finally, the overall quantity of pheromone on one given edge (i, j) for the current step defined in equation 2 is modified as follow:

$$\Delta\tau_{ij} = \Delta\tau_{ij} + \Delta\tau_{i,j}^{friendly} - \Delta\tau_{i,j}^{conflict} \tag{3}$$

3.3 Transition Rule

When an ant is on vertex i, the choice of the next vertex to be visited must be carried out according to a defined probability rule.

According to the method classically proposed in ant algorithms, the choice of a next vertex to visit is influenced by 2 terms. The first is a local heuristic based on local information, namely the *relative distance* between the factors (d). The second term is representative of the stigmergic behavior of the system. It is the quantity of pheromone deposited (τ).

Remark 1. Interaction between positive and negative pheromones can lead on some edges to an overall negative value of pheromone. Thus, pheromones quantities need normalization before the random draw is made. Let max be an upper bound value computed from the largest quantity of pheromones on the neighborhood of the current vertex. The quantity of pheromone τ_{ij} between edge i and j is normalized as $max - \tau_{ij}$.

The function $N(i)$ returns the list of vertices adjacent to i (its neighbors). The next vertex will be chosen in this list. The probability for an ant being on vertex i, to go on j (j belonging to $(N(i))$ is:

$$P(ij) = \frac{[\frac{1}{max-\tau_{ij}}^{\alpha} \cdot \frac{1}{d_{ij}}^{\beta}]}{\sum_{s \in N(i)} [\frac{1}{max-\tau_{is}}^{\alpha} \cdot \frac{1}{d_{is}}^{\beta}]} \tag{4}$$

The parameters α and β make it possible to balance the impact of pheromone trails relatively to the *relative distances*.

4 Analysis

First experiments on real sequences have been performed. Fig. 4 compares some results obtained by the proposed structural approach and the alignment provided by ClustalW. This sample shows that the visible aligned blocks can be regained in the results given by ClustalW. However, results are still to be evaluated. Indeed, it may happen that some blocks are found by our method while they are not detected by ClustalW. Anyway, we are aware of the necessity of additional analyses and discussions with biologists in order to validate this approach.

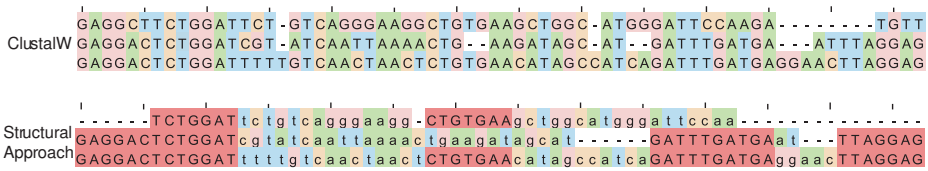


Fig. 4. Alignment of 3 sequences of TP53 regulated inhibitor of apoptosis 1 for Homo sapiens, Bos taurus and Mus musculus. Comparison of the alignment given by ClustalW and our structural approach. The red uppercase nucleotides on the structural approach are the blocks.

5 Conclusion

In this paper was proposed a different approach, for the problem of multiple sequence alignment. The key idea was to consider a problem of building and maintaining a structure in a set of biological sequences instead of considering an optimization problem. Preliminary results show that the structures built by our ant-based algorithm can be informally compared, on a pattern basis, with the results given by ClustalW.

The outlook for the project is now to prove the efficiency and the relevance of the method, in particular, an important chunk of future work will concern the comparison of the differences between conserved regions provided by ClustalW and other well-known multiple sequence alignment methods and our approach. The second perspective focuses on the performance of the method. Indeed, the way blocks are built and intermediate results allow us to consider a kind of divide-and-conquer parallel version of this tool. Most recent results and advances will be made available on `www.litislab.eu`.

References

1. S. F. Altschul. Gap costs for multiple sequence alignment. *Journal of Theoretical Biology*, 138:297–309, 1989.
2. Y. Chen, Y. Pan, J. Chen, W. Liu, and L. Chen. Partitioned optimization algorithms for multiple sequence alignment. In *Proceedings of the 20th International Conference on Advanced Information Networking and Applications - Volume 2 (AINA'06)*, volume 2, pages 618–622. IEEE Computer Society, 2006.
3. M. Dorigo and G. Di Caro. *New Ideas in Optimization. D. Corne and M. Dorigo and F. Glover eds*, chapter The Ant Colony Optimization Meta-Heuristic, pages 11–32. McGraw-Hill, 1997.
4. P.-P. Grassé. La reconstruction du nid et les coordinations inter-individuelles chez belicositermes natalensis et cubitermes s.p. la théorie de la stigmergie : essai d'interprétation du comportement des termites constructeurs. *Insectes sociaux*, 6:41–80, 1959.
5. S. Kurtz, A. Phillippy, A. L. Delcher, M. Smoot, M. Shumway, C. Antonescu, and S. L. Salzberg. Versatile and open software for comparing large genomes. *Genome Biology*, 5(2):R12, 2004.

6. J. D. Moss and C. G. Johnson. An ant colony algorithm for multiple sequence alignment in bioinformatics. In David W. Pearson, Nigel C. Steele, and Rudolf F. Albrecht, editors, *Artificial Neural Networks and Genetic Algorithms*, pages 182–186. Springer, 2003.