*Sequence analysis*

# UnaCloud MSA: a web application to analyze multiple sequence alignments exploiting an opportunistic grid infrastructure

Arthur A. Oviedo[1,*], Harold E. Castro[1] and Diego M. Riaño-Pachón[2]

[1]Department of Systems and Computer Engineering, Universidad de Los Andes, Bogotá, Colombia.

[2]Department of Biological Sciences, Universidad de Los Andes, Bogotá, Colombia.

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

**ABSTRACT**

**Summary:** We present UnaCloud MSA, a web application for generating and analyzing multiple sequence alignments running on an opportunistic infrastructure. Because of the difficulties that arise in the generation of a good multiple sequence alignment, UnaCloud MSA incorporates different existing and novel applications in an automated workflow in order to provide the user with more useful information to select the final MSA.

**Availability:** UnaCloud MSA is freely available at http://unacloudmsa.uniandes.edu.co

**Contact:** unacloudmsa@uniandes.edu.co

## 1 INTRODUCTION

One of the most important tasks in sequence analysis is the generation of a Multiple Sequence Alignment (MSA). However, generating the best alignment is an intractable problem and therefore different programs that employ a variety of heuristic approaches are available.

Despite of the variety of programs available to generate a MSA, there is not an easy way to compare alignments to determine which is the most appropriate, or the best, given a set of input sequences provided by a user.

Generating a good alignment not only depends on the program that is used, but also depends on the sequences that are introduced. If a subset of the input sequences is very similar, the alignment will assign more weight to this group and this can affect the alignment of the rest of sequences (Chica *et al*., 2008).

When generating a MSA, the user has to deal with all of these problems. Even though there are different online tools readily available to analyze and generate a MSA, they are usually no integrated and the user must go from website to website collecting result files. Moreover, as the computational cost to run these tools scales rapidly with the number of sequences, the application of different solutions to the same problem might be prohibitive.

The main objective of UnaCloud MSA is to provide an integrative option to the user, generating and analyzing MSAs, through an automatic executable workflow and by taking advantage of the available resources in a distributed infrastructure available at our university and elsewhere. UnaCloud MSA provides a simple to use web interface allowing researchers to focus on their research rather than on the underlying technologies.

*To whom correspondence should be addressed.

## 2 DESIGN AND IMPLEMENTATION

### 2.1 Workflow MSA

In order to provide a better understanding of the input dataset and the alignments that are generated, an automatic workflow was implemented using the workflow management tool Loni Pipeline (Dinov *et al.*, 2009). The different jobs that are performed in the workflow can be grouped according their functionality:

- Generation:

Depending on the number of the sequences in the input file, the workflow runs different programs for the creation of the MSA. If the number of sequences is 100 or smaller the programs that are executed are: ClustalW (Thompson *et al.*, 2007), MAFFT (Katoh *et al.*, 2009), MUSCLE (Edgar, 2004), T-Coffee (Notredame *et al.*, 2000) and ProbCons (Do *et al.*, 2005). If the number of sequences is bigger, T-Coffe and Probcons are not executed, and are replaced with the KAlign (Lassman *et al.*,2009 ) application.

These programs were selected because each one employs a different strategy (Progressive, Progressive using a Fast Fourier Transformation, Iterative Improvement, Consistency and Probability Distributions) in order to obtain a MSA. The main reason for this separation is performance. T-Coffee and ProbCons are applications that have a very high computational complexity and take long time for a relatively small number of sequences.

- Distance Analysis:

The Workflow MSA also performs a distance analysis of the input sequences. It starts by running the EMBOSS program `dist` which generates a distance matrix of the sequences. Since the number of entries in the matrix is $O(n^2)$, where n is the number of sequences, and becomes difficult to analyze as the number of sequences increases, we implemented the application `matrix_plotter`. `matrix_plotter` generates an image of the matrix with a color scale, where values closest to 0 appear in red and values closest to 100 appear in green. With this image it becomes easier to identify groups of sequences that are highly similar

A second analysis is based on a graph representation of the distance matrix. The matrix data are parsed into a graph format file, which is then rendered by the program GraphViz (Gansner *et al*, 2010) where nodes represent the original sequence and the lengths

of the arcs are proportional to their weight. This visualization allows the user to easily identify groups of sequences that are highly similar and which could be eliminated from the alignment.

- Logo Creation:

A sequence logo is a graphical representation of a multiple sequence alignment consisting of stacks of characters. Each stack represents a column of the alignment and the height of the character in a stack represents its frequency in the alignment. For each one of the alignments, a sequence logo is generated using the program Weblogo(Crooks *et al* 2004).

- Consensus

A consensus sequence takes the most frequent symbol in each column of the alignment. It summarizes the complete alignment in a single sequence. For each one of the generated alignment, the workflow executes the EMBOSS program `cons`.

- Alignment Comparison:

Since the programs that generate alignments employ a variety of strategies, different areas may be aligned in different ways. In order to compare the results of these alignments, the workflow takes the result of each consensus and aligns these sequences using the MAFFT program. After obtaining this alignment, the workflow runs the implemented program `alignment_comparator`. This program takes each column of this alignment and identifies the most frequent symbol and its number of occurrences. If the number of occurrences is 4 or 5, it indicates that most of the programs agreed in that character. If the number of occurrences is 2 or 1, this indicates that this part of the alignment is ambiguous and that the user should treat that region carefully in downstream analyses. In order to help with the visualization of this result, `alignment_comparator` plots an image with the entire consensus and paint each character with a color: Green characters are the ones that have a high score (4 or 5 number of occurrences), yellow ones have a score of 3 and red ones have a score of 1 or 2.

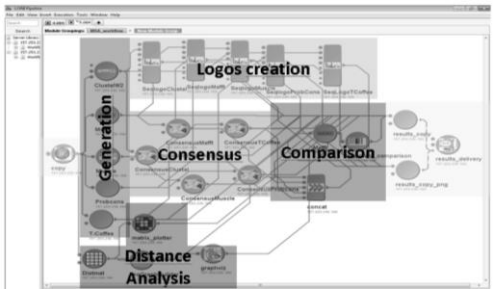


**Fig. 1.** Group of tasks in the implementation of the Workflow MSA using Loni Pipeline.

## 2.2 Web Interface:

UnaCloud MSA is accessible through a web browser. The interaction is simple: The user uploads a FASTA formatted file with the sequences he/she wants to align and analyze, introduces his/her email address and clicks the submit button. As soon as the job is completed, an email is sent to the user indicating the url where the results are. The result webpage presents the following items: Each one of the performed alignments can be visualized with Jalview (Waterhouse *et al.*, 2009), as well as the sequence logos, the distance matrix, the distance based graph, the consensus for each one of the alignments and the comparison among them. The results page also offers the possibility to download the results to the local desktop.

## 2.3 Infrastructure:

UnaCloud MSA employs the UnaCloud (Castro *et al.*, 2010, Rosales et al., 2011) project which takes advantage of idle computational resources. For UnaCloud MSA a Customizable Virtual Cluster (CVC) was deployed in the computer laboratories in the University of Los Andes. This cluster employs the SGE scheduling software in order to manage the jobs that are received from the web interface and the JGDI library to coordinate with the Loni Pipeline server. In order to provide a basic level of quality of service, UnaCloud MSA was also deployed in a VMWare ESX dedicated server. In this way, UnaCloud MSA takes advantage of both kinds of infrastructure: dedicated servers and opportunistic virtual machines using the idle resources available in desktop computers. To be able to coordinate the dedicated and opportunistic infrastructure, we developed `Opportunistic Deployer`, a middleware taking care of monitoring the load of the CVC, computing the number of new virtual machines that should be turned on and communicating this information to UnaCloud. These deployed virtual machines run on the background of the host computers and with low priority. With this scheme, UnaCloud MSA can achieve a high level of performance while processing a high number of concurrent jobs using the existing idling computing infrastructure.

The use of such an infrastructure is a key concept in UnaCloud MSA. The high computing power demanded by each of the integrated tools, would make of this type of workflow somewhat prohibitive for researchers with no access to a large computing facility. And even so, the cost of running different tools for "just" similar results may be too high. Using idle capacity makes this project viable and of interest of researchers independently of the kind of resources they have access to.

## REFERENCES

Castro, H. *et al.* (2010) UnaGrid: On Demand Opportunistic Desktop Grid. Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, pages 661-666

Chica, C. *et al.* (2008) A tree-based conservation scoring method for short linear motifs in multiple alignments of protein sequences. *BMC Bioinformatics* **9**, 229.

Crooks, G. *et al* (2004) WebLogo: A sequence logo generator. *Genome Research*, **14,** 1188-1190.

Dinov I.D., *et al.* (2009) Efficient, Distributed and Interactive Neuroimaging Data Analysis using the LONI Pipeline. *Frontiers in Neuroinformatics*, **3**, 22.

Do, C.B. *et al*. (2005) Probabilistic Consistency-based Multiple Sequence Alignment. *Genome Research*, **15,** 330-340

Edgar, R.C. (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.*, **32**, 1792-1797.

Gansner, E. *et al*. (2000) An open graph visualization system and its applications to software engineering. Software – Practice and Experience, 30, 1203-1233.

Katoh, K. *et al.* (2009) Multiple Alignment of DNA Sequences with MAFFT. Methods in Molecular Biology, 537, 39-64

Lassman, T. *et al.* (2009) Kalign: high-performance multiple alignment of protein and nucleotide sequences allowing external features. *Nucleic Acids Res*, **37,** 858-865

Notredame, C. *et al.* (2000) T-Coffee: a novel method for fast and accurate multiple sequence alignment. *J. Mol. Biol.*, **302,** 205-217

Rosales E., *et al.* UnaCloud: Opportunistic Cloud Computing Infrastructure as a Service. The Second International Conference on Cloud Computing, GRIDs, and Virtualization (CLOUD COMPUTING 2011), Roma, Italia, Septiembre 2011.

Thompson, J.D. *et al.* (2007) Clustal W and Clustal X version 2.0. *Bioinformatics*, **23,** 2947-2948

Waterhouse, A.M., *et al.* (2009) Jalview Version 2 - a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, **25,** 1189-1191.