JOBIM 2021

06 > 09 JUIL

INSTITUT PASTEUR | PARIS

Proceedings

- Keynotes
- Conférences
- Symposiums
- Sponsors

HTTPS://JOBIM2021.SCIENCESCONF.ORG | @JOBIM_2021

ORGANISÉ PAR

sfbi société française de bioinformatique

INSTITUT PASTEUR

cea

GENOSCOPE

PARTENAIRES

FRANCE GENOMIQUE

FRANCE MÉDECINE GÉNOMIQUE 2025 aviesan

GDR GÉNOMIQUE ENVIRONNEMENTALE

cnrs GDR Groupement de recherche BIM Bioinformatique moléculaire

GENOPOLE

Google Research

ifb INSTITUT FRANÇAIS DE BIOINFORMATIQUE

illumina

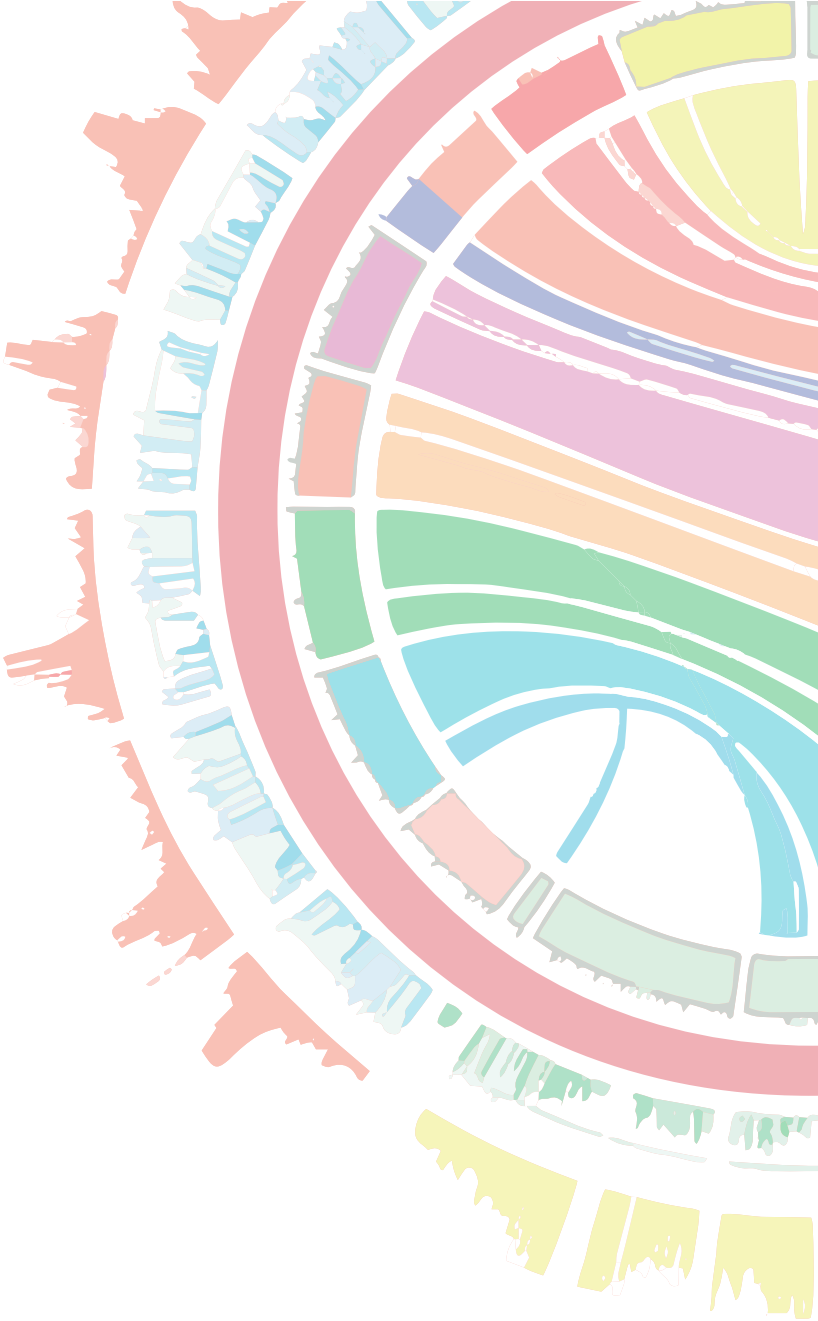
INCEPTION

INRAE

INVESTIS POUR L'AVENIR

MUSÉUM NATIONAL D'HISTOIRE NATURELLE

PSL UNIVERSITÉ PARIS | life



Keynotes

MultiMAP: Dimensionality Reduction of Multiple Datasets by Manifold Approximation and Projection

Sarah TEICHMANN

Wellcome Sanger Institute, Wellcome Genome Campus, Hinxton, Cambridge, UK
St9@sanger.ac.uk

Multi-modal data sets are growing rapidly in single cell genomics, as well as other fields in science and engineering. We introduce MultiMAP, an approach for dimensionality reduction and integration of multiple datasets. MultiMAP embeds multiple datasets into a shared space so as to preserve both the manifold structure of each dataset independently, in addition to the manifold structure in shared feature spaces. MultiMAP is based on the rich mathematical foundation of UMAP, generalizing it to the setting of more than one data manifold. MultiMAP can be used for visualization of multiple datasets as well as an integration approach that enables subsequent joint analyses. Compared to other integration for single cell data, MultiMAP is not restricted to a linear transformation, is extremely fast, and is able to leverage features that may not be present in all datasets. We apply MultiMAP to the integration of a variety of single-cell transcriptomics, chromatin accessibility, methylation, and spatial data, and show that it outperforms current approaches in run time, label transfer, and label consistency. On a newly generated single cell ATAC-seq and RNA-seq dataset of the human thymus, we use MultiMAP to integrate cells across pseudotime. This enables the study of chromatin accessibility and TF binding over the course of T cell differentiation.

Deep learning for biological sequences

Jean-Philippe VERT

Google Research, Brain team, 75009 Paris, France
jpvert@google.com

Abstract

In recent years, deep learning has revolutionized natural language processing (NLP), and is increasingly used to analyze biological sequences including DNA, RNA and proteins. While many deep learning architectures and techniques successful in NLP can be directly applied to biological sequences, there are also specificities in biological sequences that should be taken into account to adapt NLP techniques to that context. In this talk I will discuss several such specificities, including the fact that 1) biological sequences have no natural separation as a sequence of words, 2) a double-stranded DNA sequence can be represented by two reverse-complement sequences, and 3) a natural way to compare homologous biological sequences is to align them. In each case, I will show how the biological constraints can lead to specific models, and illustrate empirically the benefits of incorporating such prior knowledge on several tasks such as metagenomics read binning, protein-DNA binding prediction, or protein annotation.

References

1. V. Mallet and J.-P. Vert. Reverse-complement equivariant networks for DNA sequences. *Technical report bioRxiv* 446953, 2021.
2. R. Menegaux and J.-P. Vert. Embedding the de Bruijn graph, and applications to metagenomics. *Technical report bioRxiv* 980979, 2020.
3. R. Menegaux and J.-P. Vert. Continuous embeddings of DNA sequencing reads, and applications to metagenomics. *Journal of Computational Biology*, 26(6):509-518, 2019.

Wolbachia population genomes from individual *Culex pipiens* ovaries reveal a plasmid-like extrachromosomal circular element

Julie REVEILLAUD

MIVEGEC, Institut de Recherche pour le Développement (IRD), 911, Avenue Agropolis, BP 64501, 34394 Montpellier Cedex 5, France, reveillaud.j@gmail.com

The burden of mosquito-transmitted diseases such as malaria, dengue, West Nile virus, or Zika fever continue to increase globally, representing one of the most significant public health threats. The widespread intracellular bacterium *Wolbachia*, which can block the transmission of pathogens and manipulate the mosquito reproduction, represent one of the most promising tools to control the spread of diseases. However, our understanding of *Wolbachia*'s mobilome beyond its bacteriophages remains incomplete. We studied four wild *Culex pipiens* individuals captured in Southern France from a single collect, and generated an average 70 million Illumina paired-end sequences from the ovaries of each individual through shotgun metagenomics. Using state-of-the-art assembly and binning strategies, we were able to reconstruct near-complete *Wolbachia* genomes from each individual, along with their phage WO variants. While our pangenomic analysis suggested high level of genomic conservation across the bacterial part of *Wolbachia* chromosomes, there was notable variation between individual mosquitoes due to differences in prophage WO and other viral genes. In addition, we identified a putative plasmid that we named pWCP for plasmid of *Wolbachia* in *Culex pipiens*. We validated its presence using additional PCR, long-read sequencing, and screening of available metagenomes. These data open news windows for further genomic analyses and the potential genetic manipulation of a fastidious, widespread genus of obligate intracellular bacteria that is so far recalcitrant to genetic manipulation.

Computer-aided protein design

Thomas SCHIEX

Université Fédérale de Toulouse, ANITI, INRAE, UR 875, Toulouse, France
Thomas.Schiex@inrae.fr

Proteins are the ubiquitous molecular agents that support all the reigns of life, from viruses and bacteria to plants, animals and humans. After millions of years of evolution, Nature has built a large catalog of proteins that transport molecules, convert chemical energy into mechanical work, catalyze chemical reactions, or defend against foreign or infectious agents (often using proteins themselves). We already massively rely on this catalog for applications in health, green chemistry, food and feed, bio/nanotechnologies and cosmetics for example. But this catalog remains too narrow to fulfill all our needs.

In the line of directed evolution (2018 Chemistry Nobel price), computational protein design aims at providing original proteins with improved (or radically new) capacities, but without the restraints of experimental approaches. Indeed, with 20 natural amino acids, designing even a simple protein of 100 amino acids requires to find a suitable amino acid sequence in a huge space of 20^{100} possible sequences. A space from which only a minute fraction can be explored by experimental assays (usually far less than 10^9).

In this talk, I will present the seminal NP-hard "fixed backbone" computational protein design problem and how it can be solved using various algorithmic approaches [1], including quantum computers relying on adiabatic quantum annealing [2]. I will then present some of the many approaches that try to account for protein flexibility during design [3,4], with their associated limitations. This will be illustrated with associated experimentally tested designs in the health, environment and nanotechnology [5] domains.

Acknowledgements

The author would like to thank those that contributed to the results presented in this talk: Simon de Givry, Sophie Barbe, David Simoncini, George Katsirelos, David Allouche, Juan Cortes, Arnout Voet, Aurélien Olichon, Bruce Donald, Manon Ruffini, Jelena Vucinic, Clément Viricel, Seydou Traoré,… as well as the funding bodies(TWB, INSERM/INRAE transfert, ANR, Occitannie Région,...) and companies that supported us.

References

1. Simoncini, D., Allouche, D., de Givry, S., Delmas, C., Barbe, S., Schiex, T. Guaranteed Discrete Energy Optimization on Large Protein Design Problems. *Journal of Chemical Theory and Computation, 2015.*
2. Mulligan, Vikram Khipple, et al. Designing peptides on a quantum computer. *bioRxiv* (2020): 752485.
3. Vucinic, J., Simoncini, D., Ruffini, M., Barbe, S., & Schiex, T. (2020). Positive multistate protein design. *Bioinformatics*, 36(1), 122-130.
4. Bouchiba, Y., Cortés, J., Schiex, T., & Barbe, S. Molecular flexibility in computational protein design: an algorithmic perspective. *Protein Engineering, Design and Selection*, 34, 2021.
5. Noguchi, H., Addy, C., Simoncini, D., Wouters, S., Mylemans, B., Van Meervelt, L., Schiex, T.,Zhang K.Y.J. and Voet, A. R. (2019). Computational design of symmetrical eight-bladed β-propeller proteins. IUCrJ, 6(1), 46-55.

FAIR Computational Workflows

Carole GOBLE[1]

[1] Department of Computer Science, The University of Manchester,
Oxford Road, Manchester, M13 9PL, UK
Email: carole.goble@manchester.ac.uk

Abstract

Computational workflows capture precise descriptions of the steps and data dependencies needed to carry out computational data pipelines, analysis and simulations in many areas of Science, including the Life Sciences. The use of computational workflows to manage these multi-step computational processes has accelerated in the past few years driven by the need for scalable data processing, the exchange of processing know-how, and the desire for more reproducible (or at least transparent) and quality assured processing methods. The SARS-CoV-2 pandemic has significantly highlighted the value of workflows.

This increased interest in workflows has been matched by the number of workflow management systems available to scientists (Galaxy, Snakemake, Nextflow and 270+ more) and the number of workflow services like registries and monitors. There is also recognition that workflows are first class, publishable Research Objects just as data are. They deserve their own FAIR (Findable, Accessible, Interoperable, Reusable) principles and services that cater for their dual roles as explicit method description and software method execution [1]. To promote long-term usability and uptake by the scientific community, workflows (as well as the tools that integrate them) should become FAIR+R(eproducible), and citable so that author's credit is attributed fairly and accurately.

The work on improving the FAIRness of workflows has already started and a whole ecosystem of tools, guidelines and best practices has been under development to reduce the time needed to adapt, reuse and extend existing scientific workflows. An example is the EOSC-Life Cluster of 13 European Biomedical Research Infrastructures which is developing a FAIR Workflow Collaboratory based on the ELIXIR Research Infrastructure for Life Science Data Tools ecosystem. While there are many tools for addressing different aspects of FAIR workflows, many challenges remain for describing, annotating, and exposing scientific workflows so that they can be found, understood and reused by other scientists.

This keynote will explore the FAIR principles for computational workflows in the Life Science using the EOSC-Life Workflow Collaboratory as an example.

References

[1] Carole Goble, Sarah Cohen-Boulakia, Stian Soiland-Reyes,Daniel Garijo, Yolanda Gil, Michael R. Crusoe, Kristian Peters, and Daniel Schober FAIR Computational Workflows Data Intelligence 2020 2:1-2, 108-121 https://doi.org/10.1162/dint_a_00033.

The epigenetic logic of gene activation

Roderic GUIGÓ

Centre de Regulació Genòmica, C/Dr. Aiguader 88, E-08003, Barcelona
roderic.guigo@crg.cat

A large body of data strongly supports a crucial role for histone modifications in the regulation of gene expression [1,2], and highly predictive models have been developed that infer gene expression from histone modification levels [3.4]. An increasing number of cases, however, are being reported in which changes in gene expression occur without changes in histone modifications [5,6]. To provide a framework where to properly investigate these apparently contradictory observations, here we have generated gene expression profiles and maps of nine histone modifications at twelve time-points along a controlled cellular differentiation process: the induced transdifferentiation of human B-cells into macrophage, a process that occurs with massive transcriptomic changes.

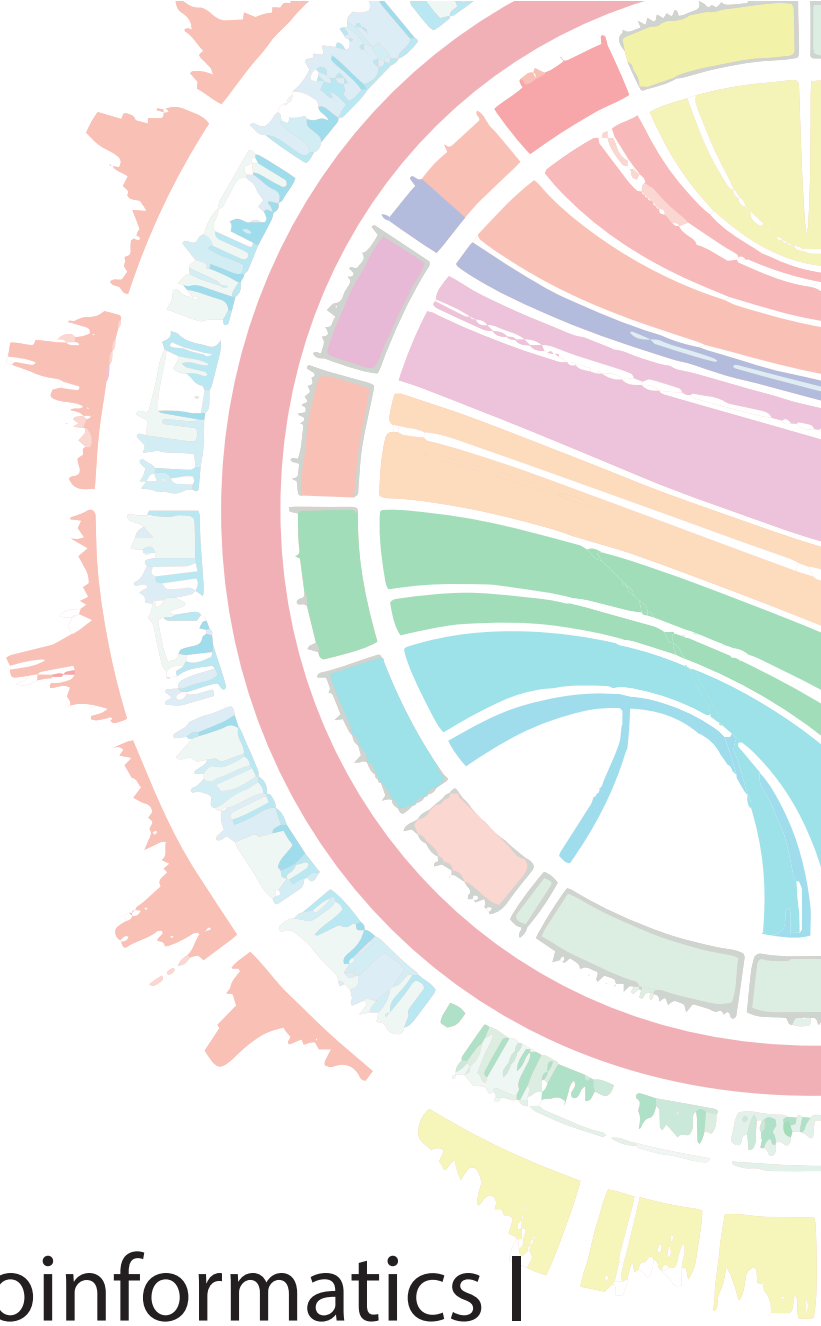
Analysis of these data reveals that the large steady-state associations between gene expression and chromatin marking previously reported are partially artifactual, and mainly arise from the constrained nature of the transcriptome and the epigenome. When measured over time, these correlations are globally weak and, remarkably, in the case of H3K9me3, run in the opposite direction that previously thought.

We found that, in contrast to the histone code hypothesis, only a limited number of combinations of histone modifications are actually marking the genes, defining the major genic chromatin states in the human genome. Genes tend to remain in the same state throughout the entire transdifferentiation process, even those that change expression substantially. We have also observed substantial chromatin changes that are not necessarily accompanied by changes in gene expression, suggesting that epigenetic modifications contribute to cell state in a manner that cannot be fully recapitulated by gene expression.

We did find, however, a strong association between chromatin marking and expression at the time of initial gene activation. We have been able to determine the precise order of histone modifications at that time, and found that only H3K4me1 and H3K4me2 appear to be deposited prior to gene activation. Further changes in gene expression, comparable or even stronger than those at gene activation, seem to be mostly uncoupled from changes in histone modifications

References

1. Hon, G., Wang, W. & Ren, B. Discovery and Annotation of Functional Chromatin Signatures in the Human Genome. *PLoS Computational Biology* 5:e1000566, 2009.
2. Barski, A.et al. High-Resolution Profiling of Histone Methylations in the Human Genome. *Cell* 129: 823–837, 2007.
3. Karlic, R., Chung, H. R., Lasserre, J., Vlahovicek, K. & Vingron, M. Histone modification levels are predictive for gene expression. *Proceedings of the National Academy of Sciences of the United States of America*, 107:2926–2931, 2010.
4. Dong, X. et al. Modeling gene expression using chromatin features in various cellular contexts. *Genome Biology* 13: R53, 2012.
5. Hodl, M. & Basler, K. Transcription in the absence of histone H3.2 and H3K4 methylation. *Current Biology* 22:2253–2257 (2012).
6. Perez-Lluch, S. et al. Absence of canonical marks of active chromatin in developmentally regulated genes. *Nature Genetics* 47:1158–1167, 2015.



> Session 1
Structural Bioinformatics I

Flexible protein structural alignment for non trivial comparisons

Gabriel Cretin[1,2], Charlotte Perin[1,2], Nicolas Zimmermann[1,2],

Tatiana Galochkina[1,2], Jean-Christophe Gelly[1,2]

[1] Université de Paris, Inserm UMR_S 1134 BIGR, INTS, 6 rue Alexandre Cabanel, 75015 Paris, France

[2] Laboratoire d'Excellence GR-Ex, 75015 Paris, France

Corresponding Author: gabriel.cretin@u-paris.fr and jean-christophe.gelly@u-paris.fr

Protein structure alignment is one of the most basic operations for the study of protein structures. Structure alignment is fundamental to analyze and understand evolutionary, structural and functional mechanisms by highlighting similarities that exist between proteins. Protein structure alignment is classically performed by methods based on rigid superposition. However in the case of non-trivial structural similarities, due to the inherent flexibility of protein structures or to evolutionary events disrupting the organization of the protein architecture, structure alignment remains challenging for classical algorithms. We propose a new method: ICARUS, based on a preliminary partitioning of protein into structural units that are subsequently iteratively aligned to a target structure. Our method outperforms both classical and flexible structural alignment reference methods on difficult structural alignment cases.

Keywords Structural alignment, flexibility, protein structure, Protein Units

1. Introduction

The comparison between two protein structures is useful to characterize their similarity at an atomic level in order to highlight similarities of structure and function, evolutionary links. As the structure is more conserved than the sequence [1], the alignment of the structures of two distant homologues allows to highlight the mutation tolerance of the folding but also its flexible areas as well as the positions of residues important for their function. Moreover, there are many protein families with a small number of folds [2–5]. Thus, there are many evolutionarily unrelated but similarly folding proteins: analogues. The structural alignment of two analogues reveals not only the capacity of a protein sequence to fold, but also the positions of the residues that are fundamental for it [6].

Thus, structural alignment allows study of many aspects of proteins and this is why many methods offer the possibility to automatically superimpose and determine equivalent positions of the two structures [7]. The main metrics used to quantify the distance between the two structures are the Root Mean Square Deviation (RMSD) and Template Modeling score (TM-score) [8]. TM-score was designed to be more meaningful than RMSD by being independent of protein size and less sensitive to large local deviations that strongly penalize RMSD and make it less relevant for comparing two structures.

The problem of structural alignment is to superimpose optimally two protein structures represented by their atoms to minimize distance between aligned positions. The most used methods are Combinatorial Extension (CE) [9, 10], and TM-align [11]. Both methods are based on rigid sequential alignment. The problems consist of finding the optimal rotation and translation of one structure to minimize distances between superposed structures and it mostly relies on least-squares fitting algorithms. However, there are complex evolutionary events that make homologous protein structures difficult to align. These include a different number of repeats of the same subunit, circular permutation or large insertions [12]. In addition, the same protein may possess a high degree of

flexibility and thus have several resolved structures with a variety of conformations. In order to highlight the structural similarity in these particular cases, classical alignment methods are not sufficient; only flexible structural alignment methods such as FATCAT [13, 14] can highlight these complex evolutionary relationships. FATCAT is the reference method for flexible structural alignment. Compared to other methods, it is at least as competitive in terms of performance and has become *de facto* the algorithm to which all other methods are compared to [15–17].

FATCAT, just as for the vast majority of flexible methods, relies on the detection of "hinge" positions around which rigid subsets of the structure to be aligned orient themselves relative to each other in order to achieve the best possible overall alignment. The first step in FATCAT consists in the identification of aligned fragment pairs (AFP) from the two proteins to compare. Two fragments of fixed length L (8 in the original paper) form an AFP if the RMSD value of the superposed fragments is below a given threshold. In the second step, FATCAT builds the global structural alignment by combining the sequentially aligned AFPs by dynamic programming. The combination of different consecutive AFPs is determined by rotations/translations between AFPs. The scoring function is mainly based on the amplitude of the rotations/translations, the size as well as the RMSD values of the AFPs. Different post-processing steps are then performed to optimize the global alignment.

Here, we present ICARUS, a non sequential flexible alignment method that uses the Protein Peeling algorithm [18, 19] to define Protein Units (PUs) as independent rigid regions to be aligned.

2. Materials and Methods

The basic principle of the ICARUS algorithm is based on iterative alignments of small compact regions called Protein Units (PU). The first step consists of applying the Protein Peeling algorithm on one protein considered as the “query” for the identification of the rigid regions. Then ICARUS builds a number of iterative structural alignments between Protein Units of the first protein and the second protein (the “target”) which remains unaltered.

Using one of the proteins as the target, Protein Peeling is applied on the query protein in order to subdivide it into PUs: compact fragments with high density of internal contacts and low number of contacts between each other. Protein Peeling will segment the query protein into either 4 or 5 consecutive PUs or 4 or 5 PUs attached by non-PU “hinge” portions. Then, ICARUS performs subsequent alignments of the identified PUs one by one using TM-align [11] onto the target protein. At each stage of the process, the next PU is aligned to the portion of the protein which has not yet been associated with any previously aligned PU. We explore all the possible alignment strategies that can be obtained by changing the order of PU alignments. This strategy results in an exhaustive tree-like exploration in which every branch represents all possible successive alignments between the 4 or 5 PUs and the target. The algorithm complexity is thus $O(n!)$ where n is the number of PUs. To optimize the number of calculations, we have limited the exploration of tree branches using the branch-and-bound algorithm. It does not reduce complexity in the worst case, and it is impossible to accurately determine the effect of the algorithm in the average case. However, in practice, it speeds up the tree search greatly.

Considering the query protein as consecutive PUs ensures a more flexible representation of the structure and allows addressing problems of protein fragment insertions, circular permutations or repetitions as PUs can be aligned to any portion of the target on which no PU was aligned yet. Finally, ICARUS repeats the procedure after switching target and query proteins. Once all the intermediate alignments are explored, the best global alignment is chosen on the basis of the TM-score [8] between the newly formed query protein and the target, normalized by the length of the smallest protein.

3. Results

The proposed strategy has been tested on the RIPC dataset [7] of 40 pairs of similar but complex structures to align due to non-trivial evolutionary relationships or high flexibility. The cases identified in this database are protein pairs that diverge due to Repetitions, Insertions, Permutations,

Conformational Variability (RIPC) and mixtures of these cases.

Our results were compared to the reference rigid alignment tool TM-align and to the reference flexible alignment method FATCAT [13, 14]. We used TM-score normalized by the shortest sequence length to evaluate the similarity between two aligned structures. TM-score adopts values between 0 and 1, where 1 means that two structures are identical. Two structures are considered to share the same fold if their TM-score is higher than 0.5. Above this threshold value, similarity increases with the increase of TM-score [8].

ICARUS clearly outperforms both TM-align and FATCAT methods on the RIPC dataset (Tab. 1). Indeed, in all cases except one, ICARUS outperforms both tools. ICARUS obtains an average TM-score of 0.74, while TM-align and FATCAT obtain 0.53 and 0.66 respectively. FATCAT obtains a TM-score 0.04 higher than ICARUS on only one case of Circular permutation and repetition (d1b5ta_ vs d1k87a2) of the dataset. ICARUS outperforms FATCAT in 34 out of 40 cases with an average gain of TM-score of 0.08 (Fig. 1). The average difference between ICARUS and FATCAT is statistically significant (P-value = 4 x 10^{-6}, Wilcoxon paired signed-rank test).

Structural relations types	**Number of protein pairs**	**FATCAT**	**TM-align**	**ICARUS**
Conformational variability	4	0.91	0.53	**0.92**
Circular permutation	4	0.59	0.57	**0.77**
Circular permutation and insertion	5	0.55	0.37	**0.64**
Insertion	12	0.58	0.55	**0.65**
Insertion and conformational variability	6	0.57	0.50	**0.67**
Insertion and repetition	5	0.62	0.57	**0.69**
Circular permutation and conformational variability	2	0.62	0.59	**0.72**
Circular permutation and repetition	1	**0.75**	0.56	0.71
Conformational variability and repetition	1	0.78	0.53	**0.91**
Average		0.66	0.53	**0.74**

Tab. 1: Performance in terms of mean values of TM-score of FATCAT, TM-align and ICARUS for different categories of structural and evolutionary events from the RIPC dataset (40 protein pairs).

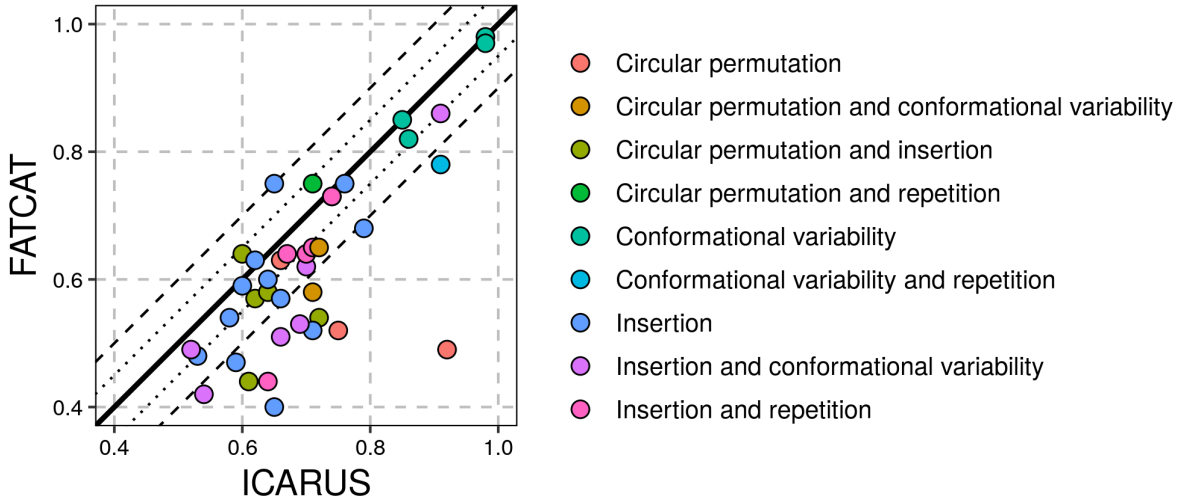


Fig. 1: Comparison of ICARUS and FATCAT TM-scores obtained on all pair proteins of RIPC set. Dotted and dashed diagonal lines represent differences of 0.05 and 0.1 of TM-score respectively.

ICARUS is the only tool able to detect structural similarity even for the particularly complex targets

of the RIPC dataset and their alignment score never fell below 0.5. We show an example of an output alignment of ICARUS of domains d1adl__ and d1mup__ (a complex case of Insertion and conformational variability) in Figure 2. We observe that even very complex evolutionary events can be successfully identified by ICARUS. Indeed, as shown in Figure 2.C4. ICARUS was, for instance, able to successfully determine a fragment represented by the PU4, which is constructed by moving the query positions 23-36 (helix) to the end of the query sequence, representing a beta-coil-helix pattern, whereas TM-align and FATCAT were unable to get this much flexibility to grasp this large insertion.

A

Aligned Protein Units:

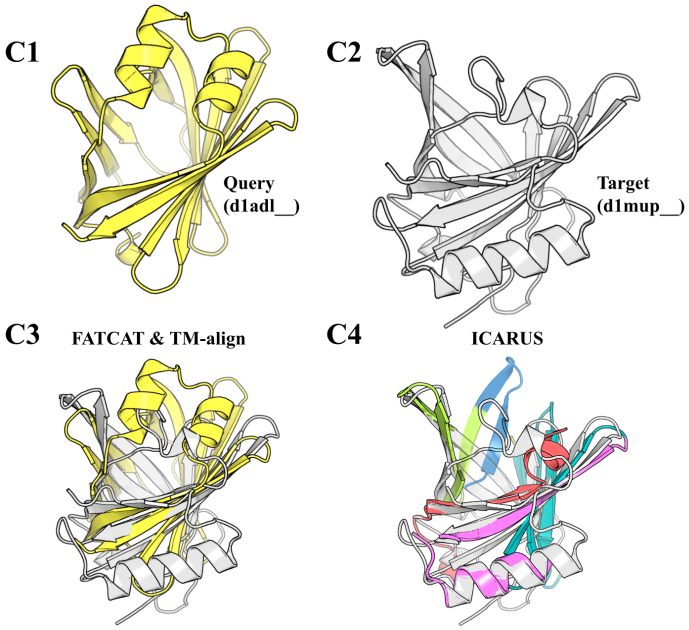
```
PU 2   (1adl):CDAFVGTWKLVSSENfDDymkE
              ::|||||||||||||  :
Target (1mup):VEKINGEWHTIILASd-Krek-

PU 5   (1adl):nMIISVNGDLVTIRSESTFKNTEISFRKRDGD------KLVVECVMK
               ::|...  :|||||||. .||||   ||||..      :||||||||:
Target (1mup):vADKTEK--AGEYSVTY-DGFNT---FTIPKTDYDNFLMAHLINEKD

PU 4   (1adl):GVTSTRVYERA--VGVGFATRKVAGMA---K
              :|||||||||:|  .|||||||||||||   .
Target (1mup):GETFQLMGLYGREPDLSSDIKERFAQLCEEH

PU 1   (1adl):vKSIITLDGG-----ALVQVQKWDGKSTTIKp
               .:|:..||:     ||||||||||||||||
Target (1mup):fRLFLEQIHVLENSLVLKFHTVRDEECSELSm

PU 3   (1adl):KLGVEFD-eITADDRK
                 ..  |.
Target (1mup):-----IEDnGN-----
```



B

Full alignment:

```
PUs    (1adl):                  PU2               PU3                  PU1
connect      :          +--------------------++--------------++-----------------------------++------------------
QUERY        :          CDAFVGTWKLVSSENfDDymkEKLGVEFD-eITADDRKvKSIITLDGG-----ALVQVQKWDGKSTTIKpnMIISVNGDLVTIRSESTF
match        :          ::|||||||||||||  :         ..  |.      .:|:..||:     ||||||||||||||||  ::|...  :|||||||. .
Target (1mup):EEASSTGRNFNVEKINGEWHTIILASd-Krek------IEDnGN-----fRLFLEQIHVLENSLVLKFHTVRDEECSELSmvADKTEK--AGEYSVTY-D

PUs    (1adl):  PU5                              PU4
connect      :---------------------------++-----------------------------+
QUERY        :KNTEISFRKRDGD------KLVVECVMKGVTSTRVYERA--VGVGFATRKVAGMA---K
match        :||||   ||||..      :|||||||::|||||||:|  .|||||||||||||   .
Target (1mup):GFNT---FTIPKTDYDNFLMAHLINEKDGETFQLMGLYGREPDLSSDIKERFAQLCEEHGILRENIIDLSNANRC
```

Fig. 2: **A and B.** ICARUS output alignment for superposition of a complex case of Insertion and conformational variability (d1adl__ vs. d1mup__). Matching symbols correspond to distances (d) of alignment between query and target residues: '|' (d <= 1Å), ':' (d <= 2Å), '.' (d <= 4Å) and ' ' (d > 4Å). **A.** Alignment of individual Protein Units against the matching portions of the target protein. **B.** Full alignment of the query against the target. Matching PUs and their connections are also represented. **C1.** Query protein d1adl__ from RIPC dataset (yellow). **C2.** Target protein d1mup__ from RIPC dataset (grey). **C3.** Output alignment of TM-align and FATCAT (identical for both tools). The query d1adl__ and the target d1mup__ are in yellow and grey respectively. **C4.** Output alignment of ICARUS. The target d1mup__ is shown in grey, and the query is coloured after the corresponding PUs represented in **A.** and **B.**

4. Conclusion

In case of high conformational variability, or complex evolutionary events such as insertions, circular permutations or repetitions alignment of protein structures are challenging. For these complex cases, classical methods are not efficient and the only way to superpose correctly two complex structures is to use flexible alignment methods. We presented here the ICARUS approach which is based on the partitioning of protein structures into Protein Units to determine the rigid regions which

are reoriented during the alignment procedure. Our method demonstrates excellent performance on the RIPC dataset with higher mean TM-scores obtained for 8/9 structural relation categories as compared to the FATCAT reference method. Finally, ICARUS highlights the structural similarity of each protein couple more clearly than FATCAT.

The efficiency of the ICARUS algorithm comes both from the definition of the rigid fragments as PUs and from the way PUs are aligned. In contrast to FATCAT, ICARUS does not keep the structure sequential, thus allowing the algorithm to detect similarities between protein structures related by such complex evolutionary events as permutation and insertion. Moreover, the use of Protein Units for the identification of rigid regions and hinge positions in the protein structure contributes significantly to the algorithm performance. PUs were previously shown to compose rigid and stable structural units [18–21]. In the current study, PUs allow us to obtain high quality structural alignments for related proteins linked by complex evolutionary events corresponding to reorganizations/repetitions/growths of structural modules of intermediate size. Therefore, it can be expected that PUs play a role as evolutionary and/or structural modules like secondary structures and domains.

References

1. Illergård, K., Ardell, D.H., Elofsson, A.: Structure is three to ten times more conserved than sequence--a study of structural response in protein cores. Proteins. 77, 499–508 (2009)
2. Wolf, Y.I., Grishin, N.V., Koonin, E.V.: Estimating the number of protein folds and families from complete genome data. J. Mol. Biol. 299, 897–905 (2000)
3. Coulson, A.F.W., Moult, J.: A unifold, mesofold, and superfold model of protein fold use. Proteins. 46, 61–71 (2002)
4. Koonin, E.V., Wolf, Y.I., Karev, G.P.: The structure of the protein universe and genome evolution. Nature. 420, 218–223 (2002)
5. Leonov, H., Mitchell, J.S.B., Arkin, I.T.: Monte Carlo estimation of the number of possible protein folds: effects of sampling bias and folds distributions. Proteins. 51, 352–359 (2003)
6. Sierk, M.L., Kleywegt, G.J.: Déjà vu all over again: finding and analyzing protein structure similarities. Structure. 12, 2103–2111 (2004)
7. Mayr, G., Domingues, F.S., Lackner, P.: Comparative analysis of protein structure alignments. BMC Struct. Biol. 7, 50 (2007)
8. Xu, J., Zhang, Y.: How significant is a protein structure similarity with TM-score = 0.5? Bioinformatics. 26, 889–895 (2010)
9. Shindyalov, I.N., Bourne, P.E.: Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. Protein Eng. 11, 739–747 (1998)
10. Holm, L.: DALI and the persistence of protein shape. Protein Sci. 29, 128–140 (2020)
11. Zhang, Y., Skolnick, J.: TM-align: a protein structure alignment algorithm based on the TM-score. Nucleic Acids Res. 33, 2302–2309 (2005)
12. Grishin, N.V.: Fold change in evolution of protein structures. J. Struct. Biol. 134, 167–185 (2001)
13. Ye, Y., Godzik, A.: Flexible structure alignment by chaining aligned fragment pairs allowing twists. Bioinformatics. 19 Suppl 2, ii246–55 (2003)
14. Li, Z., Jaroszewski, L., Iyer, M., Sedova, M., Godzik, A.: FATCAT 2.0: towards a better understanding of the structural diversity of proteins. Nucleic Acids Res. 48, W60–W64 (2020)
15. Salem, S., Zaki, M.J., Bystroff, C.: FlexSnap: Flexible Non-sequential Protein Structure Alignment, http://dx.doi.org/10.1186/1748-7188-5-12, (2010)
16. Daniluk, P., Lesyng, B.: A novel method to compare protein structures using local descriptors, http://dx.doi.org/10.1186/1471-2105-12-344, (2011)
17. Terashi, G., Takeda-Shitaka, M.: CAB-Align: A Flexible Protein Structure Alignment Method Based on the Residue-Residue Contact Area. PLoS One. 10, e0141440 (2015)
18. Gelly, J.-C., de Brevern, A.G., Hazout, S.: “Protein Peeling”: an approach for splitting a 3D protein structure into compact fragments. Bioinformatics. 22, 129–133 (2006)
19. Gelly, J.-C., Etchebest, C., Hazout, S., de Brevern, A.G.: Protein Peeling 2: a web server to convert protein structures into series of protein units. Nucleic Acids Res. 34, W75–8 (2006)
20. Gelly, J.-C., de Brevern, A.G.: Protein Peeling 3D: new tools for analyzing protein structures. Bioinformatics. 27, 132–133 (2011)
21. Postic, G., Ghouzam, Y., Chebrek, R., Gelly, J.-C.: An ambiguity principle for assigning protein structural domains. Sci Adv. 3, e1600552 (2017)

MEDUSA: deep learning based protein flexibility prediction

Yann Vander Meersche[1,2], Gabriel Cretin[1,2], Alexandre G. de Brevern[1,2],

Jean-Christophe Gelly[1,2] and Tatiana Galochkina[1,2]

[1] Université de Paris, Inserm UMR_S 1134 BIGR, INTS, 6 rue Alexandre Cabanel, 75015 Paris, France

[2] Laboratoire d'Excellence GR-Ex, 75015 Paris, France

Corresponding Author: jean-christophe.gelly@u-paris.fr and tatiana.galochkina@u-paris.fr

Paper Reference: **Vander Meersche *et al.* (2021) MEDUSA: Prediction of Protein Flexibility from Sequence, Journal of Molecular biology, 2021, 166882. https://doi.org/10.1016/j.jmb.2021.166882**

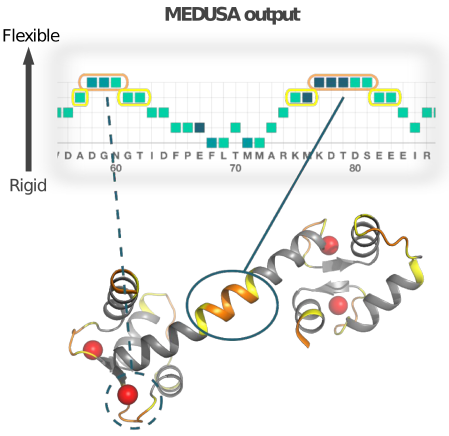
1. Introduction

Biological function of proteins is determined by their structure and dynamics. For today, experimental determination of the protein dynamical properties remains difficult and costly. Thus, development of computational prediction tools have potential to provide information on dynamics properties for the proteins without resolved structure. Recent advances in the field of artificial intelligence have demonstrated the potential of machine learning for structural bioinformatic problems. In the current study we developed a deep learning based prediction tool named MEDUSA for the protein flexibility prediction from amino acid sequence.

2. Materials and Methods

We have considered protein flexibility in terms of B-factor obtained in X-ray crystallography. B-factor reflects the attenuation of X-ray scattering caused by thermal motion and is the most common experimental descriptor of protein flexibility. We have developed four classification models predicting the degree of the expected protein flexibility in two, three and five classes. The algorithm uses evolutionary information extracted from homologous protein sequences combined with amino acid physico-chemical properties as input for a convolutional neural network for flexibility class prediction at each protein sequence position. Predictions are rated by a confidence index based on the network output probability. The performance of our model was estimated in 10-fold cross validation on a dataset filtered by structural similarity to ensure independence between train and test datasets.

3. Results and conclusion



MEDUSA is available as a web server (https://www.dsimb.inserm.fr/MEDUSA) as well as a standalone utility (https://github.com/DSIMB/medusa). MEDUSA outperforms the state-of-the-art method PROFbval [1] for the binary prediction problem. As we demonstrate in multiple biological examples, MEDUSA predictions successfully identify the potentially highly deformable protein regions for the proteins with known dynamical properties (the example of predictions for the calmodulin molecule is shown in the figure from Vander Meersche et al. 2021). Moreover, MEDUSA provides information on the presence of the locally rigid fragments for the proteins without stable fold and thus complements the information provided by the disorder prediction tools. Finally, we assess the impact of the quality of experimental data on the predictive model performance and provide possible solutions to overcome the limitations of the chosen descriptors.

References

1. Avner Schlessinger, Guy Yachdav and Burkhard Rost. PROFbval: predict flexible and rigid residues in proteins, *Bioinformatics*, 22:7, pages 891–893, 2006

A Graph-based Similarity Approach to Classify Recurrent Complex Motifs from their Context in RNA Structures

Coline GIANFROTTA[1,2], Vladimir REINHARZ[3], Dominique BARTH[1] and Alain DENISE[2,4]
[1] Université de Versailles Saint-Quentin-en-Yvelines, Université Paris-Saclay, DAVID lab, Versailles, France
[2] Université Paris-Saclay, CNRS, Laboratoire Interdisciplinaire des Sciences du Numérique, Orsay, France
[3] Department of Computer Science, Université du Québec à Montréal, Québec, Canada
[4] Université Paris-Saclay, CNRS, I2BC, Orsay, France

Corresponding author: coline.gianfrotta@ens.uvsq.fr

Reference paper: **Gianfrotta *et al.* (June 2021) A Graph-based Similarity Approach to Classify Recurrent Complex Motifs from their Context in RNA Structures. *19th Symposium on Experimental Algorithms.*** https://drops.dagstuhl.de/opus/volltexte/2021/13791

Introduction RNA molecules intervene, along with proteins, in all major cellular processes. An RNA molecule is composed of a sequence of nucleotides (A, C, G, U) which folds in space into a three-dimensional structure. The function of an RNA molecule is strongly related to its three-dimensional structure. This structure is composed of a rigid skeleton, a set of canonical interactions called the secondary structure. On top of the skeleton, the nucleotides form an intricate network of interactions that are not captured by present thermodynamic models [1]. This network has been shown to be composed of modular motifs, that are linked to function, and have been leveraged for better prediction and design [2], [3]. A peculiar subclass of structural motifs are those connecting RNA regions far away in the secondary structure. They are crucial to predict since they determine the global shape of the molecule, therefore important for the function.

Method This article proposes to use an RNA graph similarity metric, based on the Maximum Common Edge Subgraph (MCES) resolution problem [4], to compare structural contexts of this kind of motifs, represented as subgraphs of RNA graphs. We define the structural context of a motif as the set of canonical and non canonical interactions that appear at a distance k around the motif. We rely on a new modeling by graphs of these contexts, at two different levels of granularity, and obtain a classification of these graphs.

Results We explore the cases of three known motif families to validate our approach: the A-minor motif, which is frequently found in RNA 3D structures and involved in crucial cellular mechanisms [5], and two other 3D motifs from the database CaRNAval [6] (RIN 51 and 56). Those three motifs are not predictable by current computational methods, to the best of our knowledge. On these examples, the similarity in our new graph representation correlates with the geometric distance between the 3D models, while reducing the computation time in relation to a classical graph representation. Furthermore the classification induced by this similarity metric segregates well the structural contexts of the motifs. This study then shows that the structural context matters for those motifs that bind distant regions of RNA, and could be leveraged for the prediction of their location.

References

[1] A. Lescoute and E. Westhof. The interaction networks of structured RNAs. *Nucleic acids research*, 34(22):6587–6604, 2006.

[2] C. Oliver, V. Mallet, R. Sarrazin-Gendron, V. Reinharz, W. L. Hamilton, N. Moitessier, and J. Waldispühl. Augmented base pairing networks encode RNA-small molecule binding preferences. *Nucleic acids research*, 48(14):7690–7699, 2020.

[3] G. Chojnowski, T. Waleń, and J. M. Bujnicki. RNA Bricks—a database of RNA 3D motifs and their interactions. *Nucleic acids research*, 42(D1):D123–D131, 2014.

[4] J. Raymond, E. Gardiner, and P. Willett. RASCAL: Calculation of Graph Similarity using Maximum Common Edge Subgraphs. *Computer Journal*, 45:631–644, April 2002.

[5] A. Lescoute and E. Westhof. The A-minor motifs in the decoding recognition process. *Biochimie*, 88(8):993–999, August 2006.

[6] V. Reinharz, A. Soulé, E. Westhof, J. Waldispühl, and A. Denise. Mining for recurrent long-range interactions in RNA structures reveals embedded hierarchies in network families. *Nucleic Acids Research*, 46(8):3841–3851, May 2018.

Feature extraction for the clustering of small 3D structures: application to RNA fragments

Alix DELANNOY[1], Antoine MONIOT[1] , Yann GUERMEUR[1], Isaure CHAUVOT DE BEAUCHENE[1]

[1] LORIA (UL – CNRS – INRIA), 54000 Nancy, France

Corresponding Author: isaure.chauvot-de-beauchene@loria.fr

Abstract Structural libraries of fragments are commonly used to model or design the 3D structure of biomolecules (drugs, peptides, nucleic acids). They typically approximate all possible local conformations of these molecules within a given precision, by a set of well-chosen representative fragments. Such a set can be obtained by clustering a larger set of fragments whose structures have been solved experimentally, using suitable clustering algorithm and measure of dissimilarity between fragments. A commonly used measure of dissimilarity in structural biology is the root mean square deviation (RMSD), whose exact computation requires a pairwise structural alignment. But this alignment is highly time-consuming and not applicable for a very large initial set of fragments.

We propose here an approach based on feature extraction to perform an effective clustering, while avoiding a computationally expensive full pairwise alignment. Using as example poly-A RNA fragments of 3 nucleotides (3-nt), we searched for internal coordinates whose differences can best approximate the RMSD between two fragments without any superposition. We found that the simple differences of internal distances and angles can provide a lower bound on the RMSD, allowing us to filter out pairs of which the RMSD does not need to be computed. We can then compute the exact values for only the small RMSDs, and use it to apply more effective clustering methods.

We present this strategy and its application on 39431 RNA 3-nt, which could be approximated by only 3258 representative prototypes with 1 Å accuracy.

Keywords Fragment-based modeling, Structural library, Clustering, RNA 3D structure.

1 Introduction

Fragment-based methods are commonly used for modeling flexible polymers (protein loops, RNA...). They can exploit a discrete representation of the local conformations of the molecule in the form of a structural library [1], which contains an ensemble of conformers for each type of fragment. As an example, we use a library of trinucleotide (3-nt) conformations for fragment-based docking of ssRNA on proteins [2]. A straightforward approach to create structural libraries suitable for a given modeling task is to take all existing experimental structures of similar targets, extract all their fragments, and create a representative subset, by means of clustering. The objective is then to have as few prototypes as possible, while approximating the whole set with a given precision (governed by the application).

One common clustering criterion for the building of structural libraries is the root mean square deviation (RMSD), whose minimum value obtained after structural alignment is called conformational RMSD (cRMSD) [3]. Using this cRMSD raises problems reporting to both statistics and computational complexity. Indeed, there is no guarantee that the measure still exhibits all the properties of a metric, and its computation for all pairs of fragments can be time-consuming. We previously addressed both problems by aligning all fragments on one of them selected randomly before computing the RMSD, as an approximation of the cRMSD. But the resulting values are larger than the cRMSD, with the consequence that too many clusters/prototypes are generated.

Our present contribution provides a solution to both problems, based on feature extraction. Those new features, which do not require any structural alignment for their comparison, can be seen as internal coordinates. With these new descriptions at hand, we construct libraries of 3-nt prototypes such that every conformation is at most at 1Å of a prototype (according to the cRMSD). Compared to the previous

algorithm, our new method basically decreases the number of prototypes, under an acceptable computing time.

The problem is formalized in Section 2. The original contribution is introduced in Section 3. Finally Section 4 is devoted to the comparative experiments.

2 Problem statement

Our data are fragments x which belong to a subset $\mathcal{X}$ of an Euclidean space $\mathbb{R}^{3n}$, where n is the number of atoms. Their dissimilarity is measured by means of the normalized ℓ_2 distance (RMSD) computed after the application of a structural alignment. It is thus given by the following formula:

$$\forall\ (x, x') \in (\mathbb{R}^{3n})^2,\ d(x, x') = \sqrt{\frac{\sum_{k=1}^{3n} ((\phi(x))_k - (\phi(x'))_k)^2}{n}}$$

where $\Phi(x')$ in $\mathbb{R}^{3n}$ is the image of x' by the alignment. We consider two instances of the function Φ :

- Φ` is associated with the *one against all* strategy (all fragments are aligned on one single fragment, the reference fragment).
- Φ* is associated with the *one against one* strategy (the alignments are performed pairwise).

We assume that we are given m fragments x^i. Their matrix of dissimilarities $D = (d_{i,j})_{1 \le i,j \le m}$, given by $d_{i,j} = d(x^i, x^j)$, is used to produce the set of prototypes $\{\bar{x}\}$ through clustering. Let d` and d* be respectively the dissimilarity measures associated with $\Phi$` and Φ*. The prototypes must satisfy the constraint:

$\forall$ i, $1 \le i \le m$, $\exists\ \bar{x} : d^*(x^i, \bar{x}) \le$ threshold.

Given the fact that the constraints involve d*, using the matrix of dissimilarities associated with Φ`raises an obvious difficulty. If we focus on the kind of libraries we are especially interested in (3-nt RNA conformations for fragment-based docking), then it appears that the values of the RMSD after alignement on a reference (d`) and of the cRMSD (d*) can vary up to 7Å. Symmetrically, using the matrix associated with Φ* restricts the choice of the clustering methods, since it is no longer a matrix of distance. Furthermore, the computation of this second matrix is far more time consuming than the previous one, since its complexity is quadratic in the number of fragments.

3 Methods

3.1 Representation in Cartesian and internal coordinates

We extracted from the Protein Data Bank all the overlapping 3-nt RNA fragments in all structures of RNA-protein complexes obtained by X-ray crystallography (with resolution < 3Å) or solution NMR, using our in-house protNAff tool [https://github.com/isaureCdB/ProtNAff]. We then convert them into the coarse-grain representation defined in ATTRACT, which replaces sets of 3-4 heavy atoms by one pseudo-atom, resulting in 7 pseudo-atoms per purine nucleotide (Fig 1).

To define relevant internal coordinates, taking inspiration from existing methods [4, 5], we selected and computed 6 distances and 9 dihedral angles:

- the 3 pairwise distances between bases, using for each base the pseudo-atom the farthest from the backbone (GA4)
- the distance between 5'-GS2 (sugar) and 3'-GA4 (base)
- the distance between 5'-GA4 (base) and 3'-GS2 (sugar)
- the length of the backbone, from 5'-GP (phosphate) to 3'-GS1 (sugar)
- the 3 backbone angles between pseudo-atoms GP and GS1 of consecutive nucleotides
- the 3 μ angles between sugar and base of each nucleotide, using the pseudo-atoms GS1 – GS2 – GA1 – GA2.
- the 3 χ angles between the sugar-base axis GS2 – GA1 of two nucleotides.

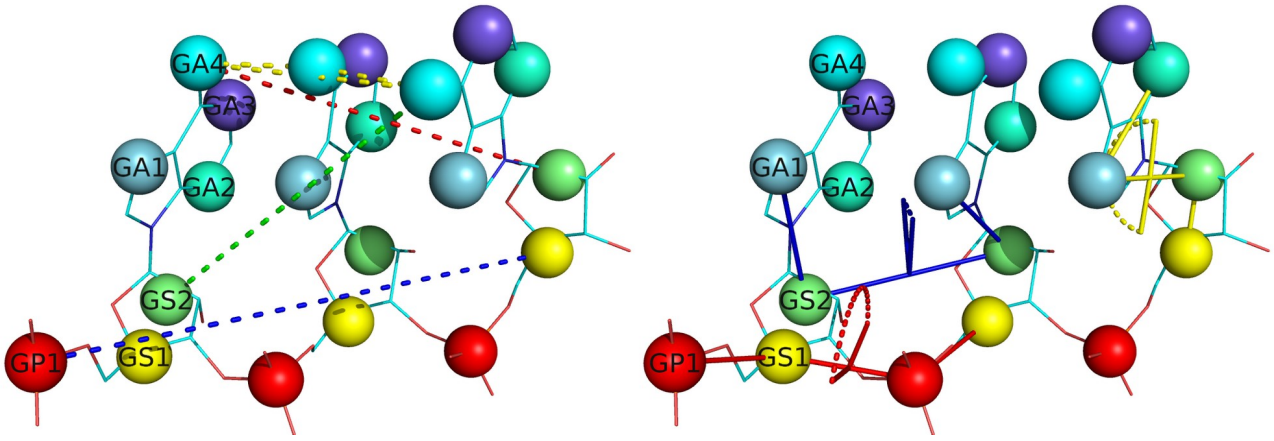


Fig 1. Selected internal coordinates: distances (left) and angles (right) on a trinucleotide in all-atoms (sticks) and coarse-grained (beads) representations, with the name of the pseudo-atoms on the 3' nucleotide.

3.2 Connection between internal coordinates and RMSD

We analysed the distribution of its values among the fragments, and evaluated how to connect the differences between two fragments measured either by the cRMSD or by the difference in each internal coordinate. We selected four times a random sample of 10 % of the full set of fragments, and computed for all pairs of fragment (i) the pairwise cRMSD after fitting, (ii) the difference between each internal coordinate, and (iii) the sum of the differences over the internal either distances or angles. We selected, for each of the 15 coordinates and 2 sums of coordinates, the threshold value above which all pairs of fragments have a cRMSD above 1 Å. In practice, to make the filtering more stringent, we allowed 1 false negative per 1000 positives (meaning that 1/1000 pairs with cRMSD < 1 Å are above that threshold). We then computed the 17 average threshold values over the 4 random samples.

We applied those thresholds on the full set of 39431 fragments. We computed all the internal coordinates and their pairwise (sum of) differences, then selected the pairs with all 17 values below the corresponding threshold. The pairwise alignment and computation of the cRMSD value were done only on that subset of pairs. For all other pairs, the cRMSD was considered as above 1 Å.

3.3 Choice of clustering methods with the full RMSD matrix

Three clustering algorithms are described below, that are compatible with our dissimilarity matrix. One fast clustering using only a subset of approximate RMSD values was applied on the full set of fragments. The two others use the full pairwise cRMSD matrix and were applied and compared in 2 cases: First, on the prototypes obtained by fast clustering with approximate RMSD values, in order to evaluate the potential gain in the number of clusters by using more effective clustering algorithms on cRMSD values. Second, we applied them on the full set of fragments, using the RMSD matrix obtained after filtering by differences of internal coordinates.

Fast Clustering with approximate RMSD

We first align each fragment on one fragment randomly chosen. All pairwise structural alignments in this study are done with the Kabsch algorithm, using the *fit.py* protocol of ATTRACT. We then use the fastcluster protocol from ATTRACT, whose algorithm goes as follows : Initialization is performed by randomly choosing (using a uniform law) a fragment as the 1st cluster prototype. Then, for each fragment is measured the distance to each of the prototypes in the current set after alignment. If one of these distances is less than the chosen threshold, then the fragment is assigned to that cluster. Otherwise, it is added to the set of prototypes.

Hierarchical agglomerative clustering (HAC)

This type of clustering is a "bottom-up" approach. At the start, each fragment is a prototype, then the two closest clusters (depending on the chosen linkage) are agglomerated, and this is iterated until reaching the linkage threshold, resulting in a hierarchy of clusters. The number of clusters obtained by the method is dependent on the threshold applied on the linkage. We applied it with a complete linkage of 1 Å, meaning that two clusters are agglomerated if the maximum distance between two members from each cluster is

below 1 Å. We used the Agglomerative Clustering function of the sklearn python module. Finally, the prototype for each cluster can be chosen as the averaged conformation from all members.

Star-shape clustering

We also applied a star-shape clustering algorithm, creating clusters with all distances of each member to a central element below a given threshold. We first select the fragment with the highest number of connected fragments (with RMSD below the threshold), assign this fragment and its connections to the first cluster and remove them from the pool, then repeat. If several fragments have the maximal number of neighbors, one of them is picked randomly. Given this stochastic aspect, the clustering was run three times on the full set of fragments, and the cluster set with smallest cardinality was kept.

4 Results

4.1 Re-clustering of prototypes with cRMSD values

By re-clustering the 4771 AAA prototypes with the pairwise cRMSD values using HAC, we obtained 3307 new clusters: The current fast clustering method with approximated RMSD is indeed non optimal, and the number of clusters can be reduced by at least 30% with more accurate methods. We also tested to apply a star-shape clustering method, and obtained 3248 clusters. As the number of clusters obtained by both re-clustering methods is quite similar, we decided to test both on the full set of fragments, after filtering by internal coordinates.

4.2 Connection between internal coordinates and RMSD

We computed all 15 internal coordinates in the full set of fragments, and plotted their distributions (Fig 2). Among distances, the base-base distance show a large variance, while the 3'-sugar – 5'-base distance is more conserved among fragments. Among angles, the χ angles representing the relative orientation of two nucleotides show a large variance, while the μ angles between sugar and base of each nucleotide are much more conserved.

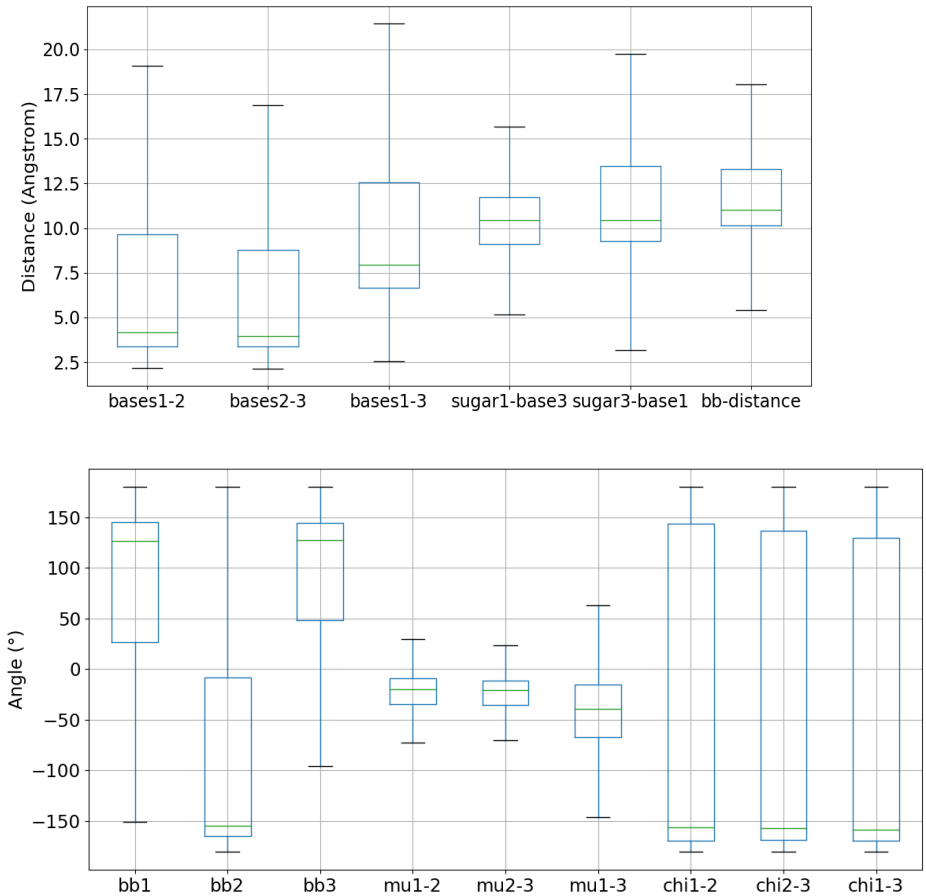


Fig 2. Distribution of the selected internal coordinates among the 39431 fragments.

We then analysed the link between the RMSD and the differences in internal coordinates for the 4 random samples of fragments (see part 3.2). We looked at which conformations are closer than 1 Å cRMSD, and we mostly found pairs below a certain difference threshold, for each internal coordinate (Fig 3). For differences above this threshold, only 0.1% of the cRMSD values below 1 Å are found.

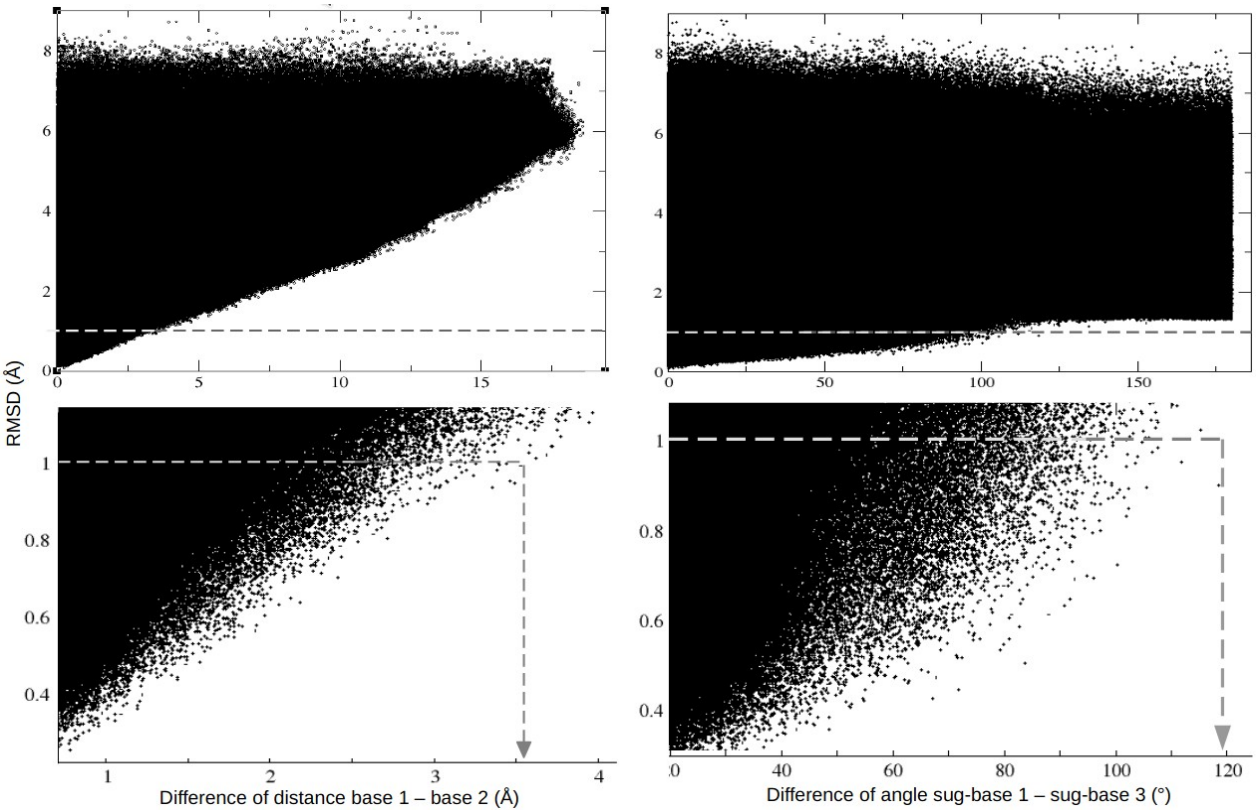


Fig 3. Correlation between pairwise RMSD and difference in some internal distances/angles.

Distances								
base 1-2	base 2-3	base 1-3	sugar 1 – base 3	sugar 3 – base 1	bb	sum	
43 %	47 %	43 %	57 %	49 %	52 %	23 %	
Angles								
bb1	bb2	bb3	mu1-2	mu2-3	mu1-3	chi1	chi2	chi3	sum
76 %	67 %	70 %	49 %	52 %	45 %	80 %	77 %	89 %	28 %

Table 1. Percentage of pairs that are under the threshold holding 99.9% of the compatible pairs, for each internal coordinate, in the 39431 fragments.

When looking at each individual threshold, the most efficient filtering is provided by the sum of distances, the sum of angles and the base-base distances, while the χ angles give the least efficient filters.

4.3 Clustering with internal coordinates filters

We tested the combination of the 17 thresholds (see 3.2) on the four random samples. The real percentage of cRMSD values under 1 Å is in range 8.6 - 9.7 % (average 9.2 %) in each sample, and is assumed to be in the same range for the full set of fragments. We found that the proportion of pairs for which all values are below the 17 thresholds is in range 14 - 16 % in the samples, meaning that we can reduce the number of pair

alignments to only ~15 % of all pairs. Among the pairs kept, 54 - 62 % were real positives. This set of thresholds was then applied to the full set of 39431 fragments. As expected, 15 % of the 1.6×10^9 pairs were identified as potentially under 1 Å cRMSD. Those were selected for pairwise alignment and RMSD computation. For the other pairs, the cRMSD was considered as above 1 Å.

Using the pre-filtered full RMSD matrix resulted in 3258 and 5483 clusters with the star-shape and agglomerative clustering algorithms respectively. The agglomerative clustering requires an upper bound on the RMSD between members from two clusters to agglomerate them. This results in clique clusters, with all members within 1 Å from each other. This is more stringent than our initial objective to have all members at a maximal distance from the cluster center, and might explain the higher number of clusters obtained by agglomerative versus star-shape clustering.

4.4 CPU times

To estimate the gain of pre-filtering with internal coordinates in terms of CPU time, we computed the full c RMSD-matrix for the 4 samples, either with or without pre-filtering, on 1 CPU. The computation of the internal coordinates and of their pairwise differences takes less than 1''. The cRMSD-matrix calculation for 4773 AAA fragments takes ~ 23' for all pairs, and < 5' for the pre-filtered pairs.

On the full set of fragments, the computation of the internal coordinates and of their pairwise differences takes 2'' and 4' respectively. The clustering with the pre-filtered RMSD matrix takes ~1' for agglomerative clustering and ~ 45' for star-shape clustering, each on 1 CPU.

5 Conclusion and future work

We showed that it is possible to overcome both the statistical and the computational problems associated with clustering fragments based on their cRMSD, by extracting features from the Cartesian coordinates. Those internal coordinates are used to evaluate if a structural alignment is needed to calculate the cRMSD between two fragments. Using this filter, the cRMSD matrix can be computed and used for new clustering methods. While this paper presents an application on RNA trinucleotides, the approach can be extended to different RNA structures, and different molecules such as peptides.

We are now developing a specific clustering method based on the hierarchical clustering, but with a different linkage. The idea is to calculate the smallest enclosing ball, containing the two linked clusters. Its center is the prototype of the new cluster, whose RMSD after alignment to all other prototypes are computed. The reduction of calculation time shown in this paper is a great help for this new method.

To refine even more the fragment libraries, the use of other dissimilarities may be explored. The current normalised ℓ_2 distance takes into account deviations globally rather than locally. However, local deviations might have a significant impact on the relevance of the RNA models created from the fragments. Other dissimilarities measures (from mixed standards...) can take this constraint into account.

Acknowledgements

We thank Sjoerd de Vries for insightful discussions on this work.

References

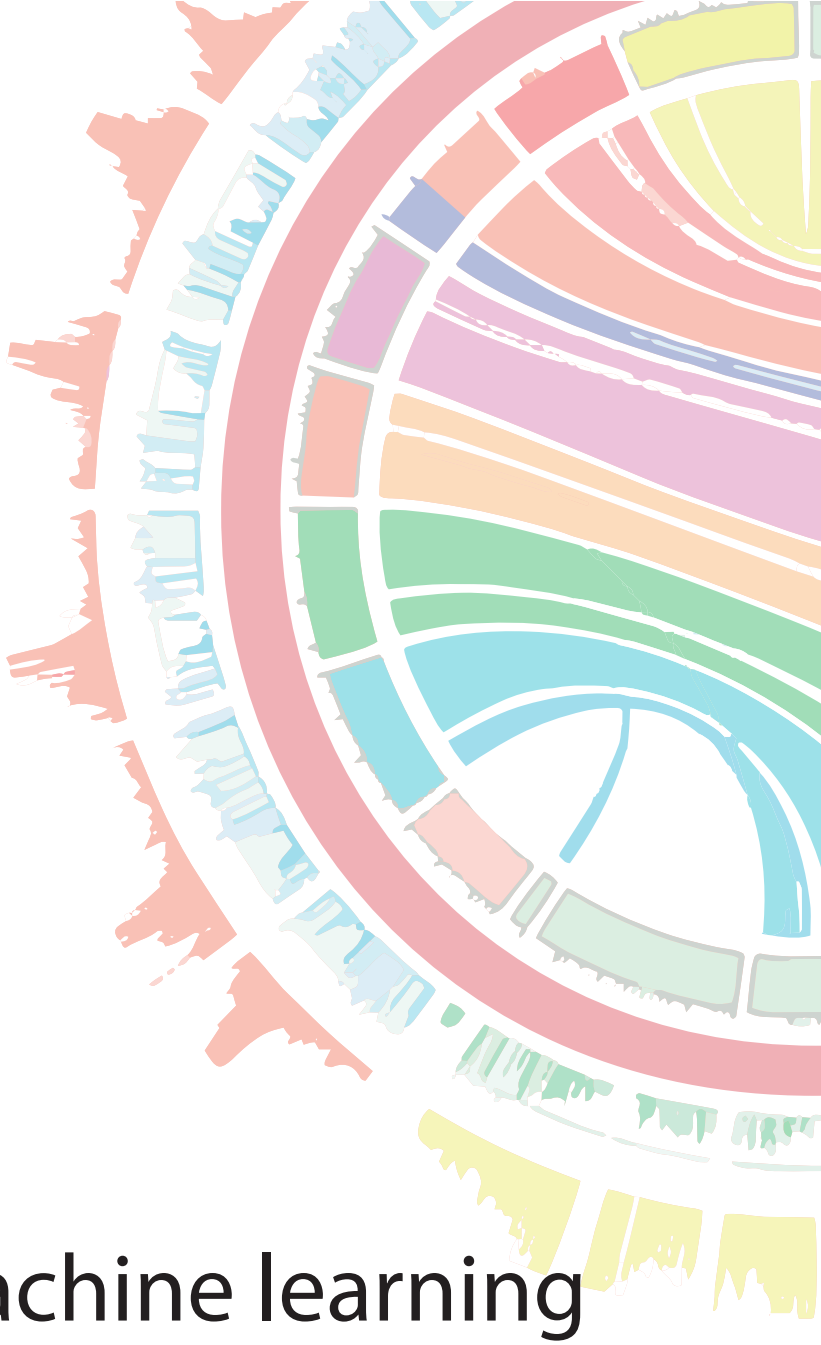
[1] Erik Verschueren, Peter Vanhee, Almer M van der Sloot, Luis Serrano, Frederic Rousseau, Joost Schymkowitz. Protein design with fragment databases. *Curr Opin Struct Biol*, 21(4):452-9, 2011.

[2] Isaure Chauvot de Beauchene, Sjoerd Jacob de Vries, Martin Zacharias. Fragment-based modelling of single stranded RNA bound to RNA recognition motif containing proteins. *Nucleic Acids Res*, 44(10):4565-80, 2016.

[3] Shuai Cheng Li. The difficulty of protein structure alignment under the RMSD. *Algorithms Mol Biol*, 4;8(1):1, 2013.

[4] Tomasz Zok, Maciej Antczak, Martin Riedel, David Nebel, Thomas Villmann, Piotr Lukasiak, Jacek Blazewicz, Marta Szachniuk. Building the library of RNA 3D nuclotide conformations using the clustering approach. *International Journal of Applied Mathematics and Computer Science*, *25*(3): 689-700, 2015.

[5] Jiří Černý, Paulína Božíková, Jakub Svoboda, Bohdan Schneider. A unified dinucleotide alphabet describing both RNA and DNA structures. *Nucleic Acids Res*, 48(11):6367-6381, 2020.



Session 2

Statistics, machine learning & artificial intelligence I

Spliceator: A new strategy for splice site prediction using deep learning algorithm

Nicolas SCALZITTI[1], Anne JEANNIN-GIRARDON[1], Pierre COLLET[1], Olivier POCH[1], Julie THOMPSON[1]

[1] Complex Systems and Translational Bioinformatics (CSTB), ICube laboratory, UMR7357, University of Strasbourg, 1 rue Eugène Boeckel, 67000, Strasbourg, France

Corresponding Author: thompson@unistra.fr

Paper Reference: **Scalzitti *et al.* (2020) A benchmark study of ab initio gene prediction methods in diverse eukaryotic organisms, BMC Genomics 21, 293 https://doi.org/10.1186/s12864-020-6707-9**

1. Abstract

High-throughput technologies are constantly generating huge amounts of genomic sequences that represent an essential source of information for studying and understanding living organisms. However, without a crucial "*annotation*" step to add information, these raw sequences are difficult to exploit and sometimes even useless. One of the main challenges of annotation is to identify genes and characterize their internal structure in the genome[1]. In Eukaryotes, this step is very complex in particular for protein coding genes. Indeed, the architecture of these genes is organized in a mosaic of exons and introns[2] delimited by boundaries called *splice sites*. There are two types of splice sites, the 5' (donor) and 3' (acceptor) sites, which are respectively the junction between exon-introns and intron-exons[3]. The splice sites are mainly characterized by the presence of GT (5') and AG (3') dinucleotides, embedded in a longer, more divergent pattern of about ten nucleotides. To help identify these sites, many prediction programs based on machine learning algorithms have been developed [4, 5]. Unfortunately, these programs are often dedicated to model organisms (e.g A. *Thaliana*, C. *Elegans*) or to human, and still generate too many annotation errors[6] that can affect downstream studies.

In this context, we have developed a new multi-species splice site prediction tool, based on the G3P0 dataset [7]. G3PO was specifically established for our study, and special attention was paid to data quality, thanks to the implementation of a protocol based on multiple sequence alignments. G3PO contains more than 147 organisms (ranging from humans to protists) and we exploited it to train a universal convolutional neural network called Spliceator [8] (http://www.lbgi.fr/spliceator). The latter has been trained with high quality data from G3PO which allows it to obtain high performances, with an accuracy of 95.3% for the Donor model and 94.9% for the Acceptor model. Different evaluations have also been performed on independent benchmarks [9] of several organisms (human fish, fly, worm and plant) and Spliceator has obtained equivalent or even better performances than the current state of the art programs.

References

1. Salzberg SL. Next-generation genome annotation: we still struggle to get it right. Genome Biology. 2019;20:92.
2. Mudge JM, Harrow J. The state of play in higher eukaryote gene annotation. Nat Rev Genet. 2016;17:758–72.
3. Matera AG, Wang Z. A day in the life of the spliceosome. Nat Rev Mol Cell Biol. 2014;15:108–21.
4. Jaganathan K, Kyriazopoulou Panagiotopoulou S, McRae JF, Darbandi SF, Knowles D, Li YI, et al. Predicting Splicing from Primary Sequence with Deep Learning. Cell. 2019;176:535-548.e24.
5. Wang R, Wang Z, Wang J, Li S. SpliceFinder: ab initio prediction of splice sites using convolutional neural network. BMC Bioinformatics. 2019;20:652.
6. Drăgan M-A, Moghul I, Priyam A, Bustos C, Wurm Y. GeneValidator: identify problems with protein-coding gene predictions. Bioinformatics. 2016;32:1559–61.
7. Scalzitti N, Jeannin-Girardon A, Collet P, Poch O, Thompson JD. A benchmark study of ab initio gene prediction methods in diverse eukaryotic organisms. BMC Genomics. 2020;21:293.
8. Scalzitti N, Kress A, Orhand R, Weber T, Moulinier L, Jeannin-Girardon A, Collet P, Poch O, Thompson JD. Spliceator: multi-species splice site prediction using convolutional neural networks under revision (Under revision)
9. Sonnenburg S, Schweikert G, Philips P, Behr J, Rätsch G. Accurate splice site prediction using support vector machines. BMC Bioinformatics. 2007;8:S7.

MISTIC: A prediction tool to reveal disease-relevant deleterious missense variants

Thomas WEBER, Kirsley CHENNEN, Xavière LORNAGE, Arnaud KRESS, Johann BOHM, Julie THOMPSON, Olivier POCH and Jocelyn LAPORTE
ComplexSystemsandTranslationalBioinformatics(CSTB),ICubelaboratory-CNRS,Fe ́de ́rationde Me ́decine Translationnelle de Strasbourg (FMTS), University of Strasbourg, Strasbourg, France,
thomas.weber@unistra.fr

The advent of high-throughput sequencing methods has made it possible to study these Mendelian diseases at the highest resolution available, the nucleotide level. A patient's genome can now be fully sequenced in order to study its genetic variations1. However, the technical limitations today lie in the exploitation of the data produced, particularly due to its volume and complexity.

Approximately five million single nucleotide variations (SNV)2 are present in each human genome resulting from both evolution and inter-individual diversity. Among these, 150,000 are found in the protein-coding regions, called exome, with an impact on gene products close to zero. Nevertheless, in rare cases, SNVs can lead to functional or structural modifications that result in Mendelian disease.

To date, the most studied SNV is the class of missense variations. A missense corresponds to a SNV leading to a change of amino acid in the peptide sequence product during translation. Unlike most damaging classes, which can for example, result in a stop codon, missense SNVs are difficult to study due to their variable consequences. Out of the 150,000 SNVs present in each human exome, the number of missense variations is estimated to be around 1,500 and finding the causative variation for a rare disease in an exome study is like searching for a needle in a haystack. Prediction of the impact of a missense is currently based on multiple parameters: frequency in the general population (large genomic databases), conservation during evolution, physico-chemical properties of the reference and the new amino acid, location in the protein (domain).

With the increase of computational power and emergence of artificial intelligence methods, different algorithms have been developed to help both researchers and physicians to find disease-causing variations in clinical studies.

We present MISsense deleTeriousness predICtor (MISTIC), a new original prediction tool based on an original combination of two complementary machine-learning algorithms that integrates 115 features, ranging from multi-ethnic minor allele frequencies and evolutionary conservation, to physiochemical and biochemical properties of amino acids. Our approach also uses training sets with a wide spectrum of variant profiles, including both positive (deleterious) and negative (benign) variants. Compared to recent state-of-the-art ensemble prediction tools in various benchmark tests, MISTIC exhibits the best and most consistent performance, notably with the highest AUC value (0.95). Importantly, MISTIC maintains its high performance in the specific case of discriminating deleterious variants from rare benign variants (allele frequency <1%) or population-specific benign variants (no allele frequency). In a clinical usage context, MISTIC drastically reduces the list of candidate variants (<30%) and has a median ranking of the “causative” deleterious variants among the top 25 variants.Pages must NOT be numbered. Final pagination will be set by the editors of the proceedings.

The list of references is headed *References*, it should be placed at the end of your contribution. It should be in *Times New Roman* 10-point font. Please do not insert a page break before the list of references. For citations in the text, please use square brackets [1] and consecutive ordered numbers [2,3] in list of references. Please find below examples on how to format references corresponding to articles [1], books [2], book chapters and proceedings [3].

Discovery of widespread transcription initiation at microsatellites predictable by sequence-based deep neural network

Mathys Grapotte[1,2,3,11], Manu Saraswat[1,2,11], Chloé Bessière[1,2,11], Christophe Menichelli[1,4], Jordan A. Ramilowski[5], Yoshihide Hayashizaki[5], Yoshihide Hayashizaki[6], Masayoshi Itoh[6], Michihira Tagami[5], Mitsuyoshi Murata[5], Miki Kojima-Ishiyama[5], Shohei Noma[5], Shuhei Noguchi[5], Takeya Kasukawa[5], Akira Hasegawa[5], Harukazu Suzuki[5], Hiromi Nishiyori-Sueki[5], Martin C. Frith[7,8,9], FANTOM consortium, Clément Chatelain[3], Piero Carninci[5], Michiel J.L. de Hoon[5], Wyeth W.Wasserman[10], Laurent Bréhélin[1,4] and Charles-Henri Lecellier[1,2,4]

[1] Institut de Biologie Computationnelle, Montpellier, France
[2] Institut de Génétique Moléculaire de Montpellier, University of Montpellier, CNRS, Montpellier, France
[3] SANOFI R&D, Translational Sciences, Chilly Mazarin, 91385 France
[4] LIRMM, Univ Montpellier, CNRS, Montpellier, France
[5] RIKEN Center for Integrative Medical Sciences, Yokohama, Kanagawa, Japan
[6] RIKEN Preventive Medicine and Diagnosis Innovation Program, Wako, Saitama, Japan
[7] Artificial Intelligence Research Center, AIST, Tokyo 135-0064, Japan
[8] Graduate School of Frontier Sciences, University of Tokyo, Chiba 277-8568, Japan
[9] AIST-Waseda University CBBD-OIL, AIST, Tokyo 169-8555, Japan
[10] Centre for Molecular Medicine and Therapeutics at the Child and Family Research Institute, Department of Medical Genetics, University of British Columbia, Vancouver BC, Canada
[11] contributed equally to this work

Corresponding author: mathys.grapotte@igmm.cnrs.fr

Paper Reference: **Grapotte, Saraswat, Bessière *et al.* Discovery of widespread transcription initiation at microsatellites predictable by sequence-based deep neural network. Nature Communications 2021. *in press.*** https://doi.org/10.1101/2020.07.10.195636

Abstract

Using the Cap Analysis of Gene Expression (CAGE) technology, the FANTOM5 consortium provided one of the most comprehensive maps of Transcription Start Sites (TSSs) in several species [1] . Strikingly, 72% of them could not be assigned to a specific gene and initiate at unconventional regions, outside promoters or enhancers. Here, we probe these unassigned TSSs and show that, in all species studied, a significant fraction of CAGE peaks initiate at microsatellites, also called Short Tandem Repeats (STRs) [2]. To confirm this transcription, we develop Cap Trap RNA-seq, a technology which combines cap trapping and long read MinION sequencing. We train sequence-based deep learning models able to predict CAGE signal at STRs with high accuracy. These models unveil the importance of STR surrounding sequences not only to distinguish STR classes, but also to predict the level of transcription initiation. Importantly, genetic variants linked to human diseases [3] are preferentially found at STRs with high transcription initiation level, supporting the biological and clinical relevance of transcription initiation at STRs. Together, our results extend the repertoire of non-coding transcription associated with DNA tandem repeats and complexify STR polymorphism.

References

[1] Forrest A. R. et al. A promoter-level mammalian expression atlas. *Nature*, 2014.

[2] Gymrek et al. Abundant contribution of short tandem repeats to gene expression variation in humans. *Nature Genetics*, 2016.

[3] Landrum MJ. Lee JM. Benson M. Brown GR. Chao C. Chitipiralla S. Gu B. Hart J. Hoffman D. Jang W. Karapetyan K. Katz K. Liu C. Maddipatla Z. Malheiro A. McDaniel K. Ovetsky M. Riley G. Zhou G. Holmes JB. Kattman BL. Maglott DR. Clinvar: improving access to variant interpretations and supporting evidence. *Nucleic Acids Res.*, 2018.

Processing of Proton Transfer Reaction Time-of Flight Mass Spectrometry (PTR-TOF-MS) data for untargeted biomarker discovery in exhaled breath: application to COVID-19 intubated ventilated patient

Camille Roquencourt[1*], Stanislas Grassin-Delyle[2*], Pierre Moine[3], Gabriel Saffroy[3], Stanislas Carn[3], Nicholas Heming[3], Jérôme Fleuriet[3], Hélène Salvator[2], Emmanuel Naline[2], Louis-Jean Couderc[2], Philippe Devillier[2], Djillali Annane[3], Etienne A. Thévenot[4]

[1]CEA, LIST, Laboratoire Sciences des Données et de la Décision, Gif-sur-Yvette, France
[2]Hôpital Foch, Exhalomics, Département des maladies des voies respiratoires, Suresnes, France
[3]Intensive Care Unit, Raymond Poincaré Hospital, APHP, Garches, France
[4]Département Médicaments et Technologies pour la Santée (DMTS), Université Paris-Saclay, CEA, INRAE, MetaboHUB, Gif-sur-Yvette, France
Corresponding Author: camille.roquencourt@cea.fr,
*contributed equally to this work

Paper Reference: **Grassin Delyle *et al.* (2020) Metabolomics of exhaled breath in critically ill COVID-19 patients: A pilot study, EBioMedicine, 2020, 63. https://doi.org/10.1016/j.ebiom.2020.103154**

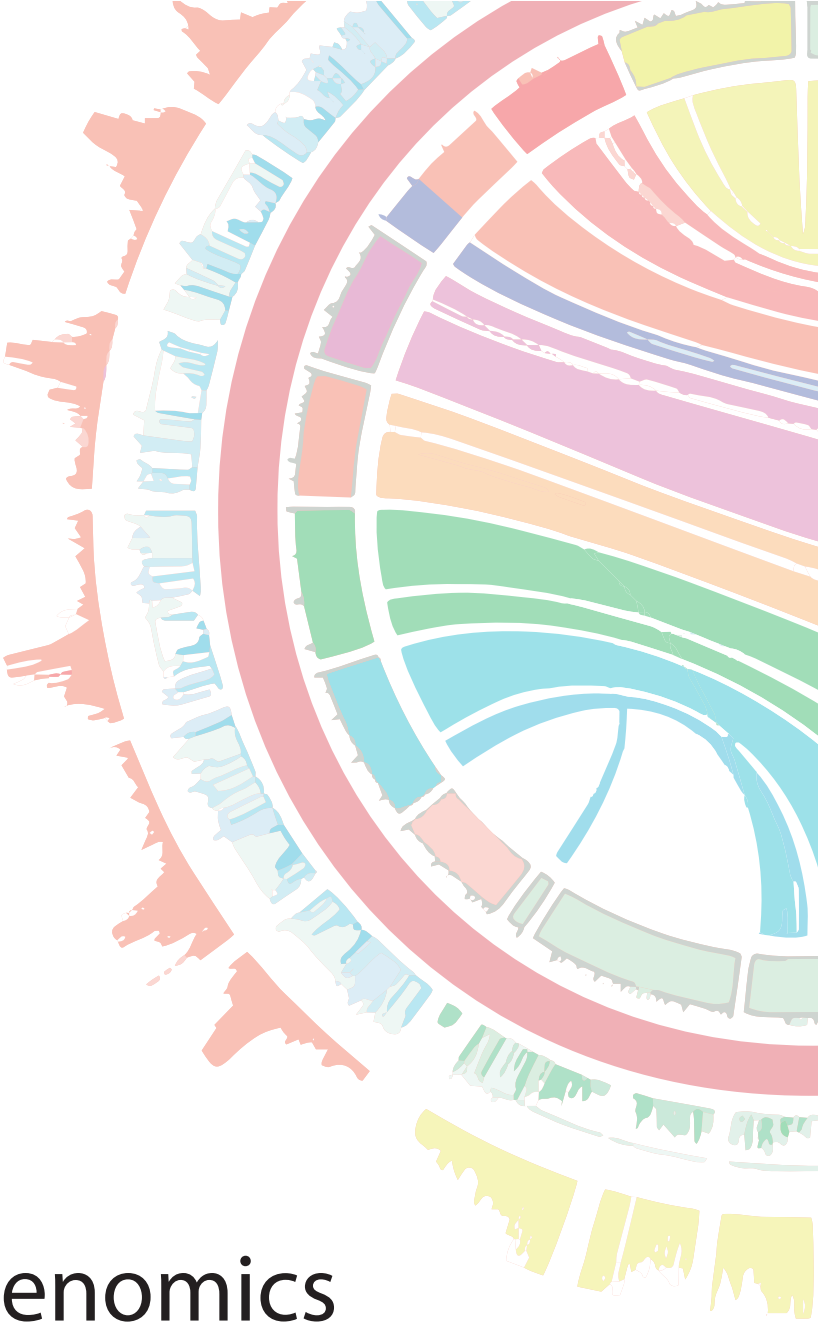
The analysis of Volatile Organic Compound (VOCs) in exhaled air is a promising non-invasive method for early diagnosis and therapeutic monitoring [1]. Proton Transfer Reaction Time-Of-Flight Mass Spectrometry (PTR-TOF-MS) has recently emerged as an innovative technology for the real time analysis of exhaled VOCs [2]. However, there is currently a lack of methods and software tools for the processing of such breath data from cohorts [3].

We therefore developed a suite of algorithms that process the raw data and build the table of feature intensities in all samples, through expiration and peak detection, quantification, alignment between samples, missing value imputation, feature annotation. Notably, we developed an innovative 2D peak deconvolution method based on penalized splines signal regression, which enables efficient denoising and estimation of the temporal evolution, even in the case of peaks with close m/z values. The methods were validated on simulated, experimental (calibration gas containing specific VOCs in known quantities), and clinical data. The software tool is publicly available as the *ptairMS* R package on GitHub (and submitted to Bioconductor) and includes a graphical user interface for interactive visualization and monitoring of the processing of cohort samples.

We applied our methodology to the characterization of exhaled breath from mechanically ventilated adults with COVID-19 infection. Analyzes of exhaled breath from 28 patients with COVID-19 ARDS and 12 patients with non-COVID-19 ARDS were performed daily from the hospital's entry to the discharge. First, using the closest available acquisition to the hospital entry, models predicting the infection status were developed: a high accuracy (93%) was obtained with all three machine learning approaches used (Random forest, Elastic Net and SVM). Second, the longitudinal evolution of each VOC as a function of the hospitalization time was analyzed by mixed-effects modeling [4]. Splines function was used for the fixed effect (infection status), and an intercept per patient for the random effect. After feature ranking [5] and selection, four biomarkers of COVID-19 infection could be identified. Altogether, these results highlight the value of the PTR-TOF-MS data and ***ptairMS*** software for the biomarker discovery in exhaled breath.

References

1. Rattray NJ, et al, Taking your breath away: metabolomics breathes life in to personalized medicine. Trends Biotechnol 2014;32 (10):538–48
2. Hansel A, Jordan A, et al Proton transfer reaction mass spectrometry: on-line trace gas analysis at the ppb level. Int J Mass Spectrom Ion Process 1995;149-150:609–19
3. Muller M, et al, A new software tool for the analysis of high resolution PTR-TOF mass spectra Chemometrics and Intelligent Laboratory Systems 2013:127:158-165
4. Berk M., et al, A statistical framework for biomarker discovery in metabolomics time course data, Bioinformatics, 2011:27: 1979–1985
5. Pihur V, et al . RankAggreg, an R package for weighted rank aggregation. BMC Bioinformatics 2009;10:62



Session 3
Functional Genomics

Dissection of intercellular communication using the transcriptome-based framework ICELLNET

Floriane NOËL[1,2], Lucile MASSENET-REGAD[1], Irit CARMI-LEVY[2], Antonio CAPPUCCIO[2], Maximilien GRANDCLAUDON[2], Coline TRICHOT[1,2], Yann KIEFFER[2], Fatima MECHTA-GRIGORIOU[2] and Vassili SOUMELIS[1,2,3]

[1] Inserm U976-HIPI IRSL, 1 avenue Claude Vellefaux, 75010, Paris, France

[2] Institut Curie, 26 rue d'Ulm, 75005, Paris, France

[3] APHP, AP-HP, Hôpital Saint-Louis, 75010, Paris, France

Corresponding Author: vassili.soumelis@aphp.fr

Paper Reference: **Noël, F., Massenet-Regad, L., Carmi-Levy, I. et al. Dissection of intercellular communication using the transcriptome-based framework ICELLNET. Nat Commun 12, 1089 (2021).**
https://doi.org/10.1038/s41467-021-21244-x

Cell-to-cell communication is at the basis of the higher-order organization observed in tissues, organs, and organisms, at steady state and in response to stress. A sender cell can exchange information by either secreting small biological molecules such as cytokines or chemokines, or expressing surface markers on its membrane. These molecules, called ligands, can bind specific receptors in « target » cells, that will lead to pathway activation and specific target cell response. The availability of large-scale transcriptomic datasets from several cell types has opened the possibility of reconstructing cell-cell interactions based on co-expression of ligand-receptor pairs. Several methods [1,2,3] have recently been publish to decipher cell communication, leading to interesting biological hypotheses. Still, important challenges remain, including the global integration of cell-cell communication, biological interpretation, the inference of communication between cell types not necessarily represented in the same dataset. Thus, we developed ICELLNET, a transcriptomic-based framework to dissect cell communication in a global manner. It integrates: 1) a manually curated database of 543 ligand-receptor interactions taking into account multiple subunits expression and that is exhaustive on cytokines, immune checkpoints and chemokines interactions with their respective receptors, 2) an R package to compute communication scores between cell types in a quantitative and global manner, 3) the possibility to connect multiple cell populations of interest with 31 reference human cell types [4], 4) the implementation of three visualization modes to facilitate biological interpretation. We have applied ICELLNET to dissect the communication of breast cancer-associated fibroblasts with other components of the tumor microenvironment revealing difference of communication channels used by CAF subsets. We also analyzed LPS-activated human dendritic cells (DC) communication and identified autocrine IL-10 as a key molecule controlling DC communication with up to 12 other cell types. Four of them are further tested and experimentally validated. Hence, ICELLNET is a global, versatile, biologically validated, and easy-to-use framework to dissect cell communication from individual or multiple cell-based transcriptomic profiles.

References

1. Vento-Tormo R, et al. Single-cell reconstruction of the early maternal-fetal interface in humans. *Nature.* 2018;563:347–353. doi: 10.1038/s41586-018-0698-6.
2. Browaeys R, Saelens W, Saeys Y. NicheNet: modeling intercellular communication by linking ligands to target genes. *Nat. Methods.* 2019 doi: 10.1038/s41592-019-0667-5.
3. Jin S, et al. Inference and analysis of cell-cell communication using CellChat. Preprint at. *bioRxiv.* 2020 doi: 10.1101/2020.07.21.214387.
4. Wu C, et al. BioGPS: an extensible and customizable portal for querying and organizing gene annotation resources. *Genome Biol.* 2009;10:R130. doi: 10.1186/gb-2009-10-11-r130.

Topoisomerase I prevents transcription-replication conflicts at transcription termination sites

Yaqun LIU [1], Alexy PROMONET [2], Ismaël PADIOLEAU [2], Yea-Lih LIN [2], Philippe PASERO [2] and Chunlong CHEN [1]

[1] Institut Curie, Univ PSL/Sorbonne, CNRS UMR3244, 26 rue d'Ulm, 75005, Paris, France
[2] Institut de Génétique Humaine, 141 Rue de la Cardonille, 34396 Montpellier, France

Corresponding Author: chunlong.chen@curie.fr

Paper Reference: **Promonet *et al.* (2020), Topoisomerase 1 prevents replication stress at R-loop-enriched transcription termination sites. *Nat Commun* 11, 3940. https://doi.org/10.1038/s41467-020-17858-2**

The transcription and replication machineries share the same DNA template, which render head-on (HO) or co-directional (CD) collisions between them inevitable. HO collisions are considered as more deleterious to the genomic stability [1]. Moreover, transcription-replication conflicts (TRCs) can also be caused by the three-stranded nucleic acid structures called R-loops, containing a RNA:DNA hybrid and a displaced DNA strand. R-loops are formed co-transcriptionally when the nascent RNA reanneals with the template DNA strand, leaving the non-coding strand unpaired. R-loops have been proposed to play both positive and negative roles in gene expression and other chromosome functions [1]. However, the mechanism by which R-loops interfere with fork progression and promote genomic instability in human cells remains poorly understood.

In order to study the direction of replication fork movement and TRCs, we developed a R-based bioinformatics toolkit OKseqHMM, by using Hidden Markov Model (HMM) algorithm, to direct measure the replication fork directionality (RFD) as well as replication initiation and termination, along genomes obtained by sequencing of Okazaki fragments (OK-Seq) [2]. Furthermore, we have gathered and analyzed OK-seq data of different human and mouse cell types, to generate high-quality RFD profiles and initiation zones and termination zones (all tool and data are available at https://github.com/CL-CHEN-Lab/OK-Seq). By combining OK-seq with the mappings of RNA:DNA hybrids (DRIP-seq), replication protein A32 subunit phosphorylated on S33 (p-RPA), phosphorylation of histone variant H2AX on S139 (γ-H2AX) and DNA double-strand breaks (DSBs, by i-BLESS; double strand Breaks Labelling, Enrichment on Streptavidin and next-generation Sequencing), we found that although R-loops are enriched at both transcription start site (TSS) and transcription termination site (TTS) of highly expressed genes, p-RPA was only detected at TTS, where forks mostly progress in a HO orientation relative to the direction of transcription. In topoisomerase I (TOP1)-deficient cells, we also observed a broad γ-H2AX signal at active genes and the presence of DSBs at TTS enriched in R-loops and p-RPA. Since p-RPA is a mark of ATM-Rad3-related (ATR) pathway activation at paused forks and γ-H2AX is a mark of collapsed forks and DSBs, these data indicate that forks transiently pause at TTS but do not break, whereas prolonged fork pausing and DSBs occur in TOP1-deficient cells, presumably because of unresolved torsional stress. The impact of R-loops in this process was further confirmed by the overexpression of RNase H1, which partially alleviated replication stress in TOP1-deficient cells [3].

Altogether, these results provide a global picture of how the functional organization of the human genome limits the deleterious consequences of fork collisions with transcription and R-loops. In this model, the preferential co-directional orientation of replication and transcription at highly expressed genes and the controlled pausing of replication forks at TTS are both important to prevent HO collisions. The molecular mechanisms ensuring stable fork pausing and restart at TTS is currently unclear, but it may require a tight control of DNA torsional stress as it is perturbed in TOP1-deficient cells. The activation of the ATR pathway, may also actively slow down fork progression to prevent further head-on collisions and maintain genome integrity in TOP1-deficient cells [3].

References

1. Hamperl, S. *et al*. Transcription-Replication Conflict Orientation Modulates R-Loop Levels and Activates Distinct DNA Damage Responses. *Cell* 170, 774-786.e19 (2017).
2. Petryk, N. *et al*. Replication landscape of the human genome. *Nat. Commun.* 7, 10208 (2016).
3. Promonet, A. *et al*. Topoisomerase 1 prevents replication stress at R-loop-enriched transcription termination sites. *Nat. Commun.* 11, 3940 (2020).

Automated Quality Control of NGS Data using Machine Learning

Steffen ALBRECHT[1], Maximilian SPRANG[1], Miguel A. ANDRADE-NAVARRO[1] and Jean-Fred FONTAINE[1]

[1] Johannes Gutenberg-Universität Mainz, Biozentrum I, Hans-Dieter-Hüsch-Weg 15, 55128, Mainz, Germany

Corresponding Author: fontaine@uni-mainz.de

Paper Reference: **Albrecht *et al.* (2021) seqQscorer: automated quality control of next-generation sequencing data using machine learning, Genome Biology, 2021, 22:75. https://doi.org/10.1186/s13059-021-02294-2**

1. Abstract

The versatility and power of next-generation sequencing (NGS) applications make the sequencing technology a popular tool in biology and medicine. Yet, the complexity to evaluate data quality leads to non-optimal results. Quality control (QC) of the data is of crucial importance to filter out low-quality data files that would have a negative impact on downstream analyses. In a clinical context, patient data of unnoticed low-quality can lead to wrong diagnosis or ill-suited treatment. Filtering out or editing a small portion of sequencing reads within a file or applying more sophisticated bias mitigation methods may be not enough or detrimental to the downstream analysis [1, 2]. Common QC tools analyze the data files to derive numerous highly specific quality features for manual review. As the usefulness of many features was never demonstrated, a large majority of NGS scientists is still not confident about classifying a sequencing file by quality.

To address this problem, we have statistically characterized 47 common NGS quality features and tested 10 classification algorithms, including tree-based and deep learning algorithms. The training set was composed of 2642 human and mouse functional genomics NGS files related to RNA-seq, ChIP-seq and DNase-seq, single- and pair-ended, from the ENCODE database (FastQ files: 5.6 TB). Predictive models were tuned by a comprehensive grid search. In combination with parameter settings specific to each algorithm, a total of 19,417 different models were trained and evaluated within the grid search for each classification case such as human single-end ChIP-seq or mouse paired-end DNase-seq. Furthermore, for each parameter setting, we applied three different feature selection methods prior to the classification. External validations were performed on 700 FastQ files from 38 datasets referenced in the GEO or Cistrome databases [3].

Results show that NGS quality features highly depends on assays and experimental conditions. We were able to build unbiased optimal models to accurately predict the quality of NGS data files. Tree-based algorithms such as random forest and gradient boosting generated the most accurate models during the grid search. A generic model trained on data from any species and assay performed similarly to models specialized by species and assays during internal tests (average area under ROC curve = 0.925). Generalization of this generic model and some specialized models was confirmed with external data, including ATAC-seq data not used for training. Relevance of the models in clinical applications was demonstrated using 6 external datasets for which the automatic identification and filtering of low-quality samples resulted in improved clustering of disease and control samples, with potential positive impact to derive marker genes and disease classifiers. Provided the limited usefulness of publicly available guidelines to categorize data files with respect to quality, our derived statistical guidelines and predictive models represent a valuable resource for users of NGS data to better understand quality issues and perform automatic quality control. We strongly encourage researchers to share both high- and low-quality data with the community. Availability: https://github.com/salbrec/seqQscorer.

References

1. Meyer CA, Liu XS. Identifying and mitigating bias in next-generation sequencing methods for chromatin biology. Nat Rev Genet. 2014;15:709–21.
2. Williams CR, Baccarella A, Parrish JZ, Kim CC. Trimming of sequence reads alters RNA-Seq gene expression estimates. BMC Bioinformatics. 2016;17:103.
3. Mei S, Qin Q, Wu Q, Sun H, Zheng R, Zang C, Zhu M, Wu J, Shi X, Taing L, et al. Cistrome data browser: a data portal for ChIP-Seq and chromatin accessibility data in human and mouse. Nucleic Acids Res. 2017;45:D658–62.

PenDA, a rank-based method for personalized differential analysis: application to lung cancer

Clémentine DECAMPS[1], Magali RICHARD[1], and Daniel JOST[1,2]

[1] TIMC-IMAG, UMR 5525, Univ. Grenoble Alpes, 38700, Grenoble, France
[2] LBMC, UMR5239, ENS de Lyon, 46 Allée d'Italie, 69007, Lyon, France

Corresponding Author: clementine.decamps@univ-grenoble-alpes.fr

Paper Reference: **Richard *et al.* (2020) PenDA, a rank-based method for personalized differential analysis: Application to lung cancer, PLoS Computational Biology, 2020.**
https://doi.org/10.1371/journal.pcbi.1007869

1 Background

A current goal in medicine is to achieve precision and *personalized medicine*: the genetic, genomic, and molecular information of each patient would be integrated to develop personalized diagnosis and treatment [1]. This is particularly useful in cancer, where each individual tumor may be viewed as an independent disease, with specific and variable responses to generic therapeutic treatments. Such challenging perspectives will be only possible with the development of efficient and robust methodological tools that allow the identifications of deregulation patterns at the individual level.

Many statistical or bioinformatic methods do already exist to identify deregulated genes at the population level, like DESeq2 [2] or edgeR [3]. Although very effective in detecting typical deregulation patterns, these methods are not designed to provide precise information at the individual level and are usually very sensitive to batch effects. Few promising techniques already allow to extract interpretable information from personalized omics data [4], but they show very high false discovery rates or need matched samples.

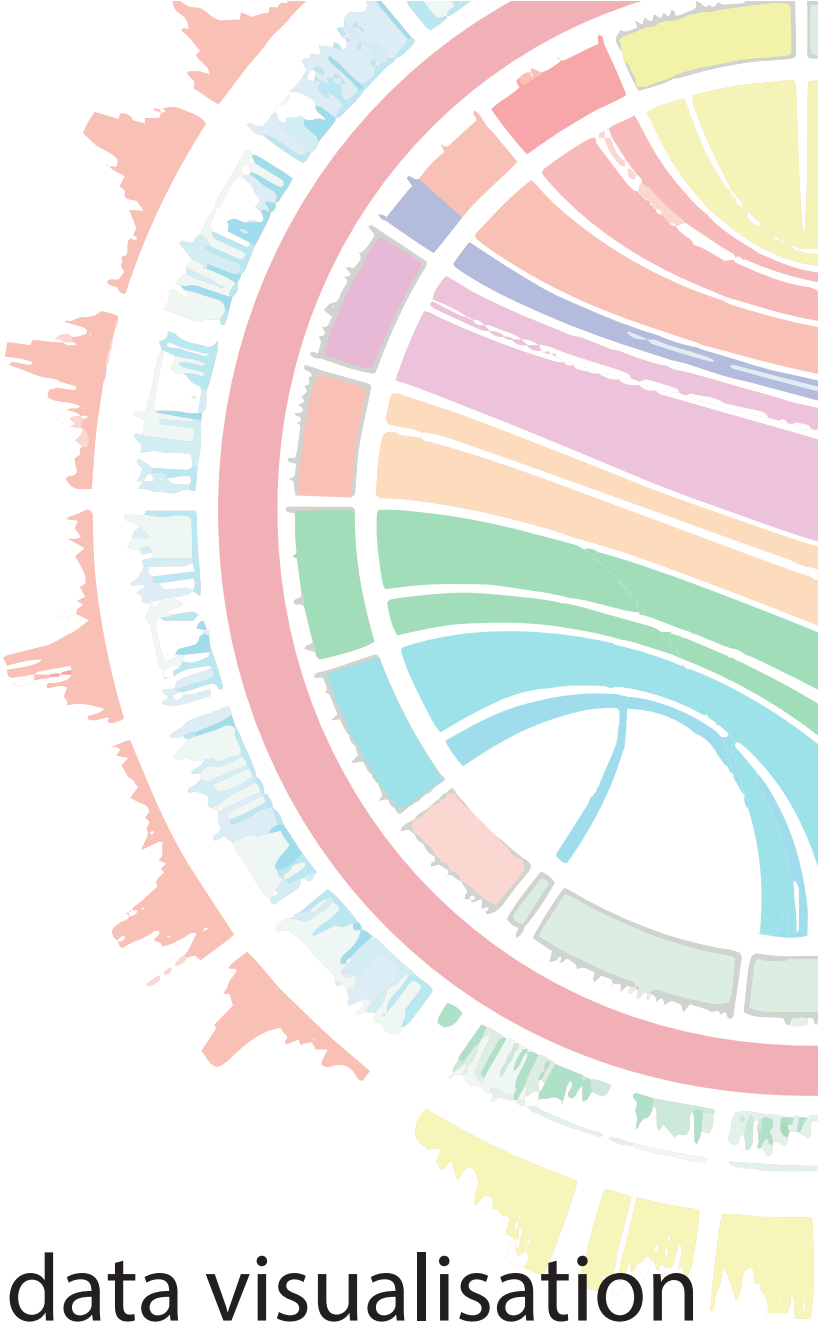
2 The PenDA method

To overcome these limitations, we developed PenDA, for Personalized Differential Analysis, a *rank-based* method, *robust* to batch and normalization effects. The method works in two steps: first it uses information extracted from a reference dataset (e.g., control – non-tumorous – data) to determine a relative ordering for each gene. Then, it uses this ordering to infer the deregulation status of genes in each individual sample of interest (e.g., tumors samples).

Based on a realistic benchmark of simulated tumors, we demonstrated that PenDA reaches very high efficiency in detecting sample-specific deregulated genes. We then applied the method to two large cohorts associated with lung cancer. A detailed statistical analysis of the results allowed to isolate genes with specific deregulation patterns, like genes that are up-regulated in all tumors or genes that are expressed but never deregulated in any tumors. In particular, we were able to identified 37 new biomarkers associated to a bad prognosis, that we validated on two independent cohorts.

References

[1] Yi-Fan Lu *et al.*. Personalized Medicine and Human Genetic Diversity. *Cold Spring Harbor Perspectives in Medicine*, 4:a008581-1, 2014.
[2] Michael I Love *et al.*. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol.* 15:550, 2014.
[3] Mark D Robinson *et al.*. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*. 26:139-40, 2010.
[4] Francesca Vitali *et al*.. Developing a "personalome" for precision medicine: emerging methods that compute interpretable effect sizes from single-subject transcriptomes. *Briefings in Bioinformatics*, 63:2889, 2017.



> Session 4
Databases & data visualisation

CALR-ETdb, the database of calreticulin variants diversity in essential thrombocythemia

Nora EL JAHRANI[1], Gabriel CRETIN[1] and Alexandre G. DE BREVERN[1]

[1] Université de Paris, INSERM UMR_S 1134, Université de la Réunion, INTS, Laboratoire d'Excellence GR-Ex, 6, rue Alexandre Cabanel, 75015, Paris, France

Corresponding Author: alexandre.debrevern@u-paris.fr

Paper Reference: **El Jahrani *et al.* (2021) CALR-ETdb, the database of calreticulin variants diversity in essential thrombocythemia, Platelets, in press. https://doi.org/10.1080/09537104.2020.1869712**

Essential thrombocythemia (ET) is a blood cancer belonging to the Myeloproliferative Neoplasms (MPNs) family. ET is characterized by an increase in the production of blood platelets. The high level of platelets associated with ET lead to complications such as thrombosis, clots (agglutination of platelets), which could partially or totally obstruct a blood vessel, or even haemorrhages. This disease is uncommon, with 2.3 cases per 100,000 people each year [1]. Several mutations are associated with ET. The calreticulin (CALR) gene corresponds to 25-30% of patients suffering from ET; it encodes an endoplasmic reticulum (ER)-localized molecular chaperone [2]. CALR protein ends with a highly flexible domain that (i) binds to a large proportion of Ca^{2+} of the ER and (ii) is ended by the ER retention peptide. A novel carboxyl-terminal sequence is generated by a frameshift mutation in CALR implied in ET (named CALR-ET), losing the ER retention peptide. CALR-ET therefore tends to go out of the ER. CALR-ETs mediate intermolecular interactions to form homodimers, bind MPL and activates it, leading to ET phenotype.

The two most common CALR mutations are type 1 and type 2 mutations characterized by a 52 base pairs (bp) deletion (c.1099_1150del; p.Leu367fs*46) and a 5 bp insertion (c.1154_1155insTTGTC; p.Lys385fs*47), respectively. Other mutations are classified as type 1-like or type 2-like according to their impact on the carboxy-terminal sequence of the protein, the remaining are associated to the 'other' type. Type 1 mutations are more frequent, accounting for about 50% of CALR-mutated cases in ET, and are associated with a better prognosis. CALR type 1-like patients appear to have a more complex molecular landscape; the allele burden increase of CALR mutations is associated with disease progression.

In this study, we have compiled variants taken from COSMIC database and literature leading to 155 different variants. This large number of variants allowed redefining 5 new classes extending the classification of type 1-like and type 2-like to a finer description. These analyses showed that last class, named E, corresponding to more than 10% of CALR variants seemed not attached to ET. Similarly, the new class B takes only 2/3 of the variants formerly associated with type 2-like, which has a significant impact, this classification being used to follow the prognosis of patients.

All the compiled and refined information had been included into a freely dedicated database CALR-ETdb (https://www.dsimb.inserm.fr/CALR-ET). It provides information for the variants with their general information (nucleic acid notation, protein, COSMIC code, type of mutation, type, class, category, ...). The structural information generated is also presented (prediction of the secondary structure, and 3D modeling), with the possibility of downloading the sequence in FASTA format and the 3D model in PDB format. The references of the variant are also indicated. The search for variants is possible by their COSMIC code, type of mutation, type, class or by nucleic or protein sequence. The class and type of the sequence are given in case no variant is found in our database. Statistical data describing the database is also available. They allow visualization of the specificities of the variants at the level of the protein or nucleotide sequence, or else according to their type or class. Database is updated as soon as a new sequence is detected.

References

1. Arie Regev, P Stark , Dorit Blickstein, Meir Lahav. Thrombotic complications in essential thrombocythemia with relatively low platelet counts. *Am J Hematol*, 56:168-172, 1997.
2. Joan How, Gabriella S Hobb, Ann Mullally. Mutant calreticulin in myeloproliferative neoplasms. *Blood*, 134:2242-2248, 2019.

Genoscapist: online exploration of quantitative profiles along genomes via interactively customized graphical representations

Sandra Dérozier[1,2], Pierre Nicolas[1], Ulrike Mäder[3] and Cyprien Guérin[1]

[1] Université Paris-Saclay, INRAE, MaIAGE, 78350, Jouy-en-Josas, France

[2] Université Paris-Saclay, INRAE, BioinfOmics, MIGALE bioinformatics facility, 78350, Jouy-en-Josas, France

[3] Interfaculty Institute for Genetics and Functional Genomics, University Medicine Greifswald, Germany

Corresponding Author: sandra.derozier@inrae.fr

Paper Reference: **Dérozier *et al.* (2021) Genoscapist: online exploration of quantitative profiles along genomes via interactively customized graphical representations, Bioinformatics, 2021.**
https://doi.org/10.1093/bioinformatics/btab079

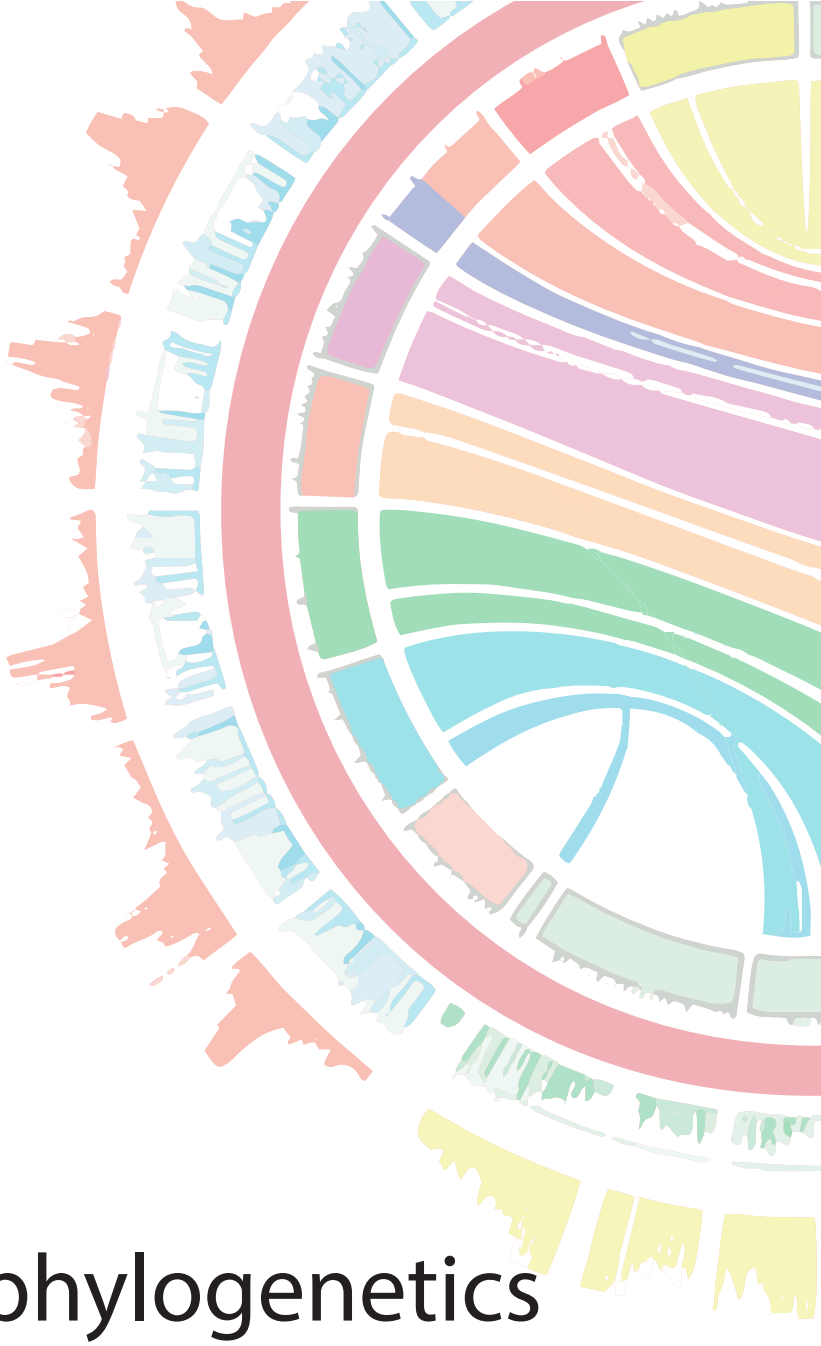
Analysis of quantitative genome-wide data, and particularly transcriptome data [1], allows to constantly improve structural and functional annotation of bacterial genomes. Examination of non-aggregated condition-dependent or strain-dependent transcription profiles along genomes is essential for delineation of transcription units and characterization of new genes such as antisense and other non-coding RNAs. Many genome browsers have been developed such as IGV [2] and JBrowse [3]. Some allow visualization of multiple profiles stacked on one track but do not combine interactive profile selection and colouring with the easy access of an online tool, as useful to efficiently browse large sets of profiles.

Here we present Genoscapist that answers this specific need. To have maximum freedom for the development and integration of graphical representation and browsing, we developed Genoscapist as an independent application instead of a plugin to another preexisting tool. Genoscapist runs on an Apache webserver. It is written in HTML5/Javascript (client-side) and Python with Flask web framework (server-side). Using AJAX reduces response time by minimizing data transfers, by sending simultaneous server requests, and by taking advantage of the processing capability of the clients. The graphical rendering follows the Scalable Vector Graphics (SVG) Web standard. Genome annotations and quantitative profiles are stored in a PostgreSQL database.

Genoscapist views provide an integrated framework to interactively explore datasets by navigating along relationships in the expression space and along the genome sequence. To demonstrate its relevance, we deployed Genoscapist (http://genoscapist.migale.inrae.fr/) on data from transcriptome-based reannotation studies of *Bacillus subtilis* [4] and *Staphylococcus aureus* [5] since customizable views of these condition-dependent transcription profiles was a demand of the respective scientific communities. As illustrated on these data sets, the tool provides an intuitive and powerful interface to select relevant profiles, set their associated colors, change parameters of the graphical representation (zoom in/out, normalization method of the profiles, display/hide gene names, …). A particular attention was paid to features directly useful for scientific communication : links can be obtained to share (or save) customized views and exported SVG files can serve as a basis to prepare high-quality figures.

References

1. Hör, J. et al. Bacterial RNA Biology on a Genome Scale. *Mol. Cell*, 70, 785-799, 2018.
2. James, T. et al. Integrative Genomics Viewer. *Nat. Biotechnol.*, 29, 24-26, 2011.
3. Buels, R. et al. JBrowse: a dynamic web platform for genome visualization and analysis. *Genome Biol.*, 17, 66, 2016.
4. Nicolas, P. et al. Condition-dependent transcriptome reveals high-level regulatory architecture in Bacillus subtilis. *Science*, 335, 1103-6, 2012.
5. Mäder, U. et al. Staphylococcus aureus transcriptome architecture: From laboratory to infection-mimicking conditions. *PLOS Genet.*, 12, e1005962I, 2016.



Session 5
Evolution & phylogenetics

AvP: a software package for automatic phylogenetic detection of candidate horizontal gene transfers.

Georgios D. Koutsovoulos[1], Solène Granjeon Noriot[1], Marc Bailly-Bechet[1], Etienne G. J. Danchin[1] and Corinne Rancurel[1]
[1] Université Côte d'Azur, INRAE, CNRS, Institut Sophia Agrobiotech, 400 Route des Chappes, 06903, Sophia Antipolis, France

Corresponding author: gdkoutsovoulos@gmail.com

Abstract *Horizontal gene transfer (HGT) is the transfer of genes between species outside the transmission from parent to offspring. Due to their impact on the genome and biology of various species, HGTs have gained broader attention, but high-throughput methods to robustly identify them are lacking. One rapid method to identify HGT candidates is to calculate the difference in similarity between the most similar gene in closely related species and the most similar gene in distantly related species. Although metrics on similarity associated with taxonomic information can rapidly detect putative HGTs, these methods are hampered by false positives that are difficult to track. Furthermore, they do not inform on the evolutionary trajectory and events such as duplications. Hence, phylogenetic analysis is necessary to confirm HGT candidates and provide a more comprehensive view of their origin and evolutionary history. However, phylogenetic reconstruction requires several time-consuming manual steps to retrieve the homologous sequences, produce a multiple alignment, construct the phylogeny and analyze the topology to assess whether it supports the HGT hypothesis. Here, we present AvP which automatically performs all these steps and detects candidate HGTs within a phylogenetic framework.*

Keywords bioinformatics, phylogenetics, horizontal gene transfer

1 Introduction

The acquisition of genes through horizontal gene transfer (HGT) is mostly observed in prokaryotes, where they play a significant role in adaptation to new environments (e.g. antibiotic resistance). To a lesser degree, cases of HGT have also been observed in eukaryotes with important consequences in the biology of the organism [1]. The increase of new genomes being sequenced and the prediction of new gene sets, represents an opportunity to detect additional HGT cases and to characterize more precisely the possible donors. To sustain these needs, high-throughput yet robust HGT detection methods are required.

One method to predict potential HGTs is to calculate the difference in similarity using BLAST [2] between closely related and distant species. The Alien Index (AI) metric uses the difference in e-value between the best hit from closely (ingroup) and distantly (donor) related taxa [3]. Positive AI means that the gene is more similar to a distant taxon and indicates a potential HGT. In the past, different values of AI have been used as a cutoff to decrease false positives but with the potential of missing HGTs. Similarly, the HGT Index (h) [4] uses the difference in bit scores but is hampered by the same limitations in terms of a trade-off between reducing false positives without missing valid cases. Furthermore, tracking these false positives from homology search results alone is not possible.

Even if different cutoffs are applied to AI, the underlying best BLAST hit analysis is an oversimplistic method for the evolutionary complexity of HGT. A more robust method is to extract the results from the BLAST analysis and infer a phylogenetic tree. The phylogenetic position of the potential HGT candidate in relation to the other genes and their taxonomy will provide an evolutionary framework and will validate or reject the HGT hypothesis. However, manually producing then checking each phylogenetic tree is a labour-intensive and time-consuming process. In addition, contamination or symbionts in genome sequencing, unless handled properly, can provide false positives that pass both AI and phylogenetic analysis [5]. External information, such as the target gene structure, taxonomic affiliation of genes near the target gene, and support by transcription data are necessary to eliminate

such false positives. Combining all information will lead to a more accurate prediction of putative HGTs.

Methods exist to reconcile gene tree-species tree to detect xenologs (i.e HGTs) as well as duplication events and gene loss [6,7]. These methods are able to distinguish genes that were transferred horizontaly with or without duplication events. To achieve this, a species tree together with the gene tree is required. Therefore, testing hundred of genes produces an extra overhead of either creating different species trees according to the input sequences or compare everything against the whole NCBI tree of life wth hundreds of thousands of branches.

In this study, we present *AvP* (short for 'Alienness vs Predictor') to automate the robust identification of HGTs at high-throughput. *AvP* extracts all the information needed to produce input files to perform phylogenetic reconstruction, evaluate HGTs from the phylogenetic trees, and combine multiple other external information for additional support (e.g. gff3 annotation file, transcript quantification file). Our method does not rely on an explicit reference species tree and only uses a simplified take on the species phylogeny, according to the organism tested. This allows for a rapid phylogenetic detection of HGTs that can then be used as input for more sophisticated analyses.

2 Software description

AvP performs automatic detection of HGT candidates within a phylogenetic framework. The pipeline comprises two major steps: (i) *prepare*, and (ii) *detect*, and three optional steps: (iii) *classify*, (iv) *evaluate*, and (v) *hgt_local_score* (Fig. 1).

2.1 Input files

AvP requires three primary files, (i) a fasta file containing the proteins of the species being studied, (ii) a tabular results file of similarity search (e.g. BLAST or DIAMOND [8]) against a protein database, and (iii) an AI features file. Furthermore, the user must provide two config files, one with information on the taxonomic ingroup in the study (defining which group of species is considered closely related and which group is distantly related) and one defining multiple software parameters. If the database is NR, the AI features file can be created with the Alienness webserver [9]. Using a different database requires providing taxonomic information linked to the database with this step being detailed in the software documentation. Then, the AI features file can be created with the script *calculate_ai.py* which can be found in the repository.

2.2 *AvP prepare*

The software collects all protein sequences corresponding to significant hits from the database based on the tabular results file of the homology search and groups the query species sequences based on the percentage of shared hits (by default 70%) using single linkage clustering. Alternatively, the user can specify a file containing user-generated groups of queries and hits (e.g. from OrthoFinder [10] or protein domain analysis). For each group, a fasta file is created containing the query species sequences and their respective database hits. Each file is then aligned using MAFFT [11] with an option for alignment trimming with trimAl [12].

2.3 *AvP detect*

There are two options available for phylogenetic inference within *AvP*: (i) FastTree [13], and (ii) IQ-TREE [14]. The defaults for these programs are [-gamma -lg] for FastTree and [-mset WAG,LG,JTT -AICc -mrate E,I,G,R] for IQ-TREE. The user can change the IQ-TREE parameters in the config file. These two approaches vary in time and compute requirements, and consequently in tree reconstruction accuracy [15]. Alternatively, the software can utilise user-generated phylogenetic trees using the alignment files created with *AvP prepare* with any program that can produce a valid Newick tree format file. By default, *AvP* does not take into account branch support values. However, the user can define a support threshold in the config file under which branches collapse into polytomies.

Each phylogenetic tree is then processed (midpoint rooting) and each query sequence is classified into one of the following three categories: HGT candidate (✓), Complex topology (?), No evidence

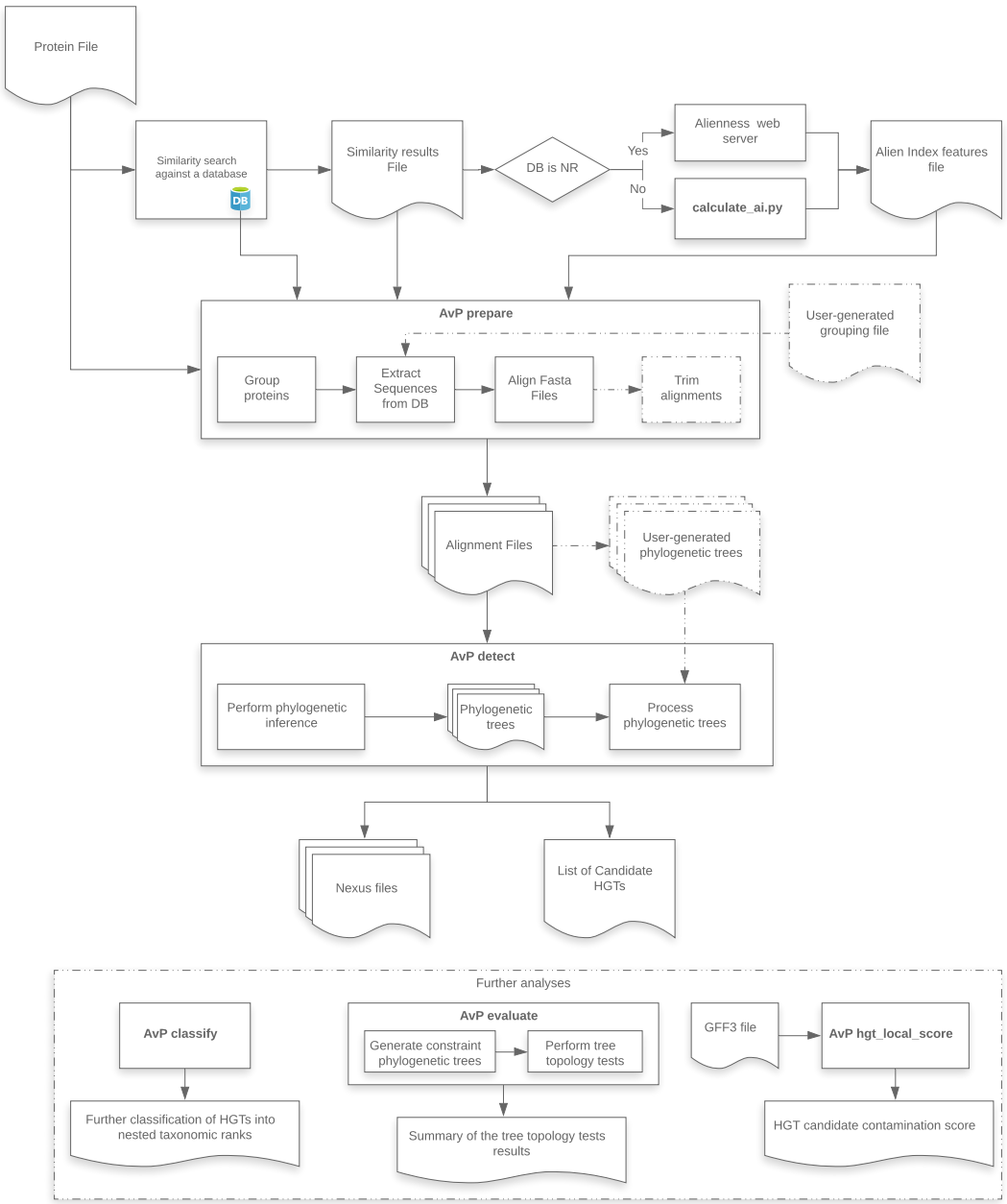


Fig. 1. *AvP* workflow. Dashed lines indicate optional routes and analyses.

for HGT (X). The taxonomic assignment of genes and their position in the tree relative to the query gene are used to characterise the gene as HGT or not. Two branches are taken into account, the sister branch of the gene of interest and the ancestral sister branch (Fig. 2). Both of these branches are tagged independently depending on the included sequences to either Donor (i.e distantly related species), Ingroup (i.e closely related species), or both. Ingroup is defined by the user and Donor is all not in Ingroup. The Ingroup tag is applied if most of the sequences (default 80%) belong to taxa inside the taxonomic group closely related to the species studied. Consequently, the Donor tag is applied if most of the sequences belong to taxa that fall outside of the Ingroup taxonomic clade. If the branch contains taxa from both groups at a ratio higher than 1 to 5, then the branch is tagged as both. The tags of these two branches are then processed according to Tab. 1. For example, if we are searching in a eukaryotic species for HGT originating from prokaryotic species, the Ingroup is set to Eukaryota and the Donor to non Eukaryota (bacteria, viruses etc). If the sister branch of the query contains sequences that belong to eukaryotic species, it is tagged as Ingroup and the gene is not considered as an HGT. In another example, if both the sister branch and the ancestral sister branch contain sequences from non eukaryotic species, both of the branches are tagged as Donor ant the gene is characterised as a potential HGT.

For each query sequence, the software produces a nexus formatted file containing the phylogenetic tree, the taxonomic information for each sequence, and each sequence coloured by the taxonomic affiliation for quick visual parsing. The nexus file can be visualised with the tree visualisation software FigTree [16].

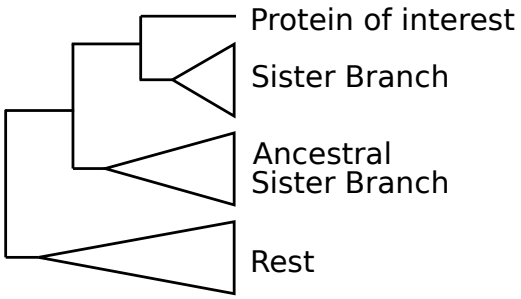


Fig. 2. Sister branch positions on the phylogenetic tree.

	Sister branch (SB)		
Ancestral SB	Donor	Ingroup	Donor + Ingroup
Donor	✓	X	?
Ingroup	?	X	X
Donor + Ingroup	?	X	?
Not present	✓	X	?

Tab. 1. Detection table whether the gene tested is an HGT candidate (described in section *AvP detect*)

2.4 *AvP classify*

This step allows the further classification of HGT candidates into user-generated nested taxonomic ranks for their putative origins. It follows the same logic as in the step *AvP detect* described previously in terms of tagging the clades to a specific taxonomic affiliation. For example, the HGTs can be classified based on their origin, such as Fungi, Viridiplantae, Viruses etc., according to the NCBI taxonomy.

2.5 *AvP evaluate*

For each HGT candidate, the topology is constrained to form a single monophyletic group containing the query sequence and all the Ingroup sequences. A phylogenetic tree is inferred with FastTree or IQ-TREE and the likelihoods of the initial and constrained topologies are compared with IQ-TREE, which supports several tree topology tests. This step can inform whether the topology supporting HGT is more likely than the alternative constrained topology that does not support HGT.

2.6 *AvP hgt_local_score*

Given a gff3 file containing the genomic location of the genes of the query species and the results of the *AvP* analyses, this step calculates a score for each HGT candidate that corresponds to whether the HGT candidate is surrounded by genes from the query genome or 'alien' genes, including possible contaminants. The score ranges between -1 and +1, with -1 indicating strongly a contamination while +1 indicating strongly a HGT candidate (Fig. 3). The rationale is that a candidate HGT surrounded by genes that were also detected as candidate HGT might be part of a contaminant insertion in the genome assembly (although HGT of a whole block of genes or duplications after acquisition are also possible). Hence, this step allows alerting the user on possible contaminations. On the opposite, if the candidate HGT is surrounded by genes that were more likely inherited vertically, the contamination hypothesis can be reasonably ruled out.

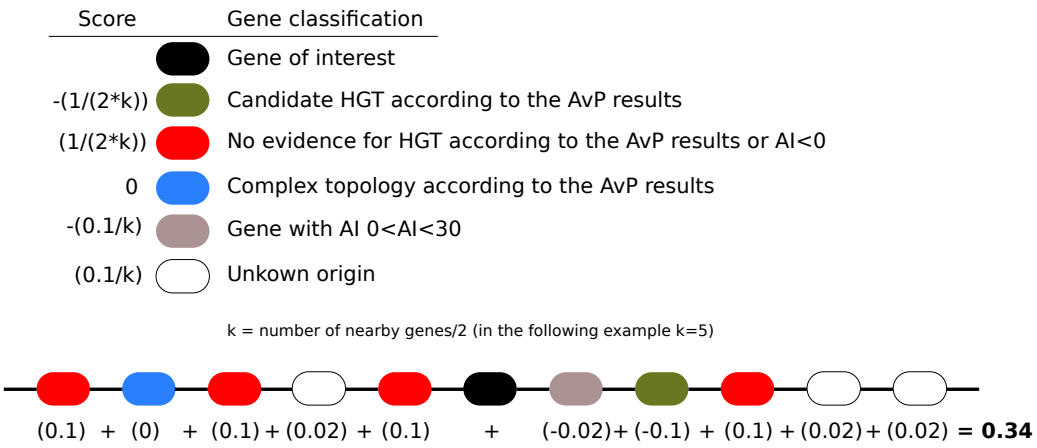


Fig. 3. Calculation of the hgt local score based on nearby genes. Each neighbouring gene contributes to the score based on its classification getting a value described in the top left panel. In the example, the score is equal to 0.34, most likely indicating an HGT insertion. Overall, a score above 0 indicates an HGT insertion, while a score below 0 indicates a possible contamination or HGT rich region.

3 Results

We tested our pipeline using the dataset for the tardigrade species *Hypsibius dujardini* [17]. We used the database NCBI nr instead of SwissProt+TrEMBL libraries, used in the original publication, and selected candidates with $AI > 30$ instead of $h_{ST} > 30$ (HGT Index), while the phylogenetic inference was performed with FastTree instead of RAxML [18]. The final selection was 401 proteins (386 genes) compared to 463 proteins (463 genes), and based on the phylogenetic trees, we detected a total of 379 candidate HGTs (95%) instead of 357 (77%). Overall, 342 candidate HGTs were common to *AvP* and the previously published analysis, the ones not identified by our pipeline having an AI below 30. We then evaluated the candidate HGTs by comparing the likelihoods of the original HGT-supporting trees to those of constrained trees in which tardigrade and other metazoan proteins were forced to form a monophyletic group. Equally likely topologies were observed for 27 proteins bringing the total number of strongly supported candidate HGTs to 352 (1.7% of the total proteins present in the genome). To assess the effect of using different databases, we performed two more searches against SwissProt (SP) and Uniref90 (UR). A total of 196 / 333 / 401 proteins were selected when using SP / UR / NR resulting in 127 / 292 / 352 candidate HGTs after alternative topology tests (*AvP evaluate*). Hence, depending on the sampling of the sequence diversity present in the sequence database, the number of detectable candidate HGT varies considerably.

In the publication describing Alien Index (AI) [3], the authors considered $AI > 45$ to be a good indication of foreign origin while genes with $0 < AI < 45$ were designated intermediate. However, this AI threshold value was originally defined on one single species only, the bdelloid rotifer, and further analyses on plant-parasitic nematodes have shown that an $AI > 45$ might be too stringent, leaving several true positives undetectable [9]. Here, we calculated the F1 score (Equation (1)) for all N with $AI > 0$ in *H. dujardini* to decide the optimal threshold between precision and sensitivity. We found

that selecting genes with $AI > 10$ represented an optimal balance between sensitivity and precision (Fig. 4). Therefore, we propose to perform *AvP* with $AI > 0$ with FastTree option to minimize the risk of missing HGT cases and utilise the scripts provided to calculate the F1 score and based on that, decide the optimal AI threshold (which is 10 for tardigrade example) for more sophisticated analyses.

$$F1_N = 2 \times \frac{HGT_{AI>N}}{HGT_{AI>0} + Genes_{AI>N}} \quad (1)$$

where:
HGT = genes confirmed to be HGT by *AvP*
$Genes$ = all the genes tested

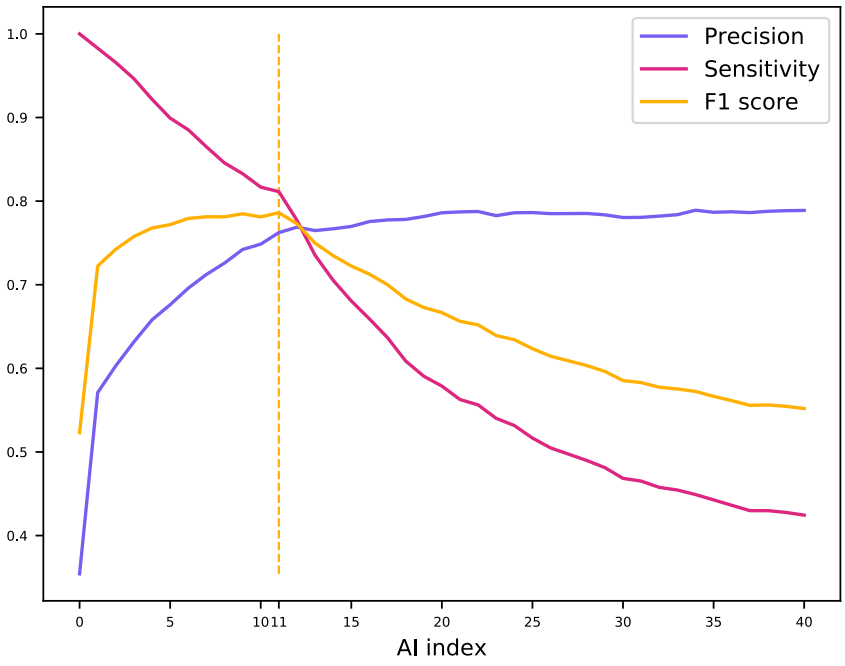


Fig. 4. Sensitivity, Precision, and F1 Score were calculated for Alien Index (AI) up to 40 for the proteins of the tardigrade *Hypsibius dujardini*. The dashed line indicates the AI with the highest F1 score indicating the most accurate AI threshold.

4 Perspectives

We propose *AvP* to facilitate the identification and evaluation of candidate HGTs in sequenced genomes across multiple branches of the tree of life. The most common methods used so far have been based on the difference of similarity between donor and ingroup sequences. Performing phylogenetic reconstruction and alternative topology evaluation creates a framework under which more robust HGT analyses can be performed. Furthermore, calculating the hgt_local_score can help identify contamination and HGT hot spots in the genome. Future extensions will include alignment evaluation to eliminate weak prediction of HGTs, a more precise traverse of complex phylogenetic topologies, and adding additional criteria for the initial selection of sequences (e.g HGT Index). Finally, we aim to incorporate a basic module of *AvP* to the Alieness webserver.

5 Availability

AvP is written in Python and is available online under GNU General Public License v3.0 at (https://github.com/GDKO/AvP).

6 Acknowledgements

We are grateful to the genotoul bioinformatics platform Toulouse Occitanie (Bioinfo Genotoul, doi: 10.15454/1.5572369328961167E12) for providing computing resources.

7 Funding

This work has been supported by the French government, through the UCA-JEDI "Investments in the Future" project managed by the National Research Agency (ANR) with the reference number ANR-15-IDEX-01. GDK has received the support of the EU in the framework of the Marie-Curie FP7 COFUND People Programme, through the award of an AgreenSkills+ fellowship (under grant number 609398).

Confict of Interest: none declared.

References

[1] Etienne G. J. Danchin. Lateral gene transfer in eukaryotes: tip of the iceberg or of the ice cube? *BMC Biology*, 14(1):101, Nov 2016.

[2] Christiam Camacho, George Coulouris, Vahram Avagyan, Ning Ma, Jason Papadopoulos, Kevin Bealer, and Thomas L. Madden. BLAST+: architecture and applications. *BMC Bioinformatics*, 10(1):421, Dec 2009.

[3] Eugene A. Gladyshev, Matthew Meselson, and Irina R. Arkhipova. Massive horizontal gene transfer in bdelloid rotifers. *Science*, 320(5880):1210–1213, 2008.

[4] Chiara Boschetti, Adrian Carr, Alastair Crisp, Isobel Eyres, Yuan Wang-Koh, Esther Lubzens, Timothy G. Barraclough, Gos Micklem, and Alan Tunnacliffe. Biochemical diversification through foreign gene expression in bdelloid rotifers. *PLOS Genetics*, 8(11):1–13, 11 2012.

[5] Georgios Koutsovoulos, Sujai Kumar, Dominik R. Laetsch, Lewis Stevens, Jennifer Daub, Claire Conlon, Habib Maroon, Fran Thomas, Aziz A. Aboobaker, and Mark Blaxter. No evidence for extensive horizontal gene transfer in the genome of the tardigrade *Hypsibius dujardini*. *Proceedings of the National Academy of Sciences*, 113(18):5053–5058, 2016.

[6] Edwin Jacox, Cedric Chauve, Gergely J. Szöllősi, Yann Ponty, and Celine Scornavacca. ecceTERA: comprehensive gene tree-species tree reconciliation using parsimony. *Bioinformatics*, 32(13):2056–2058, 02 2016.

[7] Charlotte A Darby, Maureen Stolzer, Patrick J Ropp, Daniel Barker, and Dannie Durand. Xenolog classification. *Bioinformatics*, 33(5):640–649, 12 2016.

[8] Benjamin Buchfink, Chao Xie, and Daniel H. Huson. Fast and sensitive protein alignment using DIAMOND. *Nature Methods*, 12(1):59–60, Jan 2015.

[9] Corinne Rancurel, Ludovic Legrand, and Etienne G. J. Danchin. Alienness: Rapid detection of candidate horizontal gene transfers across the tree of life. *Genes*, 8(10), 2017.

[10] David M. Emms and Steven Kelly. Orthofinder: phylogenetic orthology inference for comparative genomics. *Genome Biology*, 20(1):238, Nov 2019.

[11] Kazutaka Katoh and Daron M. Standley. MAFFT multiple sequence alignment software version 7: Improvements in performance and usability. *Molecular Biology and Evolution*, 30(4):772–780, 2013.

[12] Salvador Capella-Gutiérrez, José M. Silla-Martínez, and Toni Gabaldón. trimAl: a tool for automated alignment trimming in large-scale phylogenetic analyses. *Bioinformatics*, 25(15):1972–1973, 2009.

[13] Morgan N. Price, Paramvir S. Dehal, and Adam P. Arkin. FastTree 2 – approximately maximum-likelihood trees for large alignments. *PLOS ONE*, 5(3):1–10, 03 2010.

[14] Bui Quang Minh, Heiko A Schmidt, Olga Chernomor, Dominik Schrempf, Michael D Woodhams, Arndt von Haeseler, and Robert Lanfear. IQ-TREE 2: New Models and Efficient Methods for Phylogenetic Inference in the Genomic Era. *Molecular Biology and Evolution*, 37(5):1530–1534, 02 2020.

[15] Xiaofan Zhou, Xing-Xing Shen, Chris Todd Hittinger, and Antonis Rokas. Evaluating fast maximum likelihood-based phylogenetic programs using empirical phylogenomic data sets. *Molecular Biology and Evolution*, 35(2):486–503, 2018.

[16] Andrew Rambaut. FigTree v1.4.4. http://tree.bio.ed.ac.uk/software/figtree/. 2020.

[17] Yuki Yoshida, Georgios Koutsovoulos, Dominik R. Laetsch, Lewis Stevens, Sujai Kumar, Daiki D. Horikawa, Kyoko Ishino, Shiori Komine, Takekazu Kunieda, Masaru Tomita, Mark Blaxter, and Kazuharu Arakawa. Comparative genomics of the tardigrades *Hypsibius dujardini* and *Ramazzottius varieornatus*. *PLOS Biology*, 15(7):1–40, 07 2017.

[18] Alexandros Stamatakis. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics*, 30(9):1312–1313, 01 2014.

ConDor: Site-wise convergence detector in large protein alignments

Marie MOREL[1,2], Frédéric LEMOINE[1,3] and Olivier GASCUEL[1,4]

[1] Unité de Bioinformatique Évolutive - Département Biologie computationnelle, Institut Pasteur, 25-28 rue du Dr Roux 75015 Paris, France
[2] Université de Paris, 5 rue Thomas Mann 75013 Paris, France
[3] Hub de Bioinformatique et Biostatistique - Département Biologie computationnelle, Institut Pasteur, Paris, France
[4] Institut de Systématique, Evolution, Biodiversité (ISYEB - UMR 7205), CNRS & Muséum National d'Histoire Naturelle, 57 rue Cuvier 75005 Paris, France

Corresponding Author: marie.morel@pasteur.fr

Current methods to detect evolutionary convergence at the molecular level, aim to decipher which amino-acid changes are related to a convergent phenotype emerging from the same environmental constraints. The hypothesis is that convergent phenotypic traits may commonly arise from identical genetic changes.
Here we propose a simulation-based method to detect positions under convergent evolution in large protein alignments, without prior knowledge of the phenotype or environmental constraints. To this aim, a phylogeny is inferred from our data and then used in simulations to estimate the expected number of genetic changes (or emergence-event of mutations) in stable evolutionary constraints (null model). Similarly, we count the emergence-events of mutations in our data and test if they are occurring more often than expected under the null model.
We apply our method on a real data set of HIV reverse transcriptase and on HIV-like simulated data sets. On simulated data, with known events of convergent evolution, we detect on average two third of truly convergent events, with a low fraction of false positives. With HIV data, we know a priori that drug resistance mutations (DRMs) are convergent. Even without any knowledge on the treatment status of the patients, we retrieve more than 70% of positions corresponding to known DRMs.
These results demonstrate the potential of the method to target specific mutations to be further studied experimentally or using a dN/dS approach, for example.

Keywords molecular evolution, phylogenetics, selection and adaptation, HIV, resistance to drugs

1. Introduction

Convergent evolution can be defined as the independent acquisition of similar traits in distinct lineages over the course of evolution. The studied traits can be behavioural, morphological, molecular, etc. In each category, traits can be quantitative (size, length, etc.), binary (presence or absence of a given phenotype) or categorical (a trait is subdivided into several classes). Recent studies focused on the molecular level, following the hypothesis that convergent phenotypes generally result from the same genetic changes [1-3]. At the protein level, we commonly distinguish parallel mutations (a change towards the same amino acid is observed from the same ancestral amino acids), convergent mutations (change towards the same amino acid, from different ancestral amino acids) and reversions (mutations that restore an amino acid previously lost during evolution).

Several methods have been developed to detect convergent evolution at the molecular level [4-9]. They are all based on an a priori knowledge or on the observation of a convergent phenotype and aim to identify protein mutations that correlate with the presence of the converging trait. To assess whether the identified amino-acid changes are the result of some adaptation, one must determine whether they are due to chance and to what extent they explain the observed phenotype [10]. The various methods

differ among other things in the way they define a null model, i.e., the model that allows determining whether observations are due to chance. These methods are commonly applied to large eukaryotic and prokaryotic genomes and use genome-wide analyses to select convergent genes by considering simultaneously all sites of the protein sequences. Only one method uses statistical tests at the resolution of the site, but still requires a priori knowledge of convergence at the phenotypic level [8].

Testing the significance of convergent (or parallel or revertant) changes for a given site may have interesting applications. In the case of complex eukaryotic or bacterial organisms, there are few examples of a single amino-acid change that could explain a convergent phenotype [3]. However, in the case of viruses with rapid evolution, and whose (small) genomes are strongly constrained, often only a few possible amino-acid changes are possible at a given position. Determining which of the many parallel and convergent changes stand out from expectations could allow to sort out the mutations resulting from adaptive phenomena. This is in fact what was observed in the SARS-CoV-2 genomes, where we first identified non-silent mutations in the Spike protein, which were spreading within the viral population and appeared multiple times independently, before being classified as evolutionary advantageous for the virus [11-13]. Indeed, in viruses it is often easier to identify a mutation of interest than to observe the effects of that mutation given how difficult the phenotype of a virus or the environmental conditions in which it evolves are to access.

Here we propose a method designed to detect site-wise convergent evolution in large amino-acid alignments without prior knowledge of phenotype. This method performs detailed analysis at the gene/protein level, with typical application to viruses, but also to specific genes known to be involved in phenotypic convergence [14]. We are interested in changes towards a target amino acid regardless of the ancestral amino acids that lead to the difference in the amino acid sequences. In other words, parallel, convergent and revertant mutations are considered indifferently and we consider different target amino acids as different events. The observed number of amino-acid changes is estimated with ancestral character reconstruction, and their expected number using computer simulations. In the following sections, we describe this approach that we implemented in a software named ConDor (Convergence Detector). Its performance is assessed on HIV-like simulated data sets and on a real HIV reverse transcriptase dataset.

2. Methods

2.1. A simulation-based approach

ConDor takes as input a multiple protein sequence alignment. It then performs a site-wise analysis and identifies for each site (or position) the amino-acid mutations emerging several times in distinct lineages and occurring significantly more frequently than expected under a null model of evolution. The detailed analysis pipeline is presented Figure 1. It is made of four main steps: (1) estimate the parameters of the null model from the observed data (phylogenetic tree, substitution model parameters, site-wise evolutionary rates, etc.); (2) infer ancestral amino acids and count the number of observed emergence events of mutations (EEMs) in the observed data; (3) simulate new datasets under the inferred null model and count simulated EEMs; and finally (4) compare the observed and simulated number of EEMs and determine which mutations occur significantly more often in the observed dataset than in the simulations. Such mutations are considered as convergent events.

The null substitution model and its parameters, the evolutionary rates per site and the phylogenetic tree are all inferred from the input alignment. The selected substitution model, along with amino-acid frequencies, tree topology, branch lengths and site-wise evolutionary rates are assumed to model the data without convergence. We make this assumption because using large alignments (>1000 sequences), we consider that mutations resulting from convergent evolution are rare enough to only have a negligible influence on tree and parameter inference. The reconstructed phylogeny is then rooted using a provided outgroup. This is essential to infer the ancestral sequence at the root of the tree, run simulations starting from this sequence, and count simulated EEMs. Ancestral character reconstruction (ACR) is achieved using a maximum likelihood approach, implemented in PastML [15]. We use the "maximum *a posteriori*" (MAP) method in which the state with the highest marginal posterior is selected at each

node. Once all ancestral sequences are reconstructed and associated to the nodes in the phylogeny, we identify where independent amino-acid changes occurred in the tree and count them as explained in the subsection “Counting emergence events”. This corresponds to the observed number of EEMs for each alignment position and amino acid under study, that is, those that are observed often enough at a given position (≥12 sequences in our HIV experiments).

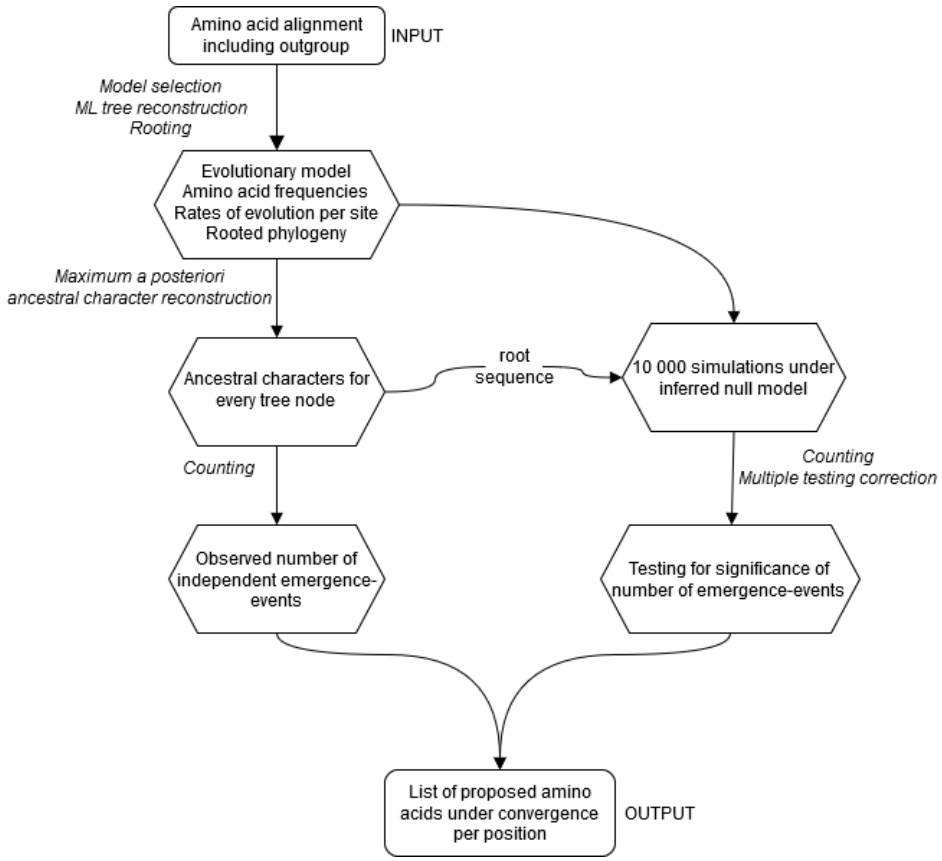


Figure 1: Simplified flowchart of ConDor, our proposed simulation-based method.
ConDor takes an amino-acid alignment as input, used for inference of the null substitution model, tree building and ancestral reconstruction. The reconstructed tree and root sequence are used to simulate 10,000 alignments under the null model. The output is a list of amino-acid changes per position detected as convergent, as they emerge more often in the input alignment than in simulations.

We then simulate the expected evolution without convergence of each site of the alignment many times (10,000 in our experiments). We do not use the root sequence reconstructed by ACR as a start but draw amino-acid characters in the vector of posterior probabilities. Taking only the amino acid with the highest posterior could bias the simulations, especially if the reconstruction is uncertain (for example two amino acids with posteriors of 0.55 and 0.45). If we proceed to 10,000 draws in the posteriors, we have a better representation of the root sequence. Simulations are done along the inferred tree, and then we count the simulated numbers of EEMs (10,000 values per site and per studied amino acid) using the algorithm explained below. For example, let us consider the mutation M41L from the real HIV data where at position 41, a Methionine (M) is substituted by a Leucine (L) in 211 sequences. The observed number of EEMs towards L is 47, which is smaller than 211 as in some subtrees all tips have L, corresponding to only 1 EEM. This number is compared to the distribution of the number of EEMs towards L, starting from an M at the tree root every time (no ambiguity in ACR), among 10,000 simulations; this number of simulated EEMs ranges from 0 to 31 with an average of 12. From the observed number of EEMs and the distribution of simulated EEMs we estimate a p-value for each observed mutation, which is equal to 0 in our M41L example. After considering a correction for multiple testing, mutations with p-values lower than the rejection criterion are considered as resulting from convergent evolution.

2.2. Counting independent emergence events of mutations (EEMs)

In our approach, the observed number of EEMs is inferred from ACR based on the input sequences, while the expected number of EEMs is inferred from many simulations evolving from the probabilistic root sequence along the inferred tree. In the simulations, changes may appear that could not be inferred by ACR, if they are not transmitted to any leaf for example. In this case, the estimated number of changes artificially deviates from the observed number. This effect is even more important on sites with rapid evolution since more changes are expected. Thus, only the changes transmitted to at least one leaf are counted in our method, since they are the only ones that could be found by ACR. Note, moreover, that generally we are only interested by the amino acids present in the available, actual sequences, and rarely by those that are never observed. Even though we count EEMs without making the difference between convergent, parallel or revertant events, we retain the information during the counting process for interpretation afterwards. All these algorithms (ACR as implemented in PastML, counting EEMs, simulations) have a time complexity that is linear in the number of tree tips, thanks to carefully orchestrated tree traversals [15].

2.3. Creation of a synthetic HIV-like dataset

To our knowledge, there is no convergent evolution model allowing to simulate thousands of sequences without prior knowledge of the phenotype or environmental constraints. We therefore created our own convergent data set inspired from a real case of convergence. Drug resistance mutations (DRMs) are mutations that occur independently in patients receiving a drug treatment and are therefore a perfect example of evolutionary convergence. In HIV, they are well characterised and studied, since their emergence can lead to treatment failure and transmission of resistant viruses. They are preferentially found in proteins targeted by the antiretroviral treatment: the protease, the reverse transcriptase, and the integrase. The list of DRMs on these proteins is publicly available at (https://hivdb.stanford.edu/). DRMs are written as “XposY” with X the ancestral amino acid, pos the position of the substitution in the protein alignment, with numbering based on the reference sequence HXB2, and Y the mutated amino acid.

Using a real HIV polymerase amino-acid dataset (250 sites, 3,387 sequences), we extracted positions and sequences with DRMs and replaced them with gaps in the multiple sequence alignment (MSA). DRMs were retrieved from the “Essential DRM Data” section of the Stanford University Drug resistance database (https://hivdb.stanford.edu/pages/poc.html). We then reconstructed a tree and inferred the rates of substitution per site. This tree represents relationships between sequences in the real data and its topology is not affected by the DRMs. We then simulated the evolution of the HXB2 reference sequence (reverse transcriptase only) along this tree. We performed the simulations five times for reproducibility purposes resulting in five multiple sequence alignments (MSAs) without convergence. DRMs were then manually added in the same sequences and positions as where they were found in the real MSA. This ensures that the way we simulate convergence is realistic. We did this for each of the 37 tested DRMs which were the most common (present in ≥ 12 sequences). Thusly, the five resulting MSAs have no convergent events, but the “realistic” added DRMs.

3. Results

3.1. Synthetic HIV-like data set

The data consists of five MSAs of 3,387 sequences and 250 amino acids each, mimicking HIV reverse transcriptase and simulated under HIVb model of evolution [16]. In total, 37 DRMs were placed in each of the MSAs over 27 positions (see above). These DRMs are found in at least 12 sequences and about 20% of sequences have at least one DRM. The most common one, M184V, is found in 273 (8%) sequences. However, 19 DRMs are found in less than 1% of the sequences (i.e., in 12 to 33 sequences) so they are expected to be difficult to identify.

The model inferred from the data sets by ModelFinder [17] was HIVb, which is the model we used for generating them. Thus, tree reconstruction, ACR and simulations were all done with HIVb, which is the true model of substitution in these simulations.

We tested on average 441 mutations per dataset, the ones present in at least 12 sequences and with more than 2 EEMs. On average 27.4 mutations were found to be convergent according to our method. Among them 26 were true DRMs (among 37).

In these synthetic data sets, the numbers of EEMs for the DRMs range from 9 (K101P) to 225 (M184V). We better detect DRMs with the highest number of EEMs and especially with more than 30 EEMs as we find more than 90% of the DRMs with more than 30 EEMs on average. If there are several DRMs for one position, we generally detect the most frequent one(s) only. For example, on position 219 we detect mutations towards Q and E but not N. Similarly, on position 215, we do not detect mutations towards S and D, while we detect mutation T215C, although there are fewer EEMs towards C. This is explained by the substitution rate between T and C that is very low, and thus few changes are expected from T towards C: 2 EEMs are expected on average in the 10,000 simulations between T and C at position 215, while 17 are found in average in the five synthetic data sets.

On average we find 1.4 false positives per data set, some of which exhibit a very high evolutionary rate: as the evolutionary rate increases, more changes are observed at the given site and thus more variability and uncertainty in the simulations. Thus, very fast sites can bias convergence detection and lead to the detection of false positives. As expected, we observe very few false positives when analysing our synthetic datasets with the true model of evolution. If we focus on detecting positions with convergence (e.g., 219) rather than DRMs (e.g., K219Q, K219E, K219N, etc.) we increase in accuracy and detect on average 22 of the 27 convergent positions for all datasets, while the number of false positive remains equal to 1.4 on average (Tab. 1).

		DRMs		Non-DRMs		Total	
	Model	Mutation	Position	Mutation	Position	Mutation	Position
Detected	HIVb	26 (±1.2)	22 (±1.2)	1.4 (±1.14)	1.4 (±1.1)	27.4 (±1.5)	23.4 (±1.3)
	JTT	25.2 (±0.8)	22.4 (±0.5)	17.2 (±3)	15.8 (±2.6)	42.4 (±2.5)	38.2 (±2.3)
Not detected	HIVb	11 (±1.2)	5 (±1.2)	402.8 (±6.8)	91.2 (±1.6)	413.8 (±7.15)	96.2 (±2.5)
	JTT	11.8 (±0.8)	4.6 (±0.5)	392.8 (±13.7)	70.2 (±4.3)	404.6 (±14)	74.8(±4.3)
Total	HIVb	37	27	404.2 (±6)	92.6 (±1.7)	441.2 (±5.6)	113.2 (±1.5)
	JTT	37	27	410 (±14.3)	86 (±3.4)	447 (±14.3)	113 (±3.4)

Table 1: Method accuracy with synthetic data.
We show the number of mutations or positions detected on the synthetic HIV-like MSAs, analysed with HIVb and JTT model of evolution. Average for the 5 datasets are reported and standard deviation is given between parentheses. True positives are at the intersection between detected and DRMs, and false positives at the intersection between detected and non-DRMs. Non-DRMs are mutations resulting from the evolution of the root sequence under the null model; they exhibit more than 2 EEMs and are found in at least 12 sequences.

Since we never have the true model of evolution with real data, we tested the effect of model violation on the synthetic dataset. We fixed the program to run all analyses with JTT for tree reconstruction, ACR and simulations, instead of letting the program infer and use the best model of evolution (here HIVb).

The strongest effect can be seen on the number of false positives, which increases from 1.4 to 17.2 (Tab. 1). Compared to the number of negatives, this remains very low, with 4% of falsely detected random mutations (i.e., mutations resulting from the evolution under the null model) among more than 400. Moreover, we observe again the same tendency with high-rate mutation events among false positives. The detected mutations with the highest rates are always false positives. Since the model does not represent exactly the synthetic dataset, we tend to detect more mutations as convergent, but this does not impact the detection of DRMs. DRM detection remains accurate and is robust to model violation. However, based on these results, we expect false positives with real data, representing a substantial fraction of detections (40% on average in Tab. 1 with JTT model). True positives tend to be mutations with the most EEMs, low substitution rate between amino acids and on sites with medium evolutionary rate. On the opposite, fast sites tend to be detected as convergent, even if they are not.

3.2. Real HIV data set

This dataset consists in a MSA of truncated polymerase from HIV-1 subtype B. It was retrieved from Lemoine *et al* [18]. It contains 3,581 sequences of 1,043 nucleotide sites that were translated into 347 amino acids. Among these 347 amino acids, 250 code for the reverse transcriptase and are analysed here. Slightly more than 20% of the sequences have at least one DRM and on average the known DRMs (https://hivdb.stanford.edu/pages/poc.html) are found in 11 sequences. The most common one, M184V is found in 273 sequences. There are 37 DRMs present in at least 12 sequences, the same ones as in the synthetic datasets. They are distributed on 27 positions. We focus on these 37 DRMs to assess the performance of our approach as we test for convergence mutations present in at least 12 sequences and with more than 2 EEMs. As already explained, we expect detecting other mutations, some being truly convergent and some corresponding to false positives, likely due to model misspecification and positioned on fast sites.

The model inferred with ModelFinder [17] on this dataset is HIVb with 'free rates' and 9 rate categories. We detect 74 convergent events after applying the Benjamini Hochberg correction (corrected p-value threshold of 0.0004, corresponding to an alpha level of 5%). Among these detections, 20 are DRMs which represents more than half (54%) of true DRMs. The non-DRM detected events correspond to 11 mutations on fast sites (Fig. 2), which are likely false positives and other events for which we cannot conclude. Regarding false predictions, 17 DRMs are not detected, 7 of which having a p-value lower than 0.005 but higher than the significance threshold. If we focus on positions instead of mutations, we detect 65 positions as convergent including 19 (70%) of the positions with DRMs.

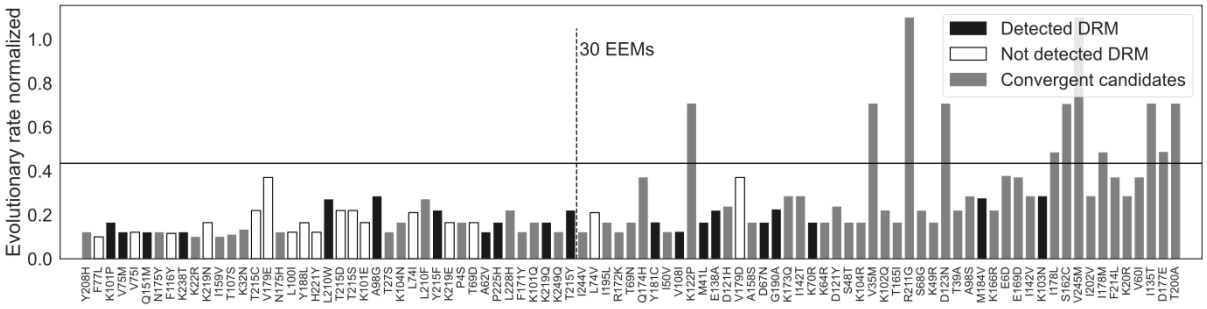


Figure 2: DRMs detection and convergent candidates, real HIV data.
We display DRMs that are detected or not and convergent candidates predicted by ConDor on the real HIV-1 MSA, analysed with the HIVb substitution model. Events are sorted by their number of EEMs on the x-axis. On the y-axis we report the evolutionary rate of the site of each mutation. This evolutionary rate is normalized between 0 and 1. The plain horizontal line represents the limit of the 5% fastest sites of the whole dataset; above this threshold the mutations are likely due to homoplasy and do not reflect any convergence.

4. Discussion

We have shown in this work that we can detect with fair accuracy evolutionary convergence at the resolution of a site, even without prior knowledge of the phenotype or environmental constraints. This is possible since we are working at the scale of a single protein with thousands of sequences, which provides sufficient signal and detection power. By working on thousands or even millions of sites, ConDor would lack the statistical power to work at the scale of a single site due to multiple testing. We do not consider the phenotype of the studied organisms, because we have designed this method for the study of specific genes, typically from viruses and microorganisms where this data is rarely available. One could however think about adapting this method to select significantly convergent sites with regards to their presence in organisms previously annotated as convergent.

In some ways, our approach presents similarities with the detection of sites under positive selection. We are indeed looking for mutations which could be advantageous as they are found more often than expected under a neutral model of evolution. Positive selection can be inferred at a site if the number of non-synonymous substitutions exceeds the number of synonymous substitutions. These substitutions can be towards a particular amino acid or any change from the original amino acid. This is, for example,

the case in immune avoidance where many amino-acid changes at the antigenic site are favourable. However, in the case of convergent evolution, and especially here with the example of DRMs, we are interested in substitutions towards one or a few specific amino acids. Positive selection could be a way to confirm some of our detections, but on the contrary, not all sites under positive selection are convergent.

Without knowledge of the phenotype, we necessarily observe some false positives as seen in our synthetic HIV-like data sets by changing the model from HIVb to JTT. Similarly, in real data, as we do not know the true model of evolution, some of our detections are likely to be false positives. Indeed, our method relies on how realistic the thousands of simulations act as a null model. Our results show that for most sites and most mutations we are close to what is observed in real data and simulations represent a satisfactory null model. However, on certain fast sites we observe that simulations tend to differ from the real data, which results in an increased rate of false positives. More involved models, e.g. based on mixtures or some ideas derived from the CAT model [19], could possibly enhance our approach.

ConDor is available at https://condor.pasteur.cloud/ and all analysis and data can be found at https://github.com/mariemorel/condor.

References

[1] D. L. Stern, « The genetic causes of convergent evolution », *Nat. Rev. Genet.*, vol. 14, nº 11, p. 751-764, nov. 2013, doi: 10.1038/nrg3483.

[2] E. B. Rosenblum, C. E. Parent, et E. E. Brandt, « The Molecular Basis of Phenotypic Convergence », *Annu. Rev. Ecol. Evol. Syst.*, vol. 45, nº 1, p. 203-226, nov. 2014, doi: 10.1146/annurev-ecolsys-120213-091851.

[3] J. F. Storz, « Causes of molecular convergence and parallelism in protein evolution », *Nat. Rev. Genet.*, vol. 17, nº 4, p. 239-250, avr. 2016, doi: 10.1038/nrg.2016.11.

[4] J. Zhang et S. Kumar, « Detection of convergent and parallel evolution at the amino acid sequence level. », *Mol. Biol. Evol.*, vol. 14, nº 5, p. 527-536, 1997.

[5] A. D. Foote *et al.*, « Convergent evolution of the genomes of marine mammals », *Nat. Genet.*, vol. 47, nº 3, p. 272-275, mars 2015, doi: 10.1038/ng.3198.

[6] G. W. C. Thomas et M. W. Hahn, « Determining the Null Model for Detecting Adaptive Convergence from Genomic Data: A Case Study using Echolocating Mammals », *Mol. Biol. Evol.*, vol. 32, nº 5, p. 1232-1236, mai 2015, doi: 10.1093/molbev/msv013.

[7] Z. Zou et J. Zhang, « No Genome-Wide Protein Sequence Convergence for Echolocation », *Mol. Biol. Evol.*, vol. 32, nº 5, p. 1237-1241, mai 2015, doi: 10.1093/molbev/msv014.

[8] O. Chabrol, M. Royer-Carenzi, P. Pontarotti, et G. Didier, « Detecting the molecular basis of phenotypic convergence », *Methods Ecol. Evol.*, vol. 9, nº 11, p. 2170-2180, 2018, doi: 10.1111/2041-210X.13071.

[9] C. Rey, L. Guéguen, M. Sémon, et B. Boussau, « Accurate Detection of Convergent Amino-Acid Evolution with PCOC », *Mol. Biol. Evol.*, vol. 35, nº 9, p. 2296-2306, sept. 2018, doi: 10.1093/molbev/msy114.

[10] J. Zhang, « Parallel adaptive origins of digestive RNases in Asian and African leaf monkeys », *Nat. Genet.*, vol. 38, nº 7, Art. nº 7, juill. 2006, doi: 10.1038/ng1812.

[11] D. P. Martin *et al.*, « The emergence and ongoing convergent evolution of the N501Y lineages coincides with a major global shift in the SARS-CoV-2 selective landscape », Infectious Diseases (except HIV/AIDS), preprint, mars 2021. doi: 10.1101/2021.02.23.21252268.

[12] L. van Dorp *et al.*, « Emergence of genomic diversity and recurrent mutations in SARS-CoV-2 », *Infect. Genet. Evol.*, p. 104351, mai 2020, doi: 10.1016/j.meegid.2020.104351.

[13] B. Korber *et al.*, « Tracking Changes in SARS-CoV-2 Spike: Evidence that D614G Increases Infectivity of the COVID-19 Virus », *Cell*, vol. 182, nº 4, p. 812-827.e19, août 2020, doi: 10.1016/j.cell.2020.06.043.

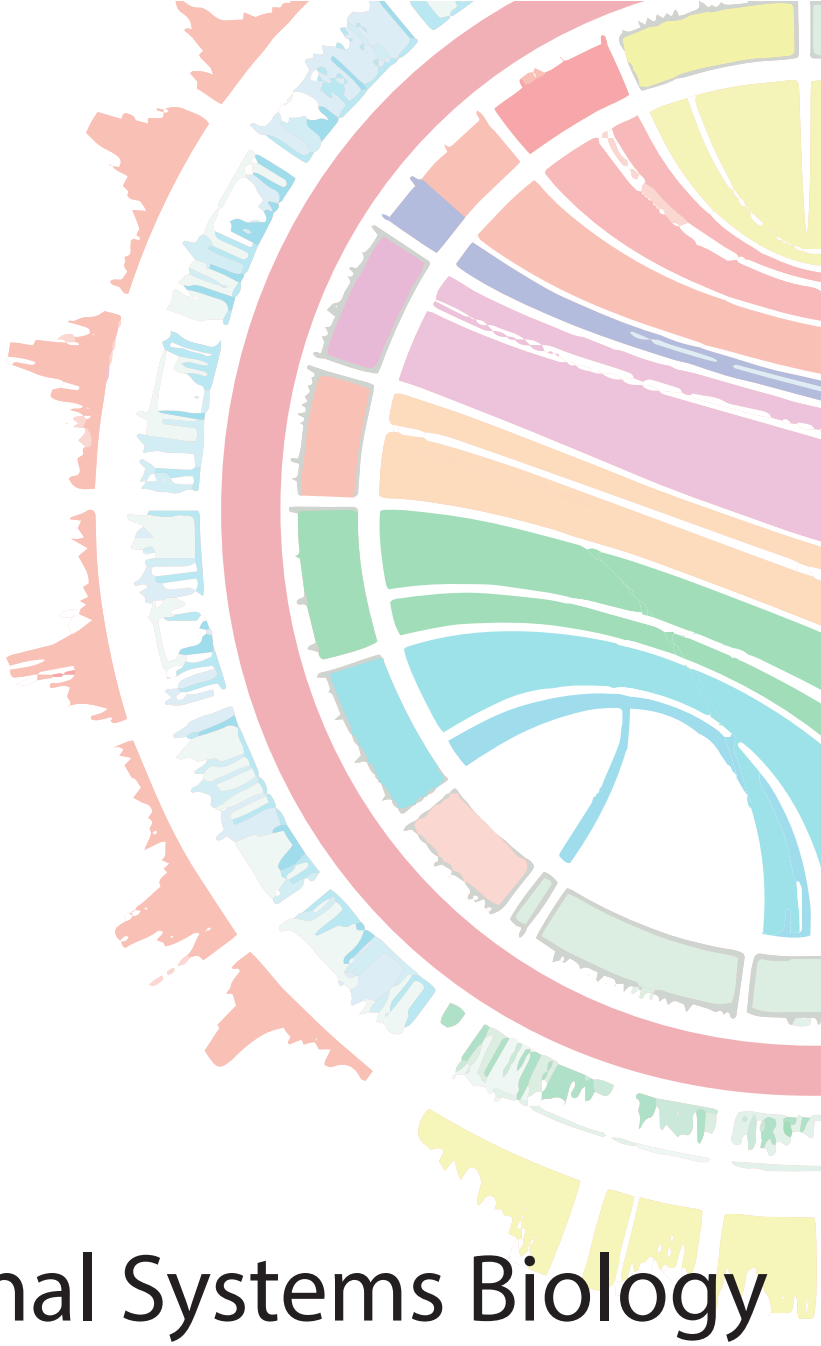
[14] J. Hill *et al.*, « Recurrent convergent evolution at amino acid residue 261 in fish rhodopsin », *Proc. Natl. Acad. Sci.*, vol. 116, nº 37, p. 18473-18478, sept. 2019, doi: 10.1073/pnas.1908332116.

[15] S. A. Ishikawa, A. Zhukova, W. Iwasaki, et O. Gascuel, « A Fast Likelihood Method to Reconstruct and Visualize Ancestral Scenarios », *Mol. Biol. Evol.*, doi: 10.1093/molbev/msz131.

[16] D. C. Nickle, L. Heath, M. A. Jensen, P. B. Gilbert, J. I. Mullins, et S. L. K. Pond, « HIV-Specific Probabilistic Models of Protein Evolution », *PLOS ONE*, vol. 2, nº 6, p. e503, juin 2007, doi: 10.1371/journal.pone.0000503.

[17] S. Kalyaanamoorthy, B. Q. Minh, T. K. F. Wong, A. von Haeseler, et L. S. Jermiin, « ModelFinder: fast model selection for accurate phylogenetic estimates », *Nat. Methods*, vol. 14, nº 6, p. 587-589, juin 2017, doi: 10.1038/nmeth.4285.

[18] F. Lemoine *et al.*, « Renewing Felsenstein's phylogenetic bootstrap in the era of big data », *Nature*, vol. 556, nº 7702, p. 452-456, avr. 2018, doi: 10.1038/s41586-018-0043-0.
[19] S. Parto et N. Lartillot, « Detecting consistent patterns of directional adaptation using differential selection codon models », *BMC Evol. Biol.*, vol. 17, nº 1, déc. 2017, doi: 10.1186/s12862-017-0979-y.



> Session 6
Computational Systems Biology

Automatic Synthesis of Boolean Networks from Biological Knowledge and Data

Athénaïs Vaginay[1,2], Taha Boukhobza[2] and Malika Smaïl-Tabbone[1]
[1] LORIA (Université de Lorraine, CNRS, Inria)
[2] CRAN (Université de Lorraine, CNRS) Contact: athenais.vaginay@loria.fr

Corresponding author: athenais.vaginay@loria.fr

Reference paper: **Vaginay *et al.* (june 2021) Automatic Synthesis of Boolean Networks from Biological Knowledge and Data *International Conference in Optimization and Learning (OLA)*, long paper accepted for publication in the OLA 2021 Springer CCIS proceedings.**
https://hal.archives-ouvertes.fr/hal-03256693

Boolean Networks (BNs) are a simple formalism used to study complex biological systems when the prediction of exact reaction times is not of interest. They play a key role in understanding the dynamics of the studied systems and predicting their disruption in case of complex human diseases. A BN consists of a set of n Boolean *transition functions* (one per components) giving the successive Boolean states of the components, depending on the previous state of the other components of the system. Here is an example of a BN of three components called A, B and C: $\mathcal{B} = \{f_A := C\ ;\quad f_B := B \wedge \neg C\ ;\quad f_C := \neg C\}$. It reads like "A will be activated is C was activated", "B will be activated if B was activated but C was not" and "C will be activated if C was not". The dynamics of a BN is obtained by applying iteratively the transition functions starting from all the 2^n possible configurations. The order of application of the transition functions is defined by the *update scheme*. In the *mixed* updated scheme, any number of components can be updated at each step. The dynamics is represented by a directed graph whose nodes are the 2^n configurations and the edges are the transitions according to the chosen update scheme. Such a graph is called *state transition graph* (STG).

BNs are generally built from experimental data and knowledge from the literature, either manually or with the aid of programs. The automatic synthesis of BNs is still a challenge for which several approaches have been proposed, such as `REVEAL` [1], `Best-Fit` [2] and `caspo-TS` [3]. In this paper, we propose `ASKeD-BN`, a new approach based on *Answer-Set Programming* (ASP) to synthesize BNs *constrained* in their dynamics by a multivariate Time Series (TS), and in their structure by a Prior Knowledge Network (PKN). A PKN is a directed graph on the components of the system. It constrains the structure of the synthesized BNs by defining which components can appear as variables in each transition function and the polarity of those variables. The synthesized BNs also have to reproduce as well as possible the sequence of configurations extracted from the given multivariate TS.

We compare `ASKeD-BN` with `REVEAL`, `Best-Fit` and `caspo-TS` according to three criteria: (i) the number of BNs returned by the approaches ran on a PKN and a multivariate TS; (ii) the median of the coverage ratios *i.e.,* the proportion of transitions extracted from the input TS that are present in the mixed STG of the BN; (iii) the standard deviation of the coverage ratios. We ran experiments on two real datasets and more than 300 synthetic datasets according to various settings (synchronous or asynchronous, with or without repetition, with or without noise), and provided empirical evidence that `ASKeD-BN` has the best trade-off on the evaluation criteria: it returns a small set of BNs which comply with the provided structural constraints, cover a good proportion of the dynamical constraints, with a small variance.

References

[1] Shoudan Liang, Stefanie Fuhrman, and Roland Somogyi. REVEAL, a general reverse engineering algorithm for inference of genetic network architectures. pages 18–29.

[2] Harri Lähdesmäki, Ilya Shmulevich, and Olli Yli-Harja. On learning gene regulatory networks under the boolean network model. 52(1):147–167.

[3] Max Ostrowski, Loïc Paulevé, Torsten Schaub, Anne Siegel, and Carito Guziolowski. Boolean network identification from perturbation time series data combining dynamics abstraction and logic programming. 149:139–153.

Distinguishing Context Dependent Events in Quotients of Causal Stories

Sébastien LÉGARÉ[1,2], Jean KRIVINE[3] and Jérôme FERET[1,2]
[1] Département d'informatique de l'ÉNS, ÉNS, CNRS, Université PSL, 75005 Paris, France
[2] INRIA, Centre de recherche INRIA de Paris, 75012 Paris, France
[3] CNRS, Université Paris Diderot, IRIF, 75013 Paris, France

Corresponding author: jerome.feret@ens.psl.eu

Abstract *Causality analysis of rule-based models allows the reconstruction of the causal paths leading to chosen events of interest. This potentially reveals emerging paths that were completely unknown at the time of creation of a model. However, current implementations provide results in the form of a collection of stories. For large models, this can amount to hundreds of story graphs to read and interpret for a single event of interest. In this work, we hence develop a method to fold a collection of stories into a single quotient graph. The main challenge is to find a trade-off in the partitioning of story events which will maximize compactness without loosing important details about information propagation in the model. The partitioning criterion proposed is relevant context, the context from an event's past which remains useful in its future. Each step of the method is illustrated on a toy rule-based model. This work is part of a longer term objective to automatically extract biological pathways from rule-based models.*

Keywords *Causality, Rule-based modelling, Visualisation, Graph folding, Concurrency*

1 Introduction

Rule-based modelling [1,2] is well suited to the construction of large models characteristic of systems biology. Rules distinguish themselves from reactions by focussing only on the part of molecules that changes during a transition, rather than fully defining the species involved. This lets rule-based models avoid combinatorial explosion issues and obviates the need to reduce models toward a predefined goal. Subsequent analysis can then reveal emerging properties that were initially completely unknown by the modellers.

Causality analysis is one of the most interesting analyses to perform on rule-based models. It allows the reconstruction of the causal paths, also called stories, leading to a chosen event of interest [3]. Considering a protein involved in some pathology for instance, it would provide a quantitative account of each upstream molecule's contribution to its activation. However, current implementations [3,4] present the results of causality analysis as a collection of individual stories. In a large systems biology model, this can amount to hundreds of story graphs to read and interpret for a single event of interest.

The goal of this work is to develop a compact representation that allows the visualisation of all the paths leading to an event of interest on a single graph. To do so, we fold a collection of stories by merging events that are deemed equivalent. The broader the definition of equivalence between events, the more compact the folded representation is. Yet, folding too much may merge events which are similar but play different roles in the propagation of information. Thus, a trade-off has to be found. Here, we refine each event in stories with some contextual information about its past, modulated by the usefulness of that context in its future. We use this additional information to define the quotient of events in families of stories. As a result, we obtain a quotient that remains compact but provides a better insight on the way information is processed in the models.

2 Initial Setting

2.1 Kappa Rule-Based Model

The toy model below is written in Kappa language [5] and will be used to illustrate the method developed in this work. Such toy model does not represent the kind of large systems that can typically benefit from rule-based modelling. However, it features events whose context is not trivially determined and is hence well suited to highlight the details of the methodology.

```
%agent:  A(x)
%agent:  B(y)
%agent:  C(x y z{u,p})
%agent:  D(z s{u,p})

'A binds C'  A(x[./1]), C(x[./1]) @ 0.01
'B binds C'  B(y[./1]), C(y[./1]) @ 0.01
'A phos C'   A(x[1]), C(x[1] z{u/p}) @ 1
'B phos C'   B(y[1]), C(y[1] z{u/p}) @ 1
'C binds D'  C(z[./1]{p}), D(z[./1]) @ 0.01
'A phos D'   A(x[1]), C(x[1] z[2]), D(z[2] s{u/p}) @ 1
'B phos D'   B(y[1]), C(y[1] z[2]), D(z[2] s{u/p}) @ 1

%init:  100 A()
%init:  100 B()
%init:  100 C()
%init:  100 D()

%obs:  'Dphos' |D(s{p})|
%mod:  [true] do $TRACK 'Dphos' [true];
```

For a description of the Kappa syntax, see [5]. Briefly, bonds are noted as a shared number between brackets. For example, `A(x[1]), C(x[1])` means that `A` and `C` are bound through their respective site `x`. A dot between brackets, like `A(x[.])`, means that a site is free of any bond. States are given between braces. For example, `C(z{p})` means here that site `z` of `C` is phosphorylated. In the definition of rules, a forward slash indicates an edit. What appears before the slash is a precondition, the binding or state value that a site must take to allow the firing of a rule. After the slash is the new binding or state value taken by a site after the firing of the rule. Brackets and braces without a forward slash inside them represent required preconditions that are unchanged by the firing of the rule.

Fig. 1 shows a sketch of the toy model, which can be described as follows. Kinases `A` and `B` can bind to protein `C`, each through a different binding site. Both `A` or `B` can phosphorylate site `z` of protein `C` once they are bound. Protein `C` can bind to `D`, but only after `C` was phosphorylated. `A` or `B` can then phosphorylate `D` as well, but only if they are in a same molecular complex as `D`. The phosphorylation of protein `D` is set as the event of interest. This simplified model is representative of scaffolding as it can occur in cell signalling. Protein `C` for instance could contain SH2 domains [6] to recruit kinases which would in turn phosphorylate multiple residues within a complex.

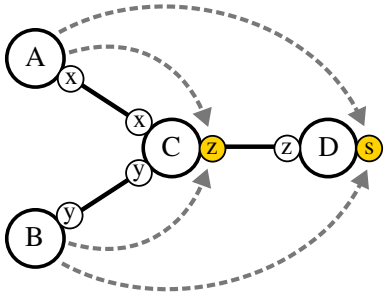


Fig. 1. Sketch of the toy model used to illustrate the method developed in this work.

2.2 Stories

Traces of computation from rule-based models can be sampled by the means of stochastic simulators [7,8]. Yet, such traces are not convenient to understand the mechanisms of signal processing because they contain satellite events that are unnecessary. They also describe precisely the order in which events occurred even when these events are causally independent. However, traces may be post-processed in order to extract relevant information. Event structures [9] abstract away the interleaving order between causally independent events. Then, irrelevant events may be discarded by using operational research techniques [3] or heuristic approaches [4]. The remaining events are the necessary steps required to reach the event of interest. Their representation in the form of a graph is called a story.

Fig. 2 shows the four stories corresponding to the four possible ways of obtaining phosphorylated `D` from the initial conditions of the toy model. Story 1 represents the case where kinase `A` phosphorylates both `C` and `D`. Story 2 is similar, but uses kinase `B` instead. Stories 3 and 4 represent cases where kinase `A` phosphorylates `C` and then kinase `B` phosphorylates `D`, or the other way around.

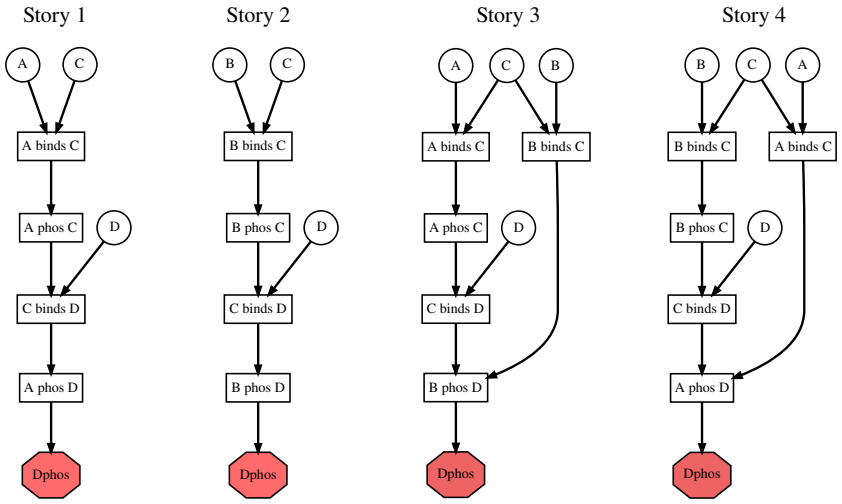


Fig. 2. The four possible stories to the phosphorylation of protein `D` in the toy model. Events are shown as rectangular nodes. Introduction nodes are shown as circles and represent the involvement of a new individual protein in the story. The event of interest appears as a red octagonal node. Edges represent precedence relationships. All graphs in this work were built with Graphviz [10].

Note that stories represent concurrency, or parallelism. When a given node has more that one incoming edge, those edges share an *and* relationship. In story 3 for instance, events `"C binds D"` *and* `"B binds C"` are both required to enable `"B phos D"`. Alternative paths cannot be represented within a story. They are rather shown as distinct stories.

3 Method

The story folding method presented in this work is implemented in the Python package KappaPathways [11]. The package is currently under development and still requires the implementation of additional functionalities before fully fledged pathway extraction from rule-based models can be performed.

3.1 Quotient of stories with concurrency

The main goal of this work is to build a compact representation of the results obtained from causality analysis of rule-based models. To do so, we seek to fold any arbitrarily large collection of stories into a single quotient graph. The first intuitive way to partition story events is according to the Kappa rules that they represent. That is, looking at the four stories from Fig. 2, all nodes with the same label are merged together.

Fig. 3, *left* shows the quotient obtained by simply partitioning story events according to their corresponding Kappa rule. While this graph does summarize some of the information from all the stories, it also clearly introduces ambiguities. For instance, readers could be mislead into thinking that completing event `"A binds C"` is sufficient to reach `"A phos D"`, while in reality `"C binds D"` is also necessary. This ambiguity arise because when a node in the quotient graph has more than one incoming edge, it is impossible to know whether each edge comes from distinct stories or concurrent branches of a same story.

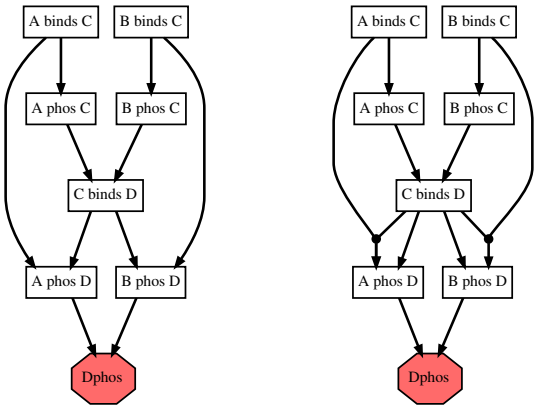


Fig. 3. Quotients of the four stories from Fig. 2. *Left*) Quotient obtained by partitioning event nodes according to their corresponding Kappa rule. *Right*) Quotient with hyperedges representing concurrency. Introduction nodes were removed for clarity.

Fig. 3, *right* illustrates a simple solution to the representation of concurrency in the quotient graph. Before the folding operation, any story edges that have the same target are regrouped in a single hyperedge with many sources and one target. The ensuing quotient allows a clear distinction between concurrent and alternative paths.

Still, the graph from Fig. 3, *right* remains misleading. It suggests paths that are not coherent with the model. For instance, the path `"A phos C"` →`"C binds D"` →`"B phos C"` seems allowed by itself. According to the model, it is instead only possible if `"B binds C"` also occurs concurrently. This inconsistent interpretation of the quotient graph tells us that a partitioning of events simply based on Kappa rules folds the stories too much. The next section presents additional event partitioning criteria that solve those ambiguities.

3.2 Relevant context

Looking back at stories 1 and 2 from Fig. 2 provides a hint into why folding too much produces a quotient with paths that are inconsistent with the model. Those two stories both pass through a same type of event, namely `"C binds D"`. However, their preceding events are different. Although the latest modification that both stories went through is the same, the accumulated state of the molecules involved up to that point is different. They may hence have different futures open to them. That is, not only the events themselves matter, but also the context that was built up in their past. Still, not all past context may be relevant. A good trade-off must be found in the information that is kept about the context of each event. Too much information leads to a blow-up in the quotient graph that may become unreadable or even too costly to compute, whereas too few information may not be discriminant enough. In this work, we propose that the appropriate amount of information to keep corresponds to the relevant context.

The relevant context of a given event consists in the context from its past which remains useful in its future. More precisely, this corresponds to the locally relevant context, which appears relevant within a same story. We also define the globally relevant context as the context from an event's past which is found useful in the future of other stories which pass through an event corresponding to the same Kappa rule. The information about context can be extracted from the trace that was used to produce the stories. The following steps describe how we obtain context for each event and determine its relevance.

3.2.1 Edit nodes First, for each story, the modifications that are performed by every event are extracted from the trace as illustrated on Fig. 4. Each individual modification is called an edit and is represented by a separate node connected to the event which it originates from. Edit nodes are labelled according to the modification that they correspond to using the Kappa syntax. Edges are then added from edit nodes to the downstream events which require them later in the story. Edges that do not correspond to any edge from the original story are referred to as transitive edges.

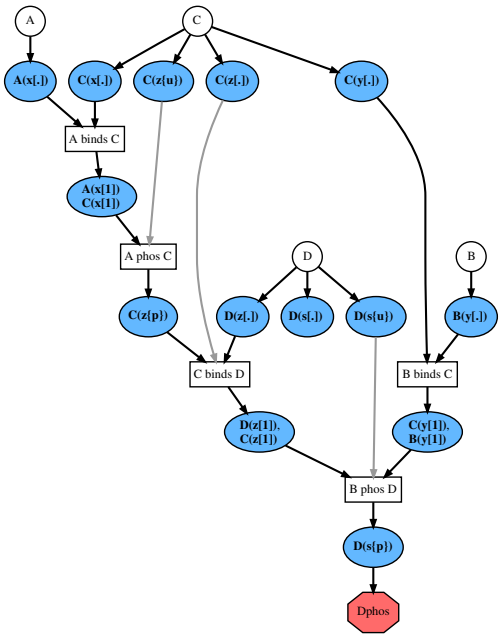


Fig. 4. Addition of edit nodes to story 3 from Fig. 2. Edit nodes are shown in blue and their labels represent modified sites in Kappa syntax. Transitive edges are shown in gray.

3.2.2 Locally relevant context Second, the locally relevant context is determined as illustrated on Fig. 5. The following substeps are performed iteratively from top to bottom on the graph with added edit nodes. a) The current node is selected as the first edit node from the top of the graph whose locally relevant context was not already computed, excluding edits coming from introduction nodes. b) The past context of the current node is gathered. It first consists in the immediate upstream edit nodes from the current node, ignoring transitive edges. Neighboring nodes are also added as the edit nodes coming from the same event or introduction node as any upstream node. Lastly, the context that was found for those upstream and neighbor nodes during previous iterations is added to the past context of the current node. c) The locally relevant context is found as any past context node that remains useful in the future of the current node. On the graph, this corresponds to any past context node that have at least one path to a node which is reachable from the current node. The path should however not pass through the current node or an edit node that is incompatible with the edit from the past context node.

Fig. 5, *left* shows the first iteration of the steps described above on story 3 from Fig. 2. It provides the locally relevant context associated with event `"A binds C"`. Two elements of context are found relevant, as displayed by the two green edges and the inscription `z[.]{u}` on the label of the highlighted current node. This indicates that the locally relevant context is that protein `C` must have its site `z` free of any bond, and also unphosphorylated. Those two edits from the past are useful for future events `"C binds D"` and `"A phos C"`, respectively. Also note the red dashed edges which indicate past context nodes whose path to the future of the current node was blocked by incompatible edits.

Fig. 5, *right* shows the second iteration of locally relevant context determination. It now focusses on the context of event `"A phos C"`. Only the edit node coming from event `"A binds C"` counts here as an immediate upstream node since the edge to `C(z{u})` is transitive. `C(z[.])` and `C(z{u})` are nevertheless subsequently added as past context because the were found as relevant during the previous iteration.

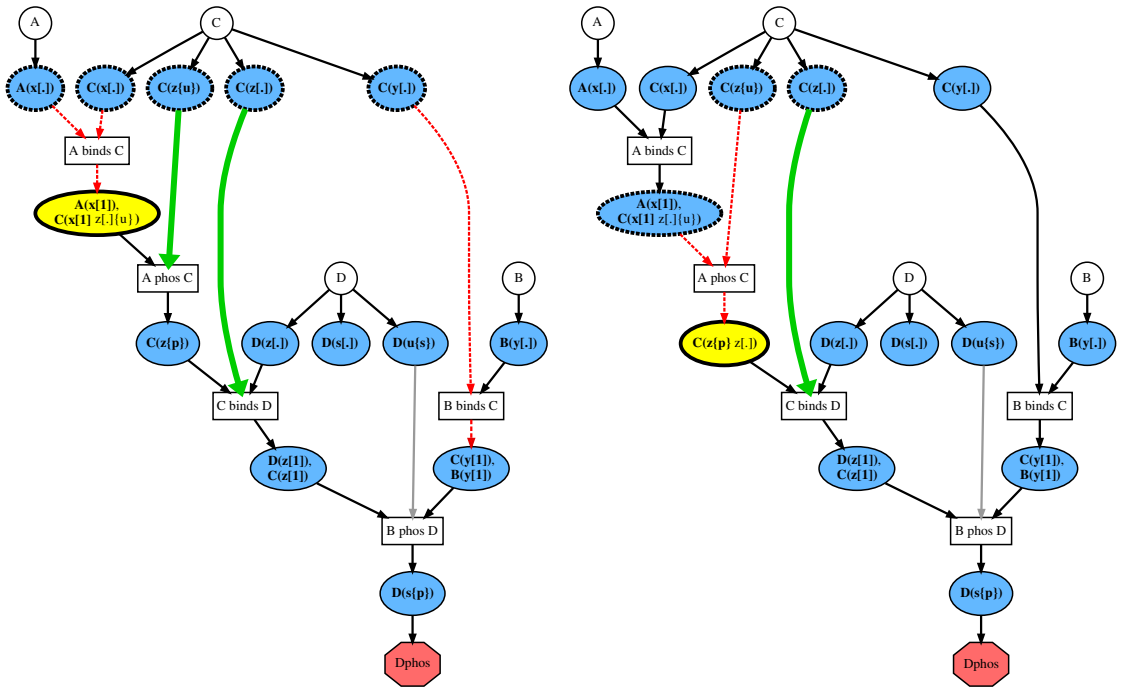


Fig. 5. Highlight of the nodes and edges involved in the determination of the locally relevant context. The first (*left*) and second (*right*) iterations are shown on story 3 from Fig. 2. The current node of each iteration is shown in yellow with bold border. Past context nodes have a bold dashed border. Paths from past context nodes to any node reachable by the current node are shown with thick green edges. Paths that are blocked by an edit node that is incompatible with the edit from the past context node are shown with dashed red edges. The locally relevant context obtained at the end of each iteration is written in thin font on the label of the current node.

3.2.3 Globally relevant context Third, the globally relevant context is evaluated. The goal is to harmonize what is considered relevant across all stories. Suppose an element of context which exists in the past of two different stories, but is found irrelevant in the first story and locally relevant in the second. Then, the first story must be revised, knowing that this given element of context is actually relevant when considering the whole system.

To do so, the total possible context of each edit is first computed. It corresponds to any context that was found locally relevant across all the edit nodes that represent a same edit among all stories. Then, the globally relevant context of a given edit node is found as the elements from the total context which exist in the past of that edit node within the story where it is found. Finally, elements of context are removed if they are found globally relevant across all edit nodes that represent a same edit.

Fig. 6 shows the four stories now with edit nodes containing the globally relevant context associated with each event. It turns out that the only relevant context is whether protein `C` was bound to kinase `A` or `B` when it got phosphorylated. Looking back at the toy model, this is precisely what was expected. Note that this context, the binding to `A` or `B`, was not locally relevant in story 3 as seen on Fig. 5. It was instead added as globally relevant from stories 1 and 2. Also, the locally relevant context `C(z[.])` from Fig. 5 was removed since it is equally present in all four stories and is hence not discriminating.

3.3 Quotient of contextualized stories

It is now possible to fold the contextualized stories from Fig. 6. Using edit nodes and their context defined in the previous, we can now partition the nodes in a way that will preserve the information propagation dictated by the model. Edit nodes across all stories are merged if they have the same edit and the same relevant context. Event nodes are merged if they correspond to the same Kappa rule and their target edit nodes also have the same relevant context.

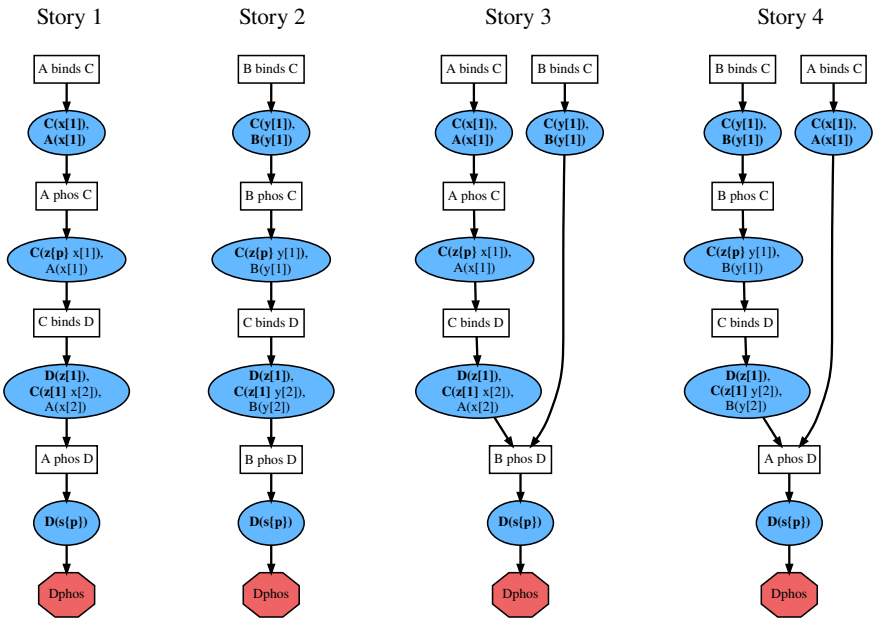


Fig. 6. The four stories from Fig. 2 annotated with relevant context on edit nodes. Introduction nodes were removed for clarity.

Fig. 7, *left* presents the resulting quotient. There are two nodes representing rule `"C binds D"`. One corresponds to the case where `C` was priorly bound to `A`, and to other to the case where it was bound to `B`. There is no way to read this graph by following a path that is not allowed by the model.

Fig. 7, *right* shows a more compact graph obtained by removing the contextual information about events that we have used to get a more precise quotient. We consider this last graph as the correct representation of the paths to the event of interest, as opposed to the two graphs from Fig. 3 which both lead to a wrong interpretation.

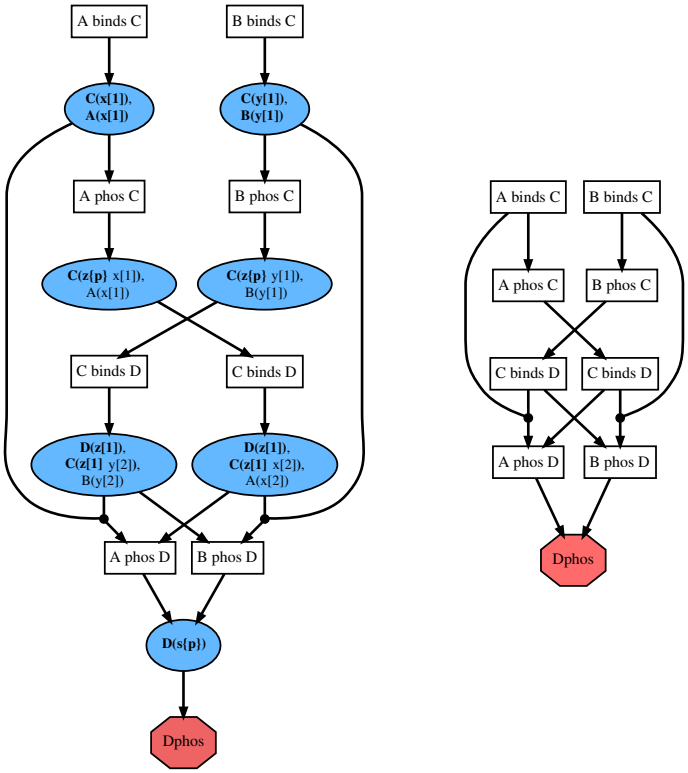


Fig. 7. Quotient graph obtained after considering relevant context. *Left*) Full version of the graph. *Right*) Compact version with edit nodes removed.

4 Conclusion

This work describes a method to build a compact representation for the visualisation of causality analysis results in rule-based modelling. A collection of stories provided by a current implementation of causality analysis [3,4] is folded into a single quotient graph. The main challenge is to find a partitioning of story events which maximizes compactness of the quotient graph without loosing important details about how information propagates in the model. The appropriate partitioning criterion suggested here is the relevant context, the context from an event's past which remains useful in its future. By folding stories based on their events type and relevant context, a quotient graph is obtained that faithfully represents all the paths to an event of interest.

Scalability of the method is important since its true usefulness lies in its application to large models. While rigorous optimization and benchmarking was not performed yet, we tested the current implementation on a model of human cell signalling comprising about three thousand Kappa rules. This model is considered large because representing it in a reaction-based setting, like in a Petri net, would lead to a combinatorial explosion with millions of nodes. Calculation time of quotient graphs for events of interest from this model is usually of the order of several minutes on an average personal computer. Those graphs are typically easy to visualize with less than fifty ordered nodes. For cases where the graph becomes larger and confusing to read, quantitative data from the model's execution can be used to prune away least impactful paths.

The story folding method reported in this paper is a milestone in a longer term objective to automatically extract biological pathways from rule-based models. Other functionalities are planned to be implemented within the KappaPathways [11] package to reach that goal. Those include counterfactual analysis [12], the representation of inhibitions from negative influences and the ordering of events participating in feedback loops.

Acknowledgements

This work has been partially supported by French ANR project DCore ANR-18-CE25-0007.

References

[1] Vincent Danos, Jérôme Feret, Walter Fontana, Russ Harmer, and Jean Krivine. Rule-based modelling of cellular signalling. In *CONCUR*, pages 17–41. Springer, 2007.

[2] Michael L. Blinov, Jin Yang, James R. Faeder, and William S. Hlavacek. Graph theory for rule-based modeling of biochemical networks. In *Transactions on Computational Systems Biology VII*, pages 89–106. Springer, 2006.

[3] Vincent Danos, Jérôme Feret, Walter Fontana, Russell Harmer, Jonathan Hayman, Jean Krivine, Chris Thompson-Walsh, and Glynn Winskel. Graphs, Rewriting and Pathway Reconstruction for Rule-Based Models. In *FSTTCS*, pages 276–288, 2012.

[4] Jonathan Laurent. KaFlow. https://github.com/jonathan-laurent/KaFlow.

[5] Pierre Boutillier, Mutaamba Maasha, Xing Li, Héctor F Medina-Abarca, Jean Krivine, Jérôme Feret, Ioana Cristescu, Angus G Forbes, and Walter Fontana. The Kappa platform for rule-based modeling. *Bioinformatics*, 34(13):i583–i592, 2018.

[6] Melany J. Wagner, Melissa M. Stacey, Bernard A. Liu, and Tony Pawson. Molecular mechanisms of sh2- and ptb-domain-containing proteins in receptor tyrosine kinase signaling. *Cold Spring Harb Perspect Biol.*, 5:a008987, 2013.

[7] V. Danos, J. Feret, W. Fontana, and J. Krivine. Scalable simulation of cellular signaling networks. In *APLAS*, pages 139–157. Springer, 2007.

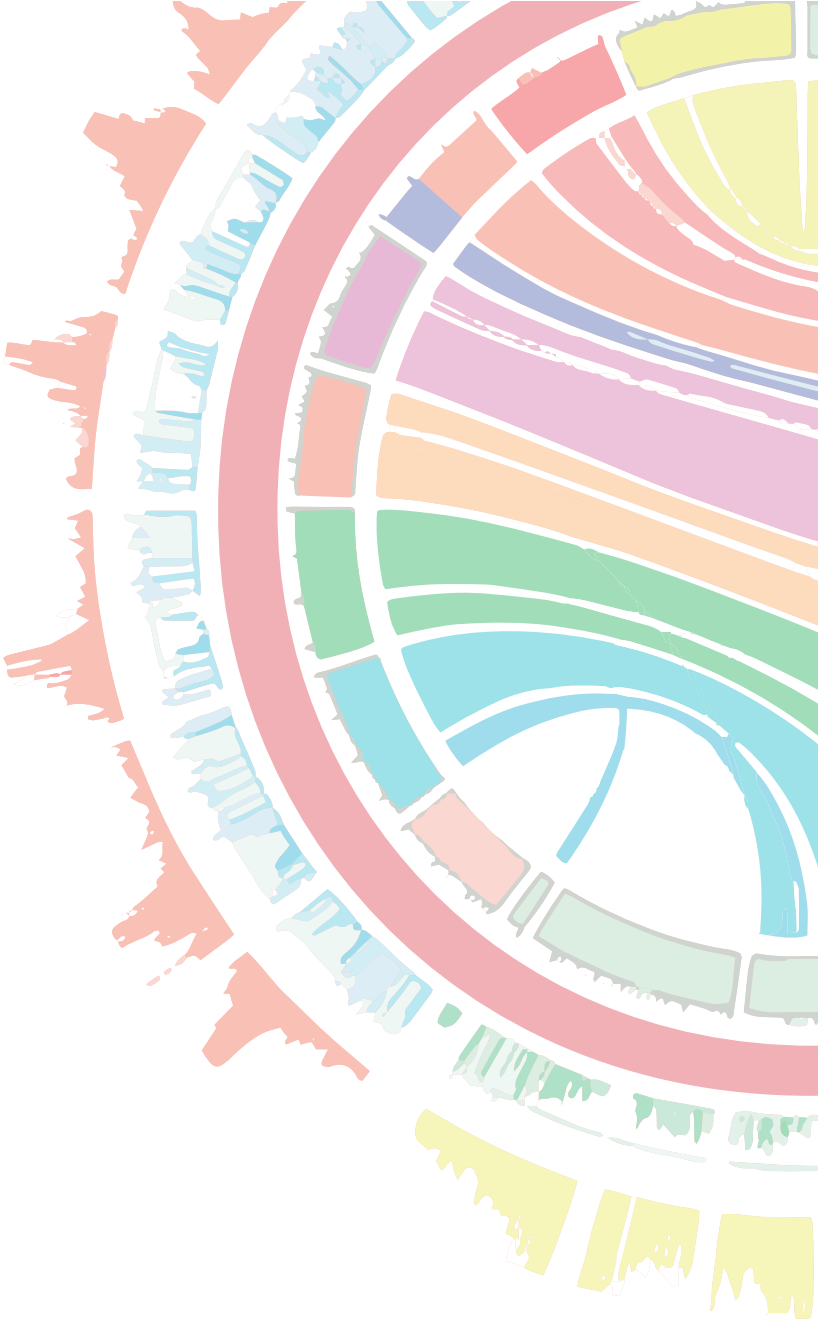
[8] James R. Faeder, Michael L. Blinov, and William S. Hlavacek. Rule-based modeling of biochemical systems with BioNetGen. *Methods in molecular biology*, 500:113–167, 2009.

[9] Glynn Winskel. Event structures. In *Advances in Petri Nets*, pages 325–392. Springer, 1986.

[10] John Ellson, Emden R. Gansner, Eleftherios Koutsofios, Stephen C. North, and Gordon Woodhull. Graphviz and dynagraph – static and dynamic graph drawing tools. In *Graph Drawing Software*, pages 127–148. Springer-Verlag, 2003.

[11] Sébastien Légaré. KappaPathways. https://github.com/slegare2/KappaPathways.

[12] Jonathan Laurent, Jean Yang, and Walter Fontana. Counterfactual resimulation for causal analysis of rule-based models. In *IJCAI*, pages 1882–1890. ijcai.org, 2018.



Session 7
Algorithms & sequence data structures I

UMI-Gen: a UMI-based read simulator for variant calling evaluation

Vincent SATER[1,3], Pierre-Julien VIAILLY[2,3], Thierry LECROQ[1], Philippe RUMINY[2,3], Élise PRIEUR-GASTON[1], Caroline BÉRARD[1] and Fabrice JARDIN[2,3]

[1] Normandie Univ, UNIROUEN, LITIS EA 4108, 76000 Rouen, France
[2] Centre Henri Becquerel, 76000 Rouen, France
[3] Normandie Univ, UNIROUEN, INSERM U1245 Team Genomics and Biomarkers of Lymphoma and Solid Tumors, 76000 Rouen, France

Corresponding Author: vincent.sater@gmail.com

Paper Reference: **Sater *et al.* (2020) UMI-Gen: a UMI-based read simulator for variant calling evaluation, Computational and structural Biotechnology Journal, 2020, 18:2270-2280.**
https://doi.org/10.1016/j.csbj.2020.08.011

Abstract

With Next Generation Sequencing becoming more affordable every year, NGS technologies asserted themselves as the fastest and most reliable way to detect Single Nucleotide Variants (SNV) and Copy Number Variations (CNV) in cancer patients. These technologies can be used to sequence DNA at very high depths thus allowing to detect abnormalities in tumor cells with very low frequencies. Multiple variant callers are publicly available and are usually efficient at calling out variants.

However, when frequencies begin to drop under 1%, the specificity of these tools suffers greatly as true variants at very low frequencies can be easily confused with sequencing or PCR artifacts. The recent use of Unique Molecular Identifiers (UMI) [1,2,3] in NGS experiments has offered a way to accurately separate true variants from artifacts. UMI-based variant callers are slowly replacing raw-read based variant callers as the standard method for an accurate detection of variants at very low frequencies. However, benchmarking done in the tool's publication are usually realized on real biological data in which real variants are not known, making it difficult to assess their accuracy.

We present UMI-Gen, a UMI-based read simulator for targeted sequencing paired-end data. UMI-Gen generates reference reads covering the targeted regions at a user customizable depth. After that, using a number of control files, it estimates the background error rate at each position and then modifies the generated reads to mimic real biological data. Finally, it will insert real variants in the reads from a list provided by the user.

References

1. Michael W. Schmitt et al. Detection of ultra-rare mutations in next-generation sequencing. *PNAS,* (36):14508-14513, 2012.
2. Yoji Kukita et al. High-fidelity target sequencing of individual molecules identified using barcode sequences: de novo detection and absolute quantification in plasma cell-free DNA from cancer patients. *DNA Research,* (4):269-277, 2015.
3. Aaron M. Newman et al. Integrated digital error suppression for improved detection of circulating tumor DNA. *Nature Biotechnology,* (34):547-555, 2016.

QUARTIC: QUick pArallel algoRithms for high-Throughput sequencIng data proCessing

Frederic Jarlier[1–4], Nicolas Joly[5], Nicolas Fedy[1–4,6], Thomas Maglalahes[1–4,6], Leonor Sirotti[1–4,6], Paul Paganiban[1–4,6], Firmin Martin[1–4,6], Michael Mc Manus[7] and Philippe Hupe[1–4,8]

[1] Institut Curie, Paris, F-75005, France
[2] U900, Inserm, Paris, F-75005, France
[3] PSL Research University, Paris, France
[4] Mines Paris Tech, Fontainebleau, F-77305, France
[5] Institut Pasteur, Paris, F-75015, France
[6] Université Paris Descartes, Paris, F-75006, France
[7] Intel Corporation, Hudson, Massachusetts, USA
[8] UMR144, CNRS, Paris, F-75005, Frances

Corresponding author: frederic.jarlier@curie.fr, philippe.hupe@curie.fr

Reference paper: **Jarlier F*et al.* (2020) QUARTIC: QUick pArallel algoRithms for high-Throughput sequencIng data proCessing. *F1000Research 2020, 9:240.*** https://doi.org/10.12688/f1000research.22954

Bioinformaticians are overwhelmed with high-throughput sequencing data. While they offer new insights to decipher the genome structure they also raise major challenges to use them for daily clinical practice care and diagnosis purposes as they are bigger and bigger. We implemented a software to reduce the time to delivery for the alignment and the sorting of high-throughput sequencing data. Our solution is implemented using Message Passing Interface and is intended for high-performance computing architecture. The software scales linearly with respect to the size of the data and ensures a total reproducibility with the traditional tools. For example, a 300X whole genome can be aligned and sorted within less than 9 hours with 128 cores with no overhead compare with the original BWA-MEM.

The alignment is based on the original BWA-MEM [1] algorithm. Some features were added to support multinodes and MPI-IO [3]. The main contribution of our software is the capability to distribute the IO and alignment works over several computing nodes avoiding the creation of intermediate files and the bottleneck of merging then in final step. The scalability is linear and a significant speed-up can be obtained by adding more ressources in term of CPU and memory, for instance by doubling the number of servers we divide by two the walltime.

The sorting algorithm implements a parallel bitonic sorting network and a parallel merge phase. During the bitonic phase, the data is distributed to all CPUs and parallel jobs work together to sort the SAM file. The results are also merged in parallel way via a Bruck [2] algorithm. In contrast to the traditionnal merge-sort, the idle time and IO bottlenecks are avoided. When compared to the traditionnal tools Samtools or Sambamba, the MPI tool results in a significant speed-up of six. Its scalability is also linear.

In this paper, we propose an implementation of the NGS alignment and sorting that make an efficient usage of High-Performance Computing architectures. At the storage level, a parallel file system (such as BeeGFs or Lustre) improves drastically the latencies when reading and writing concurrently with the MPI-IO framework. At the computer node level, we introduce MPI inter-node communication for a better synchronization and load balancing. The software is freely available on the Institut Curie github repository (https://github.com/bioinfo-pf-curie/mpiBWA, https://github.com/bioinfo-pf-curie/mpiSORT).

References

[1] Li H. "Fast and accurate long-read alignment with burrows-wheeler transform." In: *Bioinformatics* 26(5) (2010), pp. 589–595.

[2] Bruck J. "Efficient algorithms for all-to-all communications in multiport message-passing systems." In: *Parallel and Distributed Systems* 8(11) (1997).

[3] Gropp W. "A high-performance, portable implementation of the MPI message passing interface standard." In: *Parallel Computing* 22(6) (1996), pp. 789–828.

GraphUnzip: unzipping assembly graphs with long reads and Hi-C

Roland Faure[1], Nadège Guiglielmoni[1] and Jean-François Flot[1,2]
[1] Service Evolution Biologique et Ecologie, Université libre de Bruxelles, 1050 Brussels, Belgium
[2] Interuniversity Institute of Bioinformatics in Brussels - (IB)[2], 1050 Brussels, Belgium

Corresponding author: nadege.guiglielmoni@ulb.be

Abstract *Long reads and Hi-C have revolutionized the field of genome assembly as they have made highly contiguous assemblies accessible even for challenging genomes. As haploid chromosome-level assemblies are now commonly achieved for all types of organisms, phasing assemblies has become the new frontier for genome reconstruction. Several tools have already been released using long reads and/or Hi-C to phase assemblies, but they all start from a set of linear sequences and are ill-suited for non-model organisms with high levels of heterozygosity. We present GraphUnzip, a fast, memory-efficient and flexible tool to unzip assembly graphs into their constituent haplotypes using long reads and/or Hi-C data. As GraphUnzip only connects sequences that already had a potential link in the assembly graph, it yields high-quality gap-less supercontigs. To demonstrate the efficiency of GraphUnzip, we tested it on the human HG00733 and the potato Solanum tuberosum. In both cases, GraphUnzip yielded phased assemblies with improved contiguity.*

Keywords genome assembly, phasing, long reads, Hi-C

Introduction

The field of genomics is thriving and chromosome-level assemblies are now commonly achieved for all types of organisms, thanks to the combined improvements of sequencing and assembly methods. Chromosome-level assemblies are generally haploid, regardless of the ploidy of the genome. To obtain a haploid assembly of a multiploid (i.e. diploid or polyploid) genome, homologuous chromosomes are collapsed into one sequence. However, assemblers often struggle to collapse highly heterozygous regions, which leads to breaks in the assembly and duplicated regions [1]. Furthermore, haploid assemblies provide a partial representation of multiploid genomes: ideally, multiploid genomes should be phased rather than collapsed if the aim is to grasp their whole complexity [2].

The combination of low-accuracy long reads, such as Oxford Nanopore Technologies (ONT) reads and Pacific Biosciences (PacBio) Continuous Long Reads (CLRs), with proximity ligation (Hi-C) reads has made chromosome-level assemblies accessible for all types of organisms. The latest development of PacBio, high-accuracy long circular consensus sequencing (CCS) reads (a.k.a. HiFi), is now starting to deliver highly contiguous phased assemblies [3,4,5]. Hi-C scaffolding is commonly used in genome assembly projects to obtain chromosome-level scaffolds. This approach relies on the interaction frequency in the genome and these interactions are heightened between loci belonging to the same chromosome [6]. Based on this principle, alleles can be associated using their interaction frequencies.

A first approach to phase assemblies is called trio-binning and uses sequencing data from the individual and its parents to retrieve haplotypes [7]; yet this method is unavailable when the parents cannot be identified, or for asexual species. Existing tools are able to use either long reads (Falcon-Unzip [8], WhatsHap [9]) or Hi-C reads (Falcon-Phase [10], ALLHiC [11]) for phasing assemblies, but they are limited to phasing local variants or well-identified haplotypes and are not suited for complex, highly heterozygous genomes. WhatsHap takes as input a collapsed assembly and searches for alternative haplotypes. As collapsing haplotypes can be too difficult for highly heterozygous regions, it seems more intuitive to phase these assemblies *de novo*. FALCON-Unzip and FALCON-Phase offer this alternative, yet they are dependant on the output of the FALCON assembler and cannot be combined

with other assemblers.

We present GraphUnzip, a new tool to phase assemblies using long reads and/or Hi-C. GraphUnzip implements a radically new approach to phasing that starts from an assembly graph instead of a set of linear sequences. In an assembly graph, heterozygous regions result in bubbles every time the assembler is unable to collapse the haplotypes or to choose one of them. GraphUnzip "unzips" the graph, meaning that it separates the haplotypes by duplicating homozygous regions that have been collapsed and partitioning heterozygous regions into haplotypes. This tool is based on a simple principle that was implemented in many scaffolders since SSPACE [12]: long-range data (mate-pair reads, long reads, linked reads, proximity ligation...) provide information on the linkage between contigs that can be used to group and orient them into scaffolds. As GraphUnzip takes as input and produces as output assembly graphs, it only connects contigs that are actually adjacent in the genome and yields gap-less scaffolds, i.e. supercontigs. GraphUnzip is compatible with any assembler that produces an assembly graph. We tested GraphUnzip on the genomes of the human HG00733 and the potato *Solanum tuberosum*. GraphUnzip is available at github.com/nadegeguiglielmoni/GraphUnzip.

Methods

Inputs

GraphUnzip requires an assembly graph in GFA (Graphical Fragment Assembly) format. The Hi-C input is a sparse matrix, such as the one obtained when processing the reads with hicstuff [13]. hicstuff also provides a module to convert other file formats (e.g. cool, a common Hi-C format) to a sparse matrix. The long reads are mapped to the assembly graph using GraphAligner [14].

Overview of GraphUnzip

In an assembly graph, contigs that are inferred to be adjacent or to overlap in the assembly are connected with edges. However, some of these connections between contigs may be artefacts. To discriminate correct edges from erroneous ones, GraphUnzip relies on long reads and/or Hi-C data. These data are translated into interactions between contigs: the strength of interaction between two contigs is defined as the number of long reads bridging both contigs when using long reads as input; and as the number of Hi-C contacts between the two contigs when using Hi-C as input. In both cases, a strong interaction is a sign of proximity on the genome.

GraphUnzip first builds one or two interaction matrices containing all pairwise interactions between contigs, depending on whether long-read data, Hi-C data or both are provided (Figure 1). In the next step, GraphUnzip iteratively reviews all contigs and their edges. The strength of an edge i is computed based on the strength of interaction between the contigs it connects. A high strength supports the reality of the link, while a low strength may signal an artefactual edge. When a contig has several edges at one of its extremities, these edges are compared in a pairwise fashion. This comparison uses two user-provided thresholds: the rejection threshold T_R and the acceptance threshold T_A, where $T_R < T_A$. Considering two edges X and Y and their respective strengths $i(X)$ and $i(Y)$, if $i(X) < i(Y)$, Y is considered strong; if $i(X)/i(Y) < T_R$, then X is considered weak, else, if $T_R \leq i(X)/i(Y) < T_A$, X is flagged as dubious. X is labelled as strong when $i(X)/i(Y) \geq T_A$. The algorithm thereafter considers weak edges as artefacts that do not actually exist in the genome, whereas strong edges represent true connections. If both long reads and Hi-C input data are provided, strengths based on long reads are used first because they are more reliable locally, and strengths based on Hi-C are only used if some edges are flagged as dubious.

Edges identified as weak in the previous calculation are removed. Then, every contig that has more than one strong edge and no dubious edge at one end is duplicated as many times as the number of these strong edges. Such contigs are typically collapsed homozygous regions that need to be present in several copies to be included in every haplotypes. All the copies retain the edges of the original

contig at its other end. This entails that the duplication of contigs creates many new (and potentially artefactual) edges. Contigs that are unambiguously linked are merged in supercontigs that will be handled as regular contigs thereafter.

When assessing the strength of two putative edges (S_1,S_2) and (S_1,S_3) connecting the supercontigs S_1, S_2, and S_3, the strength of these edges are calculated as the strength of interaction between contigs in S_1 and contigs present in S_2 but not in S_3 (and vice versa). For example, in the third step of Figure 1, when trying to associate supercontig a-b to either d-e or d'-f, only the interactions between the supercontig a-b and the contigs e and f are considered. Interactions between the supercontig a-b and the contigs d and d' are not considered in the calculation because d and d' actually originate from the duplication of a collapsed region.

All contigs and edges are iteratively processed s times to phase the assembly, where s is a user-provided parameter. Because extremely long contigs tend to share a significant number of Hi-C contacts even if they are not adjacent, we observed that in extreme cases the algorithm could join two chromosomes by their telomeric ends. The Hi-C matrix is used at the end of the process to detect such chimeric connections in the assembly graph, based on low Hi-C interactions, and break them.

Homo sapiens HG00733 assemblies

We used HiFi, ONT and Hi-C reads from [15]. HiFi reads were assembled using hifiasm with the parameter `-l 0`, and the resulting p_utg assembly graph was used for downstream analyses. All HiFi reads and the ONT reads longer than 30 kb were mapped to the assembly using GraphAligner with the parameter `-x vg`. Hi-C reads were processed with hicstuff using the parameters `--aligner bowtie2 --enzyme 200 --iterative`. GraphUnzip was run with parameters `--accept 0.10 --reject 0.05 --exhaustive --whole_match --minimum_match 0.8`. All non-ambiguous paths in the GFA were merged using Bandage. The assemblies were compared to the DipAsm reference [16] using QUAST v5.0.2 [17] with the parameters `-m 0 --eukaryote --large --min-identity 99.9`.

Solanum tuberosum assemblies

HiFi, ONT and Hi-C reads published in [18] were retrieved from the NCBI Sequence Read Archive with the Bioproject accession number PRJNA573826. The HiFi reads were assembled using hifiasm with the parameter `-l 0`, and the p_utg assembly graph was used for downstream analyses. All HiFi reads and the ONT reads longer than 25 kb were mapped to the assembly using GraphAligner with the parameter `-x vg`. Hi-C reads were processed with hicstuff using the parameters `--aligner bowtie2 --enzyme MboI --iterative`. GraphUnzip was run with parameters `--accept 0.40 --reject 0.10 --exhaustive --whole_match --minimum_match 0.8`. All non-ambiguous paths in the GFA were merged using Bandage. To check the output of GraphUnzip, we mapped the published assembly to the assembly graph using GraphAligner. We used calN50 (available at github.com/lh3/calN50) to compute the NG50 against the published assembly size of 1.67 Gb [18]. BUSCO v4 [19] was run with parameters `-m genome --long` against the dataset viridiplantae odb10.

Computational performance

RAM usage and CPU time were measured with the command `/usr/bin/time -v` on a desktop computer with 128 GB of RAM and a i9-9900X 3.5 GHz processor.

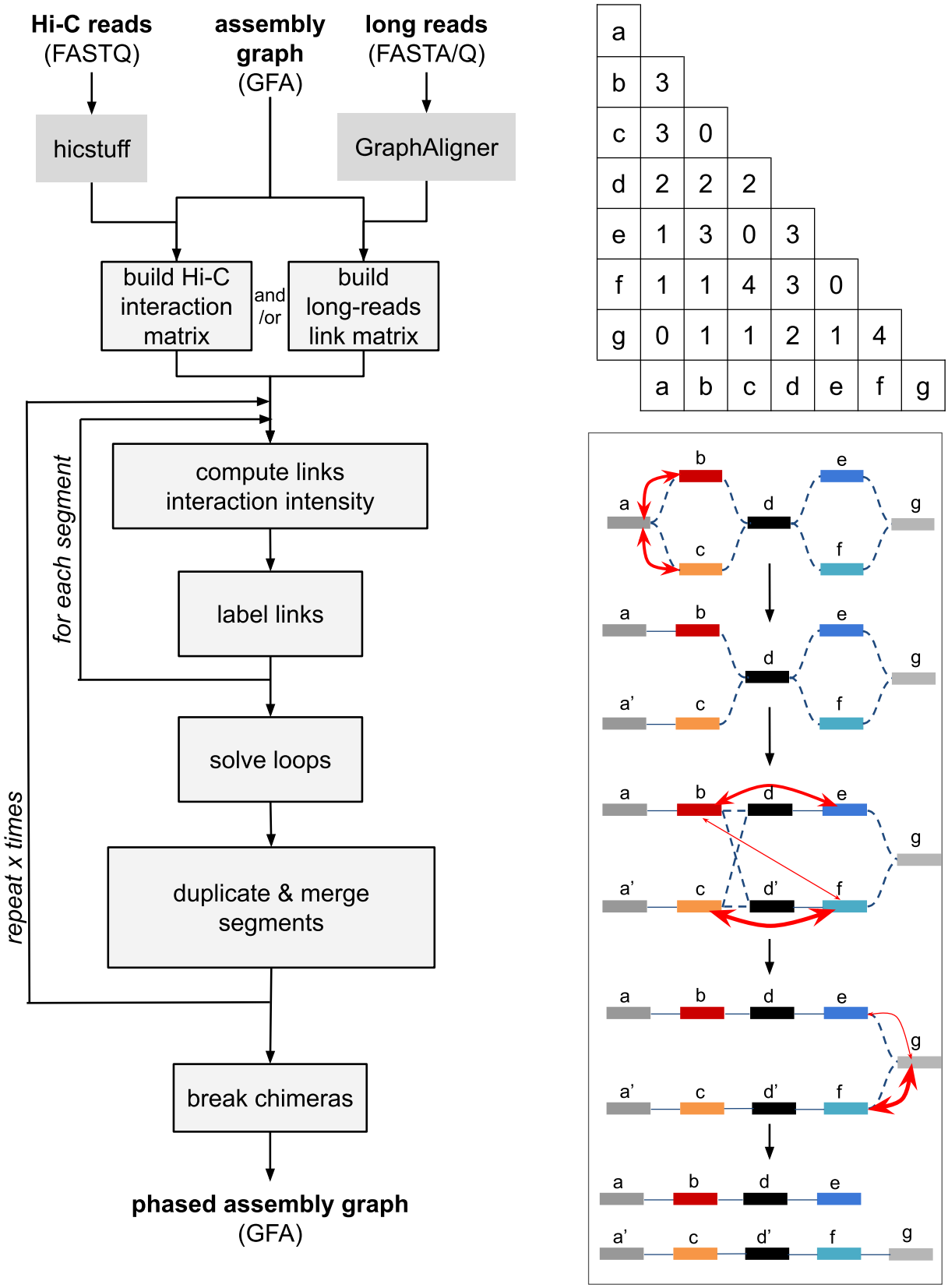


Fig. 1. Description of GraphUnzip: workflow of the program (left), interaction matrix (top right), and overview of the algorithm to discriminate links (bottom right). This example algorithm analyzes the potential links between the segments a, b, c, d, e, f, g. The red arrows represent the intensity of interactions between the segments, computed based on the values in the matrix.

Tab. 1. Assembly metrics of *Homo sapiens* HG00733 compared with the DipAsm reference.

Assembly	GraphUnzip	Size	N50	NA50	Misassemblies	CPU	RAM
Reference	-	5.9 Gb	27.8 Mb	27.8 Mb	84	-	-
hifiasm	-	5.5 Gb	397 kb	343 kb	9146	-	-
	ONT + Hi-C	6.2 Gb	1.5 Mb	1.2 Mb	8091	33min 46s	23.5 GB

Results

Homo sapiens HG00733

We compared the hifiasm + GraphUnzip assembly of the human HG00733 genome with a published reference obtained using DipAsm, based on the N50, the NA50 and the number of misassemblies. The N50 represents the contiguity of the assembly: it is defined as the length of the largest contig for which 50% of the assembly size is contained in contigs of equal or greater length. The NA50 is the N50 of the assembly broken at every misassembly (compared to a reference). GraphUnzip increased the size of the hifiasm assembly (from 5.5 Gb to 6.2 Gb), and the N50 rose as well (from 397 kb to 1.2 Mb) (Table 1). The NA50 was improved while the number of misassemblies decreased in the GraphUnzip supercontigs. Notably, the reference assembly size is only 5.9 Gb, while the GraphUnzip assembly reaches 6.2 Gb, which is the expected size for a phased human genome.

We also tried an assembly of the HiFi reads with Flye, but the draft assembly was only 2.9 Gb, little below half the expected size, which indicates that the haplotypes were nearly completely collapsed. A good candidate assembly for GraphUnzip should have uncollapsed heterozygous regions, as GraphUnzip is not able to retrieve a missing haplotype in collapsed heterozygous regions and can only duplicate the collapsed region, leading in that case to a suboptimal result.

Solanum tuberosum

Tab. 2. Assembly metrics of *Solanum tuberosum*. The NG50 values were computed based on an estimated genome size of 1.67 Gb.

Assembly	GraphUnzip	Size	NG50	BUSCO Single	BUSCO Dup.	CPU	RAM
Reference	-	1.67 Gb	66.1 Mb	21.6%	76.9%	-	-
hifiasm	-	1.51 Gb	2.2 Mb	21.2%	77.9%	-	-
	HiFi	1.69 Gb	3.7 Mb	7.1%	91.5%	16s	0.2 GB
	ONT	1.67 Gb	3.4 Mb	6.8%	92.2%	52s	0.2 GB
	Hi-C	1.69 Gb	5.6 Mb	7.8%	91.5%	38min 27s	11.5 GB
	HiFi + Hi-C	1.69 Gb	4.9 Mb	9.4%	89.4%	39min 59s	11.5 GB
	ONT + Hi-C	1.73 Gb	5.9 Mb	7.3%	91.8%	39min 10s	11.5 GB

We tested GraphUnzip on the diploid genome of the potato *Solanum tuberosum* RH89-039-16, for which a phased assembly of 1.67 Gb [18] was recently published. We assembled the HiFi reads with hifiasm and then ran GraphUnzip using the HiFi, ONT and/or Hi-C reads. The draft assembly was 1.51 Gb, and after phasing with GraphUnzip, the assembly size rose to 1.67-1.73 Gb (Table 2). In this case, we compared the NG50s, a value similar to N50 but based on a reference genome size rather than the assembly size. GraphUnzip increased the contiguity: from 2.2 Mb, the NG50 reached 3.4 to 5.9 Mb. The combination of both ONT and Hi-C reads yielded the highest NG50. Hi-C reads improved the contiguity better than long reads. The overall BUSCO completeness of the GraphUnzip supercontigs was slightly improved compared to the reference: 98.6-99.3% against 98.5% for the reference, and the number of duplicated BUSCO features was higher as well (89.4-92.2% against 76.9%). We mapped the published assembly to the GraphUnzip assembly graph obtained when using Hi-C and ONT reads. We found that there were no differences in phasing between the two assemblies. However, some regions that were phased by hifiasm and GraphUnzip were collapsed in the published assembly. This result, in conjunction with the higher number of duplicated features, indicates that GraphUnzip

led to an improved phased assembly.

Computational performance

For both the human and *Solanum tuberosum* genomes, GraphUnzip required limited computational resources as it ran in less than 1 hour on a single thread and used up to 23.5 GB of memory. For *Solanum tuberosum*, the run time was also shorter when using only long reads (less than a minute). The longer run time when using Hi-C reads was due to the building of the interaction matrix. As this interaction matrix is outputted by the program, this file can be reused for other runs, which will consequently finish faster. Therefore, users can try several sets of parameters to optimize the result, with short runtimes.

Conclusion

GraphUnzip is a flexible tool that can phase assemblies of high-accuracy long reads with long reads and/or Hi-C. A limitation of GraphUnzip is that it does not necessarily reach chromosome-level assemblies like most Hi-C scaffolders do, but it aims instead to produce more contiguous gap-less supercontigs by fully exploiting assembly graphs. As genome projects now usually include long reads and Hi-C to obtain chromosome-level assemblies, GraphUnzip can easily be integrated in assembly projects to obtain *de novo* phased assemblies for non-model organisms.

Acknowledgments

This project was funded by the Horizon 2020 research and innovation program of the European Union under the Marie Skłodowska-Curie grant agreement No 764840 (ITN IGNITE, www.itn-ignite.eu). Part of this analysis was performed on computing clusters of the Leibniz-Rechenzentrum (LRZ) and the Consortium des Équipements de Calcul Intensif (CÉCI) funded by the Fonds de la Recherche Scientifique de Belgique (F.R.S.-FNRS) under Grant No. 2.5020.11.

References

[1] Nadège Guiglielmoni, Antoine Houtain, Alessandro Derzelle, Karine Van Doninck, and Jean-François Flot. Overcoming uncollapsed haplotypes in long-read assemblies of non-model organisms. *BMC Bioinformatics*, 22(1):1–23, 2021.

[2] Xingtan Zhang, Ruoxi Wu, Yibin Wang, Jiaxin Yu, and Haibao Tang. Unzipping haplotypes in diploid and polyploid genomes. *Computational and Structural Biotechnology Journal*, 18:66–72, 2020.

[3] Mikhail Kolmogorov, Jeffrey Yuan, Yu Lin, and Pavel A Pevzner. Assembly of long, error-prone reads using repeat graphs. *Nature Biotechnology*, 37(5):540–546, 2019.

[4] Haoyu Cheng, Gregory T Concepcion, Xiaowen Feng, Haowen Zhang, and Heng Li. Haplotype-resolved *de novo* assembly using phased assembly graphs with hifiasm. *Nature Methods*, pages 1–6, 2021.

[5] Sergey Nurk, Brian P Walenz, Arang Rhie, Mitchell R Vollger, Glennis A Logsdon, Robert Grothe, Karen H Miga, Evan E Eichler, Adam M Phillippy, and Sergey Koren. HiCanu: accurate assembly of segmental duplications, satellites, and allelic variants from high-fidelity long reads. *Genome Research*, 30(9):1291–1305, 2020.

[6] Jean-François Flot, Hervé Marie-Nelly, and Romain Koszul. Contact genomics: scaffolding and phasing (meta) genomes using chromosome 3d physical signatures. *FEBS Letters*, 589(20):2966–2974, 2015.

[7] Sergey Koren, Arang Rhie, Brian P Walenz, Alexander T Dilthey, Derek M Bickhart, Sarah B Kingan, Stefan Hiendleder, John L Williams, Timothy PL Smith, and Adam M Phillippy. *De novo* assembly of haplotype-resolved genomes with trio binning. *Nature Biotechnology*, 36(12):1174–1182, 2018.

[8] Chen-Shan Chin, Paul Peluso, Fritz J Sedlazeck, Maria Nattestad, Gregory T Concepcion, Alicia Clum, Christopher Dunn, Ronan O'Malley, Rosa Figueroa-Balderas, Abraham Morales-Cruz, et al. Phased diploid genome assembly with single-molecule real-time sequencing. *Nature Methods*, 13(12):1050–1054, 2016.

[9] Murray Patterson, Tobias Marschall, Nadia Pisanti, Leo Van Iersel, Leen Stougie, Gunnar W Klau, and Alexander Schönhuth. WhatsHap: weighted haplotype assembly for future-generation sequencing reads. *Journal of Computational Biology*, 22(6):498–509, 2015.

[10] Zev N Kronenberg, Arang Rhie, Sergey Koren, Gregory T Concepcion, Paul Peluso, Katherine M Munson, Stefan Hiendleder, Olivier Fedrigo, Erich D Jarvis, Adam M Phillippy, et al. Extended haplotype phasing of de novo genome assemblies with FALCON-Phase. *bioRxiv*, 2019.

[11] Xingtan Zhang, Shengcheng Zhang, Qian Zhao, Ray Ming, and Haibao Tang. Assembly of allele-aware, chromosomal-scale autopolyploid genomes based on Hi-C data. *Nature Plants*, 5(8):833–845, 2019.

[12] Marten Boetzer, Christiaan V Henkel, Hans J Jansen, Derek Butler, and Walter Pirovano. Scaffolding pre-assembled contigs using sspace. *Bioinformatics*, 27(4):578–579, 2011.

[13] Cyril Matthey-Doret, Lyam Baudry, Amaury Bignaud, Axel Cournac, Rémi Montagne, Nadège Guiglielmoni, Théo Foutel-Rodier, and Vittore F. Scolari. koszullab/hicstuff, October 2020.

[14] Mikko Rautiainen and Tobias Marschall. GraphAligner: rapid and versatile sequence-to-graph alignment. *Genome Biology*, 21(1):1–28, 2020.

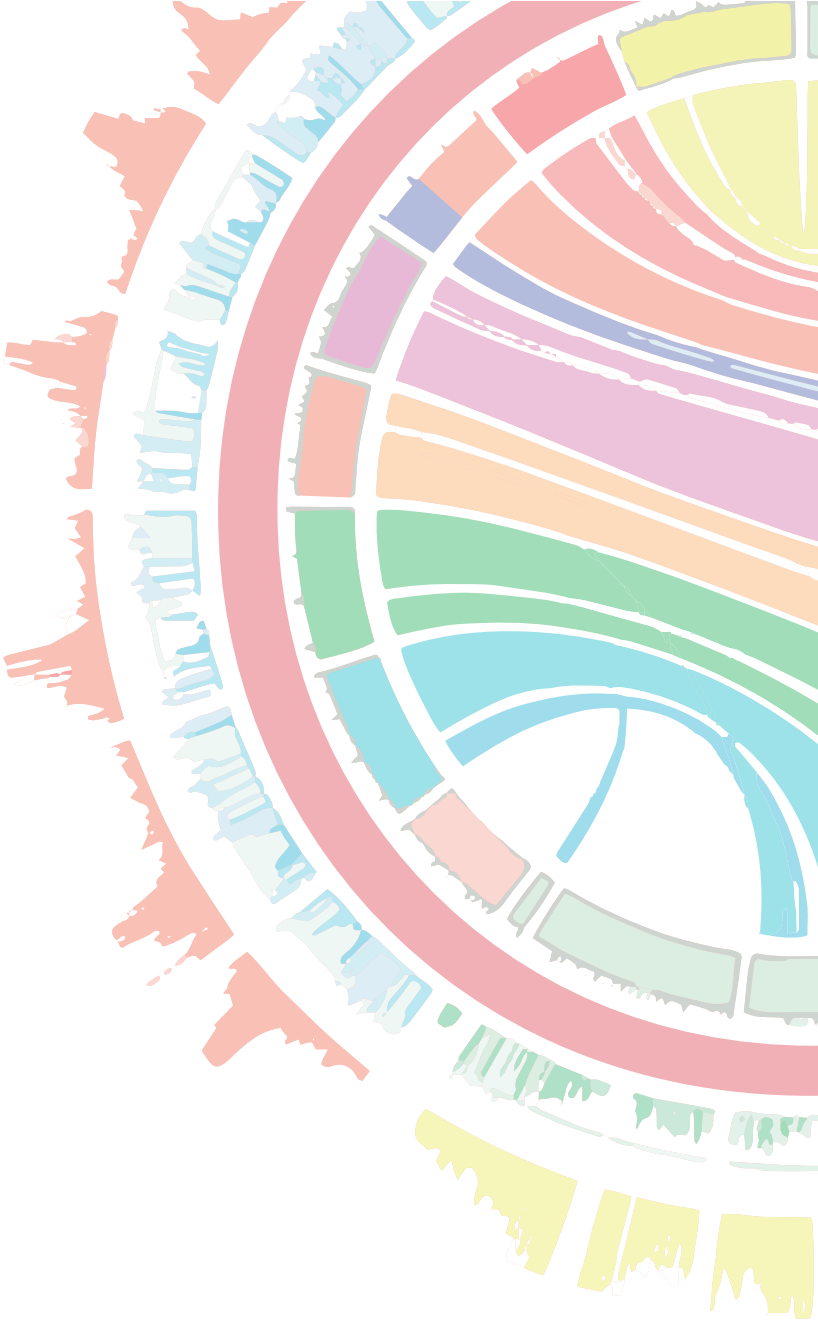
[15] David Porubsky, Peter Ebert, Peter A Audano, Mitchell R Vollger, William T Harvey, Pierre Marijon, Jana Ebler, Katherine M Munson, Melanie Sorensen, Arvis Sulovari, et al. Fully phased human genome assembly without parental data using single-cell strand sequencing and long reads. *Nature Biotechnology*, 39(3):302–308, 2021.

[16] Shilpa Garg, Arkarachai Fungtammasan, Andrew Carroll, Mike Chou, Anthony Schmitt, Xiang Zhou, Stephen Mac, Paul Peluso, Emily Hatas, Jay Ghurye, et al. Chromosome-scale, haplotype-resolved assembly of human genomes. *Nature Biotechnology*, 39(3):309–312, 2021.

[17] Alexey Gurevich, Vladislav Saveliev, Nikolay Vyahhi, and Glenn Tesler. QUAST: quality assessment tool for genome assemblies. *Bioinformatics*, 29(8):1072–1075, 2013.

[18] Qian Zhou, Dié Tang, Wu Huang, Zhongmin Yang, Yu Zhang, John P Hamilton, Richard GF Visser, Christian WB Bachem, C Robin Buell, Zhonghua Zhang, et al. Haplotype-resolved genome analyses of a heterozygous diploid potato. *Nature Genetics*, pages 1–6, 2020.

[19] Felipe A Simão, Robert M Waterhouse, Panagiotis Ioannidis, Evgenia V Kriventseva, and Evgeny M Zdobnov. BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. *Bioinformatics*, 31(19):3210–3212, 2015.



Session 8
Meta-omics & microbial genomics

No microbe is an island: metabolic complementarity in the microbiota and identification of key players

Arnaud Belcour*[1], Clémence Frioux[1,2,3], Méziane Aite[1], Anthony Bretaudeau[1,4], Falk Hildebrand[2,3] and Anne Siegel[1]
[1] Univ Rennes, Inria, CNRS, IRISA, F-35000 Rennes, France
[2] GMH Quadram Institute and Digital Biology Earlham Institute, NR4 7GJ, Norwich, United Kingdom
[3] Inria Bordeaux Sud-Ouest, France
[4] UMR IGEPP INRAE, BIPAA Rennes, and Inria IRISA Genouest Core Facility Rennes, France

Corresponding author: clemence.frioux@inria.fr

Reference paper: **Belcour*, Frioux*** ***et al.*** **(2020) Metage2Metabo, microbiota-scale metabolic complementarity for the identification of key species.** ***eLife.*** https://doi.org/10.7554/eLife.61968

Host-associated and environmental microbiotas are complex ecosystems in which a variety of interactions occur between microbial members. As the culture of all these microbes is impractical, omics experiments and in particular shotgun metagenomics are the main sources of information to decipher the web of interactions in these large communities [1]. The number of available metagenomes and the methodological improvements in reconstruction of metagenome-assembled genomes (MAGs) makes it possible to apply metabolic network modelling in microbiota [2]. However, the remaining imperfection of automatically-generated data - both for reconstructed MAGs and reconstructed metabolic networks - still impairs the applicability of such models.

In [3], we present Metage2Metabo (M2M), a pipeline dedicated to the metabolic screening of large communities of microbes, and apply it to several use-cases and datasets in order to demonstrate its versatility. M2M automatically *reconstructs and screens the metabolic potential* of thousands of microbes, considered both individually and as a community, in order to evaluate the metabolic gain brought by cooperative interactions. These metabolites that cannot be producible individually or other sets of compounds can be used as an objective to identify minimal communities predicted to sustain their producibility. As up to millions of equivalent minimal communities can exist due to the combinatorics of the problem, we solve it using Answer Set Programming in order to retrieve the key species (KS), that are all microbes occurring in at least one of such communities. We can further distinguish KS by identifying those that occur in every minimal community, thereby targeting the metabolic key players within the original microbiome. We illustrated our methods using various genomic and metagenomic datasets. Applied to 1,520 high-quality draft reference genomes of the human gut microbiota, we showed the screening potential of M2M and studied key species for several categories of metabolic end-products. In addition, we compared the robustness of M2M predictions on degraded MAGs demonstrating that M2M is *applicable to metagenomics*. Finally, we used M2M to screen the metabolism associated to the gut microbiota of individuals in a disease context.

Functionally describe and reduce the complexity of large communities is a critical matter in the journey towards a better understanding of microbiotas organisation. This work provides a step in that direction by identifying functions and species of interest in microbial ecosystems.

Acknowledgements

GenOuest bioinformatics core facility, NBI Computing infrastructure for Science (CiS), and the Bioinformatics Research Group of SRI International.

References

[1] Stephen Nayfach, Zhou Jason Shi, Rekha Seshadri, Katherine S. Pollard, and Nikos C. Kyrpides. New insights from uncultivated genomes of the global human gut microbiome. *Nature*, 568(7753):505–510, 2019.

[2] Clémence Frioux, Dipali Singh, Tamas Korcsmaros, and Falk Hildebrand. From bag-of-genes to bag-of-genomes: metabolic modelling of communities in the era of metagenome-assembled genomes. *Computational and Structural Biotechnology Journal*, 18:1722–1734, 2020.

[3] Arnaud Belcour, Clémence Frioux, Méziane Aite, Anthony Bretaudeau, Falk Hildebrand, and Anne Siegel. Metage2Metabo, microbiota-scale metabolic complementarity for the identication of key species. *eLife*, 9:e61968, 2020.

Shallow Shotgun Metagenomics as a cost-effective and accurate alternative to WGS for taxonomic profiling and clinical diagnosis

Benoit GOUTORBE[1,2,3], Anne-Laure ABRAHAM[1], Mahendra MARIADASSOU[1], Anne PLAUZOLLES[2], Ghislain BIDAUT[3], Philippe HALFON[2] and Sophie SCHBATH[1]

[1] Université Paris-Saclay, INRAE, MaIAGE, 78350, Jouy-en-Josas, France

[2] Laboratoire Alphabio, 1 rue Melchior Guinot, 13003, Marseille, France

[3] Centre de Recherche en Cancérologie de Marseille, Cibi plateform, Aix Marseille Université U105, Inserm U1068, CNRS UMR7258, Institut Paoli Calmettes, Marseille, France

Corresponding author: benoit.goutorbe@inrae.fr

Abstract *Shallow shotgun metagenomics has been recently suggested as a promising strategy to study human microbiota, providing nearly identical taxonomic profiles than deep shotgun metagenomics with a sequencing cost similar to metabarcoding. With shallow sequencing approach (typically <1M reads/samples), taxonomic profiles are directly built by mapping reads on a catalog of reference genomes, without assembly step.*

In the present study, we first used simulated data set to design a dedicated workflow in order to obtain reliable taxonomic profiles from shallow sequencing reads. We propose a novel data-driven filtering method based on machine learning techniques that largely outperformed basic filtering methods. We then used this approach on 3 real data sets, covering patients from several continents and clinical conditions. Even if one loses some information like rare taxa, our results clearly show that shallow shotgun metagenomics is able to correctly retrieve structures like differences between groups of patients and diagnosis-like classification.

Keywords Shallow shotgun metagenomics, Gut microbiota, Sequencing depth, Clinical research

1 Introduction

Allowing culture-free analysis of microbial ecosystems, high throughput sequencing revolutionized our comprehension of the role that plays human associated microbiota in health and disease. It is nowadays a very active clinical research field, with thousands of studies carried out each year, covering many diseases [1] [2] [3]. Two sequencing strategies emerged to study microbiota, and the choice depends on the sequencing cost, the size of the cohort, the expected level of taxonomic resolution and, when possible, functional annotation. On the one hand, metabarcoding, which consists in targeted sequencing of a phylogenetic marker (often rRNA gene 16S for bacteria, ITS for fungi), is a very cost-efficient way to characterize diversity within and between samples and to obtain an approximate taxonomic identification of microorganisms (often down to the genus level). On the other hand, shotgun sequencing consists in sequencing all DNA material present in an environment, which allows deeper taxonomic resolution (species, or even strain level), functional profiling (identification and quantification of genes, metabolic pathways), and *de novo* assembly of uncultured organism genomes as Metagenome Assembled Genomes (MAGs) [4].

Due to huge inter-patients variability and new insights into microbiome's plasticity [5], clinical studies need to include many patients [6], and have a longitudinal approach when possible, to extract reliable information. Despite the continuous drop in sequencing costs, metabarcoding is thus often preferred to shotgun sequencing to carry out clinical studies, providing limited information and hindering our comprehension of microbiota.

Shallow shotgun metagenomics has been recently suggested as an alternative [7], cost-competitive to metabarcoding (allowing analysis of large cohorts) and providing nearly the same information as *deep* shotgun sequencing. It consists in shotgun sequencing at much lower sequencing depth : 20M reads/sample were typically used to characterize a human gut microbiota samples with shotgun sequencing, while so called *shallow* shotgun metagenomics typically deals with fewer than 1M

reads/sample, drastically reducing sequencing costs. This is made possible by assembly-free processing of reads, thus requires an exhaustive genome reference catalog for the studied environment to process mapping [8] [9].

In the present study, we aim to provide new insights towards usage of shallow shotgun metagenomics for taxonomic profiling of human gut microbiota, assessing the reliability of information that can be recovered from shallow shotgun sequencing in comparison with deep sequencing. We first used simulations to design and calibrate filters that efficiently identify organisms genuinely present in the mapping data providing reliable taxonomic profiles at each sequencing depth. We then applied this approach to real data sets, and assessed information recovery at a sample level and a study level. Our results show that some information is lost if we want to obtain reliable profiles (rare taxa are filtered out to avoid having massive identification of spurious taxa), but that structures like differences between groups of patients and diagnosis-like classification are very well conserved using shallow shotgun metagenomics.

2 Material and Methods

2.1 Data

Simulated data sets. We retrieved taxonomic profiles of 19 human gut microbiomes from Qin 2014 [10] through *curatedMetagenomicData* [11], with a complexity of 98 $\pm$ 15 species per sample, and species' relative abundance ranging from 5.10^{-1} down to 10^{-6} (average 10^{-3}). These profiles were given using the NCBI's taxonomy and were translated into UHGG's taxonomy [12] by choosing the UHGG species with the closest taxonomic assignation to the NCBI species (if several species tied, one was chosen randomly), resulting in profiles with the exact same complexity, approximately the same phylogenetic composition and some uncultured organisms (MAGs). UHGG genomes are clustered into "species clusters" (thereafter referred as species), that share at least 95% of identity on 30% of the genomes, and one representative genome is chosen in each cluster for inclusion in the mapping catalog. We generated profiles using either the species representative genome or using a randomly selected genome belonging to the same species. The second scenario introduces noise in the mapping data because of the intra-species diversity and corresponds to the more realistic case where the genome is not necessarily in the database used for mapping. It is the one shown in the results. For each sample, we used *Grinder* [13] to simulate 10M paired end reads (length of 2*125bp, insert size normally distributed with an average of 500bp and standard deviation of 50 bp without sequencing errors) and subsampled at 5 M, 1 M, 500 K, 100 K, 50 K and 10 K reads/sample.

Real data set. We used data from 3 clinical studies, for a total of N=439 samples covering patients from several continents and clinical conditions (healthy patients, hepatic diseases at different stages, cancer patients). Loomba *et al.* (2017) [14] compares the gut microbiota of patients suffering from hepatic diseases at different stages (fibrosis vs NAFLD). Matson *et al.* (2018) [15] compares, among patients having metastatic melanoma, those who responded to anti-PD-1 immunotherapy and those who didn't. Qin *et al.* (2014) [10] compares patients having liver cirrhosis and a group of healthy controls, with a discovery and a validation cohort for both groups. We analyzed these data sets at full depth and subsampled them to mimic shallow sequencing in the remainder.

2.2 Bioinformatics pipeline

Reads were pre-processed using *trimmomatic* [16], removing low quality reads and reads shorter than 80 nucleotides. For real data sets, reads were also mapped to human genome to filter out host contamination. Remaining reads were then mapped to UHGG catalog [12] using *bwa mem* [17] local aligner. We also used *bwa aln*, and *bowtie2* [18] in its *end-to-end* and local settings for comparison in the simulated data, but we only present results for *bwa mem* as it resulted in a better trade-off between overall mapping rate, false positive and multi-mapping rate.

Multi-mapping (*ie* reads that map to several genomes) occurs frequently when mapping shotgun metagenomics reads to a catalog of reference genomes (26% and 42% of the mapped reads in simulated and real data sets respectively), due to highly conserved genes and mobile elements notably. Thus,

we split mapped reads into unambiguous reads that mapped to one genome only, and other reads. For each genome identified, we retrieved the reads count (RC) and the fraction of the genome covered (FC) by at least one read, using either all reads or unambiguous reads only (uRC and uFC), as well as a specificity ratio (SR) defined by the number of unambiguous reads divided by the total number of reads mapped to this genome.

In order to estimate species' relative abundances, we first compute the representative genomes' average coverage $C_s = \frac{1}{\ell_s} \sum_i r_{i,s}$, with ℓ_s being the length of the representative genome of species s and $r_{i,s}$ the length of read i that is unambiguously mapped to s, and then we obtain the relative abundance by normalizing across species to sum to 1 : $A_s = \frac{C_s}{\sum_j C_j}$. We refine this estimation by reallocating the ambiguous reads by randomly assigning them to one of their hits, with a probability proportional to the previously computed relative abundances.

2.3 Simulations analysis

Direct mapping of short reads on reference genomes produces false positives (genomes covered by reads but not present) that need to be filtered out. We used simulated profiles, with known composition, to determine the most efficient way to classify the genomes into true positives (TP) and false positives (FP). In order to assess methods and compare them to each other, we computed the area under the receiver operating characteristic (ROC) curve (AUC) for this classification task, using *evabic* R package. We also implemented an automated threshold search, that allows for a false discovery rate ($FDR = \frac{FP}{TP+FP}$) of at most 10%, and compared false negative (FN) rates at this threshold across methods and sequencing depths.

We first evaluated how genomes features (RC, uRC, FC, uFC and SR) can be used independently to classify the genomes, and then combined them to train classifiers. We used logistic regression, linear discriminant analysis (LDA) and random forests (RF), to perform classification, with uRC, uFC, SR and total sequencing depth as input features. Finally, we used a 4-fold cross validation process to evaluate the performance of these methods and determine suitable thresholds for each method and sequencing depths.

2.4 Real data sets analysis

We analyzed real data set using (1) RF-based filters fitted on the simulations data and thresholds that control FDR at each sequencing depth, and (2) a basic filtering that discards all species with a relative abundance beyond 10^{-4}, FC beyond 10^{-2} or uFC beyond 10^{-4}. This filtering is inspired by what was done in [8] and corresponds to currently used methods with a quite permissive threshold due to low sequencing depths on which it will be applied.

We evaluated α-diversity using species richness and Shannon diversity, and β-diversity using Jaccard distance and Bray-Curtis dissimilarity index using *phyloseq* [19]. In order to assess the impact of sequencing depth on taxonomic profiles, we evaluated the correlation between subsampled and deep α-diversity measures using Spearman and Pearson correlation as well as the correlation between species relative abundance at full depth and shallower depths. We also measured the distance between low depth samples and their full depth counterpart. Finally, for each data set, we evaluated the differences between groups of interest, according to the sequencing depth, at different levels:

— differences in α-diversity between groups, through a Wilcoxon test using different metrics described earlier,
— structure in the β-diversity matrix, through a PERMANOVA analysis using different metrics described earlier,
— biomarker discovery using a Wilcoxon test with Benjamini-Hochberg correction that shows differentially abundant species,
— patients' classification in their group of interest, using random forests trained on taxonomic profiles, with a feature selection step as performed in [14].

In order to perform unbiased comparison of p-values and AUCs across sequencing depths, we used only samples having at least 10M high quality reads per sample for Loomba-2017 ($N = 77$), Matson-

2018 ($N = 39$) and 5M reads for Qin-2014 ($N = 235$, we reduced maximum sequencing depth to include more patients).

3 Results

3.1 Filters design and performance on simulated data

Our raw mapping data from simulations showed that a small number of reads (8% of unambiguously mapped reads) are mapped to an unexpected genome, resulting in a great number of false positives genomes (FDR = 91% prior to any filter). The basic (threshold-based) filtering technique yielded in overall FNR = 0.39 and FDR = 0.45. We sought optimal thresholds on read counts (RC), but it appeared to be sub-optimal (AUC = 0.75). Using genomes' fraction covered (FC) to determine such a threshold was better (AUC = 0.85), and retrieving only unambiguous reads to compute this statistics enhanced the classification (AUC = 0.856 using uRC, and AUC = 0.896 for uFC, see table 1) thus was used for the following. As seen on figure 1A, a threshold based on uFC and/or uRC values, which would correspond to horizontal and/or vertical line to discriminate TPs and FPs, is suboptimal as it would miss the long tail of genomes with low uRC but comparatively high uFC values. These results motivated our attempt to train classifiers able to take benefit from this pattern. To train such classifiers, we used uRC, uFC, SR, as well as sequencing depth to predict genomes' status (present or absent).

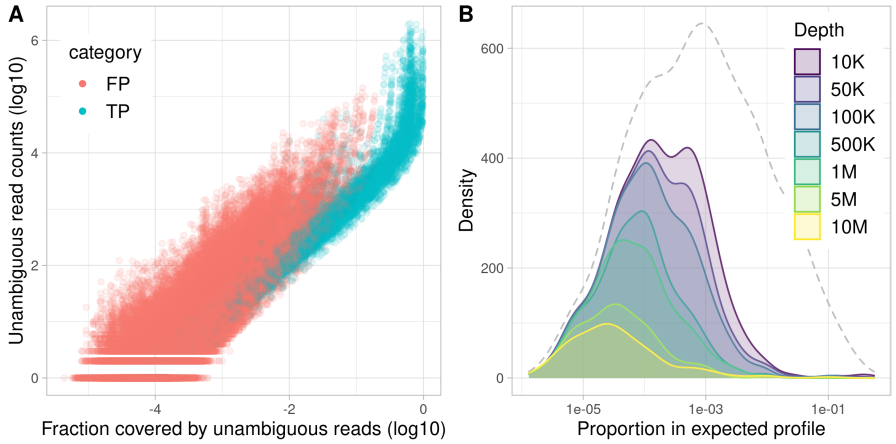


Fig. 1. Simulations results: (A) Unambigous fraction covered (uFC) and unambiguous read counts (uRC) of genomes present in the expected profiles (TPs, blue points) or absent (FPs, red points). (B) Distribution of FN species according to their relative abundances in the expected profiles, using RF-based filters with a 10% FDR on the testing set of cross validation. The dot line represents the distribution of all expected species.

method	AUC		FN rate at threshold	
	training	testing	training	testing
uRC	0.856		0.883	
uFC	0.896		0.580	
LDA	0.944 ± 0.002	0.944 ± 0.006	0.391 ± 0.010	0.391 ± 0.025
Logistic regression	0.955 ± 0.002	0.955 ± 0.006	0.386 ± 0.012	0.387 ± 0.032
Random forest	0.999 ± 0.0001	0.969 ± 0.007	0.017 ± 0.002	0.291 ± 0.040

Tab. 1. Classification of mapping hits in present and spuriously identified species : area under ROC curves and false negative rates when threshold is set to tolerate 10% FDR. For machine learning based methods, these measures are split into training and testing sets, using a 4-fold cross validation.

We can see on table 1 that sophisticated classifiers largely outperform basic filtering. LDA and logistic regression perform similarly, and yield nearly identical results in training and testing in the cross validation process, highlighting very good generalization capabilities. RF appeared to be the best method, yielding in a nearly perfect classification in training set, and still better than others in

the testing sets; RF will thus be used in the remainder. When setting a threshold that control the FDR at 10%, we can see the interest of refining this classification, to drastically lower the FN rate, although it remains quite high in the test samples.

We further characterized the information loss in the context of shallow sequencing metagenomics by plotting the distribution of expected relative abundances of species that were absent in profiles with respect to the sequencing depth, as seen on figure 1B, using RF-based filtering. We can see clearly the inflation of FN while lowering sequencing depth, but we can also notice that, as expected, the populations that are lost are relatively rare. For instance, at 500K reads/sample, all populations with relative abundance greater than 10^{-2} were detected.

While focusing on TPs, we noticed that Pearson correlation between expected and estimated species relative abundances went up from $\rho = 0.54$ to $\rho = 0.60$ by reallocating ambiguous reads.

3.2 Performance on real data sets for taxonomic profiling

Applying the RF-based filters on real data sets resulted in high quality profiles, with an average diversity of 128 ± 66 species per sample at full depth, which gradually decreased with sequencing depth, down to 45 ± 21 at 500K reads/sample for example (fig 2A). In comparison, basic filtering were more permissive, producing profiles with increased diversity and more resilient towards reduced sequencing depths (fig 2D). The Shannon diversity index (fig 2B,E) was much less impacted by sequencing depth, indicating that the species lost at low sequencing depth were mostly rare ones. Down to 500K reads per sample, the correlation between full depth and subsampled Shannon indices was nearly perfect using basic filtering and remained very high with RF-based filters. Distances between subsamples and their reference, defined as the corresponding sample at full depth, gradually augmented when decreasing the sequencing depth, and these distances were much more important using RF-based filters than basic filters (fig 2C,F, both graphs share the same scale), and showed high replicability across data sets.

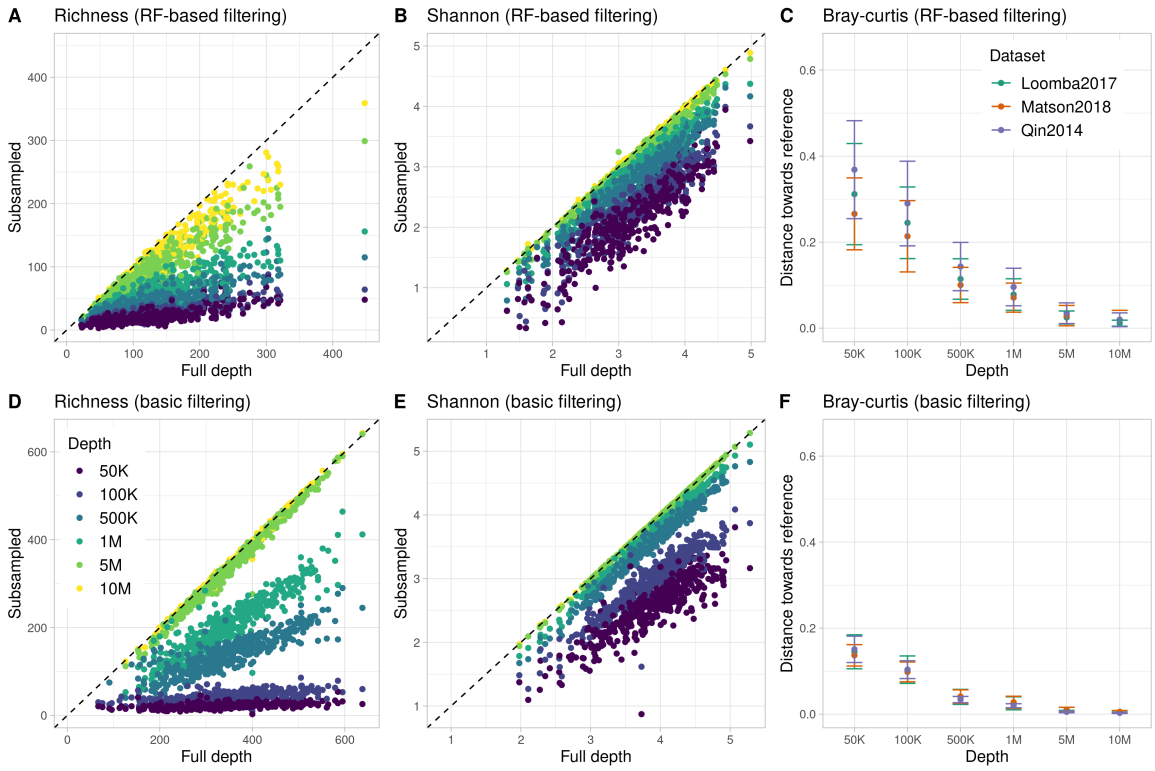


Fig. 2. Comparison between full depth and subsampled profiles for samples from the 3 data sets considered : richness observed, Shannon diversity and Bray-Curtis distance between subsampled data and reference (full depth data) using RF-based filtering (A, B and C respectively) and basic filtering (D, E and F respectively)

Comparison between filtering strategies highlighted that filtering plays a key role while dealing with shallow metagenomics data. If taxonomic profiles were less impacted by sequencing depth using basic filters than RF-based filters, we know according to our simulations that lots of false positives are present in the profiles, which introduces an important noise and jeopardizes the biological interpretation of results.

3.3 Performance on real data sets for patients stratification

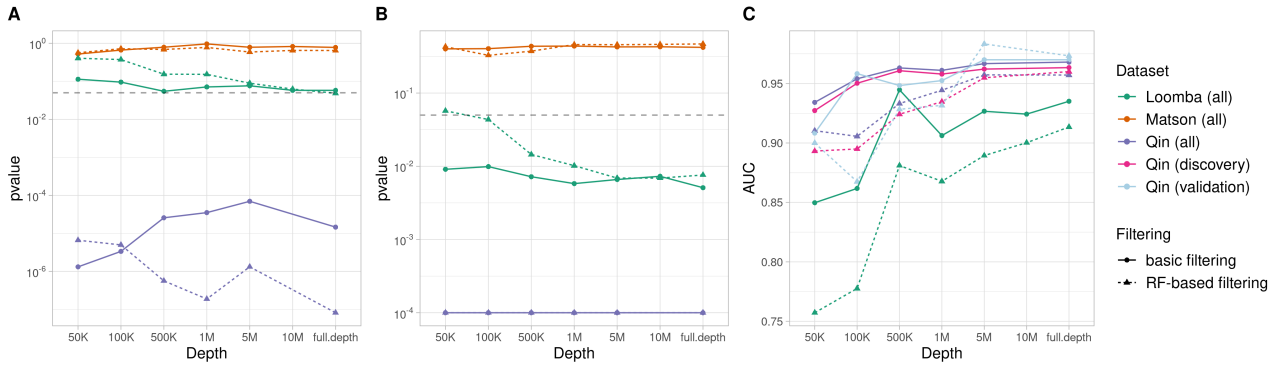


Fig. 3. Differences between patients groups in different studies : significance of inter-group difference regarding Shannon diversity index (A), PERMANOVA analysis (B). AUC corresponding to random forest classification (C) was performed in Loomba-2017 and Qin-2014, with a split between discovery and validation cohorts in Qin-2014 as performed on the original paper of this study.

Here, we assess the robustness of biological signal found in the different data sets towards sequencing depth. As expected according to previous results, differences in α-diversity between groups were maintained using shallow sequencing : p-values were concordant across sequencing depths (see Fig. 3A), with a strong difference in Qin-2014 data set, a slight difference between groups that is not significant in Loomba-2017 and no differences between groups in Matson-2018. PERMANOVA analysis led to similar results (Fig. 3B), showing that the structure of the matrix distances between samples is very marginally impacted by sequencing depths. The p-value regarding Loomba-2017 on RF-based filtered data increased, but stayed significant, even under 1M reads/sample, traducing the absence of key populations at such sequencing depths. Out of the 7 differentially abundant taxa in Loomba-2017 with RF-based filtering found at full depth ($FDR < 0.1$), 5 taxa were still identified at 1M reads/sample and 4 at 500K reads/sample. Basic filtering, allowing more taxa in the profiles, identified more differentially abundant taxa (19 at full depth, 12 at 500K reads/sample) but the reliability of these taxa is questionable. As previously discussed, signal was much more important in Qin-2014 data set: 25 differentially abundant taxa were identified at full depth with RF-based filtering ($FDR < 0.05$ in both discovery and validation cohorts), 13 taxa at 1M reads/sample and 9 at 500K reads/sample. Again, basic filtering allowed to identify more taxa (124 at full depth, 72 at 500K reads/sample). Finally, classification of patients using RF was performed in Loomba-2017 and Qin-2014 (see Fig. 3C). In Loomba-2017, we could perform a better classification using basic filtering, with an AUC similar to full depth AUC down to 500K reads/sample, while it gradually decreased as sequencing depth decrease using RF-based filters, due to some key taxa for the classification being lost. On Qin-2014 data set, we could perform a very good classification on both discovery and validation cohorts even at low sequencing depth, with performance very stable using basic filtering down to 100K reads/sample using basic filtering and that gradually decreased with RF-based filters under 5M reads/sample.

4 Discussion

Direct mapping of reads on catalogs of reference genomes was previously suggested as the most suitable way to build taxonomic profiles from shallow sequencing metagenomics data [7] [9], as it produced nearly identical taxonomic profiles across sequencing depths. Our simulations highlighted the need to refine filters on genomes identified by such mapping, and to perform depth dependent thresholds to obtain reliable profiles at each sequencing depth. This step is crucial to prevent mislead-

ing interpretations and to provide trustful biological knowledge. Controlling the false discovery rate (FDR) in the taxonomic profiles had the direct consequence of decreasing the number of identified species, especially at low sequencing depth. The benefit of random forest-based filters, and to a lesser extent other machine learning based models tested, over simple filtering based on species features independently (read counts and fraction covered, considering all reads and unambiguous reads only) was remarkable, allowing to identify more species and rarer ones for equivalent FDR. The application of such techniques, as they rely on a learning step, is by definition limited to the training conditions. In our case, usage of our random forest-based model to filter genomes should be limited to ecosystems with similar complexity, sequenced with short reads at depth included in the range used for the training and mapped to a catalog similar to representative genomes of UHGG in terms of completeness, intra and inter species diversity.

On the three real data sets considered, our analysis showed that differences between groups of patients observed at full depth where still recovered at low sequencing depth. Permissive and depth-independent filtering, as performed in previously published papers on shallow shotgun metagenomics, allowed a little improvement in structure recovery than our stringent random forest-based filters: these structures were less sensible to the noise introduced by spuriously identified species in the profiles using basic filtering, than to the removal of key species induced by our stringent random forest-based filter.

Overall, our results show that (1) one needs to perform stringent and depth-dependent filters to obtain reliable profiles in shallow sequencing data, (2) resulting taxonomic profiles are limited to most abundant taxa in shallow sequencing context, and (3) shallow shotgun metagenomics can be a suitable approach to perform diagnosis-like classification of patients even if further investigations should be led to assess generalization capability and interpretability of signatures obtained with shallow sequencing.

Shallow shotgun metagenomics requires an exhaustive reference database regarding the studied ecosystem to build taxonomic profiles. Although it produced reduced complexity profiles if we want to ensure reliability of results, it appeared to be a very good alternative for clinical studies, and sufficient to classify patients, when discrimination between groups is expected to be important and to rely on relatively dominant taxa. Therefore, it can be profitable in such cases to favour the number of patients included or to introduce a longitudinal aspect, rather than per sample sequencing depth. For other body sites (vaginal, oral or skin microbiota) host contamination should be taken into account when determining sequencing depth, as host reads will be discarded. Shallow shotgun metagenomics could also be used to perform functional analysis, for example for coarse grain identification of family of genes (like KOs), as the sequencing depth could not allow the identification of very specific genes and SNPs that require assembly, such as antibiotic resistance.

Acknowledgements

We are grateful to the INRAE MIGALE bioinformatics facility (MIGALE, INRAE, 2020. Migale bioinformatics Facility, doi: 10.15454/1.5572390655343293E12) for providing help, computing and storage resources, as well as CRCM's DISC platform. This work was financially supported by ANRT and Laboratoire Alphabio thanks to a CIFRE scholarship.

References

[1] Yong Fan and Oluf Pedersen. Gut microbiota in human metabolic health and disease. *Nature Reviews Microbiology*, 19(1):55–71, January 2021.

[2] Sunny H. Wong and Jun Yu. Gut microbiota in colorectal cancer: mechanisms of action and clinical applications. *Nature Reviews Gastroenterology & Hepatology*, 16(11):690–704, November 2019.

[3] Jose C Clemente, Julia Manasson, and Jose U Scher. The role of the gut microbiome in systemic inflammatory disease. *BMJ*, page j5145, January 2018.

[4] Donovan H. Parks, Christian Rinke, Maria Chuvochina, Pierre-Alain Chaumeil, Ben J. Woodcroft, Paul N. Evans, Philip Hugenholtz, and Gene W. Tyson. Recovery of nearly 8,000 metagenome-assembled genomes substantially expands the tree of life. *Nature Microbiology*, 2(11):1533–1542, November 2017.

[5] Jessica A. Grembi, Lan H. Nguyen, Thomas D. Haggerty, Christopher D. Gardner, Susan P. Holmes, and Julie Parsonnet. Gut microbiota plasticity is correlated with sustained weight loss on a low-carb or low-fat dietary intervention. *Scientific Reports*, 10(1):1405, December 2020.

[6] Christopher Quince, Alan W Walker, Jared T Simpson, Nicholas J Loman, and Nicola Segata. Shotgun metagenomics, from sampling to analysis. *Nature Biotechnology*, 35(9):833–844, September 2017.

[7] Benjamin Hillmann, Gabriel A Al-Ghalith, Robin R Shields-Cutler, Qiyun Zhu, Daryl M Gohl, Kenneth B Beckman, Rob Knight, and Dan Knights. Evaluating the Information Content of Shallow Shotgun Metagenomics. *mSystems*, 3(6):12, 2018.

[8] Tasha M Santiago-Rodriguz, Aaron Garoutte, Emmase Adams, Waleed Nasser, Matthew C Ross, Alex La Reau, Zachariah Henseler, Tonya Ward, Dan Knights, Joseph F Petrosino, and Emily B Hollister. Metagenomic Information Recovery from Human Stool Samples Is Influenced by Sequencing Depth and Profiling Method. *Genes*, page 17, 2020.

[9] Federica Cattonaro, Alessandro Spadotto, Slobodanka Radovic, and Fabio Marroni. Do you cov me? Effect of coverage reduction on metagenome shotgun sequencing studies. *F1000 Research*, 7:1767, 2020.

[10] Nan Qin, Fengling Yang, Ang Li, Edi Prifti, Yanfei Chen, Li Shao, Jing Guo, Emmanuelle Le Chatelier, Jian Yao, Lingjiao Wu, Jiawei Zhou, Shujun Ni, Lin Liu, Nicolas Pons, Jean Michel Batto, Sean P. Kennedy, Pierre Leonard, Chunhui Yuan, Wenchao Ding, Yuanting Chen, Xinjun Hu, Beiwen Zheng, Guirong Qian, Wei Xu, S. Dusko Ehrlich, Shusen Zheng, and Lanjuan Li. Alterations of the human gut microbiome in liver cirrhosis. *Nature*, 513(7516):59–64, September 2014.

[11] Edoardo Pasolli, Lucas Schiffer, Paolo Manghi, Audrey Renson, Valerie Obenchain, Duy Tin Truong, Francesco Beghini, Faizan Malik, Marcel Ramos, Jennifer B Dowd, Curtis Huttenhower, Martin Morgan, Nicola Segata, and Levi Waldron. Accessible, curated metagenomic data through ExperimentHub. *Nature methods*, 14(11):1023–1024, October 2017.

[12] Alexandre Almeida, Stephen Nayfach, Miguel Boland, Francesco Strozzi, Martin Beracochea, Zhou Jason Shi, Katherine S. Pollard, Ekaterina Sakharova, Donovan H. Parks, Philip Hugenholtz, Nicola Segata, Nikos C. Kyrpides, and Robert D. Finn. A unified catalog of 204,938 reference genomes from the human gut microbiome. *Nature Biotechnology*, 39(1):105–114, January 2021.

[13] Florent E. Angly, Dana Willner, Forest Rohwer, Philip Hugenholtz, and Gene W. Tyson. Grinder: a versatile amplicon and shotgun sequence simulator. *Nucleic Acids Research*, 40(12):e94–e94, July 2012.

[14] Rohit Loomba, Victor Seguritan, Weizhong Li, Tao Long, Niels Klitgord, Archana Bhatt, Parambir Singh Dulai, Cyrielle Caussy, Richele Bettencourt, Sarah K. Highlander, Marcus B. Jones, Claude B. Sirlin, Bernd Schnabl, Lauren Brinkac, Nicholas Schork, Chi-Hua Chen, David A. Brenner, William Biggs, Shibu Yooseph, J. Craig Venter, and Karen E. Nelson. Gut Microbiome-Based Metagenomic Signature for Non-invasive Detection of Advanced Fibrosis in Human Nonalcoholic Fatty Liver Disease. *Cell Metabolism*, 25(5):1054–1062.e5, May 2017.

[15] Vyara Matson, Jessica Fessler, Riyue Bao, Tara Chongsuwat, Yuanyuan Zha, Maria-Luisa Alegre, Jason J. Luke, and Thomas F. Gajewski. The commensal microbiome is associated with anti–PD-1 efficacy in metastatic melanoma patients. *Science*, 359(6371):104–108, January 2018.

[16] Anthony M. Bolger, Marc Lohse, and Bjoern Usadel. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics*, 30(15):2114–2120, August 2014.

[17] H. Li. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *arXiv:1303.3997 [q-bio]*, May 2013. arXiv: 1303.3997.

[18] Ben Langmead and Steven L. Salzberg. Fast gapped-read alignment with Bowtie 2. *Nature Methods*, 9(4):357–359, April 2012.

[19] Paul J. McMurdie and Susan Holmes. phyloseq: An R Package for Reproducible Interactive Analysis and Graphics of Microbiome Census Data. *PLoS ONE*, 8(4):e61217, April 2013.

panRGP: a pangenome-based method to predict genomic islands and explore their diversity

Adelme Bazin[1], Guillaume Gautreau[1] , Claudine Medigue[1], David Vallenet[1] and Alexandra Calteau[1]

[1]LABGeM, Génomique Métabolique, CEA, Genoscope, Institut François Jacob, CNRS, Université d'Évry, Université Paris-Saclay, Evry, France.

Corresponding Author: abazin@genoscope.cns.fr

Paper Reference: **Bazin *et al.*, panRGP: a pangenome-based method to predict genomic islands and explore their diversity, Bioinformatics, Volume 36, Issue Supplement_2, December 2020, Pages i651–i658, https://doi.org/10.1093/bioinformatics/btaa792**

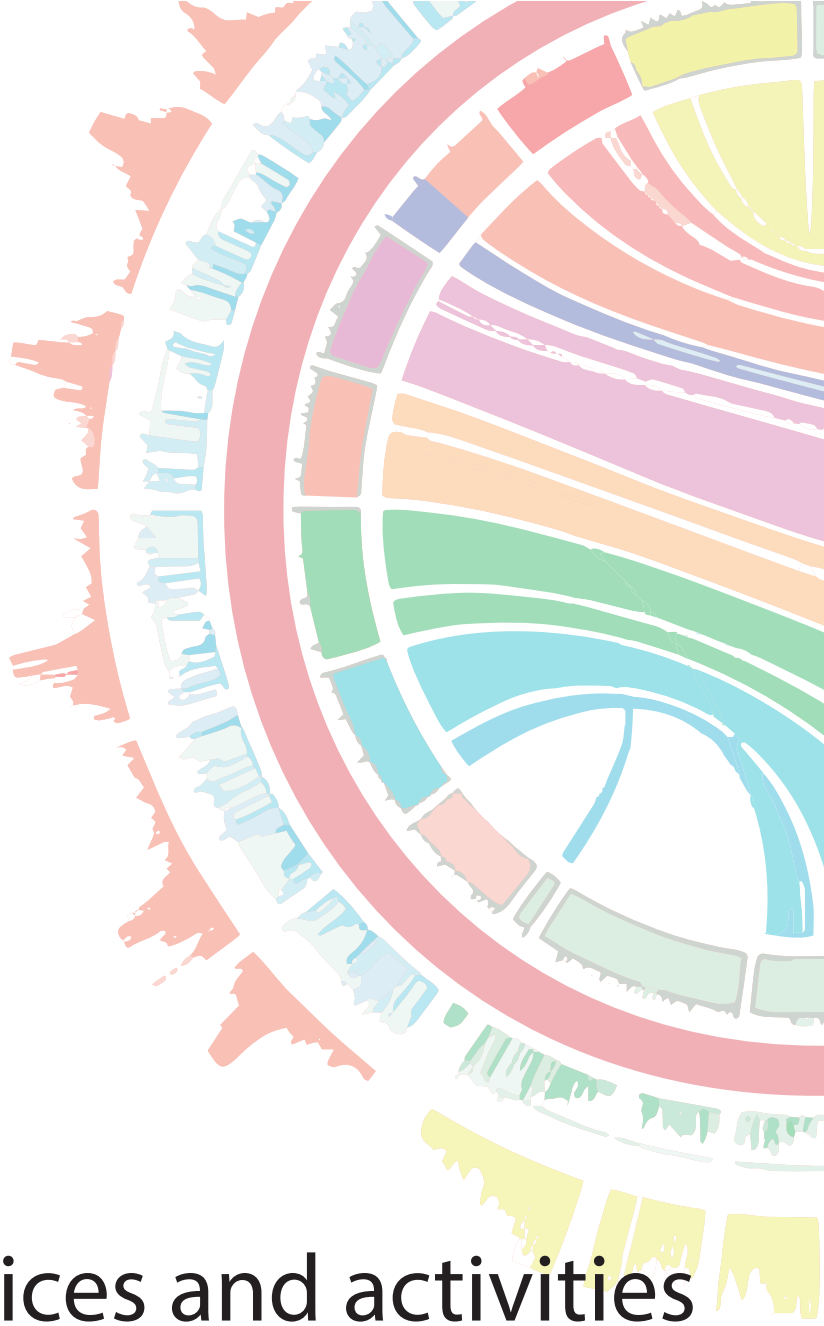
Horizontal gene transfer (HGT) is a major source of variability in prokaryotic genomes. This evolutionary process is a significant source of gene novelty [1] and allows microbes to adapt to new environments or to obtain new pathogenic capabilities [2].

Regions of genome plasticity (RGPs) are clusters of genes located in highly variable genomic regions. Most of them arise from HGT and correspond to genomic islands (GIs). As GIs carry so many genes of interest, they have been heavily studied and countless methods have been designed to detect and analyse those particular regions of microbial genomes [3]. The study of those regions at the species level has become increasingly difficult with the deluge of genomic data. To date, no methods are available to identify GIs using information from hundreds of genomes to explore their diversity and identify those that share the same genomic context.

We present here the panRGP method that predicts RGPs using pangenome graphs made of all available genomes for a given species [4]. It allows the study of thousands of genomes in order to access the diversity of RGPs and to predict spots of insertion. It gave the best predictions when benchmarked along other GI detection tools against a reference dataset. In addition, we illustrated its use on metagenome assembled genomes by redefining the borders of the leuX tRNA hotspot, a well-studied spot of insertion in *Escherichia coli* [5]. panRPG is a scalable and reliable tool to predict GIs and spots making it an ideal approach for large comparative studies. The panRGP method has been implemented in the PPanGGOLiN pangenomic software suite (https://github.com/labgem/PPanGGOLiN).

References

1. Treangen TJ, Rocha EPC (2011) Horizontal Transfer, Not Duplication, Drives the Expansion of Protein Families in Prokaryotes. PLoS Genet 7(1): e1001284.
2. Hacker J. , Carniel E. (2001) Ecological fitness, genomic islands and bacterial pathogenicity. EMBO Rep., 2, 376–381
3. Claire Bertelli, Keith E Tilley, Fiona S L Brinkman, Microbial genomic island discovery, visualization and analysis, Briefings in Bioinformatics, Volume 20, Issue 5, September 2019, Pages 1685–1698
4. Gautreau G, et al. (2020) PPanGGOLiN: Depicting microbial diversity via a partitioned pangenome graph. PLoS Comput Biol 16(3): e1007732.
5. Lescat M. et al. (2009) A module located at a chromosomal integration hot spot is responsible for the multidrug resistance of a reference strain from Escherichia coli clonal group A. Antimicrob. Agents Chemother., 53, 2283–2288.



Session 9
Platfom services and activities

From biogitflow to geniac: harmonisation of software development practices with Nextflow to support daily production in oncology

Choumouss Kamoun[1-4], Julien Roméjon[1-4], Henri de Soyres[1-4], Apolline Gallois[1-4], Elodie Girard[1-4], Fabrice Allain[1-4], Philippe Hupé[1-5]

[1] Institut Curie, Paris, F-75005, France
[2] U900, Inserm, Paris, F-75005, France
[3] PSL Research University, Paris, France
[4] Mines Paris Tech, Fontainebleau, F-77305, France
[5] UMR144, CNRS, Paris, F-75005, France

Corresponding Author: philippe.hupe@curie.fr

Keywords development workflow, bioinformatics pipeline, quality management, healthcare, deployment

The use of a bioinformatics pipeline as a tool to support diagnostic and theranostic decisions in the healthcare process requires the definition of detailed development workflow dedicated to daily production. Therefore, we present biogitflow as a protocol that describe step-by-step all the command lines and actions that the developers have to follow. Our protocols capitalized on the two powerful and widely used tools git and GitLab, and are based on gitflow, a well-established workflow in the software engineering community. They address two use cases: a nominal mode to develop a new feature in the bioinformatics pipeline and a hotfix mode to correct a bug that occurred in the production environment. The protocols are available as a comprehensive documentation at https://biogitflow.readthedocs.io.

In addition to the common development workflow, we propose geniac as a more specific set of coding conventions and tools from the prototyping step to production operations of bioinformatic pipelines using the workflow manager Nextflow. With the idea of being as less as invasive for the different expert communities (bioinformaticians, statisticians, software engineers, data managers, core facility engineers), those guidelines and utilities aims to reduce the overall development cycle, provide portable pipelines with containers (docker, singularity, …) and automatize whenever possible the building of containers. One of the main concepts of this approach is the *one container per tool* strategy which have several advantages compared to the usage of a single development or production environment for a bioinformatic pipeline. A detailed documentation along with support tools for development and deployment are respectively accessible at https://geniac.readthedocs.io/en/latest/intro.html and https://github.com/bioinfo-pf-curie/geniac.

Acknowledgements

This work was supported by the European Union's Horizon 2020 research and innovation programme and the Canadian Institutes of Health Research under the grant agreement No 825835 in the framework on the European-Canadian Cancer Network and the project French Bioinformatic Network for NGS Cancer Diagnosis, funded by the Institut National du Cancer (INCA).

References

1. Kamoun C, Roméjon J, de Soyres H et al. biogitflow: development workflow protocols for bioinformatics pipelines with git and GitLab. F1000Research 2021, 9:632 (https://doi.org/10.12688/f1000research.24714.3)

ToulligQC 2: fast and comprehensive quality control for Oxford Nanopore sequencing data

Karine DIAS[1], Corinne BLUGEON[1], Sophie LEMOINE [1], Morgane THOMAS-CHOLLIER[1], Stéphane LE CROM[2,1], Médine BENCHOUAIA[1], and Laurent JOURDREN[1]

[1]Genomics core facility, Institut de Biologie de l'ENS (IBENS), Département de biologie, École normale supérieure, CNRS, INSERM, Université PSL, 75005 Paris, France

[2]Sorbonne Université, CNRS, Institut de Biologie Paris-Seine (IBPS), Laboratory of Computational and Quantitative Biology (LCQB), F-75005, Paris

Corresponding author: jourdren@bio.ens.psl.eu

The sequencing devices developed by Oxford Nanopore Technologies (ONT) produce long DNA sequence (> 200 kb) and full-length RNA. Sequencing and primary data acquisition are driven by the MinKNOW software, developed by ONT. MinKNOW stores the raw signal data as Fast5 files. Basecalling is then performed either during or after the acquisition step. Basecalling is usually achieved by the program Guppy, the official ONT basecaller. The output sequence reads are stored in FASTQ or Fast5 format. MinKNOW produces a Quality Control (QC) report as a PDF file at the end of the run. However this report only provides estimated information as it is based on non-basecalled and non-demultiplexed data. In addition, the metrics and scales that were provided by MinKNOW when we started RNA-Seq applications in 2016 were not appropriate (unsuitable scales for RNA - which has been fixed since - and no barcode handling). It was thus necessary to develop a dedicated QC tool, flexible enough to handle both RNA and DNA sequencing.

The first version of ToulligQC is freely available since 2017, and used in production in our Genomics core Facility. It allows users to quickly estimate the quality and homogeneity of their samples before running further analyses. Easy to use, this tool provides a detailed graphical output about the quality of Nanopore runs and exploratory data analysis, in the same spirit as the well-known FastQC program for short reads [1].

We introduce ToulligQC 2, a new major version of our QC software. ToulligQC 2 produces an improved HTML report with stylish and interactive plots obtained with the Plotly [2] library. The report contains exhaustive information about the sequencing run, basecalling and demultiplexing steps, such as: read count and length distributions, homogeneity of the run, location of potential flow cell spatial biases, statistics about pass and fail reads, PHRED score distribution and density distribution across read types, length/speed/quality and number of sequences over sequencing time, length/quality and read counts for each barcode. In addition to new graph types, all plots were qualitatively improved, and some of them provide alternative visualisation mode (e.g., boxplot and violin plot).

ToulligQC 2 has a reduced memory footprint and is faster (few minutes on a laptop) than the previous version. To facilitate interpretation of the graphs, each plot displays an “info” icon directly linking to the online help page on *GitHub* [3].

Because ONT protocols and bioinformatics tools are constantly evolving, ToulligQC 2 supports all versions of Guppy and the latest sequencing protocols. It can be used with all the Oxford Nanopore sequencing devices (MinION, GridION, PromethION), and remains compatible with both 1D and $1D^2$ chemistries. It takes as input the sequencing summary file generated by the Guppy basecaller and the sequencing telemetry file, if available.

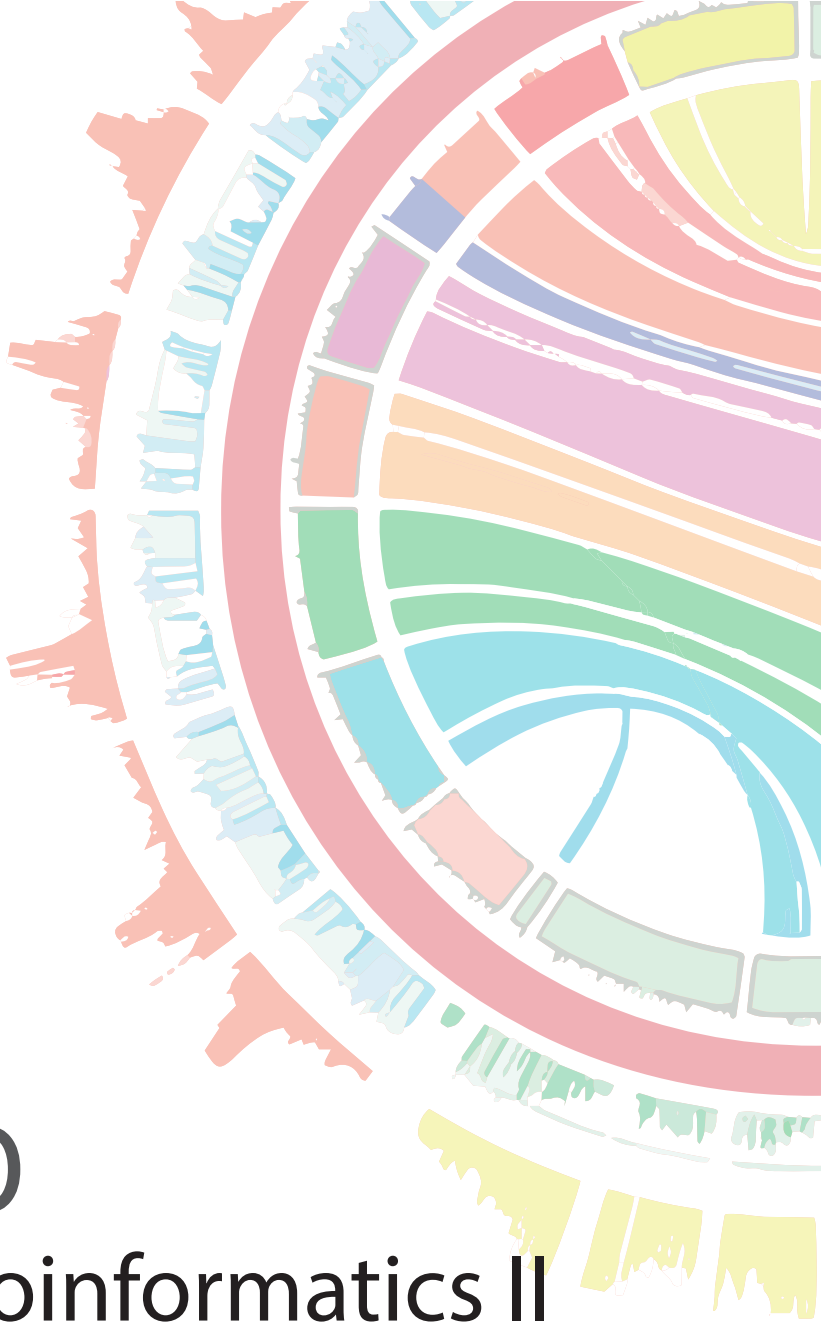
ToulligQC 2 is an *open source* software published under GPL3 and CeCILL licences. It can be freely downloaded on *Github* [3], as a *Docker image* (genomiquepariscentre/toulligqc), and as a *PyPy package* [4].

Acknowledgements

The IBENS genomics core facility was supported by the France Génomique national infrastructure, funded as part of the “Investissements d'Avenir” program managed by the Agence Nationale de la Recherche (contract ANR-10-INBS-09).

References

[1] https://www.bioinformatics.babraham.ac.uk/projects/fastqc/
[2] https://plot.ly
[3] https://github.com/GenomicParisCentre/toulligQC
[4] https://pypi.org/project/toulligqc/



Session 10
Structural Bioinformatics II

SLiMAN - un serveur pour l'analyse des interactions protéine-protéine via des motifs linéaires

SLiMAN - a webserver to analyze protein-protein interaction mediated by short linear motifs

Victor REYS [12], Gilles LABESSE [1]

[1] Centre de Biologie Structurale, 29 Rue Navacelles, 34090, Montpellier, France
[2] Université de Montpellier, 163 Rue Auguste Broussonet, 34090, Montpellier, France

Corresponding Author: gilles.labesse@cbs.cnrs.fr

Abstract *Protein-protein interactions play an important role in cell biology. Proteomics and related techniques (phosphoproteomics, transcriptomics) are providing experimental data at large scale and increasing pace. However, this mainly results in list of proteins for which direct interactions is not known. Structural biology can help in decreasing the gap between large scale studies and precise annotation of the possible physical interactions. Here we describe SLiMAN, a dynamic web software devoted to the analysis of protein-protein interactions (PPIs) involving binding of Short Linear Motifs (SLiMs) onto structured protein domains. This open the road for deeper understanding of the protein networks at play in various biological issues such as cancers.*

Keywords Peptide-domain interaction, SLiM, comparative modeling, exploration

1. Introduction

Multiple protein-protein interactions (PPI) networks play key cellular roles, regulating protein expression, cell signaling and cell behavior [1,2]. Experimental interactomic studies are now unraveling potential partners at high pace and sub-proteomic level. Interactomic analysis are usually performed using a flagged protein serving as a bait to capture its interacting partners, followed by an identification process (mass spectrometry, western-blot,…). The resulting interactants are mostly displayed as a list of identified proteins, annotated according to their biological function but this rarely lead to build the underlying PPI network. The main drawbacks of most identification processes, is the absence of knowledge of the true physical interactions.

It is known that many of the biological processes are performed by interactions through short linear motifs (SLiMs) with structured protein domains[3]. The Eukaryotic Linear Motif resource (ELM [4]) is a reference database for SLiM annotation and prediction. From 3559 publications, dealing with up to 136 non-homogenously methods, this database gather 289 manually curated SLiM classes. Classes are defined by the signature – or patterns - of the binding motifs which are also associated with interacting and structured domain as annotated from the Protein Familly (Pfam) database [5].

The ELM database is now largely known and used by the community to predict SLiM mediated PPI for each protein. The main difficulties of such analysis, is the number of false positive matches obtained from the ELM regular expressions and also the huge number (often over 100) of predicted interaction motif for a given protein.

More frequently found in intrinsincly disordered regions of proteins or in external accessible loops, SLiMs can be considered to act as peptides, that bind onto a target structured domain [6]. For this reason, the current state of the art for SLiM predictions also includes additional features, such as prediction of the disorder state of the SLiM. The software IUpred2[7] is often used to predict the probability for a given amino-acid to be in a disordered part of a protein. Still, the analysis of a given datasets for an interactomic analysis is leading to a huge number of matches. Systematically excluding SLiMs with high probability or predicted as too ordered (IUpred < 0.5) may exclude frequent SLiMs (e.g.: LIG_WD40_WDR5_VDV_2) and others that are more

structured (e.g.: TRG_NES_CRM1_1). Currently, performed on one protein at a time, this constitute a huge burden when one is analyzing an interactomic output made of hundred(s) of proteins.

Analyzing interactomic results aim to rebuild the PPI network to enravel the mechanisms of biological pathways/processes, possibly at a molecular level. PPI network reconstruction is a challenge, and some databases are devoted to help such a task. The STRING [8] database, well appreciated by the community, combines genetic interactions, text mining, text association, predicted and experimental PPI databases, but it does not provide clues on the direct physical interactions involved in the corresponding (sub-)networks. Similarly PPI databases, such as the Biological General Repository for Interaction Datasets (BioGRID[9]) gather the results of multiple interactomics studies, but are also lacking structural information.

Interactions based on domain-domain interactions can be predicted using structural information extracted for the PDB and the resulting network can be displayed. This is now made available through the web server Interactome3D [10] or Proteo3Dnet [11]. The latter is devoted to the modeling of PPI from structural information, and also from some ELM information. However, motif probability is not taken into account and the IUpred2 exclusion parameter is set very high (>0.95), which dramatically reduce the number of predicted SLiM mediated interactions. In addition, generated models are made for domain-domain interactions, leaving the motif-domain interaction structure unknown. Rarely interactomics studies include thorough predictions of useful SLiMs for PPI analysis although high throughput predictions have been implemented[12]. Indeed, interactive analysis of potential motif-domain interactions is still lacking automation for systematic study.

Here, we describe SLiMAN (Short Linear Motif ANanlysis), a webserver devoted to analysis of interactomic results from SLiM interactions. This new tool integrates information from multiple databases related to related to sequence (UniprotKB[13], Pfam, ELM), post-translational modifications (PTM) from PhosphoSitePlus® [14], structural (PDB[15]) and experimental PPI (BioGRID) data as well as tools to compute disorder region probability (IUpred2), sequence alignment (MAFFT[16], BLAST[17]) and structural homology models (SCWRL3[18]). This dynamic webserver (*http://sliman.cbs.cnrs.fr*) allows one to analyze interactomic PPI mediated by SLiMs in an interactive manner. By playing with various parameters, users are able to dig into the predicted data and build structural models for potential peptide-domain interactions, using a large set (5064) of closely related structural templates extracted from the PDB. Hence, a more precise picture of the corresponding network of protein-portein interactions can be discovered. Such an annotation opens the road for further experimental validation using for example directed mutagenesis.

2. Material and Methods

2.1. Protein Annotations

UniprotKB is used to gather protein sequences restricted to “reviewed” entries, which corresponds to 564 277 entries (20 396 human), in the last update of UniprotKB (feb 2021). Additional domain annotation is directly extracted from Pfam (domain boundaries and descriptions) for the corresponding sequences.

From the ELM database, class names, interaction domain types (Pfam annotation), regular expression and E-value are extracted for SLiMs analysis. In the current release, a total of 289 classes are defined. ELM experimental instances, ones that are used for the class regular expression definition, are also included, allowing fast validation for known interactions.

The PhosphoSitePlus® (PSP) database for post-translational modifications (PTM), is integrated to pinpoint locations of acetylation, methylation, O-GalNAc, O-GlcNAc, phosphorylation, sumoylation and ubiquitination sites over the amino acid sequence, for 46 096 proteins (from which 18 021 are humans).

2.2. Disorder Predictions

IUpred2 is used to predict disorder along the amino-acid sequence of a protein. The disorder scores (DS) (ranging from 0 – most order - to 1 – most disordered) are predicted at the residue level and includes several predictors; local disorder (short and long window size), presence of structured domain (short and long windows) and ANCHOR2 (probability to be part of an interacting segment). The final scores attributed to the motif are obtained by averaging the scores over the residues constituting a given motif. An additional binary

value is computed (StrictDisorder). It is set to 1 if all residues from a motif have their short and long DS predictions above the 0.5 (else 0).

2.3. Protein-protein Interaction Data

For complementary source of information, SLiMAN intergrates the lastest release (current 4.3.195) of the BioGRID dataset, from which only physical interactions were retained, and splited into low and high throughput experiments. The mapping between Uniprot entry names and BioGIRD data is achieved using the Uniprot mapping API (*https://www.uniprot.org/uploadlists/*).

2.4. Structural Support Extraction

For each of the possible 291 ELM/PFam associations, PDB is parsed in search for structural information, using the *pdb_pfamA_reg* database to select the corresponding domain chains. For each referenced domain chains, other chains are converted to FASTA format, and associated ELM regular expressions are used to parse the sequences. During the FASTA conversion, modified residues (MLZ, MLY, M3L, ALY, SEP, TPR, MSE, MNN, DA2, SEC, TPO and PTR) found in the structure are converted to the one-letter code of the corresponding unmodified amino-acids (e.g.: SER for SEP). To check actual peptide-domain interaction, Cα from residues that matched the regular expression must be found under a 10 Å threshold from any atoms of the domain chain, and the chain containing the matched motif should be a peptide shorter than 35 residues. In this case, we avoid most crystallographic artefacts. Then, contact distances bewteen residues belonging to the peptide-domain interface are computed to split residues in 4 distance categories (> 7Å, < 7Å, < 5.5Å and < 4Å). Distance categories are then converted into sequences of contact-scores (respectively 0, 1 ,2 and 3), allowing a one-dimention encoding of the interface over the sequences of the domain and the peptide. After template extraction, domain amino-acid sequences are placed in a FASTA file and the BLAST® database generator (*makeblastdb*) is used to setup the corresponding blastable database of the domains, for future alignment queries.

2.5. Sequence Alignments

Alignments between the domain sequence and template structure sequence is performed by two software. MAFFT is used with the local-pair alignment option, limited to 1000 max-iterations (L-INS-i). BLAST (*blastp*) is used with the pre-computed blastable databases, previously described.

For peptide alignments, only MAFFT is used. To limit mis-alignment of parts of the peptides in the multi-alignment process, the gap opening and extention penalty options are increased to 10 and 0.2 respectively.

2.6. Alignment Metrics

Five different alignment metrics are computed to identify the suitable templates for comparative modeling.

Sequence Identity (%Ident), corresponding to the sum of identical aligned amino acids devided by the number of aligned amino-acids.

Query coverage (%QueryCoverage), corresponds to the sum of aligned amino acids from the query devided by the query length, and represent the percentage of the query amino-acids that will be modeled.

Template coverage (%TemplateCoverage), corresponds to the sum of aligned amino acids from the template devided by the template length, and represent the percentage of the template amino-acids that will be used for the modeling.

Contact conservation score (CCS), corresponds to the sum contact-scores of aligned amino acids from the template. This metric allows fast discrimination of alignments that will lead to peptide-domain interactions.

These metrics help faster discrimination of optimal sequence-structure alignments.

2.7. Comparative Modeling

Tri-dimension model of the motif-domain interaction, can be built from the sequence structure alignment computed by MAFFT and BLAST®. Only aligned amino-acids are used for the modeling by SCWRL 3.0, with amino-acid backbone atoms and side-chains of strictly conserved residues kept fixed. Modeling of the

selected complex is then preformed in three steps. On the first step, the queried domain is modeled, using the original extracted domain as template and the peptide as constraint to model domain substituted amino-acids side chains. Then, the peptide model is generated using the previously generated domain model as a constraint. The domain and the peptide models are then combined to build the final peptide-domain complexe.

2.8. SLiMAN Workflow

From a simple list of Uniprot accession numbers or entry names, SLiMAN will analyze the data with 3 successive levels. First, possible ELM/Pfam pairing are highlighted. Then, for each hit prediction, the corresponding sequence can then be aligned to matched templates and used for comparative modeling interactively (Fig 1.).

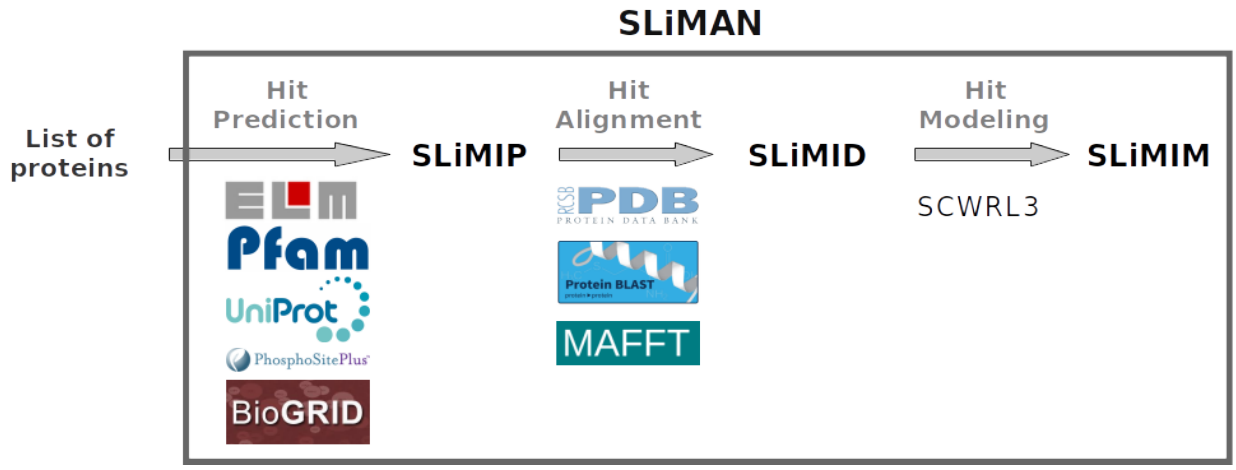


Fig 1. SLiMAN workflow annotated with used databases and software for each steps; SLiM Interaction Prediction (SLiMIP), SLiM Interacting with Domains (SLiMID) and SLiM Interaction Modeling (SLiMIM).

2.8.1. Hit Predictions

For each of the input proteins, fasta sequences and annotated domains are extracted. Regular expressions from the linear motifs referenced in ELM are used to parse the corresponding sequences. For each regular expression match, all Pfam domain matching the ELM class are extracted. For each motif, IUpred2 is used to compute the average disorder scores (Short, Long, ShortDom, LongDom, Anchor2). In addition, for the two partners paired, the PPI database (BioGRID) is searched and high and low throughput experimental interactions are counted. The final prediction, for a given motif-domain association (hit), is a set of 17 descriptors (ELM-class-name, matched_motif, motif_uniprotId, motif_start, motif_end, associated_Pfam, Pfam_uniprotId, ELM_experimental_evidence, ELM_E-Value, ShortDS, LongDS, ShortDomainDS, LongDomainDS, ANCHOR2DS, StrictDisorder, LowTBioGRIDInteractionsCount and HighTBioGRIDInteractionsCount). The four most important parameters (ELM_E-Value, StrictDisorder, TotalBioGRID count and LowTBioGRIDInteractionsCount) are used to computed a SLiMIP confidence score. All predicted associations (hits) are written in a tabular separated value file.

2.8.2. Hit Alignments

For a given hit, motif and domain sequences are aligned (c.f. 2.5 - Sequence alignments) with corresponding extracted templates. Alignment metrics are computed, to help template selection. The corresponding sequence-structure alignments can be visualized with a color code indicating either the conservation (query sequences) and the contacts (template sequences) for both the domain and the peptide motif.

2.8.3. Hit Modeling

For each of the templates previously selected, a model of the complex is generated (c.f. 2.7 - Comparative modeling). It can be visualized through the JSmol[19] applet and dowloaded for further analyses.

3. Results

3.1. PDB Template Extraction

The parsing of the PDB in search for suitable templates for the 291 ELM/Pfam associations led into the extraction of 5064 templates (from 2228 structures). The resulting database allows fast template selection and motif-domain modeling for 201 ELM/Pfam associations that could find at least one template. For the remaining 90 associations, no template could be extracted and therefore alignments/modeling can not be performed.

3.2. SLiMAN - Web Interface

SLiMAN (*http://sliman.cbs.cnrs.fr*) was designed as a web application, enabling a visual and interactive representation of the results and its use by the community from web browser. SLiMAN webserver is devided in 4 major sections: SLiMAN, SLiMIP, SLiMID and SLiMIM.

On the SLiMAN homepage, users are asked to input a list of proteins, as a fasta file and/or a list of uniprot accession numbers or entry names separated by a coma, to start a new SLiMAN project. First, input entries are checked, and then the initialization of the new project is triggered to find motifs and matching domains.

3.2.1. SLiMIP – Short Linear Motif Interaction Prediction

From the inputed list of proteins, possible pairings are searched and the resulting predictions are displayed in a table, in which motifs are displayed in columns and interacting domains in rows. Residues corresponding to an ELM motifs are displayed with the corresponding PTM from PhosphoSitePlus® annotation, when available. On the same page, a parameter pannel is displayed, allowing the user to change the paramerters for hit filtering based on 20 different criteria (ELM(x8), Iupred(x6), BioGRID(x3) and SLiMAN(x3)). Additionally, ordering modification of the filtered hits (input, alphabetic or cluster order) is possible. Parameters can be modified at will to navigate into the predicted results with distinct stringency. Displayed hits (filtered in), are colorized according to the confidence score, and links to SLiMID and SLiMIM are displayed when templates are available. By clicking the ‘Alignments’ link, a SLiMID query of the corresponding hit is launched.

3.2.2. SLiMID – Short Linear Motif Interacting with Domains

Triggered from the SLiMIP result table, SLiMID will preform sequence alignments of the hit (peptide and domain) with corresponding template sequences matching the same ELM/Pfam association, and alignment metrics are computed. Matched residues from the motif and the domain are highlighted, in green, on the full protein sequence, to better identify the region of interest. In addition domain/peptide boundaries can be modified at will. At any time, results can be downloaded for further analyses. Resulting alignments are displayed in a table, which can be sorted by the different alignment metrics, and corresponding templates can be selected. Once the selection is made (for at least one template), users can launch a SLiMIM query, sending the corresponding peptide-alignments, domain-alignments and selected templates to comparative modeling.

3.2.3. SLiMIM - Short Linear Motif Interaction Modeling

On a submited query (hit, alignments and template selections), SLiMIM will perform comparative modeling of the peptide-domain complexe. Generated models are accessible in a table, can be visualized online using the Jsmol applet, or directly downloaded. User are allowed to validate or discard models according to their expertise. Validated models are forwarded to the SLiMIP result table.

3.3. BioGRID extention

An other feature proposed by SLiMAN in the input section, is the BioGRID extention analysis. On this extention, the BioGRID database is analyzed with the inputed proteins, and the resulting interactors (for each entry) are displayed in a table, sorted by number of interactions. Such extracted list can be submitted to SLiMAN, using the ‘QuickLaunch’ button (or by copy-paste of the list of interactants to the SLiMAN input section).

A

ELM ANALYSIS					FILTERED UNIPROT LIST	
					GSK3B_HUMAN P49841	XPO1_HUMAN O14980
UNIPROT	ELM_Class	Motif	Location	Class_RegEx	PF00069 Pkinase	PF08389 Xpo1
FRAT2_HUMAN O75474	TRG_NES_CRM1_1	DRLVAQIGETLQLD	41-54	([DEQ].{0,1}[LIM].{2,3}[LIVMF] [^P]{2,3}[LMVF].[LMIV].{0,3}[DE])| ([DE].{0,1}[LIM].{2,3}[LIVMF] [^P]{2,3}[LMVF].[LMIV].{0,3}[DEQ])		Alignments(8) Models
	DOC_MAPK_gen_1	KARPPAVPL	85-93	[KR]{0,2}[KR].{0,2}[KR].{2,4} [ILVM].[ILVF]	Alignments(46)	
	DOC_MAPK_MEF2A_6	KARPPAVPL	85-93	[RK].{2,4}[LIVMP].[LIV].[LIVMF]	Alignments(10)	
	DOC_MAPK_DCC_7	KARPPAVPL	85-93	[RK].{2,4}[LIVP]P.[LIV].[LIVMF]| [RK].{2,4}[LIVP].P[LIV].[LIVMF]	Alignments(4)	

B

Selector	PDB ID	Entry	Reso Å	Motif / Domain	ChainID	%Ident	%QueryCov / %TemplCov	Nb Contacts (%)	Aligned Sequences
☑	6CIT	6CIT_C_cPep_80_93_D.pdb	2.03	ELM	D	**21.4**	100.0 100.0	40 100.0	DRLVAQIGETLQLD-- DEMTKKFGTLTIHD--
				PFam	C	**76.2**	100.0 100.0	36 (**100.0**)	WKFLKTVVNKLFEFMHETHD(WNFLRTVILKLFEFMHETHE(

C

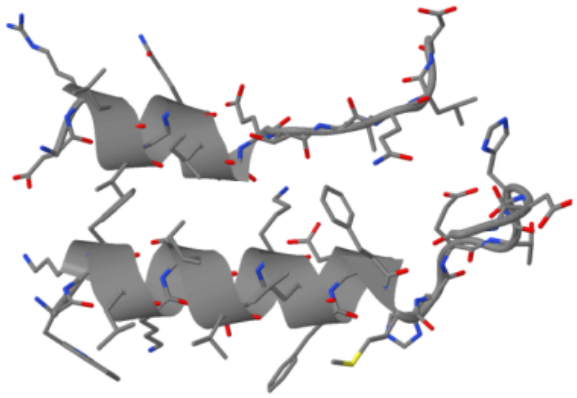


Fig 2. A) Part of the SLiMIP results table. FRAT2 NES motif able to bind to XPO1 (top-right). Eight templates are available for alignment and modeling. B) Part of the SLiMID alignments table, where the FRAT2 NES motif and the XPO1 domain are aligned with a closely related template, which is selected for comparative modeling. C) JSmol view of the downloaded comparative modeling generated by SLiMIM of the FRAT2 NES motif (top) interacting with XPO1 (bottom) using a template extracted from the PDB 6CIT structure.

3.4. Case study – XPO1 and FRAT2

FRAT2 is an inhibitor of GSK3-beta (GSK3beta), which is a well-studied protein-kinase known to shuttle between the nucleus and the cytoplasm. Interogating usual databases like Uniprot, PubMed, BioGRID or STRING does not provide any clue for the precise mechanism involved in this translocation. From BioGRID, a total of 14 interactors can be retrieved for FRAT2, including XPO1 and GSK3beta. With STRING databases, queries with FRAT2 retrieve GSK3beta, but not XPO1, while request with XPO1 simply retrieve neither GSK3beta nor FRAT2.

To find a possible mechanism of translocation, we used SLiMAN to analyze all possible SLiMs involved in the system (Fig 2). To tackle this task, we used the BioGRID extention to gather all known interactors of FRAT2. In a few clicks, we were able to identify a specific Nuclear Export Signal (NES) motif present in FRAT2. GSK3beta and FRAT2 have no documented NES motif in Uniprot. But SLiMAN straightforwardly highlights that FRAT2 harbors such a motif and that FRAT2 has been found to interact with a major exportin XPO1. Indeed, one high-throughput experiment detected an interaction between XPO1 and FRAT2[20]. An experimental validation using directed mutagenesis of the NES signal in FRAT2 was performed almost a decade ago[21] but this information did not make its way to most databases. Interestingly, homology

modeling of the corresponding interface with XPO1 (Fig 2.D), using SLiMAN, reveals an interaction close to a position in XPO1 (E591) that is mutated in chronic lymphocytic leukemia[22]. Accordingly, not only a clearer picture of the shuttling mechanism of GSK3beta was unraveled but also a new hypothesis could be drawn for the role of the mutation of XPO1 in an particular cancer.

4. Discussion

The webserver SLiMAN (*http://sliman.cbs.cnrs.fr*) provides an unprecedented tool for rapid and user-friendly survey of possible physical interactions between proteins. As such it represents a novel opportunity to analyze more deeply data from any type of interactomic studies.

Acknowledgements

Jean-Luc Pons is acknowledge for installation and survey of the server hardware and La Ligue contre le Cancer for granting the doctoral funds of Victor Reys.

References

1. Hein MY, Hubner NC, Poser I, *et al,*. A human interactome in three quantitative dimensions organized by stoichiometries and abundances. *Cell*, 163(3):712-723, 2015.
2. Van Roey Kim *et al.*, Short linear motifs: ubiquitous and functionally diverse protein interaction modules directing cell regulation. *Chemical reviews*, 114,13:6733-78, 2014.
3. Davey NE, Van Roey K, Weatheritt RJ, *et al.*, Attributes of short linear motifs. *Mol Biosyst*, 8(1):268-281, 2012.
4. Kumar M, Gouw M, Michael S, *et al.*, ELM-the eukaryotic linear motif resource in 2020. *Nucleic Acids Res.*, 48(D1):D296-D306, 2020.
5. Mistry J, Chuguransky S, Williams L, *et al.*, Pfam: The protein families database in 2021. *Nucleic Acids Res*., 49(D1):D412-D419, 2021.
6. Fuxreiter M, Tompa P, Simon I. Local structural disorder imparts plasticity on linear motifs. *Bioinformatics*, 23(8):950-956, 2007.
7. Erdős G, Dosztányi Z. Analyzing Protein Disorder with IUPred2A. *Curr Protoc Bioinformatics*, 70(1):e99, 2020.
8. von Mering C, Huynen M, Jaeggi D, Schmidt S, Bork P, Snel B. STRING: a database of predicted functional associations between proteins. *Nucleic Acids Res*., 31(1):258-261, 2003.
9. Oughtred R, Rust J, Chang C, *et al.*, The BioGRID database: A comprehensive biomedical resource of curated protein, genetic, and chemical interactions. *Protein Sci*., 30(1):187-200, 2021.
10. Mosca R, Céol A, Aloy P. Interactome3D: adding structural details to protein networks. *Nat Methods*, 10(1):47-53, 2013.
11. Postic G, Marcoux J, Reys V, *et al.*, Probing Protein Interaction Networks by Combining MS-Based Proteomics and Structural Data Integration. *J Proteome Res*., 19(7):2807-2820, 2020.
12. Zhang QC, Petrey D, Garzón JI, Deng L, Honig B. PrePPI: a structure-informed database of protein-protein interactions. *Nucleic Acids Res*., 41(Database issue):D828-D833, 2013.
13. UniProt Consortium. UniProt: the universal protein knowledgebase in 2021. *Nucleic Acids Res*., 49(D1):D480-D489, 2021.
14. Hornbeck PV, Zhang B, Murray B, Kornhauser JM, Latham V, Skrzypek E. PhosphoSitePlus, 2014: mutations, PTMs and recalibrations. *Nucleic Acids Res.*, 43(Database issue):D512-D520, 2015.
15. Burley SK, Bhikadiya C, Bi C, *et al.*, RCSB Protein Data Bank: powerful new tools for exploring 3D structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotechnology, bioengineering and energy sciences. *Nucleic Acids Res*., 49(D1):D437-D451, 2021.
16. Katoh K, Misawa K, Kuma K, Miyata T. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Res.*, 30(14):3059-3066, 2002.
17. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol*., 215(3):403-410, 1990.
18. Wang Q, Canutescu AA, Dunbrack RL Jr. SCWRL and MolIDE: computer programs for side-chain conformation prediction and homology modeling. *Nat Protoc.*, 3(12):1832-1847, 2008.
19. Jmol: an open-source Java viewer for chemical structures in 3D (*http://www.jmol.org/)*.
20. Kırlı K, Karaca S, Dehne HJ, *et al.*, A deep proteomics perspective on CRM1-mediated nuclear export and nucleocytoplasmic partitioning. *Elife*, 4:e11466, 2015.
21. Bechard M, Trost R, Singh AM, Dalton S. Frat is a phosphatidylinositol 3-kinase/Akt-regulated determinant of glycogen synthase kinase 3β subcellular localization in pluripotent cells. *Mol Cell Biol*., 32(2):288-296, 2012.
22. Walker JS, Hing ZA, Harrington B, *et al.*, Recurrent XPO1 mutations alter pathogenesis of chronic lymphocytic leukemia. *J Hematol Oncol*., 14(1):17, 2021.

Structural analysis of interaction between SARS-CoV-2 spike protein and the human ACE2 receptor

S.Naceri [1*], M.Ghoula[1*], S.Sitruk[1], G.Moroy[1], D.Flatters[1], A-C.Camproux[1]

[1] Université de Paris, BFA, UMR 8251, CNRS, ERL U1133, Inserm, F-75013 Paris, France

Corresponding Author: sarah.naceri@etu.u-paris.fr

Paper Reference: Molecular Dynamics Simulations of Influenza A Virus NS1 Reveal a Remarkably Stable RNA-Binding Domain Harboring Promising Druggable Pockets.Abi Hussein H, Geneix C, Cauvin C, Marc D, Flatters D, Camproux AC. Viruses. 2020 May 14;12(5):537. doi: 10.3390/v12050537.

Abstract:

In 2019, the emergence of the highly pathogenic SARS-CoV-2 coronavirus, which spread rapidly in 2020, led to an intensive search on this virus for a rapid development of vaccines and became a global public health priority. At the same time, the search for drug candidates to inhibit the virus mechanism and reduce the overall infection has also become a priority. The Spike protein is a target protein of interest because it allows the virus to enter human cells by interacting with human ACE2 (Angiotensin-converting enzyme 2). In addition, this protein is the target of antibodies produced by the host after infection. Several three-dimensional structures of the Spike protein, alone or in interaction with different protein partners, are currently available. These structures were mainly resolved by cryo-electron microscopy, but also to a lesser extent by X-ray crystallography. Thus, at the end of March 2020, 10 structures of the spike protein were available from the Protein Databank (PDB) **[1]**, reaching now more than 336 available structures. The aim of this work was to analyze and characterize the interaction between the Spike Receptor Binding Domain (RBD) which is located on the S1 subunit of the Spike protein and the human ACE2. Then, the surface properties of the RBD was explored to identify pockets that could be recognized by therapeutic molecule to block the RBD-ACE2 interaction. The druggability of a target (its ability to bind drug-like molecules), specifically of its binding site, can be predicted from its 3D structure **[2]** using physicochemical and geometrical parameters to characterize the pockets. In our study, two RBD-ACE2 complex structures (6M0J and 6LZG PDB) were used to understand the interaction mechanism of both proteins. As proteins are known to be highly flexible **[3],** the RBD-ACE2 complex and the isolated RBD domain were studied through Molecular Dynamics simulations (MD) using GROMACS software **[4]** in order to identify the proteins's movements, predict pockets emergence of the isolated RBD during MD and characterize key residues involved in the interaction of the complex. We ran 20 simulations of 100ns each to cover a wide range of trajectories and to sample different conformational spaces of our different systems. An extensive pocket search was conducted to detect druggable pockets in the RBD protein along the simulations using the PockDrug software **[5].** A multivariate statistical method has been applied to analyze the protein pockets extracted throughout the MD. The free binding energy of the complex were computed using the Molecular Mechanics Poisson-Boltzmann Surface Area (MM/PBSA) method **[6]** to identify key residues (hotspots) in this RBD-ACE2 interaction. Altogether, our study helped us to identify interesting druggable pockets comprising crucial key residues for the RBD-ACE2 interaction and that can be easily targeted by efficient inhibitors in order to prevent the virus infection.

References

[1] Berman, H. M. et al., *The protein data bank*. Nucleic Acids Res, 2000

[2] Abi Hussein, H. et al., *Global vision of druggability issues, applications and perspectives*, DRUG Discov Today, 2017

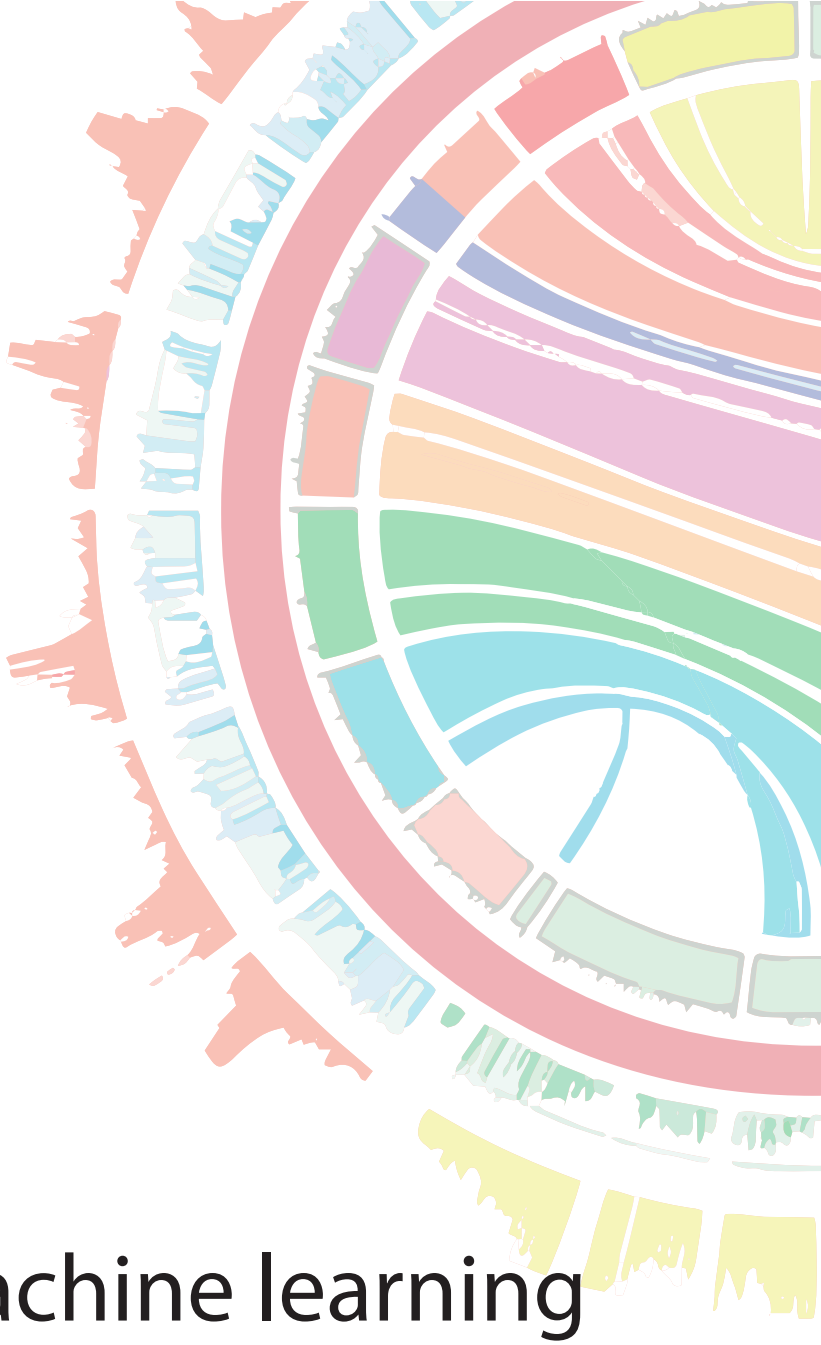
[3] Regad, L. et al., *Exploring the potential of structural alphabet-based tool for mining multiple target conformations and target flexibility insight.* PLoS One, 2017

[4] Abraham, M. J. et al., *GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers*. SoftwareX, 2015

[5] Borrel, A. et al., *PockDrug-Server: a new web server for predicting pocket druggability on holo and apo proteins*. Nucleic Acids Research, 2015

[6] Kumari, R. et al., *g_mmpbsa – A GROMACS tool for hight-throughput MM-PBSA calculations*. J. Chem. Inf. Model, 2014

[1]



Session 11

Statistics, machine learning & artificial intelligence II

Benchmarking joint multi-omics dimensionality reduction approaches for the study of cancer

Laura Cantini[1], Pooya Zakeri[2,5], Celine Hernandez[1,6], Aurelien Naldi[1,7], Denis Thieffry[1],Elisabeth Remy[3] Anaïs Baudot[2,4]

[1]Computational Systems Biology Team, Institut de Biologie de l'Ecole Normale Supérieure, CNRS, INSERM, Ecole Normale Supérieure, Université PSL, 75005Paris, France.
[2]Aix Marseille Univ, INSERM, MMG, Marseille Medical Genetics, CNRS, Turing Center for Living Systems, Marseille, France.
[3]Aix MarseilleUniv, CNRS, Centrale Marseille, I2M, Turing Center for Living Systems, Marseille, France.
[4]Barcelona Supercomputing Center (BSC), Barcelona 08034,Spain.
[5]Present address: Centre for Brain and Disease Research, Flanders Institute for Biotechnology (VIB), Leuven, Belgium and Department ofNeurosciences and Leuven Brain Institute, KU Leuven, Leuven, Belgium.
[6]Present address: Université Paris-Saclay, CEA, CNRS, Institute for Integrative Biology of the Cell (I2BC), 91198 Gif-sur-Yvette, France.
[7]Present address: Inria Saclay Ile de France, EP Lifeware, Palaiseau, France.

Corresponding Author: laura.cantini@ens.psl.eu

Paper Reference: **Cantini, L., et al. (2021). Benchmarking joint multi-omics dimensionality reduction approaches for the study of cancer.** ***Nature communications*****,** ***12*****(1), 1-12. https://doi.org/10.1038/s41467-020-20430-7**

High-dimensional multi-omics data are now standard in biology [1]. They can greatly enhance our understanding of biological systems when effectively integrated. To achieve this multi-omics data integration, Joint Dimensionality Reduction (jDR) methods are among the most efficient approaches [2,3,4]. However, several jDR methods are available, urging the need for a comprehensive benchmark with practical guidelines.

We performed a systematic evaluation of nine representative jDR methods using three complementary benchmarks. First, we evaluated their performances in retrieving ground-truth sample clustering from simulated multi-omics datasets. Second, we used TCGA cancer data to assess their strengths in predicting survival, clinical annotations and known pathways/biological processes. Finally, we assessed their classification of multi-omics single- cell data.

From these in-depth comparisons, we observed that intNMF performs best in clustering, while MCIA offers a consistent and effective behavior across many contexts. The full code of this benchmark is implemented in a Jupyter notebook - multi-omics mix (momix) - to foster reproducibility, and support data producers, users and future developers.

References

1. Cancer Genome Atlas Research Network et al. The Cancer Genome Atlas Pan-Cancer analysis project. Nat. Genet. 45, 1113–1120 (2013).

2. Bersanelli, M. et al. Methods for the integration of multi-omics data: mathematical aspects. BMC Bioinformatics 17 Suppl 2, 15 (2016).

3. Huang, S., Chaudhary, K. & Garmire, L. X. More Is Better: Recent Progress in Multi-Omics Data Integration Methods. Front. Genet. 8, 84 (2017).

4. Meng, C. et al. Dimension reduction techniques for the integrative analysis of multi-omics data. Brief. Bioinform. 17, 628–641 (2016).

The SgenoLasso and its cousins for selective genotyping and extreme sampling

Charles-Elie Rabier[1,2] and Céline Delmas[3]
[1] Institut Alexander Grothendieck Montpellier Institute (IMAG), Université de Montpellier, CNRS, France
[2] Institut des Sciences de l'Evolution (ISEM), Université de Montpellier, CNRS, IRD, EPHE, Montpellier, France
[3] INRAE, UR MIAT, Université de Toulouse, Castanet-Tolosan, France

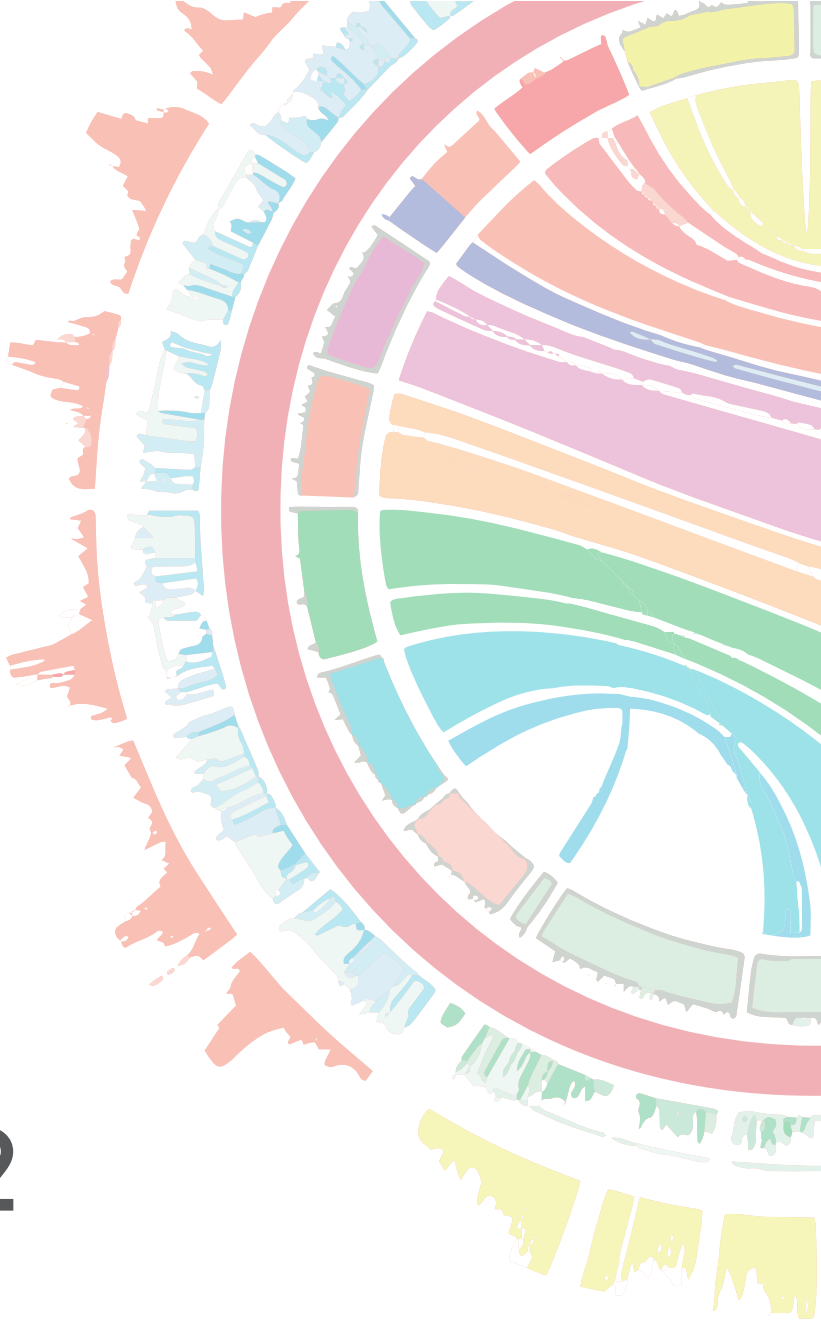
Corresponding author: charles-elie.rabier@umontpellier.fr

Reference paper: **Rabier *et al.* (2021). The SgenoLasso and its cousins for selective genotyping and extreme sampling: application to association studies and genomic selection. *Statistics: A Journal of Theoretical and Applied Statistics, 55(1).***
https://www.tandfonline.com/doi/full/10.1080/02331888.2021.1881785

Keywords: Selective Genotyping, Genomic Selection, Variable Selection, Prediction Accuracy, High Dimension, Lasso, Rice data

Context: In a seminal paper, Lebowitz et al. (1987) showed that the extreme observations of a given trait (i.e. the highest or the lowest observations) contain most of the signal on Quantitative Trait Loci, so-called QTL (genes influencing a quantitative trait which is able to be measured). As a consequence, the authors suggested to genotype only the individuals with extreme phenotypes. This concept is called selective genotyping and it was formalized later by Lander and Bostein (1989). Genome Wide Association Study (GWAS) and Genomic Selection (GS) are today two research topics using the selective genotyping methodology.
We denote some recent association studies using selective genotyping in plants (e.g. sugarcane, Gutierrez et al. 2018; tomatoes, Ohlson et al. 2018) in animals (e.g. dairy cattle, Kurz et al. 2019), and in humans (e.g. on intelligence, Zabaneh et al. 2018). Selective genotyping is particularly rewarding for finding QTLs: by considering the extremes, the signal is significantly increased. The second application field of selective genotyping is Genomic Selection (GS) (Hayes et al., 2001), which is nowadays a very popular topic in genomics (e.g. strawberry, Gezan et al. 2017; banana, Nyine et al. 2018). The main goal of GS is to select individuals (i.e. candidates) by means of genomic predictions. Since predictions can be performed as soon as the DNA is available, GS accelerates significantly the genetic gain. In GS, the learning model has to be recalibrated over time, otherwise it leads to unreliable predictions (see Goddard et al. 2009). As a result, when updating the model, candidates selected at the previous steps are used to train the model. This way, the model is learned on extreme individuals, which is highly linked to selective genotyping.

Results: We introduce here a new variable selection method, called SgenoLasso (for Selective genotyping Lasso), that handles extreme data. SgenoLasso allows to estimate the number of QTLs, their positions and their effects. It differs from the classical Lasso (Tibshirani 1996) since it models explicitly the extremes. SgenoLasso enjoys all known statistical properties of Lasso since the problem has been replaced in a L1 penalized regression framework. As its famous ancestor Lasso, SgenoLasso has multiple cousins: we can cite for instance SgenoElasticNet (a mixture of L1 and L2 penalties) and SgenoGroupLasso (penalty by group).
We propose a comparison with existing methods in a GWAS context, on simulated data and on rice data. SgenoLasso and its cousins outperformed existing methods (Lasso, Group Lasso, Yuan and Lin 2006, Elastic Net, Zhou and Hastie 2005, RaLasso, Fan et al. 2017, and BayesianLasso, Park and Casella 2008), specially when a unidirectional selective genotyping was performed (i.e. we genotype only the so-called best individuals with the largest phenotypes).
In GS, Zhao et al. (2012) highlighted the "drastic reduction" in terms of predictive ability when only the best individuals were used in the learning model in GS. Interestingly, Brandariz and Bernardo (2018) have shown recently that it is crucial to include a few worst individuals in the training set, to keep GS efficient. However, keeping the poorest lines in a breeding program has a non negligible cost. In this context, we show on simulated data that SgenoLasso and its cousins do not suffer from this drawback: they give satisfactory results even when only best individuals are considered.



Session 12
Algorithms & sequence data structures II

MTG-Link: filling gaps in draft genome assemblies with linked read data

Anne Guichard[1,2], Fabrice Legeai[1,2], Denis Tagu[1] and Claire Lemaitre[2]
[1] INRAE, Agrocampus Ouest, Université de Rennes, IGEPP, F-35650 Le Rheu, France
[2] Univ Rennes, CNRS, Inria, IRISA - UMR 6074, F-35000 Rennes, France

Corresponding author: anne.guichard@irisa.fr

Abstract *De novo genome assembly is a challenging task, especially for large non-model organism genomes. Low sequence coverage, genomic repeats and heterozygosity often create ambiguities in the assembly, and result in undefined sequences between contigs called "gaps". Hence, filling gaps in draft genomes has become a natural sub-problem of many de novo genome assembly projects. Even though there are several tools for closing gaps, to our knowledge none uses the long-range information of the linked read data. Linked read technologies have a great potential for filling gaps in draft genomes as they provide long-range information while maintaining the power and accuracy of short-read sequencing. In this work, we present MTG-Link, a novel gap-filling tool dedicated to linked read data. Taking advantage of the barcode information contained in the linked read dataset, a subsample of reads is first selected for each gap. These reads are then locally assembled and the resulting gap-filled sequences are automatically evaluated. We validated our approach on a real 10X genomics linked read dataset, on a set of simulated gaps, and showed that the read subsampling step of MTG-Link enables to get better gap assemblies in a time/memory efficient manner. We also applied MTG-Link on individual genomes of a mimetic butterfly (Heliconius numata), where it significantly improved the contiguity of a 1.3 Mb locus of biological interest.*

MTG-Link is freely available at https://github.com/anne-gcd/MTG-Link.

Keywords High throughput sequencing, Genome assembly, Gap-filling, Linked reads

1 Introduction

The fast development of both second and third generation sequencing technologies have been accompanied by an increased growth of the number of de novo genome assemblies, with better quality. Complete genome assemblies are crucial for downstream analysis as they enable to get better genome annotations, less genotyping errors and provide valuable information on structural variations [1].

Long-read sequencing technologies such as Pacific Biosciences and Oxford Nanopore are expected to greatly improve the quality of the assembled draft genomes. Indeed, these technologies offer much longer reads than short-read sequencing technologies (10-200 kb vs. 100-250 bp), giving the ability to span repetitive regions, define haplotypes and resolve structural rearrangements [2,3]. However, relative to short-read sequencing, long-read sequencing suffers from high error rates (10-15% vs. $\leq$ 0.3%) [4] and lower throughput [5]. Synthetic long-read sequencing approaches can also be used for genome assembly, as they provide all the benefits of short-read sequencing, besides incorporating information from long strands of DNA [6]. These include linked reads, which can be employed in synergy with true long reads to get accurate and complete genome assemblies.

With linked read technologies, such as the 10X Genomics Chromium platform, every short reads that have been sequenced from the same long DNA molecule (around 30-50 Kb) are tagged with a specific molecular barcode. Non-contiguous reads sharing the same barcode are referred to as linked reads. By linking the short reads together via a shared barcode, linked read technology provides long-range information while maintaining the power and accuracy of short-read sequencing [7,8]. Low-cost, low-input and high-accuracy linked read technologies have many applications: de novo genome assembly [8], haplotype identification [9] and structural variant calling [10]. The 10x Chromium Genomics company, which popularized this technology [9], recently stopped producing such data.

However, large volumes of data were produced and still need to be properly analyzed, and other linked read technologies such as TELL-Seq [11] and Haplotagging [12] emerged.

Complete and accurate reconstruction of large non-model organism genomes remains challenging with the current technologies and assembly tools. Problems generally reside at regions that are highly repetitive, highly heterozygous or have low coverage. All these features create ambiguities in the overlap detection between reads, resulting in undefined sequences between contigs of unknown or estimated lengths, called *gaps*.

Gap-filling methods aim at recovering the gap sequence between contigs, by performing a local assembly of the sequencing reads between the flanking sequences. Several tools have been developed for local assembly or gap-filling with short read data, such as GapCloser [13], Sealer [14], GapFiller [15], GAPPadder [16] and MindTheGap [17]. Implemented algorithms are quite different: some rely on De Bruijn graphs, others on iterative extensions based on read overlaps. While some methods use the whole input read set for assembly, others select reads of interest based on mate anchoring of paired-end or mate pair reads. Therefore, the former have difficulty assembling repeat-rich gaps while the latter are limited in the gap size. Even though there are several tools for closing gaps with short read data, to our knowledge, there is currently no tool that uses the long-range information of the linked read data, although this type of information has proven to be very useful for assembly issues.

In this work, we present MTG-Link, a novel gap-filling tool for draft genome assemblies dedicated to linked read data. The main feature of MTG-Link is that it takes advantage of the linked-read barcode information to get a subsample of reads of interest for the local assembly of each gap. It also automatically tests different parameters values and performs a qualitative evaluation of the obtained solutions. We validated our approach on a real 10X genomics dataset, in which gaps were simulated, and compared it to MindTheGap, that does not use the barcode information. We showed that the read subsampling step of MTG-Link enables to get better gap assemblies in less CPU time. We then applied our tool on several individual genomes of a mimetic butterfly (*Heliconius numata*) to improve the contiguity of a 1.3 Mb locus of biological interest.

2 Materials and Methods

2.1 Gap-filling with linked read data

Pipeline overview We propose a method, called MTG-Link, that aims at filling gaps in draft genome assemblies using linked read data. The method takes as input a set of linked reads, a GFA file with gap coordinates and an indexed BAM file obtained after mapping the linked reads onto the draft assembly. It outputs the set of gap-filled sequences in FASTA format, as well as an assembly graph file in GFA format, containing the original contigs and the obtained gap-filled sequences of each gap, together with their overlapping relationships.

The method described in this work relies on a three-step pipeline, where each gap is processed independently from the others. The first step uses the barcode information of the linked read dataset to get a subsample of reads of potential interest for gap-filling. The second step performs local assembly using this subsample of linked reads. Two different assembly algorithms are implemented and can be interchangeably used. The first one, called hereafter the *De Bruijn Graph (DBG) algorithm*, uses a de Bruijn graph data structure, and the second one, called the *Iterative Read Overlap algorithm*, is based on on-the-fly computations of read overlaps. The third step evaluates the obtained gap-filled sequence and annotates it with a quality score. The main steps are illustrated in Fig. 1.

Read subsampling The first step requires an indexed BAM file of linked reads mapped on the draft assembly and an indexed Fastq file. For each gap, it extracts the linked reads whose barcode is observed in chunk regions surrounding the gap, using the thirdparty tool LRez [18]. The chunk region size can be defined by the user, the default value being 5,000 bp. To increase specificity, we keep only the barcodes for which the number of occurrences in the union set from the two flanking sequences is larger than a user-defined parameter *-f* (by default 2). The goal of this step is to get a subsample of reads that will be used in the local assembly step, instead of using the whole set of reads, thus reducing the complexity of the assembly graph and the running time.

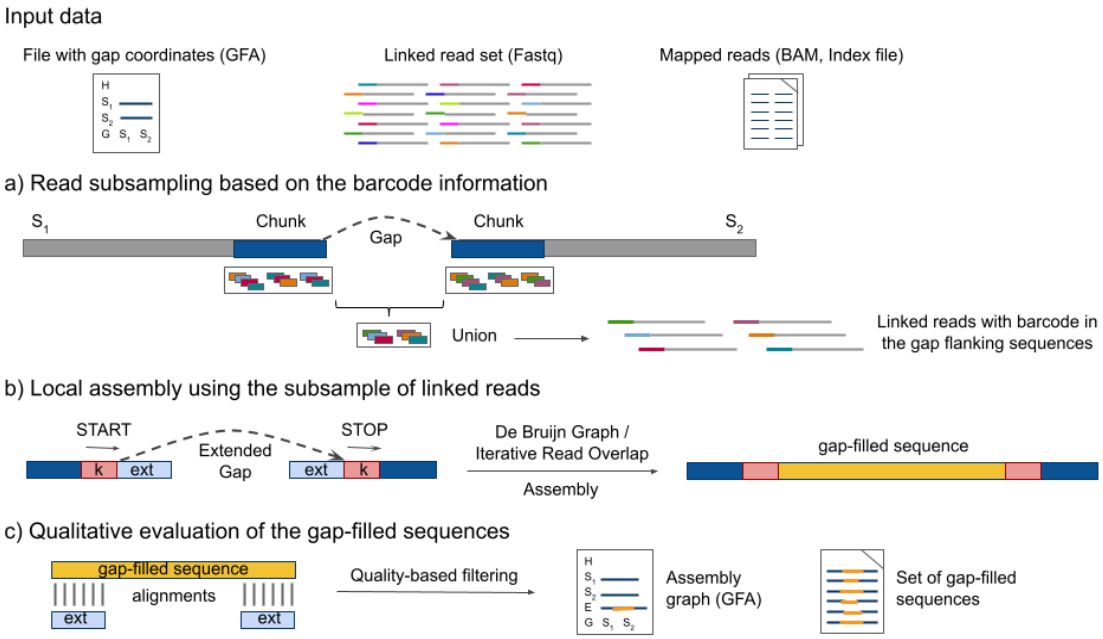


Fig. 1. Overview of the MTG-Link gap-filling pipeline. a) Linked reads whose barcode is observed in chunk regions surrounding the gap are extracted, and constitute the read subsample used in the local assembly step. b) The local assembly is performed on an extended gap, from the k-mer *START* (source) to the k-mer *STOP* (target), using the subsample of linked reads obtained in (a). c) A quality score is assigned to the gap-filled sequence according to its alignment against the gap flanking sequences. Only the gap-filled sequences with good quality scores are returned.

Local assembly To fill the gap between two contigs, we perform a local assembly using the subsample of linked reads obtained during the first step. The goal is to find a path between the source sequence and the target sequence surrounding the gap, using an assembly algorithm. To be able to further evaluate the obtained gap-filled sequence, we extend the gap on both sides by *-ext* bp (by default 500 bp). Thus, MTG-Link will perform the local assembly between the sequences surrounding the extended gap, e.g. from the k-mer *START* (source) to the k-mer *STOP* (target). Two assembly algorithms can be used during this step: the *DBG algorithm* or the *Iterative Read Overlap algorithm*.

The *DBG algorithm* is performed with the *fill* module of the software MindTheGap [17]. MindTheGap was originally developed for the detection and assembly of insertion variants, but it also includes an efficient local assembly module (*fill* module) that relies on a De Bruijn graph data structure to represent the input read sequences. Basically, starting from a source k-mer, it performs a breadth-first traversal of the De Bruijn graph, building a contig graph. The traversal is halted when the contig graph becomes too complex. Then, all the contigs in the graph are searched for the presence of the target k-mer. If one or more contigs are found containing the target k-mer, it returns all possible sequence paths between both k-mers. In MTG-link, it is then used to perform a local assembly for each pair of gap-flanking k-mers. In MindTheGap, as in any De Bruijn graph based assembly, two parameters have major impacts on the quality of the assembly: the k-mer size and the k-mer abundance threshold for including a k-mer in the graph (solid k-mer threshold). These parameters are usually set in accordance with the expected sequencing depth. In the case of MTG-link, the latter may vary depending on the efficiency of the barcode-based subsampling step. Hence for higher sensitivity, MTG-Link automatically tests different values for these two parameters, starting with the highest ones and decreasing the values if no inserted sequence with good quality is found.

MTG-Link integrates another assembly algorithm: the *Iterative Read Overlap algorithm*. This algorithm is based on on-the-fly computations of read overlaps and iterative extensions of the current assembly sequence. Overlapping reads are reads whose prefix (or reverse complement of the suffix) aligns with the suffix of the current assembly sequence with at most *-dmax* differences (including substitutions and indels) over at least $-O_{min}$ bp. These overlaps are found using a seed-and-extend

schema, combining a seed indexing with a hash table and a banded dynamic programming semi-global alignment algorithm. At each iteration, several possible extensions may be found, due to sequencing errors and/or repeats. In this case, the algorithm groups the overlapping reads together according to their extension sequence, and gives the priority to the longest overlap. To avoid including sequencing errors, only extensions that are supported by a minimum number of reads (parameter *-a*, by default 2) are considered. Then, another extension phase begins. When no overlapping read is found, or if there is no extension shared by a sufficient number of reads, or if the maximal assembled sequence size (user defined parameter) is reached, then the algorithm backtracks and tries other extensions previously encountered but not yet explored. Finally, if during an extension phase, the k-mer *STOP* is found, the assembly sequence is returned and the exploration ends.

Qualitative evaluation Each gap-filled sequence obtained during the local assembly step is evaluated to infer its quality and provide a score that might help filtering out putative erroneous sequences. The evaluation is based on the comparison of the gap-filled sequence to the gap flanking sequences, e.g. the sequences corresponding to the extensions of the gap *-ext*. Alignments are performed with *Nucmer* [19]. Then, MTG-Link assigns a two-letters quality score to each gap-filled sequence. The first letter represents the alignment to the left flanking sequence, and the second letter represents the alignment to the right flanking sequence. To have a good quality score, the gap-filled sequence must be larger than twice *-ext bp*, and it must align on at least 90% of the lengths of the gap flanking sequences. Otherwise, the gap-filled sequence obtained is assigned a bad quality score and is considered as erroneous. Only the gap-filled sequences with a good quality score are returned.

Implementation and availability MTG-Link is written in Python 3. In order to speed up the process, it uses a trivial parallelization scheme by giving each gap to a separate thread. MTG-Link is available on GitHub (`https://github.com/anne-gcd/MTG-Link`) under the GNU Affero GPL licence, and as a Bioconda package (`https://anaconda.org/bioconda/mtglink`). Additional Python scripts for converting input and output files to the desirable formats are also provided.

2.2 Validation of the method with simulated gaps

Simulated gaps We evaluated our method with a real linked read dataset but with simulated gaps in the assembly, for which we know the true sequence to be assembled (hereafter called reference sequence) in order to assess the quality of the results. One individual genome of the butterfly *Heliconius numata* was sequenced with the 10X Genomics Chromium technology and was assembled with Supernova [8] in a draft genome assembly (genome size of ∼320 Mb) [20] (BioProject PRJNA676017, individual 37). The number of reads in the dataset is approx. 110 million, with an effective read depth of 40X. We tested MTG-Link on four different gap sizes (1, 5, 10 and 20 Kbp). For each gap size, we simulated 57 gaps in the draft assembly.

MTG-Link parameters MTG-Link was used in version 1.1.0 with the same set of parameters for all gaps. For the read subsampling step, we tested different chunk sizes (5, 10 and 15 Kbp). For the local assembly step, we used the *DBG algorithm*, with a k-mer size of [61, 51, 41, 31, 21] and a solid k-mer threshold of [3, 2]. The extension size chosen was 500 bp.

Evaluation In order to evaluate the quality of the results, we performed *Blastn* [21] alignments of each obtained gap-filled sequence to the corresponding reference sequence. The gap-filled sequences having more than 85% identity and coverage with the reference sequence are labelled as "successful". However, if they have less than 85% identity and coverage with the reference sequence, they are considered as "erroneous". The "no gap-fillings" represent those for which no gap-filled sequence with a good quality score was found, e.g. no solution was returned by MTG-Link.

Comparison with other approaches To assess the impact of the read subsampling on the quality of the gap-filling, the running time and the memory consumption, we compared the results obtained with MTG-Link to those obtained with MindTheGap. As MTG-Link was run with the *DBG algorithm*, the local assembly step is the same in both approaches. The two approaches differ by the read subsampling and the qualitative evaluation steps which are specific to MTG-Link. Besides, as the read coverage can be highly variable in MTG-Link due to the read subsampling step, different *DBG* parameters values are automatically tested. On the contrary, as the whole set of reads is used for the local assembly in

MindTheGap, it was run with a unique parameter set: k-mer size (-k) of 51 and solid k-mer threshold (-a) of 3.

2.3 Application on real gaps of *Heliconius numata* genomes

We applied MTG-Link to the gap-filling of the Supergene P locus (1.3 Mbp) of the butterfly *Heliconius numata*. Twelve individuals genomes with different haplotypes were sequenced with the 10X Genomics Chromium technology and were assembled in draft genome assemblies with the Supernova assembler [20]. The number of reads in each dataset is approx. 110 million, with an effective coverage ranging from 20X to 47X (BioProject PRJNA676017). We attempted to fill the gaps between scaffolds of the Supergene P locus in eight individuals, for which this locus was fragmented. For this purpose, we re-scaffolded this locus by analyzing shared barcodes between scaffolds, and performed gap-filling with MTG-Link. MTG-Link was used with the *DBG algorithm*, with a k-mer size of [61, 51, 41, 31, 21] and a solid k-mer threshold of [3, 2]. For all other parameters, default values were used.

3 Results

3.1 Validation with simulated gaps

MTG-Link was assessed on simulated gaps of various sizes from a real linked read dataset of one *H. numata* genome. For each gap size (1, 5, 10 and 20 Kbp), we applied our tool on a GFA file containing 57 gaps. The results obtained with MTG-Link are represented by the right bars on each subplot in Fig. 2.

Among all tested gap sizes (228 gaps in total), 189 gaps were completely filled with MTG-Link and returned with a good quality score. Among them, 170 gaps have a correct assembled sequence (e.g. >85% identity and coverage with the reference sequence), hereafter referred as successful gap-fillings. Thus, MTG-Link has a precision of 90% with a recall of 75%. As we can observe in Fig. 2, the quality of the gap-filling depends primarily on the gap size. The gap-filling is mostly successful for small gaps (1 and 5 Kbp), but it is more difficult to close larger gaps (10 and 20 Kbp).

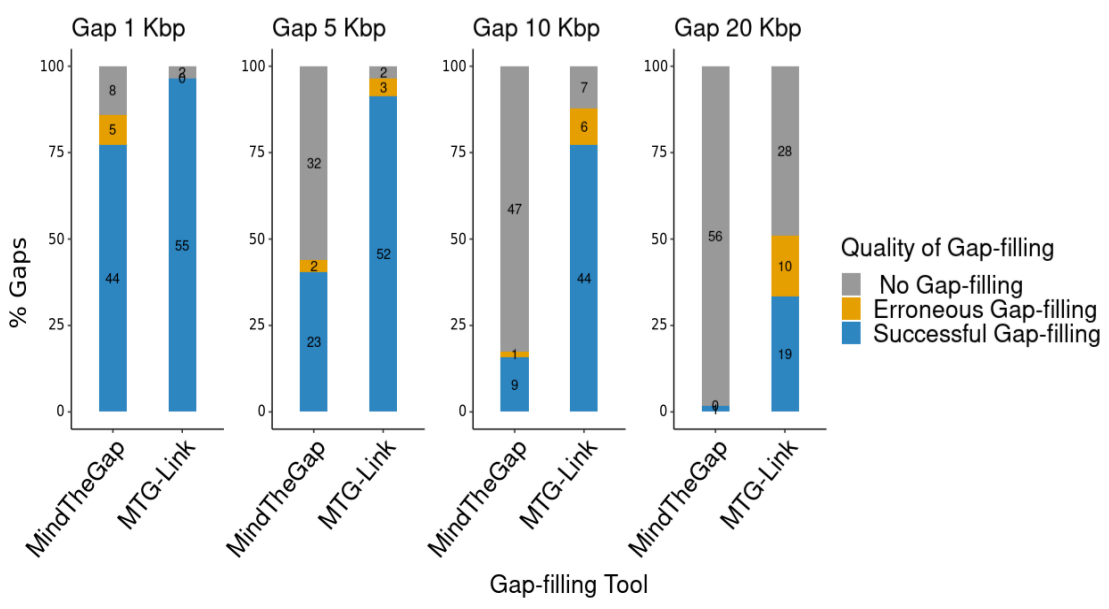


Fig. 2. Comparison of two gap-filling tools on several sets of simulated gaps. MTG-Link and MindTheGap were applied on four sets with different gap sizes, each composed of 57 simulated gaps. MTG-Link was run with the *DBG algorithm* and a chunk size of 5 Kbp.

Interestingly, we noticed that when there is no solution returned by MTG-Link (e.g. "no gap-fillings"), in some cases the number of barcodes observed in chunk regions surrounding the gap is very small ($\leq$ 500) (Fig. 3A). However, a higher number of barcodes does not guarantee that the gap will be successfully filled. Indeed, increasing the chunk size, and consequently getting a larger number of barcodes, does not improve the gap-filling (Fig. 3B). More precisely, we observed that the gaps labelled

as "no gap-fillings" but having a number of barcodes higher than 500 are those for which MTG-Link finds a solution but with a bad quality score.

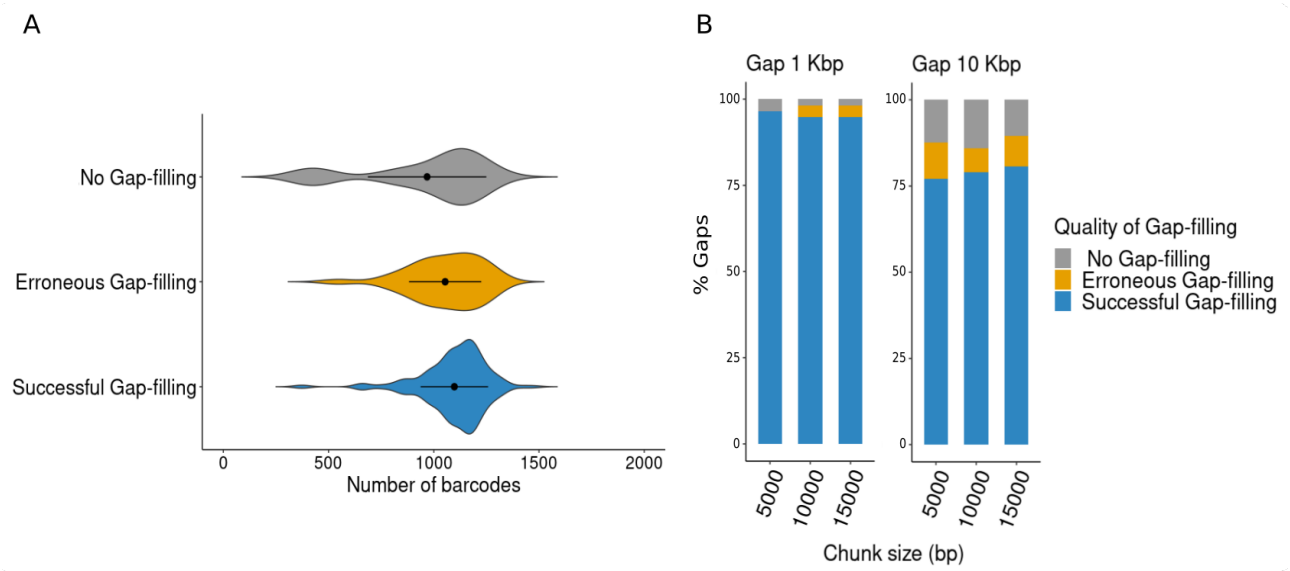


Fig. 3. Influence of two variables of the quality of the gap-filling performed by MTG-Link. A) Influence of the number of barcodes on the gap-fillings. The results shown here are obtained for all tested gap sizes (1, 5, 10 and 20 Kbp) and with a chunk size of 5 Kbp. B) Influence of the chunk size on the gap-fillings. Three different chunk sizes were tested for 1 Kbp and 10 Kbp gaps.

The erroneous gap-fillings were manually investigated. Most of the gap-fillings showed high sequence similarities with the reference sequence, but were incomplete. In several cases, we observed the presence of direct repeats in the reference sequence, generating a cycle in the De Bruijn graph whose sequence (between repeat copies) is lost in the assembly. Preliminary results obtained with the *Iterative Read Overlap algorithm* showed that this algorithm allows the correct gap-filling of some of these gaps.

The quality score assigned by MTG-Link during the qualitative evaluation step does not give a perfect auto-evaluation of the gap-filling, but it still improves its accuracy. Among the 228 tested gaps, the quality score filter enabled to discard 9 erroneous gap-fillings at the expense of losing 6 false negatives. In our method, we chose to favor precision over recall (precision of 90% with the filter vs. 86% without the filter).

Comparison with MindTheGap The gap-fillings performed by MTG-Link were compared to those obtained with MindTheGap, the tool used in the local assembly step of our pipeline. By comparing these two gap-filling tools, we are able to assess the impact of the read subsampling and the qualitative evaluation steps on the gap-filling results. Results are presented in Fig. 2. As expected, MTG-Link outperforms MindTheGap by returning more successful gap-fillings, for all tested gap sizes. Only 34% of gaps were successfully filled with MindTheGap, against 75% with MTG-Link. The differences tend to increase with the gap size. Therefore, the read subsampling and the qualitative evaluation steps greatly improve the gap-filling.

	Gap 1Kbp		Gap 5 Kbp		Gap 10 Kbp		Gap 20 Kbp	
	Time	Memory	Time	Memory	Time	Memory	Time	Memory
MTG-Link	1min27s	2.7 G	1min38s	3.1 G	2min2s	5.0 G	2min26s	13.4 G
MindTheGap	3min23s	15.1 G	3min26s	15.0 G	3min35s	15.7 G	3min34s	15.2 G

Tab. 1. Comparison of resources used by two gap-filling tools on several sets of simulated gaps. For each gap size, MTG-Link and MindTheGap were applied on a set of 57 simulated gaps. MTG-Link was run with the *DBG algorithm*. The values reported in this table are the average runtime for one gap, and the memory peak reached during each run of 57 gaps.

Importantly, MTG-Link is also significantly faster than MindTheGap. The average runtime of MTG-Link is comprised between 1.5 and 2.4 minutes per gap, which is approx. two times smaller than MindTheGap runtime (approx. 3.5 minutes per gap), as shown in Tab. 1. Although MTG-Link tests several parameters values contrary to MindTheGap, it remains faster thanks to the read subsampling step. Hence, MTG-Link is a time/memory efficient gap-filling tool.

3.2 Application on real gaps of *Heliconius numata* genomes

We applied MTG-Link on real gaps from real linked read datasets to improve the contiguity of the Supergene P locus of the butterfly *Heliconius numata*. The Supergene P locus is a locus of biological interest in *H.numata* as it controls the mimetic wing colour pattern and is subject to rearrangement polymorphism [20]. Out of the twelve individual genomes sequenced and assembled in this study, the Supergene P locus was reconstructed as a single scaffold for four individual genomes. For the other eight individual genomes, the assembly of this locus was fragmented into several scaffolds (61 gaps in total). For each of these eight individuals, we attempted to fill the gaps between the scaffolds using MTG-Link. We succeeded in reducing the number of scaffolds in the Supergene P locus for all *H. numata* individuals. For two of them, the Supergene P locus was reconstructed as a single scaffold in one step of gap-filling. For the others, the assembly was still fragmented and it required additional steps of extra contigs recruitment. Finally, after all these steps, we succeeded in filling 43 out of the 61 initial gaps with MTG-Link. This improved contiguity will allow a finer analysis of the genomic structural diversity in this locus.

4 Discussion and Conclusion

In this work, we provide a novel gap-filling tool for linked read data, called MTG-Link. This tool is composed of three main steps: read subsampling, local assembly and qualitative evaluation. To our knowledge, this is the first gap-filling tool for draft genome assemblies, dedicated to linked read data. We have therefore compared our tool MTG-Link to a generic short-read local assembly tool, MindTheGap. Both use the same *De Bruijn Graph* assembly algorithm, allowing to assess the benefit of the additional read subsampling step of MTG-Link prior to local assembly. We have shown that MTG-Link outperforms MindTheGap, in terms of both time and gap-filling quality.

Therefore, this analysis highlights the main benefit of using linked read data for the gap-filling of draft genomes, as the barcode information contained in the reads allows the enrichment of reads originating from the gap region in the read set used for the assembly. By discarding a large fraction of reads originating from other regions of the genome, we reduce the noise and complexity in the assembly graph, thus making the search for the gap-filling path easier.

A valuable feature of MTG-Link is to assign a qualitative score to each gap-filled sequence. This feature allows the pipeline to automatically test several parameters values for local assembly and to select the best solution. This is important in the context of barcode-based read subsampling, as the resulting sequencing depth and thus the optimal assembly parameters values can greatly vary between gaps. Moreover, the qualitative evaluation also allows the user to choose to prioritize the precision over the recall by using a more stringent quality score, and reciprocally.

One of the characteristics of MTG-Link is that it can use either a *De Bruijn Graph (DBG) algorithm* or an *Iterative Read Overlap algorithm* in the local assembly step. For the moment, MTG-Link was mainly tested with the *DBG algorithm*, and we have shown that this algorithm performs well, especially on small gaps. However, the gap-filling is less successful on larger gaps probably due to an increased likelihood of containing some repeated regions or a drop of sequencing depth as the distance to the gap extremities grows. In this context, the *Iterative Read Overlap algorithm* appears as a promising avenue for improvement, since it allows for variable size overlaps between reads. Longer overlaps allow to disentangle repeats larger than the k-mer size used in the de Bruijn graph but smaller than the read size, whereas smaller overlaps allow the assembly of regions of the gap covered by fewer selected reads.

Acknowledgements

This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 764840, and from the French ANR ANR-18-CE02-0019 Supergene grant. We acknowledge the GenOuest bioinformatics core facility (https://www.genouest.org) for providing the computing infrastructure. We warmly thank Mathieu Joron and Paul Jay for sharing their data and results on *H. numata* Supergene locus.

References

[1] Chaisson M.J.P., Wilson R.K., and Eichler E.E. Genetic variation and the de novo assembly of human genomes. *Nat Rev Genet.*, 16(11):627–640, 2015.

[2] Chaisson M.J.P. et al. Resolving the complexity of the human genome using single-molecule sequencing. *Nature*, 517(7536):608–611, 2015.

[3] Sedlazeck F.J. et al. Accurate detection of complex structural variations using single-molecule sequencing. *Nat Methods*, 15(6):461–468, 2018.

[4] Koren S. et al. Hybrid error correction and de novo assembly of single-molecule sequencing reads. *Nat Biotechnol.*, 30(7):693–700, 2012.

[5] Lee H. et al. Third-generation sequencing and the future of genomics. *BioRxiv*, 2016.

[6] Kuleshov V., Snyder M.P., and Batzoglou S. Genome assembly from synthetic long read clouds. *Bioinformatics*, 32(12):i216–i224, 2016.

[7] Ott A. et al. Linked read technology for assembling large complex and polyploid genomes. *BMC Genomics*, 19(651), 2018.

[8] Weisenfeld N.I., Kumar V., Shah P., Church D.M., and Jaffe D.B. Direct determination of diploid genome sequences. *Genome Res.*, 27(5):757–767, 2017.

[9] Zheng G.X.Y. et al. Haplotyping germline and cancer genomes with high-throughput linked-read sequencing. *Nat Biotechnol.*, 34(3):303–311, 2016.

[10] Spies N. et al. Genome-wide reconstruction of complex structural variants using read clouds. *Nat Methods*, 14(9):915–920, 2017.

[11] Chen Z. et al. Ultra-low input single-tube linked-read library method enables short-read second-generation sequencing systems to routinely generate highly accurate and economical long-range sequencing information. *Genome Res.*, 30(6):898–909, 2020.

[12] Meier J.I. et al. Haplotype tagging reveals parallel formation of hybrid races in two butterfly species. *BioRxiv*, 2020.

[13] Luo R. et al. SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler. *GigaScience*, 1(1):18, 2012.

[14] Paulino D. et al. Sealer: a scalable gap-closing application for finishing draft genomes. *BMC Bioinformatics*, 16(1):230, 2015.

[15] Nadalin F., Vezzi F., and Policriti A. GapFiller: a de novo assembly approach to fill the gap within paired reads. *BMC Bioinformatics*, 13(Suppl 14):S8, 2012.

[16] Chu C., Li X., and Wu Y. GAPPadder: a sensitive approach for closing gaps on draft genomes with short sequence reads. *BMC Genomics*, 20(Suppl 5):426, 2019.

[17] Rizk G., Gouin A., Chikhi R., and Lemaitre C. MindTheGap: integrated detection and assembly of short and long insertions. *Bioinformatics*, 30(24):3451–3457, 2014.

[18] Morisse P., Lemaitre C., and Legeai F. LRez: C++ API and toolkit for analyzing and managing Linked-Reads data. arXiv:2103.14419, 2021.

[19] Marçais G. et al. MUMmer4: A fast and versatile genome alignment system. *PLoS Comput Biol.*, 14(1):e1005944, 2018.

[20] Jay P., Chouteau M., Whibley A., Bastide H., Parrinello H., Llaurens V., and Joron M. Mutation load at a mimicry supergene sheds new light on the evolution of inversion polymorphisms. *Nature Genetics*, 53(3):288–293, 2021.

[21] Altschul S.F. et al. Basic local alignment search tool. *J. Mol. Biol.*, 215(3):403–410, 1990.

LEVIATHAN: efficient discovery of large structural variants by leveraging long-range information from Linked-Reads data

Pierre Morisse[1], Fabrice Legeai[1,2] and Claire Lemaitre[1]
[1] Univ Rennes, Inria, CNRS, IRISA, 35000, Rennes, France
[2] INRAE, Agrocampus Ouest, Université de Rennes, IGEPP, F-35650 Le Rheu, France

Corresponding author: pierre.morisse@inria.fr

Abstract *Linked-Reads technologies, popularized by 10x Genomics, combine the high-quality and low cost of short-reads sequencing with a long-range information by adding barcodes that tag reads originating from the same long DNA fragment. Thanks to their high-quality and long-range information, such reads are thus particularly useful for various applications such as genome scaffolding and structural variant calling. As a result, multiple structural variant calling methods were developed within the last few years. However, these methods were mainly tested on human data, and do not run well on non-human organisms, for which reference genomes are highly fragmented, or sequencing data display high levels of heterozygosity. Moreover, even on human data, most tools still require large amounts of computing resources. We present LEVIATHAN, a new structural variant calling tool that aims to address these issues, and especially better scale and apply to a wide variety of organisms. Our method relies on a barcode index, that allows to quickly compare the similarity of all possible pairs of regions in terms of amount of common barcodes. Region pairs sharing a sufficient number of barcodes are then considered as potential structural variants, and complementary, classical short reads methods are applied to further refine the breakpoint coordinates. Our experiments on simulated data underline that our method compares well to the state-of-the-art, both in terms of recall and precision, and also in terms of resource consumption. Moreover, LEVIATHAN was successfully applied to a real dataset from a non-model organism, while all other tools either failed to run or required unreasonable amounts of resources. LEVIATHAN is implemented in C++, supported on Linux platforms, and available under AGPL-3.0 License at https://github.com/morispi/LEVIATHAN.*

Keywords Linked-Reads, structural variants, variant calling, genome sequencing, sequencing data analysis

1 Introduction

Structural variants (SVs) represent variations in the structure of an organism's genome. Detecting such events is crucial, since many of them are associated with genetic diseases. Classical short-read SV calling methods usually rely on their alignment against a reference genome, and on the detection of discordant paired-read or split read signals, in order to determine the breakpoints and types of the SVs. However, due to the limited size of the short-reads, many SVs remain undetected, while many False Positive calls are reported by such methods [1].

Linked-Reads technologies rely on partitioning and barcoding of diluted high-molecular-weight DNA using a microfluidic device prior to classical short-read sequencing. Molecule sizes usually range between 10 and 50 kbp on average. However, the short-reads coverage of each molecule is usually low. Indeed, for a typical 30x sequencing depth experiment, the coverage of the reference genome by the large molecules is of about 150x, but each molecule displays a weak short-read coverage of about 0.2x. 10x Genomics popularized this technology [2], but since discontinued the sales of their Linked-Reads product lines. However, large volumes of data were produced and still need to be properly analyzed, and other technologies such as TELL-Seq [3] and Haplotagging [4] emerged, and also allow the sequencing of Linked-Reads.

Thanks to the barcodes, the origin of the short-reads fragments can be determined, and long-range information can be inferred. Such reads thus combine the high-quality of the short-reads with the

long-range information of the long-reads. As a result, Linked-Reads are particularly useful for various applications, such as genome scaffolding [5], and especially SV calling, on which we further focus below.

1.1 Related works

Since their inception, several methods were developed to detect SVs using Linked-Reads. These methods mainly focus on the detection of large SVs (around 10 kb) by leveraging the long-range information of the Linked-Reads. As of today, the nine following tools are available: Long Ranger [2,6], GROC-SVs [7], LinkedSV [8], NAIBR [9], VALOR/VALOR2 [10,11], ZoomX [12], Novel-X [13] and the NUI-pipeline [14]. Most of these methods rely on pairwise comparison of regions of the reference genome, in order to retrieve pairs of distant regions that share a higher number of barcodes than what would be expected based on their distance. Indeed, such region pairs indicate regions that actually appear close to each other on the resequenced genome, since adjacent regions are expected to share more barcodes than distant ones, and thus represent potential SV evidence. However, these methods do not make use of efficient barcode indexing strategies. As a result, they either need to store the barcodes of each region, which can be extremely memory consuming, or extract the barcodes from the same region multiple times, which can be highly time consuming.

Moreover, all of the aforementioned methods were mainly designed for human data, and especially, all of them were tested exclusively on human datasets in their respective publications. As a result, this focus on human data, and the lack of indexing strategies lead to scalability issues and to a poor applicability to non-model organisms, for which reference genomes are highly fragmented or sequencing data display high levels of heterozygosity. For example, Long Ranger displays an error message and cannot run on reference genomes composed of more than 1,000 contigs, while tools such as LinkedSV and VALOR2 can require up to more than 1 TB of RAM, and others such as GROC-SVs and NAIBR sometimes undergo an indefinite sleep after running for a few days.

1.2 Contribution

We introduce LEVIATHAN, a new Linked-Reads based SV calling method that aims to overcome these limitations, and especially mitigate resource consumption, and allow applications to non-model organisms. To achieve scalability, our method relies on a new indexing strategy, that allows to record the occurrence positions of the different barcodes through the input BAM file. This index allows to quickly and efficiently compute the number of common barcodes between all possible pairs of region that share at least one barcode. The numbers of shared barcodes between region pairs thus give a first hint as to where SVs might be located. In a second step, classical short reads methods, such as discordant paired-reads and split reads analysis are applied to region pairs sharing a sufficient number of barcodes, to further filter out false-positives, and accurately determine the types and breakpoints of actual SVs.

Using default parameters, our method can detect large SVs of at least 1,000 bp, including deletions, duplications, inversions and translocations. However, it has no support for novel insertions yet. Our experiments on simulated data show that it compares well to the state-of-the-art in terms of recall and precision, and also in terms on resource consumption. Moreover, experiments on real data also show that it manages to run on non-model organisms on which other tools either fail to run or require unreasonable amounts of resources. LEVIATHAN thus allows to analyze a wider range of datasets than the state-of-the-art, and opens doors to broader analysis of SVs in a large variety of organisms.

2 Methods

2.1 Overview

LEVIATHAN takes as input a BAM file representing the alignments of the sequencing reads of interest against a reference genome. This BAM file can either be generated by a Linked-Reads dedicated mapper, such as Long Ranger, or by any other aligner. However, when using another aligner, the reads require pre-processing prior to alignment, in order to extract the barcodes from the sequences and append them to the headers. For instance, such a pre-processing can be performed using Long Ranger basic.

LEVIATHAN relies on two distinct steps. The first step relies on the computation of the amount of common barcodes between region pairs of the reference genome, in order to highlight SVs candidates. The second step then acts as a refining step, and relies on classical short read methodologies to further filter out erroneous candidates and determine the types and breakpoints of actual SVs. An overview is given in Figure 1, and the different steps are further in the following subsections.

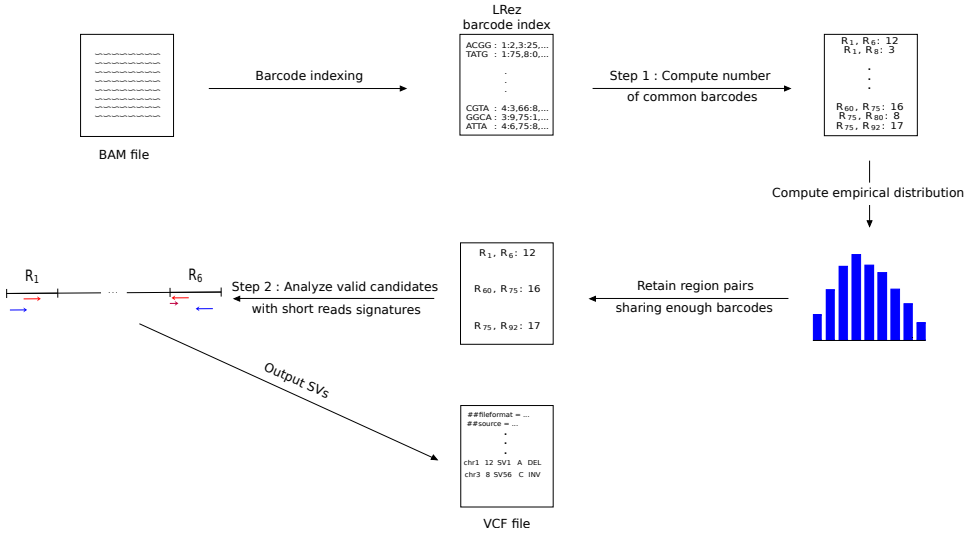


Fig. 1. Overview of the workflow of LEVIATHAN. First, the occurrence positions of the barcodes appearing in the BAM file are indexed. The first step queries the index, to identify region pairs that share at least one barcode, and compute the number of common barcodes of such pairs. The distribution is then analyzed, and a threshold above which region pairs are further considered is defined. The second, refining step, analyzes reads signatures of these pairs, in order to define the types and breakpoints of the SVs, which are output in VCF format.

2.2 Index construction

The index construction step relies on LRez [15], a tool and library designed to process Linked-Reads barcodes, which, among other functionalities, provides indexing features. LEVIATHAN thus uses LRez to build an index containing the occurrence positions of each barcode in the BAM file. This index is stored as a map, associating each barcode (in binary representation, 2 bits per nucleotide) to its list of occurrence positions in the reference genome, in format chromosome:position.

2.3 Computing the number of common barcodes between region pairs

Once the barcode index is built, the reference genome is divided in non-overlapping regions of size L ($L = 1,000$ by default, although this can be user-defined). As a result, LEVIATHAN only considers SVs whose breakpoints are located more than L bp apart on the reference genome. Iterating through the index then allows to easily identify region pairs that share at least one common barcode, as well as the exact number of common barcodes between such region pairs. This information is stored in a map, where the key is a region pair, and where the value is the number of common barcodes these regions share. This indexing and querying strategy allows to avoid the explicit comparison between every possible region pairs, and thus allows a massive speed-up. However, processing the index as such still raises a memory issue, since numerous region pairs will share only few barcodes by chance, and will need to be stored, despite the fact they will not be considered as candidates in further steps due to their weak support. To mitigate memory consumption, we iterate through the index N times ($N = 10$ by default). Given R is the total number of regions in the reference genome, for each iteration, we only compute the number of common barcodes between region pairs for which the first region is comprised between the $((i-1) * R/N + 1)$-th and the $(i * R/N)$-th region of the reference genome. At the end of each iteration, the region pairs that share less than B barcodes ($B = 1$ by default) are removed, since, as previously mentioned, they will not be considered as candidates in further steps. We set this value to 1 by default, since, for 10x Genomics Linked-Reads a given barcode does not correspond to a single

molecule, but can correspond to up to 10 different ones. As a result, it is frequent that two distant regions share a given barcode by chance, but it is much less likely that they share more. In practice, with $B = 1$, this filter allows to filter out more than 95% of the overall number of region pairs, thus greatly reducing memory consumption. During this step, we also gather statistics on the empirical distributions of the number of shared barcodes between region pairs according to their distances, which will be used in the following step.

2.4 Identifying region pairs with high numbers of common barcodes

Once the numbers of common barcodes between all the pairs of regions have been computed, we need to define a threshold above which region pairs will be considered as putative SVs candidates. Effectively, despite the fact region pairs sharing only few barcodes are dynamically filtered while querying the index, a large number of pairs will still share a low number of barcodes, simply by chance in the absence of any SV. Further analyzing all of them would thus require an unreasonable amount of resources. Moreover, it is worth noting that pairs of regions that appear close to each other on the reference genome will naturally share more barcodes than pairs of regions that either appear far from each other, or even more so, on different chromosomes. As a result, considering all region pairs as one to analyze the distribution of their numbers of shared barcodes, and thus defining a single threshold, could be misleading and lead us to ignore distant pairs of regions that contain a SV breakpoint, but share an insufficient number of barcodes.

To compensate, we consider three distinct classes of distances between regions in a pair: the pairs of regions that are located close to each other (either directly adjacent or separated by one region at most), the pairs of regions that are moderately distant (separated by two to ten regions), and the pairs of distant regions (separated by more than ten regions) or on different chromosomes.

For each distance class, we then chose the 99-th percentile of the empirical class distribution as a threshold. Candidate region pairs that share less barcodes than their associated threshold are removed and not further considered.

Finally, if a region is involved in an excessively high number of pairs ($> 1,000$ by default), all of its pairs are also removed from the candidate list. Indeed, such regions are most probably either involved in multi-mapping problems, or prone to erroneous mapping caused by repeated regions. As a result, they are thus filtered out, since the probability of a single region being involved in such a large number of SVs is feeble. Moreover, excluding such regions from further analysis once again helps us reducing computation times. Other regions passing all these filters are then independently processed.

2.5 Candidate SV processing

For each of the candidates passing the previously described filters, LEVIATHAN then investigates regular short reads signals. First, reads that map on both regions are retrieved, and only these reads are then further analyzed. Classical short-reads methods are thus applied to analyze discordant paired-read and split read signatures between these two regions, in order to further determine whether a candidate is a valid SV or not, and to identify the type and the breakpoints coordinates of the actual SVs. Figure 2 illustrates the relationship between short reads signals and SVs types.

From this analysis, multiple support values are associated to each candidate. These values register the number of shared barcodes between the two regions, the support of each SV type, the overall number of discordant paired-reads, the overall number of split reads, as well as the support of all the possible breakpoints, in each of the two regions. A candidate is then considered as a valid SV if its support values are sufficiently high. By default, the minimum required supports are at least one discordant read pair, and at least one split read, indicating the breakpoint of the SV, in each of the two regions. Candidates passing these two filters are thus considered as valid SVs.

In terms of implementation and optimization, candidates are sorted in such a way that region pairs in which a same given region is involved are gathered together, to that the alignments of that said region only need to be extracted once. Additionally, each candidate (region_1, region_2) is gathered with the other candidates of the region involved in the largest number of candidates. For instance, if region_1 is involved in three candidates, and region_2 is involved in five, the candidate (region_1, region_2) will be gathered with other candidates of region_2. This allows to further reduce the number of alignments extractions, and thus further optimize the runtime.

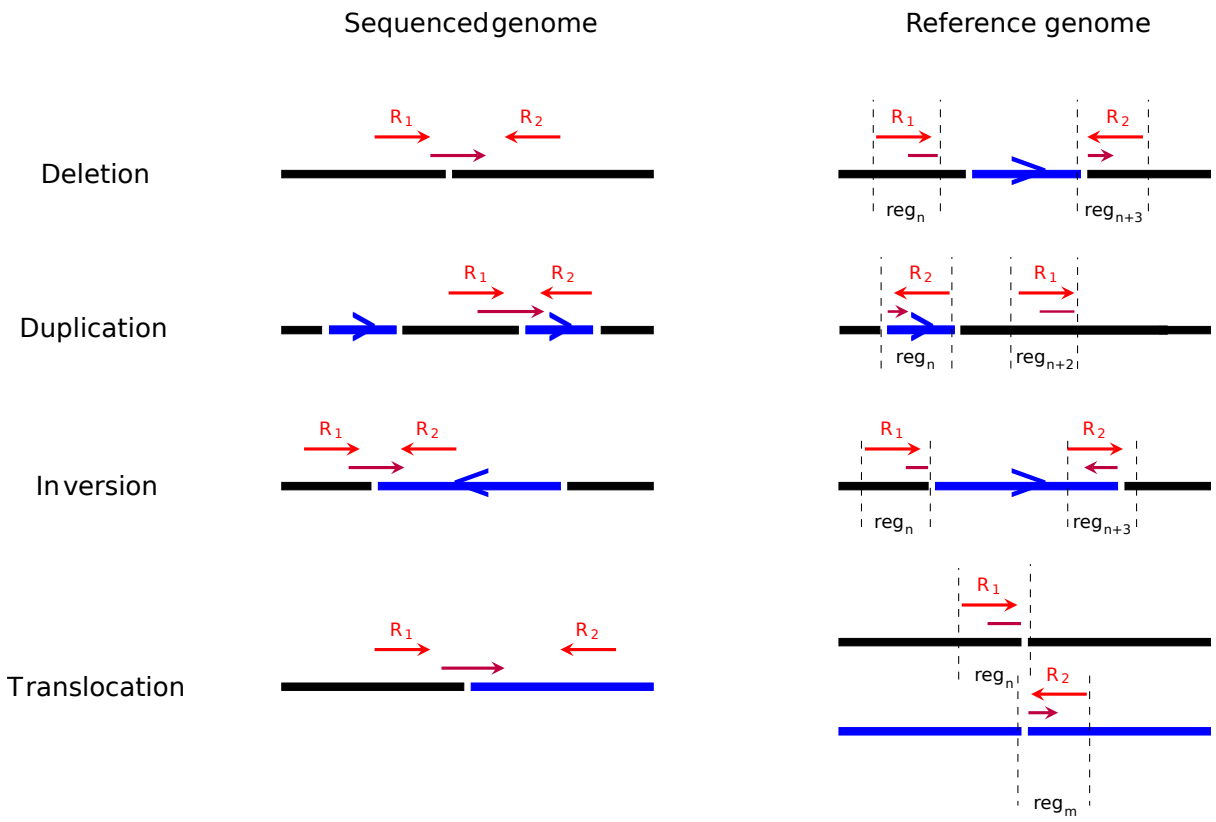


Fig. 2. Short reads signals (discordant paired-reads and split reads) used to discriminate SV types when analyzing candidates. Dashed lines represent the regions which are considered when analyzing short-reads signals.

2.6 SV filtering and output

Prior to output, a last filtering step is applied to SV candidates. It can happen that the same SV event is represented by several breakpoint pair candidates whose genomic coordinates are close (within 10 bp from each other) or identical, but with a different annotated type. In such a case, we only report the candidate with the largest cumulative support (barcode and short-read signal supports).

LEVIATHAN finally outputs its final list of SVs in VCF format, reporting detailed information regarding the SV, such as its type, its beginning and end positions, its length, the number of barcodes shared between the two involved regions, and the number of discordant paired-reads involved.

3 Results

We evaluated LEVIATHAN on simulated and real datasets. For the real data, we chose a dataset from the butterfly *Heliconius numata*, which is a non model organism and for which the discovery of structural polymorphism is of special interest. In this mimetic butterfly, it was shown that several large inversions in a 1.3 Mb locus are associated to its wing color pattern and then play a crucial role in its population biology [16]. As a non model organism, it does not have a chromosome-level reference genome, instead its 360 Mb draft genome is rather fragmented with a total of 16,950 contigs (N50: 474 kb). This genome, together with 10X genomics whole genome resequencing data of 12 individuals are available under the PRJNA676017 project ID on NCBI.

In order to properly evaluate the results quality, we also simulated data with controlled sets of SVs. To do so, we used LRSim [17] to ensure producing data that mimic the actual characteristics of Linked-Reads. We generated two datasets, one from the *H. numata* genome and one from the *H. sapiens* GrCh38 chromosome 1 (250 Mb), in order to compare the results between non-model and model organisms with similar genome sizes but different genome complexities. However, since LRSim had troubles simulating data on the highly fragmented assembly of *H. numata*, we had to filter out contigs shorter than 27,500 bp, resulting in a reference genome composed of 1,054 contigs (total size: 272 Mb and N50: 924 kb). Both datasets were simulated with a 30x coverage, and contained respec-

tively 1,348 SVs and 1,048, ranging from 1,000 to 100,000 bp, including, deletions, duplications, and inversions as well as translocations for the *H. numata* dataset. No insertions were simulated, since LEVIATHAN is not currently able to process them. Additionally, SNPs were also inserted, in order to further mimic real data.

We compared LEVIATHAN against other state-of-the-art Linked-Reads SV calling tools. All tools were run using 8 threads. LEVIATHAN was run both in fast and sensitive mode. For fast mode, the 99-th percentiles of the distributions were chosen, while for sensitive mode, the 95-th percentiles were chosen. Other tools were run with default or recommended parameters. Moreover, both simulated and real data were used in our experiments.

3.1 Validation of the method with simulated data

To precisely asses the accuracy of the different tools, we first tested them on the simulated datasets, where precise recall and precision could be computed. However, on both datasets, we could not manage to get GROC-SVs, LinkedSV and Valor to run properly. Indeed, GROC-SVs and LinkedSV crashed on both datasets, while Valor also crashed on the *H. numata* dataset, and ran for more that two days on the *H. sapiens* dataset.

For these experiments, a SV was validated as a true positive if its breakpoints were correctly predicted, within 100 bp from a true SV reported in the simulation file. Since NAIBR only reported the breakpoints of the SVs it detected, we did not take into account the SV types, in order to allow a fair comparison. Statistics of the aforementioned tools on the two simulated datasets, along with their runtime and memory consumption are reported in Table 1.

Results on the *H. sapiens* dataset show that, in terms of resource consumption, LEVIATHAN was faster than Long Ranger and NAIBR, both in fast and sensitive mode, and also required less memory, especially compared to NAIBR. However, it is worth noting that Long Ranger does not accept BAM files as an input, and that its reported runtime thus also includes reads mapping. In terms of recall, Long Ranger performed slightly better than LEVIATHAN (fast), but reached a much lower precision. Compared to NAIBR, even in fast mode, LEVIATHAN reached both higher recall and higher precision. Moreover, it is also worth nothing that, in sensitive mode, LEVIATHAN reached up to 89% of recall, while still running faster than NAIBR, and consuming the same amount of memory as in fast mode. Precision was however lower in sensitive mode than it was in fast mode, which can be explained by the fact that, in sensitive mode, a larger number of SV candidates are considered, which can increase the false-positives rate. Nonetheless, LEVIATHAN still reached more than 92% of precision in both modes, and largely outperformed both NAIBR and especially Long Ranger.

On the *H. numata* dataset, Long Ranger could not be run since it does not allow the reference genome to contain more than 1,000 contigs. Once again, the two modes of LEVIATHAN required almost three times less memory than NAIBR, and LEVIATHAN (fast) also ran faster. In terms of recall and precision, both modes of LEVIATHAN outperformed NAIBR, whose recall was particularly low, failing identifying more than half of the SVs (recall of 40.73%). In comparison, LEVIATHAN (fast) reached a recall of 63.65%, while running faster than NAIBR, and LEVIATHAN (sensitive) reached a recall of 66.54%, despite requiring a slightly larger processing time than NAIBR. In terms of precision, both modes of LEVIATHAN once again outperformed NAIBR, reaching up to 97.36% in fast mode. While LEVIATHAN still outperformed NAIBR on this dataset, its overall performance was not as good as on the *H. sapiens* dataset. Although this can be partially explained by the low quality of the reference genome we used, this still leaves us room for improvement, and future works should thus head in the direction of studying why such a proportion of SVs remained undetected.

3.2 Application to a real butterfly dataset

We then applied the different tools on the real dataset of *H. numata*. On this dataset, no tool except LEVIATHAN managed to run. Indeed, Long Ranger could not be run since the reference genome was composed of more than 1,000 contigs, GROC-SVs and NAIBR were stopped after 15

Dataset	Tool	Recall (%)	Precision (%)	Time	Memory (MB)
H. sapiens (chr 1)	LEVIATHAN (fast)	71.18	**97.14**	**11 min**	**4,603**
	LEVIATHAN (sensitive)	**89.03**	92.38	37 min	**4,603**
	NAIBR	68.13	78.12	44 min	25,764
	Long Ranger	72.04	50.23	11 h 29 min	7,062
	GROC-SVs[1]	-	-	> 6 hours	11,030
	LinkedSV[2]	-	-	> 19 min	9,311
	Valor[3]	-	-	> 2 days	-
H. numata	LEVIATHAN (fast)	63.65	**97.39**	**8 min**	**4,560**
	LEVIATHAN (sensitive)	**66.54**	92.95	13 min	**4,560**
	NAIBR	40.73	86.19	10 min	11,736
	Long Ranger[4]	-	-	-	-
	GROC-SVs[1]	-	-	> 24 min	1,403
	LinkedSV[2]	-	-	> 23 min	30,250
	Valor[3]	-	-	-	-

Tab. 1. Results reported by the different SV calling tools on the two simulated datasets. [1] GROC-SVs crashed after 6 hours on the *H. sapiens* dataset, and after 24 minutes on the *H. numata* dataset. [2] LinkedSV crashed after 19 minutes on the *H. sapiens* dataset, and after 23 minutes on the *H. numata* dataset. [3] Valor was killed after 2 days of computing on the *H. sapiens* dataset, and crashed upon start on the *H. numata* dataset. [4] Long Ranger could not be run on the *H. numata* dataset, since it does not allow the reference genome to contain more than 1,000 contigs.

days of computation, Valor crashed during processing, while LinkedSV also crashed and required more than 1 TB of RAM.

In comparison, LEVIATHAN managed to run in less than two hours, only required 18 GB of RAM, and reported a total of 50,000 SVs. On this dataset, we were especially interested in finding inversions located in the *supergene* locus, the locus associated to the wing color patterns of *H. numata*. On the particular individual we studied in this experiment, LEVIATHAN did report the 430 Mb inversion at its expected breakpoints. This inversion was initially detected with SNPs, as it is associated to strong sequence divergence between individuals that display or do not display it. It was then further confirmed via PCR, and breakpoints were refined by aligning different genome assemblies [16].

While other variants reported by LEVIATHAN still need to be properly analyzed, its ability to run on such non-model organisms, for which the reference genome are highly fragmented, without requiring an unreasonable amount of resources, represents a major improvement compared to the state-of-the-art. Moreover, the fact that the inversion of interest could be detected at its expected breakpoints is particularly promising for the analysis of the remaining reported SVs.

4 Discussion and conclusion

We presented LEVIATHAN, a new SV calling tool for Linked-Reads data. LEVIATHAN makes use of a barcode index, which allows us to quickly and efficiently identify the region pairs than share a high number of barcodes, which represent potential SVs. Complementary classical short reads methods are then applied, in order to further analyze such pairs of regions, and determine whether they are actual SVs, as well as their types and breakpoints in such cases.

Our experiments show that LEVIATHAN compares well to the state-of-the-art in terms of recall and precision, all the while being faster and consuming less memory. Moreover, LEVIATHAN also allows the analysis of non-model organisms on which other tools do not manage to run or require an unreasonable amount of resources. As a result, LEVIATHAN thus tackles the main limitations of other Linked-Reads SV calling tools, and allows to better scale to large datasets, as well as to accurately analyze non-model organisms. We thus believe LEVIATHAN could allow the discovery of new sets of SVs on a broad range of such datasets.

As future work, we are, first of all, planning to try LEVIATHAN on several other non-model datasets in order to detect new SVs. Moreover, we are also planning to integrate a local assembly feature, that would not only help us detect SV breakpoints more accurately, but that would also allow us to detect novel insertion variants. Other optimizations, such as analysis and comparison of

SVs sharing common breakpoints, to better determine their types, and in-depth study of undetected SVs in simulated data are also currently being investigated. We are also considering integrating other short-reads classical approaches to LEVIATHAN, in order to allow the detection of shorter SVs. Since most Linked-Reads based SV calling tools only focus on large SVs, this would allow to detect a broader range of SVs, while having to run a single tool. Finally, while LEVIATHAN is currently designed for and has only been tested on 10x Genomics Linked-Reads, adaptation to other barcoded Linked-Reads technologies seems to be feasible at the expense of minimal additional work. As a result, we are planning to integrate support for other technologies in the near future. In particular, the Haplotagging technology has been designed for re-sequencing a large number of individuals at a low cost [4], and its relevance has been illustrated with the discovery of large inversions in a non-model butterfly. Our tool will be particularly fitted for these applications that are likely to become widely used for population genomics analyses.

Acknowledgements

This project has received funding from the French ANR ANR-18-CE02-0019 Supergene grant. We acknowledge the GenOuest bioinformatics core facility (https://www.genouest.org) for providing the computing infrastructure. We warmly thank Mathieu Joron and Paul Jay for sharing their data and results on Heliconius inversions.

References

[1] Mahmoud M et al. Structural variant calling: the long and the short of it. *Genome Biology*, 20(1), nov 2019.

[2] Brock Medsker et al. Haplotyping germline and cancer genomes using high- throughput linked-read sequencing. *Nature Biotechnology*, 34(3):303–311, 2016.

[3] Zhoutao Chen et al. Ultra-low input single tube linked-read library method enables short-read second-generation sequencing systems to generate highly accurate and economical long-range sequencing information routinely. *Genome Research*, 2020.

[4] Joana I. Meier et al. Haplotype tagging reveals parallel formation of hybrid races in two butterfly species. *bioRxiv*, pages 1–27, 2020.

[5] Sarah Yeo et al. ARCS: Scaffolding genome drafts with linked reads. *Bioinformatics*, 34(5):725–731, 2018.

[6] Patrick Marks et al. Resolving the full spectrum of human genome variation using Linked-Reads. *Genome Research*, 29(4):635–645, 2019.

[7] Noah Spies et al. Genome-wide reconstruction of complex structural variants using read clouds. *Nature Methods*, 14(9):915–920, 2017.

[8] Li Fang et al. LinkedSV for detection of mosaic structural variants from linked-read exome and genome sequencing data. *Nature Communications*, 10(1), 2019.

[9] Rebecca Elyanow et al. Identifying structural variants using linked-read sequencing data. *Bioinformatics*, 34(2):353–360, 2018.

[10] Marzieh Eslami Rasekh et al. Discovery of large genomic inversions using long range information. *BMC Genomics*, 18(1):10–21, 2017.

[11] Fatih Karaoğlanoğlu et al. VALOR2: characterization of large-scale structural variants using linked-reads. *Genome Biology*, 21(1), 2020.

[12] Li C Xia et al. Identification of large rearrangements in cancer genomes with barcode linked reads. *Nucleic Acids Research*, 46(4):1–12, 2018.

[13] Dmitry Meleshko et al. Detection and assembly of novel sequence insertions using Linked-Read technology. *bioRxiv*, page 551028, 2019.

[14] Karen H.Y. Wong et al. De novo human genome assemblies reveal spectrum of alternative haplotypes in diverse populations. *Nature Communications*, 9(1):1–9, 2018.

[15] Pierre Morisse et al. Lrez: C++ api and toolkit for analyzing and managing linked-reads data. *arXiv*, 2021.

[16] Paul Jay et al. Mutation load at a mimicry supergene sheds new light on the evolution of inversion polymorphisms. *Nat Genet*, 53(3):288–293, 03 2021.

[17] Ruibang Luo et al. LRSim: A Linked-Reads Simulator Generating Insights for Better Genome Partitioning. *Computational and Structural Biotechnology Journal*, 15:478–484, 2017.

Freddie: Annotation-independent Detection and Discovery of Transcriptomic Alternative Splicing Isoforms Using Long-read Sequencing

Baraa Orabi[1,2], Brian McConeghy[2], Cedric Chauve[3] and Faraz Hach[2,4]
[1] Department of Computer Science, the University of British Columbia, Vancouver, B.C., Canada
[2] Vancouver Prostate Centre, Vancouver, B.C., Canada
[3] Department of Mathematics, Simon Fraser University, Burnaby, B.C., Canada
[4] Department of Urologic Sciences, the University of British Columbia, Vancouver, B.C., Canada

Corresponding author: cedric.chauve@sfu.ca

Reference paper: **Freddie: Annotation-independent Detection and Discovery ofTranscriptomic Alternative Splicing Isoforms [1] (to appear in RECOMB 2021).**

Alternative splicing (AS) is an important mechanism in the development of many cancers, as novel or aberrant AS patterns play an important role as an independent onco-driver. In addition, cancer-specific AS is potentially an effective target of personalized cancer therapeutics. However, detecting AS events remains a challenging task, especially if these AS events are not pre-annotated. This is exacerbated by the fact that existing transcriptome annotation databases are far from being comprehensive, especially with regard to cancer-specific AS. Additionally, traditional sequencing technologies are severely limited by the short length of the generated reads, that rarely spans more than a single splice junction site. Given these challenges, transcriptomic long-read (LR) sequencing presents a promising potential for the detection and discovery of AS.

We present Freddie, a computational annotation-independent isoform discovery and detection tool. Freddie takes as input transcriptomic LR sequencing of a sample and computes a set of isoforms for the given sample. Freddie takes as input the genomic alignment of the transcriptomic LRs generated by a splice aligner. It then partitions the reads to sets that can be processed independently and in parallel. For each partition, Freddie segments the genomic alignment of the reads into canonical exon segments. The goal of this segmentation is to be able to represent any potential isoform as a subset of these canonical exons. This segmentation is formulated as an optimization problem and is solved with a Dynamic Programming algorithm. Then, Freddie reconstructs the isoforms by jointly clustering and error-correcting the reads using the canonical segmentation as a succinct representation. The clustering and error-correcting step is formulated as an optimization problem – the Minimum Error Clustering into Isoforms (MErCi) problem – and is solved using Integer Linear Programming (ILP).

We compare the performance of Freddie on simulated datasets with two isoform detection tools with varying dependence on annotation databases, FLAIR [2] and StringTie2 [3]. We show that Freddie outperforms them in its recall, including those given the complete ground truth annotation. In terms of false positive rate, Freddie performs comparably to the other tools. We also run Freddie on a transcriptomic LR dataset generated in-house from a prostate cancer cell line. Freddie detects a potentially novel Androgen Receptor isoform that includes novel intron retention, cross-validated using orthogonal publicly available short-read RNA-seq datasets.

Acknowledgements

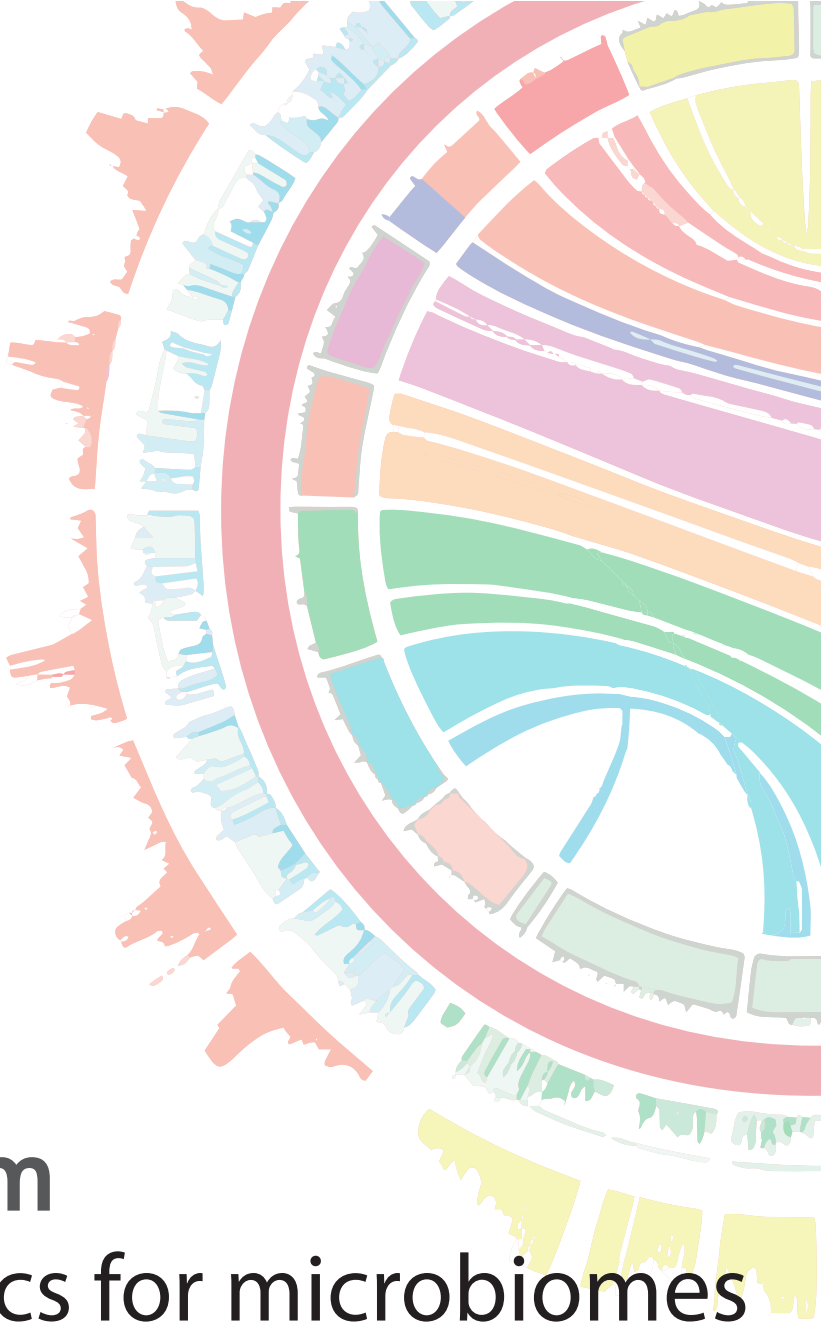
This research is funded in part by National Science and Engineering Council of Canada Discovery Grants to F.H (RGPIN-05952) and C.C (RGPIN-03986) and a Michael Smith Foundation for Health Research Scholar Award to F.H. (SCH-2020-0370).

References

[1] Baraa Orabi, Brian McConeghy, Cedric Chauve, and Faraz Hach. Freddie: Annotation-independent detection and discovery of transcriptomic alternative splicing isoforms. *bioRxiv*, 2021.

[2] Alison D Tang, Cameron M Soulette, Marijke J van Baren, Kevyn Hart, Eva Hrabeta-Robinson, Catherine J Wu, and Angela N Brooks. Full-length transcript characterization of SF3B1 mutation in chronic lymphocytic leukemia reveals downregulation of retained introns. *Nature Comm.*, 11(1):1–12, 2020.

[3] Sam Kovaka, Aleksey V Zimin, Geo M Pertea, Roham Razaghi, Steven L Salzberg, and Mihaela Pertea. Transcriptome assembly from long-read rna-seq alignments with stringtie2. *Genome Biol.*, 20(1):1–13, 2019.



Symposium
Bioinformatics for microbiomes

Translating microbiome research outputs into main stream services in MGnify

Rob FIN
European Bioinformatics Institute (EMBL-EBI), Wellcome Genome Campus, Hinxton, UK
rdf@ebi.ac.uk

Introduction

The past 5 years has witnessed a dramatic change in the knowledge that can being gleaned from metagenomic studies. Metagenomics assembly has become more routine, with the generation of metagenome assembled genomes (MAGs) providing insights into community composition from a genomic perspective. In this presentation, I will highlight some of the recent research outputs from my group, both in terms of datasets (e.g. the Unified Human Gut Genome Catalog) and tools that have been developed to access viral and eukaryotic components. Finally, I will indicate how these developments are being incorporated into the range of MGnify services, and the breadth of data that are now available.

Development of computational approaches for the prediction and analysis of microbial networksy

Lisa RÖTTJERS
Laboratory of Microbial Systems Biology, REGA Institute, KU Leuven, Herestraat 19, 3000, Leuven, Belgium
lisa.rottjers@kuleuven.be

Introduction

Microbes are able to carry out a number of interactions that can be important for ecosystem functioning. The inference of microbial networks aims to predict those interactions from microbial abundance data. Yet, there are a number of reasons why it may be challenging to infer microbial interactions. This talk will provide a brief overview of issues that affect the ability of network inference methods to predict microbial interactions. Following this discussion, several network inference tools and their performance on simulated data will be discussed. This will be concluded with a short introduction of methods for the analysis of microbial association networks that take their inaccuracy into account.

Sedimentary ancient DNA revealed irreversible plankton shifts and toxic microalgae species invasion in relation to human impact in the Bay of Brest.

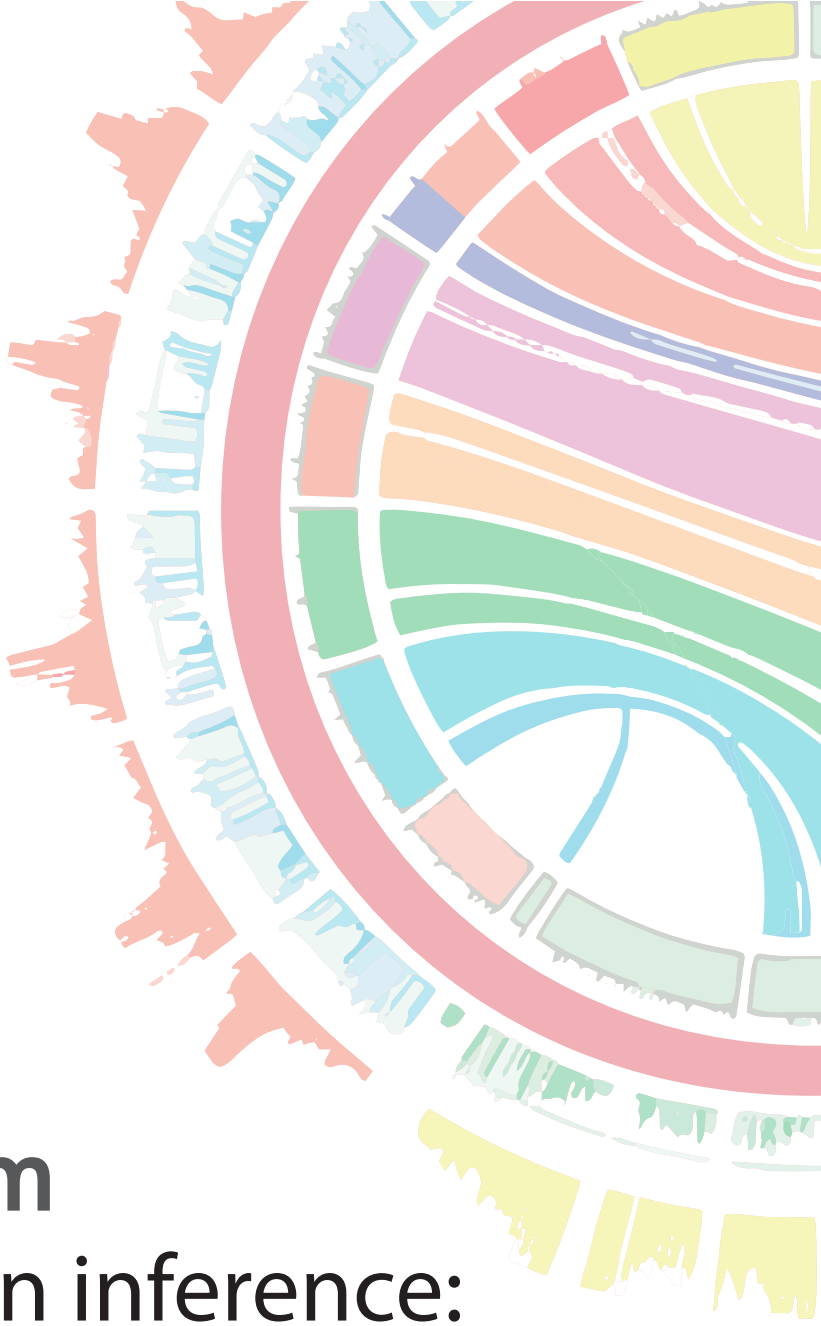
Raffaele SIANO
Ifremer - DYNECO/ Pelagos, BP70, 29280, Plouzané, France
raffaele.siano@ifremer.fr

Introduction

To evaluate he stability and resilience of coastal ecosystem communities to perturbations occurred during the Anthropocene, pre-industrial biodiversity baselines inferred from paleo-archives are needed. The study of ancient DNA (aDNA) from sediments (sedaDNA) has provided valuable information about past dynamics of specific taxa, including Harmful Algal Bloom (HABs) species, and protist communities in relation to ecosystem variations. Sediment cores collected from different sites of the Bay of Brest (Northeast Atlantic, France) allowed ca. 1400 years of retrospective analyses of the effects of human pollution on marine protists and HABs. Comparison of sedaDNA extractions and metabarcoding analyses with different barcode regions (V4 and V7 18S rDNA) revealed that protist assemblages in ancient sediments are mainly composed of species known to produce resting stages. Multivariate regression tree (MRT) analyses revealed major shifts within protist communities. Dinoflagellates and stramenopiles community variations coincided with heavy metal pollution traces in sediments ascribed to the World War II period. After the war and especially from the 1980s to 1990s, protist genera shift followed chronic contaminations of agricultural origin, showing an increase in the HAB species Alexandrium minutum across the XXth century. Community composition reconstruction over the time showed that there was no recovery to a Middle-Age baseline composition. This demonstrates the irreversibility of the observed shifts after the cumulative effect of war and agricultural pollutions. Developing a paleoecological approach, this study highlighted how human contaminations irreversibly affect the marine microbial compartments, which contributes to the debate on coastal ecosystem preservation and restoration.

Reference

Siano Raffaele, Lassudrie Duchesne Malwenn, Cuzin Pierre, Briant Nicolas, Loizeau Veronique, Schmidt Sabine, Ehrhold Axel, Mertens Kenneth, Lambert Clement, Quintric Laure, Noël Cyril, Latimier Marie, Quéré Julien, Durand Patrick, Penaud Aurélie Sediment archives reveal irreversible shifts in plankton communities after World War II and agricultural pollution . Current Biology Volume 31, Issue 12, 21 June 2021, Pages 2682 2689.e7 https://doi.org/10.1016/j.cub.2021.03.079.



Symposium
Post selection inference: valid double-dipping

Selective inference for region detection with high-throughput genomic assays

Yuval BENJAMINI[1]

[1] Department of Statistics and Data-Science, Hebrew University, Mount Scopus, 9190501, Jerusalem, Israel

Corresponding Author: yuval.benjamini@mail.huji.ac.il

1. Abstract

In both genomics and neuroscience, scientists search for spatially-coherent regions which are correlated with a covariate of interest. A popular approach is to (a) estimate a population statistic at each point, (b) threshold this map and (c) merge spatially consistent points passing the threshold into regions. However, treating these regions as if they were known a-priori would lead to biases in the estimation, especially considering the large spatial process from which they were selected.

In this talk I will present a conditional-inference approach to estimating and forming confidence intervals for the effects in regions [1]. The proposed method is based on sampling from a conditional distribution, and therefore can accommodate the non-stationary covariance in each region. The new method helps evaluate differentially methylated regions (DMRs), and is shown to have more power compared to alternatives. I will discuss challenges and ideas in adapting this framework for fMRI spatial signal detection.

References

1. Yuval Benjamini, Jonathan Taylor and Raphael Irizarry, *Selection-corrected statistical inference for region detection with high-throughput assays*, Journal of the American Statistical Association 114 (527), 1351-1365.

Inflated false discovery rate due to volcano plots: problem and solutions

Mitra EBRAHIMPOOR and Jelle J. GOEMAN

Medical statistics, Department of Biomedical Data Sciences, Leiden University Medical Center, Leiden, 2333 ZA, The Netherlands
m.ebrahimpoor@lumc.nl

Volcano plots are used to select the most interesting discoveries when too many discoveries remain after application of Benjamini–Hochberg's procedure (BH). The volcano plot suggests a double filtering procedure that selects features with both small adjusted P-value and large estimated effect size. Despite its popularity, this type of selection overlooks the fact that BH does not guarantee error control over filtered subsets of discoveries. Therefore the selected subset of features may include an inf lated number of false discoveries. Results: In this paper, we illustrate the substantially

inf lated type I error rate of volcano plot selection with simulation experiments and RNA-seq data. In particular, we show that the feature with the largest estimated effect is a very likely false positive result. Next, we investigate two alternative approaches for multiple testing with double filtering that do not inflate the false discovery rate. Our procedure is implemented in an interactive web application and is publicly available

Post-Selection Inference for Sequence Motifs

Antoine Villié[1], Philippe Veber[1], Laurent Jacob[1]
[1]Université de Lyon, Université Lyon 1, CNRS, Laboratoire de Biométrie et Biologie Évolutive UMR 5558, Villeurbanne, France

Corresponding Author: antoine.villie@univ-lyon1.fr

Genome-wide association studies aim to identify correlations between genetic variants and a trait. Standard approaches focus primarily on single-nucleotide polymorphisms or short indels, and are thus not well suited to accessory genomes, translocations, meta-genomes or repeated regions. More recent methods rely on k-mers, i.e., substrings of length k. Variants defined as the presence of k-mers within a biological sequence capture a broader category of genetic variation.

However, k-mers may not be expressive enough to represent polymorphic regions, whose presence is then diluted across several k-mers representing all possible versions of the region. A more flexible alternative is to use probabilistic sequence motifs. These motifs summarize several close k-mers by modelling the presence of a mixture of nucleotides at each site.

To quantify the presence of such a motif and a given sequence, we compute its average activation across each sequence in the panel. Our objective is then to find motifs whose activation is significantly associated with a given phenotypic trait. This task is made difficult by the fact that there is an infinite number of possible motifs, since the nucleotide proportions at each site are continuous.

In the present work, we first develop a stable step-wise procedure to select a small number of sequence motifs associated with a trait. We then take advantage of recent advances in post-selection inference to produce a well-calibrated testing procedure for the association between the selected motifs and the trait, while accounting for our selection procedure.

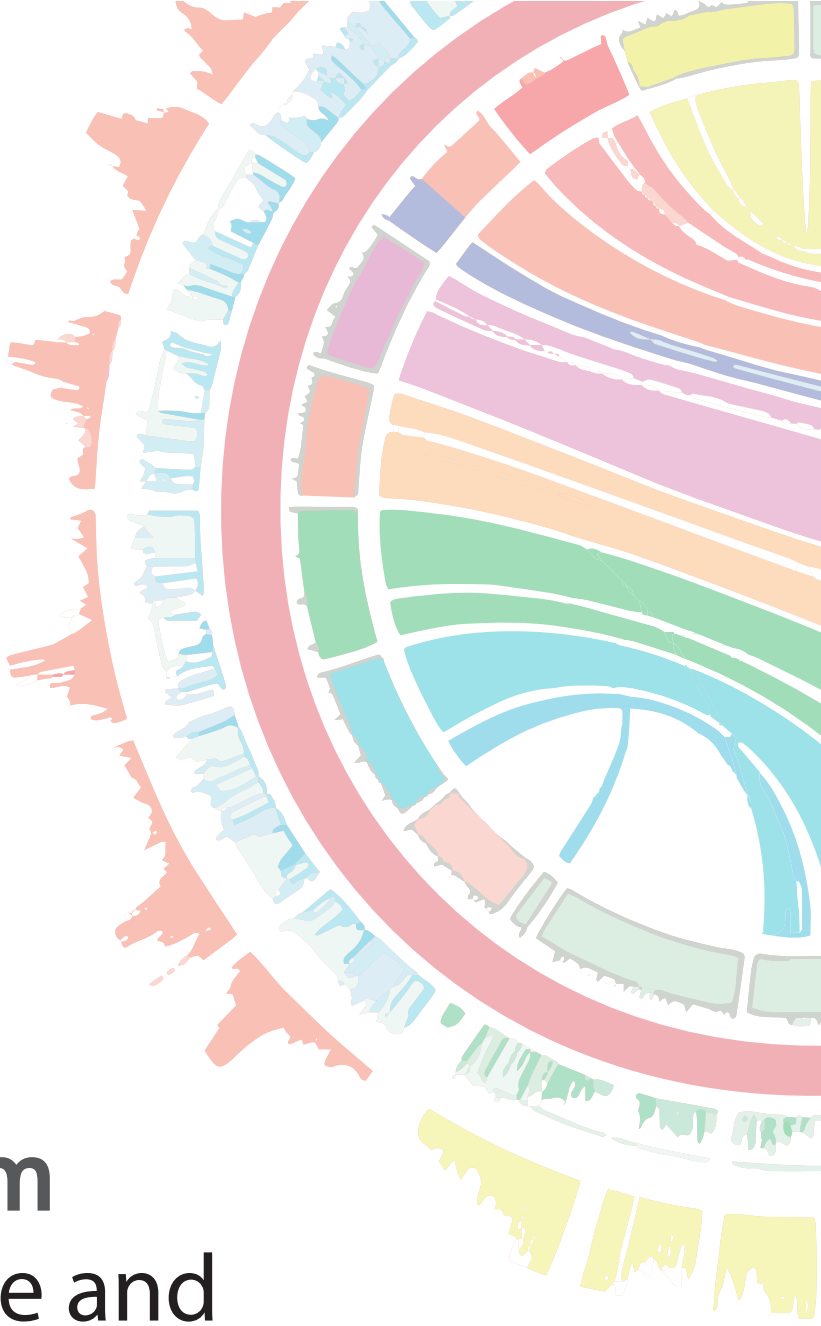
We also draw a formal link between our procedure and convolutional neural networks (CNNs) for biological sequences, which are shown to define a particular association score. Our procedure could therefore be used to perform statistical inference on the filters of a one-layer CNN.

Valid Differential Analysis for Groups Defined via Clustering

Lucy Gao[1], Jacob Bien[2] and Daniela Witten[2]
[1] University of Waterloo, Waterloo, Ontario, Canada
[2] University of Southern California, Los Angeles, California, United States of America
[3] University of Washington, Seattle, Washington, United States of America

Corresponding author: lucy.gao@uwaterloo.ca

Testing for a difference in means between two groups is fundamental to answering research questions across virtually every scientific area. Classical tests like the t-test control the Type I error rate when the groups are defined a priori. However, when the groups are instead defined via a clustering algorithm, then applying a classical test for a difference in means between the groups yields an extremely inflated Type I error rate. This has serious implications for analyses in single cell data science, where it is common practice to define putative cell types via a clustering algorithm, then use a classical test for differential expression analysis between the clusters. Notably, this problem persists even if two separate and independent data sets are used to define the groups and to test for a difference in their means. In this talk, we propose a selective inference approach to test for a difference in means between two clusters obtained from any clustering method. Our procedure controls the selective Type I error rate by accounting for the fact that the null hypothesis was generated from the data. We describe how to efficiently compute exact p-values for clusters obtained using agglomerative hierarchical clustering with many commonly used linkages. We apply our method to simulated data and to single-cell RNA-seq data.



Symposium
Open Science and Interoperability in Bioinformatics

The DisGeNET knowledge platform: facilitating the access and use of disease genomics data*

Laura I FURLONG

Research Programme on Biomedical Informatics (GRIB), Hospital del Mar Medical Research Institute (IMIM), Department of Experimental and Health Sciences, Pompeu Fabra University (UPF), Barcelona, Spain. MedBioinformatics Solutions SL, Barcelona, Spain

Laura.furlong@upf.edu

One of the most pressing challenges in genomic medicine is to understand the impact of genomic variation in health and disease. Large-scale interrogation of the human genome uncovered hundreds of thousands of disease-associated loci. However, the identification of variants of clinical relevance remains challenging, which hinders the exploitation of this information in clinical practice and drug R&D. Bioinformatic tools and resources that enable the automation of every possible step in this process are crucial.

DisGeNET is a knowledge platform that aggregates and standardizes data about disease associated genes and variants from multiple authoritative sources, complemented with the most recent findings extracted from the scientific literature by text mining. Due to its ample coverage of the disease spectrum, it can be applied to complex as well as rare diseases. The current release includes more than 30,000 diseases & traits, 21,000 genes and 195,000 variants. These data are enriched with information from other resources and with different scores and metrics to enable searching, filtering and prioritizing the data.

The DisGeNET suite of tools facilitates data exploration and analysis by different types of users and supports the development of bioinformatic workflows and pipelines enabling automation and reproducibility of the analyses. DisGeNET is an ELIXIR Recommended Interoperability Resource supporting a variety of applications in genomic medicine and drug R&D, including rare disease diagnosis, interpretation of GWAs results and prioritization of drug targets.

Acknowledgements

Funding: IMI-JU (116030: TransQST, 777365: eTRANSAFE); AGAUR (2017SGR00519), GenCat-FEDER (RIS3CAT-VEIS)

* *Invited talk for the mini-symposium Open Science and Interoperability in Bioinformatics*

The OpenLink project: a transversal gateway and dashboard for research data management tools. Demo and Roadmap

Laurent Bouri[1] and Julien Seiler[1]

[1] IGBMC, 1 Rue Laurent Fries, 67400, Illkirch-Graffenstaden, France

Corresponding Author: julien.seiler@igbmc.fr

1. Introduction

In the Life Sciences landscape, bioinformatics core facilities play a key role for many scientific communities, by providing software and reference data in a computational environment tailored for high-throughput computing. They have to handle huge amounts of data generated by scientists in the -omics era, which require an ever-increasing storage and computation capacity.

Bioinformatics platforms also play a pivotal role in the life cycle of scientific data. They are the places where raw data are analyzed and integrated before being made available to the community by deposition in international databases.

In order to help scientists to adopt best practices in data management[1], OpenLink has been selected during the "ANR Flash Données Ouvertes". This project, starting in early 2020 at IGBMC, relies on its IT department, imaging center as well as three research teams, to create a web application that will enable the establishment of a virtual link between data and metadata scattered over multiple management tools and to bring together good practices.

2. Openlink, an interoperable network of data management tools

The web application Openlink will facilitate the transversal identification of projects and their associated data, from the Data Management Plan, to the publication, through the LabGuru electronic lab notebook and data processing tool such as OMERO. The aim is to streamline the transfer of data from production to archiving, while automatically enriching data.

Transversal metadata can be managed using API (Application Programming Interface). API allows users to submit several query parameters to a server in order to fetch or send data. So, information retrieved with API provided by research tools can be used to support researchers in the process of publishing their data.

3. Conclusion

The aim of OpenLink projects is to set up dashboards and automatic procedures to support researchers in data management and guide them towards the adoption of a FAIR² approach compatible with the commitments made by the Ministry of Research in favour of open science.

References

1. Simms, S., Jones, S., Mietchen, D., & Miksa, T. (2017). Machine-actionable data management plans (maDMPs). Research Ideas and Outcomes, 3, e13086. https://doi.org/10.3897/rio.3.e1308
2. Directorate-General for Research and Innovation (2018). Turning FAIR into reality. EU publications. https://doi.org/10.2777/1524

Reproducibility in bioinformatics: take care of your code with Software Heritage

Pierre Poulain[1], Morane Gruenpeter[2] and Roberto Di Cosmo[3]
[1] Université de Paris, CNRS, Institut Jacques Monod, F-75006, Paris, France
[2] Software Heritage, Inria, France
[3] Software Heritage, Inria and University of Paris, France

Corresponding author: pierre.poulain@u-paris.fr

Reproducibility is an ongoing effort in the bioinformatics community[1]. Open science helps toward this goal with open access to the scientific literature, open data and open source research software. In 2018, more than 36% of yearly published papers were published under open access conditions[2]. In biology and bioinformatics, the recent development of preprints has acted as a leverage towards open access.

Raw data deposit in public international repositories of genomics and proteomics data is now well established and enforced by most journal editorial policies. Availability of all-purpose data repositories such as Zenodo or Figshare also fostered open data.

It is now important to establish good practices also for scientific software, going beyond the common approach of depositing code in development platforms such as GitHub or GitLab, where long-term preservation is not guaranteed.

This presentation aims to present to our community the Software Heritage (SWH) [1] archive[3]: it collects, preserves, and makes available all source codes, from the one that ran on the Apollo 11 Guidance Computer to the source code of the Gromacs molecular dynamics engine, the Bowtie 2 genomics read aligner, the Cytoscape network visualization software... Software Heritage can also archives smaller programs like the scripts commonly used in bioinformatics.

Software Heritage regularly collects source code from a growing list of code hosting platforms, and provides a powerful "Save code now" functionality [2] that allows to trigger archival for any public repository based on the Git, Mercurial or Subversion version control systems, free of charge. Any object archived in Software Heritage is assigned an intrinsic persistent identifier [3] called the SWHID[4], that can be independently verified.

We will present actionable recommendations for better referencing and indexing research source code, including best practices for providing metadata files in the code repository (AUTHOR(s) file with the list of authors, LICENSE file with the applicable license to the source code, README file with the description of the software and other valuable information) and for making it citable [4], including pointers to appropriate bibliographic styles[4].

References

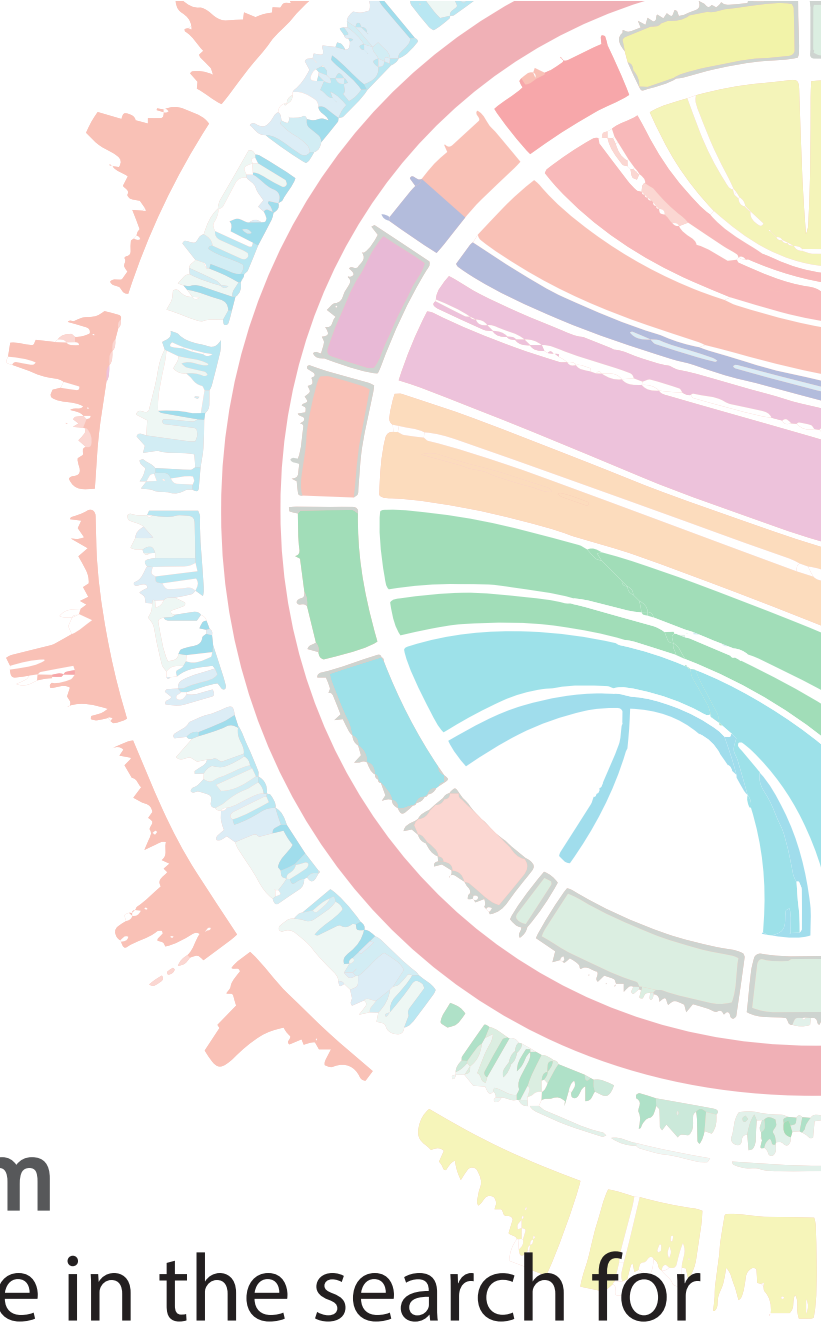
[1] Y.-M. Kim, J.-B. Poline, and G. Dumas. "Experimenting with Reproducibility: A Case Study of Robustness in Bioinformatics". en. In: *GigaScience* 7.7 (2018).

[2] European Commission. *Trends for open access to publications*. Website. Available from https://ec.europa.eu/info/research-and-innovation/strategy/goals-research-and-innovation-policy/open-science/open-science-monitor/trends-open-access-publications_en. 2018.

[3] R. Di Cosmo and S. Zacchiroli. "Software Heritage: Why and How to Preserve Software Source Code". In: *Proceedings of the 14th International Conference on Digital Preservation, iPRES 2017, Kyoto, Japan*. Available from https://hal.archives-ouvertes.fr/hal-01590958. 2017.

[4] R. D. Cosmo. "Archiving and Referencing Source Code with Software Heritage". In: *ICMS*. Volume 12097. Lecture Notes in Computer Science. Springer, 2020, pages 362–373.

1. https://www.softwareheritage.org/
2. https://archive.softwareheritage.org/save/
3. https://www.softwareheritage.org/2020/07/09/intrinsic-vs-extrinsic-identifiers/
4. https://codemeta.github.io/



Symposium

Where are we in the search for DNA motifs involved in transcriptional regulation?

Facteurs de transcription floraux: de la liaison à la régulation

François PARCY

Physiologie Cellulaire et Végétale, CNRS, CEA, UGA, INRAE, 17 av. des martyrs, 38054, GRENOBLE, France
Francois.parcy@cea.fr

Notre équipe étudie une cascade de facteurs de transcription importants au cours du développement floral (ARF, LFY, MADS) chez Arabidopsis thaliana. Nous combinons des données génomiques, biochimiques et structurales pour comprendre la régulation des cibles de ces facteurs. Les données génomiques visent à caractériser la régulation des gènes, la liaison TF/ADN et les paysages chromatiniens et proviennent de techniques de type ChIP-seq, DNase-seq, DAP-seq (et même seq-DAP-seq). Les liaisons TF/ADN sont modélisées sous forme de PWM, TFFM et k-mer (1). Nous avons ainsi montré que 1) les facteurs ARF lient l'ADN sous forme de différentes configurations dont certaines seulement favorisent la régulation (2). 2) le domaine de tétramérisation présent chez les facteurs MADS favorise la liaison de tétramères sur des configurations particulières de sites de liaison (3, 4) 3) les différences entre la liaison in vivo et in vitro du facteur LFY mettent en évidence son rôle pionnier avec une faible sensibilité à la méthylation (5).

L'ensemble de ces travaux illustrent comment l'usage de méthodes computationnelles et en particulier l'usage de différents modèles de liaison à l'ADN permettent de mieux comprendre le rôle clés des facteurs transcription floraux dans le contrôle des programmes d'expression géniques spécifiques à cette phase développementale.Pages must NOT be numbered. Final pagination will be set by the editors of the proceedings.

The list of references is headed *References*, it should be placed at the end of your contribution. It should be in *Times New Roman* 10-point font. Please do not insert a page break before the list of references. For citations in the text, please use square brackets [1] and consecutive ordered numbers [2,3] in list of references. Please find below examples on how to format references corresponding to articles [1], books [2], book chapters and proceedings [3].

References

1. 1. X. Lai, A. Stigliani, G. Vachon, C. Carles, C. Smaczniak, C. Zubieta, K. Kaufmann, F. Parcy, Building Transcription Factor Binding Site Models to Understand Gene Regulation in Plants. Mol Plant. 12, 743–763 (2019).
2. 2. A. Stigliani, R. Martin-Arevalillo, J. Lucas, A. Bessy, T. Vinos-Poyo, V. Mironova, T. Vernoux, R. Dumas, F. Parcy, Capturing Auxin Response Factors Syntax Using DNA Binding Models. Mol. Plant. 12, 822–832 (2019). 3. V. Hugouvieux, C. S. Silva, A. Jourdain, A. Stigliani, Q. Charras, V. Conn, S. J. Conn, C. C. Carles, F. Parcy, C. Zubieta, Tetramerization of MADS family transcription factors SEPALLATA3 and AGAMOUS is required for floral meristem determinacy in Arabidopsis. Nucleic Acids Res. 46, 4966–4977 (2018).
3. 4. X. Lai, A. Stigliani, J. Lucas, V. Hugouvieux, F. Parcy, C. Zubieta, Genome-wide binding of SEPALLATA3 and AGAMOUS complexes determined by sequential DNA-affinity purification sequencing. Nucleic Acids Res. 48, 9637–9648 (2020).
4. 5. X. Lai, R. Blanc-Mathieu, L. GrandVuillemin, Y. Huang, A. Stigliani, J. Lucas, E. Thévenon, J. Loue-Manifel, L. Turchi, H. Daher, E. Brun-Hernandez, G. Vachon, D. Latrasse, M. Benhamed, R. Dumas, C. Zubieta, F. Parcy, The LEAFY floral regulator displays pioneer transcription factor properties. Mol Plant. 14, 829–837 (2021).
5. We receive support from the GRAL LabEX (ANR-10-LABX-49-01) with the frame of the CBH-EUR-GS (ANR-17-EURE-0003),

Motif discovery in epigenomics datasets with RSAT peak-motifs2

Morgane THOMAS-CHOLLIER

Institute of Biology of ENS (IBENS), Department of biology, École normale supérieure, CNRS, INSERM, Université PSL, 75005 Paris, France
mthomas@bio.ens.psl.eu

Algorithms for *ab initio* or *de novo* motif discovery have been developed for 25 years. These algorithms were initially meant to study a handful of non-coding sequences upstream of genes, and predict binding of transcription factors (TF). These algorithms solely take as input a set of sequences, and detect exceptional motifs that are then considered as putative regulatory signals. These algorithms quickly became accessible to non-experts, thanks to the web interfaces developed by MEME [1] and Regulatory Sequences Analysis Tools (RSAT) [2,3].

Initiated in 1998, RSAT (**http: //www.rsat.eu/)** is a complete suite to detect *cis*-regulatory elements in genomic sequences. RSAT functionalities include *de novo* motif discovery, analyses of motif quality, motif comparisons and clustering, motif scanning to predict transcription factor binding sites (TFBSs), detection and analysis of regulatory variants [4]. Over the last 20 years, the RSAT team has maintained uninterrupted service, while extending developments prompted by the advances in the field of regulatory genomics.

A turning point 10 years ago was the quick adoption of ChIP-seq to study TF binding genome-wide. An important bottleneck for most existing tools was that the underlying algorithms were originally developed for a small set of input sequences, and could hardly treat the thousands of peaks produced by ChIP-seq experiments. We thus developed RSAT *peak-motifs* [5], motivated by the pressing need for a statistically reliable, time-efficient and user-friendly framework to analyze full datasets of ChIP-seq peaks. It has become the flagship tool of RSAT, along with its companion *matrix-clustering* tool [6]. This more recent tool enables to identifying clusters of similar motifs, and is very useful to regroup redundant motifs.

Currently, many development of new tools for motif discovery are targeted towards using machine learning approaches. Although these methods yield impressive results, they are not adressing non-experts users who need to autonomously analyze their own datasets. In addition to expertise, these approaches have hardware requirements that are not available in many institutes. There is thus still a need among users for reliable motif discovery tools, accessible through a web interface.

We are thus developing a new major version of *peak-motifs*. Novelties include input a BED file and automatically retrieve the corresponding FASTA sequences to analyze, removing low-complexity regions, an enhanced web form, and fully-revised report that takes advantage of recent web frameworks, to display all results as a dashboard [cf. poster n°76]. Current developments involve direct link to *matrix-clustering*, and a new way to rank motifs, with a metric based on central enrichment of each discovered motif within the peaks. Altogether, the six public RSAT servers jointly support >10 000 genomes from all kingdoms. The open-source code has been moved to GitHub in 2021 (https://github.com/rsa-tools). RSAT is well-documented and available through Web sites, SOAP/WSDL + REST web services, virtual machines and stand-alone programs.

Acknowledgements

Work supported by the *ITMO Cancer CID ModICeD* and the Institut Universitaire de France

References

1. Timothy L. Bailey and Charles Elkan, Fitting a mixture model by expectation maximization to discover motifs in biopolymers, *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pp. 28-36, AAAI Press, Menlo Park, California, 1994.
2. van Helden,J., André,B. and Collado-Vides,J. Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *J. Mol. Biol.*, **281**, 827–842, 1998.
3. van Helden,J., André,B. and Collado-Vides,J. A web site for the computational analysis of yeast regulatory sequences. *Yeast*, **16**, 177–187, 2000.
4. Nguyen NTT, Contreras-Moreira B, Castro-Mondragon JA, Santana-Garcia W, Ossio R, Robles-Espinoza CD, Bahin M, Collombet S, Vincens P, Thieffry D, van Helden J, Medina-Rivera A, Thomas-Chollier M. RSAT 2018: regulatory sequence analysis tools 20th anniversary. Nucleic Acids Res. 46(W1):W209-W214, 2018.
5. Thomas-Chollier M., Darbo E., Herrmann C., Defrance M., Thieffry D., van Helden J.. A complete workflow for the analysis of full-size ChIP-seq (and similar) data sets using peak-motifs. Nat. Protoc. 7:1551–1568, 2012.
6. Castro-Mondragon J.A., Jaeger S., Thieffry D., Thomas-Chollier M., van Helden J.. RSAT matrix-clustering: dynamic exploration and redundancy reduction of transcription factor binding motif collections. Nucleic Acids Res. 45:e119, 2017.

Combining TF binding profiles with ChIP-seq data to predict high quality direct TF-DNA interactions

ANTHONY MATHELIER

Centre for Molecular Medicine Norway (NCMM), EMBL partnership, University of Oslo, Norway
Department of Medicical Genetics, Oslo University Hospital, Norway
anthony.mathelier@ncmm.uio.no
@AMathelier

Abstract

Transcription Factors (TFs) are key proteins regulating when and where genes are expressed through their interaction with the DNA at specific binding sites. Hence, it is critical to locate these TF-DNA interactions to understand transcriptional regulation. Chromatin immunoprecipitation followed by sequencing (ChIP-seq) represents the most popular experimental assay to identify the genomic regions, so called ChIP-seq peaks, where TFs bind to DNA in vivo. Unfortunately, it has recurrently been shown that ChIP-seq experiments are prone to generate ChIP-seq artifacts, and delineating bona fide bound regions from experimental noise is critical to provide useful biological insights for this data. The ever-increasing number of publicly available ChIP-seq data sets provides an unprecedented opportunity to develop computational tools designed to infer the precise locations of the TFBSs within ChIP-seq peaks by combining both computational and experimental evidences of direct TF-DNA interactions. Recently, we have updated the JASPAR[1] and UniBind[2] resources to provide the community with high-quality models and maps of direct TF-DNA interactions across species. They represent fundamental resources for researchers analysing transcriptional regulation.

JASPAR (http://jaspar.genereg.net) is an open-access database of manually curated, non-redundant TF-binding profiles essentially stored as position frequency matrices (PFMs) for TFs across multiple species in six taxonomic groups. The JASPAR database is amongst the most popular and longest maintained database for TF-binding profiles, and is a standard resource in the field. As of today, the CORE collection of the 2020 release of JASPAR contains 1,646 high-quality non-redundant PFMs.

Taking advantage of the JASPAR PFMs, we recently processed ~10,000 public ChIP-seq datasets from nine species to provide high-quality TFBS predictions. The data was uniformly processed through our ChIP-eat software to specifically delineate direct TF-DNA interactions in ChIP-seq peaks and separate them from indirect or non-specific binding and ChIP-seq artifacts. Briefly, ChIP-eat combines both computational (high PWM score) and experimental (centrality to ChIP-seq peak summit) evidence to find high-confidence direct TF-DNA interactions in a ChIP-seq experiment-specific manner. After quality control, it culminated with the prediction of ~56 million TFBSs with experimental and computational evidence for direct TF-DNA interactions for 644 TFs in >1,000 cell lines and tissues. These TFBSs were used to predict >198,000 cis-regulatory modules representing clusters of binding events in the corresponding genomes. The high-quality of the TFBSs was reinforced by their evolutionary conservation, enrichment at active cis-regulatory regions, and capacity to predict combinatorial binding of TFs. Further, we confirmed that the cell type and tissue specificity of enhancer activity was correlated with the number of TFs with binding sites predicted in these regions. All the data is provided to the community through the UniBind database that can be accessed through its web-interface (https://unibind.uio.no/), a dedicated RESTful API, and as genomic tracks. Finally, we provide an enrichment tool, available as a web-service and an R package, for users to find TFs with enriched TFBSs in a set of provided genomic regions. UniBind is the first resource of its kind, providing the largest collection of high-confidence direct TF-DNA interactions in nine species.

References

1. Fornes O, Castro-Mondragon JA, Khan A, *et al.* JASPAR 2020: update of the open-access database of transcription factor binding profiles. *Nucleic Acids Res.* 2019;48: D87–D92.
2. Riudavets Puig R. *et al.* UniBind: maps of high-confidence direct TF-DNA interactions across nine species. *BMC Genomics* 2021; in press. (preprint on *bioRxiv*)

Identification of long regulatory elements in the genome of Plasmodium falciparum and other eukaryotes

Christophe MENICHELLI, Vincent GUITARD, Rafael M. MARTINS, Sophie LEBRE, Jose-Juan LOPEZ-RUBIO, Charles-Henri LECELLIER, Laurent BREHELIN

LIRMM, Univ Montpellier, CNRS, Montpellier, France
brehelin@lirmm.fr

Long regulatory elements (LREs), such as CpG islands, polydA:dT tracts or AU-rich ele- ments, are thought to play key roles in gene regulation but, as opposed to conventional binding sites of transcription factors, few methods have been proposed to formally and auto- matically characterize them. We present here a computational approach named DExTER (Domain Exploration To Explain gene Regulation) dedicated to the identification of candi- date LREs (cLREs) and apply it to the analysis of the genomes of P. falciparum and other eukaryotes. Our analyses show that all tested genomes contain several cLREs that are somewhat conserved along evolution, and that gene expression can be predicted with sur- prising accuracy on the basis of these long regions only. Regulation by cLREs exhibits very different behaviours depending on species and conditions. In P. falciparum and other Api- complexan organisms as well as in Dictyostelium discoideum, the process appears highly dynamic, with different cLREs involved at different phases of the life cycle. For multicellular organisms, the same cLREs are involved in all tissues, but a dynamic behavior is observed along embryonic development stages. In P. falciparum, whose genome is known to be strongly depleted of transcription factors, cLREs are predictive of expression with an accu- racy above 70%, and our analyses show that they are associated with both transcriptional and post-transcriptional regulation signals. Moreover, we assessed the biological relevance of one LRE discovered by DExTER in P. falciparum using an in vivo reporter assay.

PLMdetect : *de novo* mapping of functional *cis*-regulatory motifs in 5'- and 3'-proximal regions from Arabidopsis and maize

Julien Roziere[1,2,3]

[1] Université Paris-Saclay, CNRS, INRAE, Univ Evry, Institute of Plant Sciences ParisSaclay (IPS2), 91405, Orsay, France

[2] Université de Paris, CNRS, INRAE, Institute of Plant Sciences Paris Saclay (IPS2), 91405, Orsay, France

[3] Institut Jean-Pierre Bourgin, INRAE, AgroParisTech, Université Paris-Saclay, 78000, Versailles, France

julien.roziere@inrae.fr

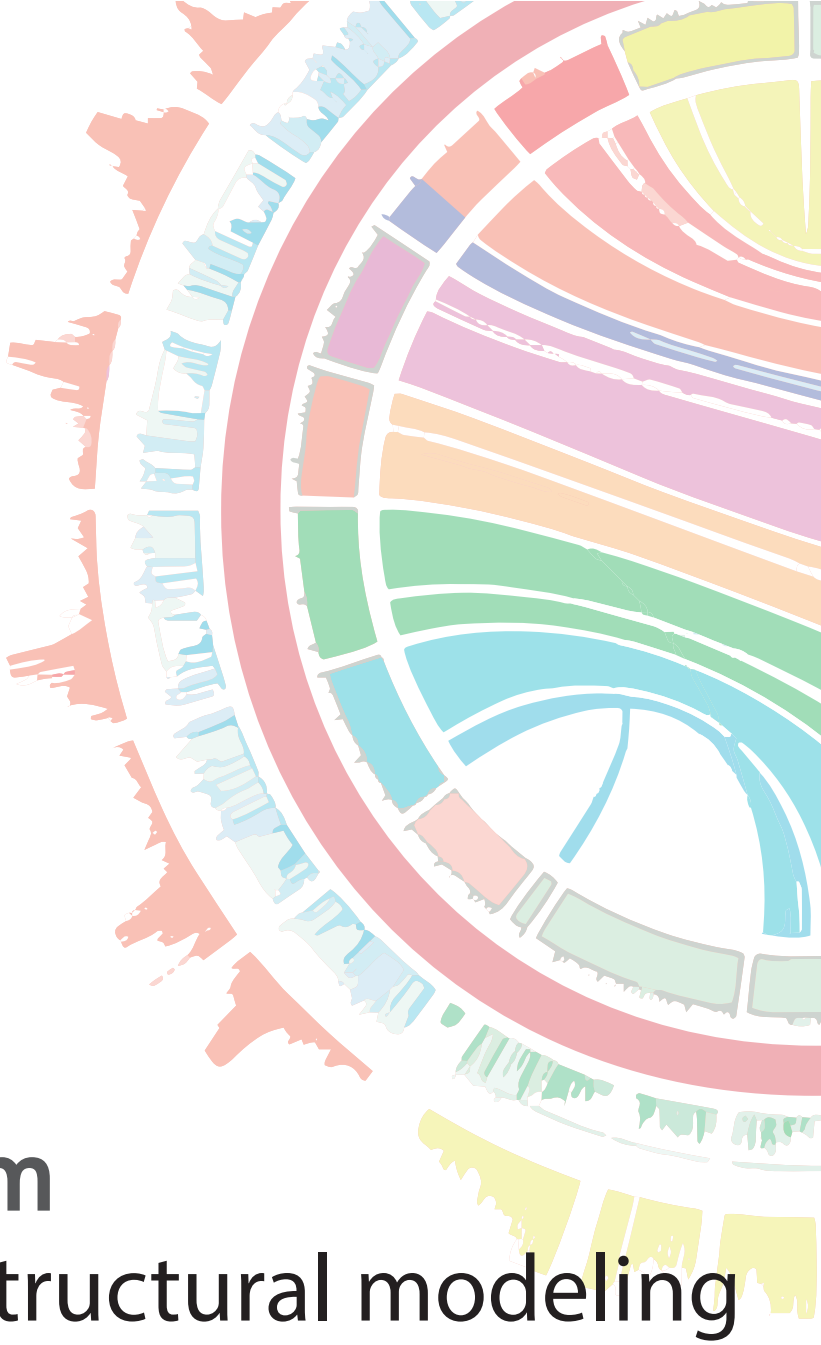
Identifying cis-regulatory motifs controlling gene expression is an arduous challenge that is actively explored to discover key genetic factors responsible for traits of agronomic interest. The Preferentially Located Motif detection (PLMdetect) method was developed to identify over-represented motifs (PLMs) in promoters at a preferred distance from the transcription start site in the model plant Arabidopsis [1]. Here, we expanded the PLMdetect method to comprehensively analyze de novo the promoters as well as the untranslated transcribed regions of Arabidopsis and the important crop maize. We sought to determine how their differences in genome content and architecture would be reflected in features of their PLMs in 5'- and 3'-proximal regions of each gene locus. We have currently identified three groups of PLMs for each species in each targeted region. An assessment of these PLMs using known plant transcription factor (TF) binding site (TFBS) data [2] revealed that a subset of these PLMs (9.4% and 7.3% in Arabidopsis and maize, respectively) are previously characterized TFBSs (tPLMs), while the others represent novel and uncharacterized motifs (uPLMs), not captured by the current collection of plant TFBSs. Positional analyses of the tPLMs revealed positional preferences of TFBSs from several TF families as previously reported in Arabidopsis [3]. Furthermore, GO term enrichment analyses showed that 15.3% of the uPLMs are able to infer functional predictions which are not provided by tPLMs. In the near future, we will add comparisons between the datasets obtained from each species. Additionally, the development of the interactive PLMviewer website will provide the plant community with a valuable resource of PLM datasets for exploitation to investigate user-specific sequences.

Acknowledgements

This research is supported by a grant from the Plant2Pro Carnot Institute. IJPB and IPS2 benefit from the support of Saclay Plant Science-SPS (ANR-17-EUR-0007).

References

1. Bernard V, Lecharny A, Brunaud V. Improved detection of motifs with preferential location in promoters. Genome. 2010 Sep;53(9):739-52. doi: 10.1139/g10-042. PMID: 20924423.
2. Fornes O, Castro-Mondragon JA, Khan A, van der Lee R, Zhang X, Richmond PA, Modi BP, Correard S, Gheorghe M, Baranašić D, Santana-Garcia W, Tan G, Chèneby J, Ballester B, Parcy F, Sandelin A, Lenhard B, Wasserman WW, Mathelier A. JASPAR 2020: update of the open-access database of transcription factor binding profiles. Nucleic Acids Res. 2020 Jan 8;48(D1):D87-D92. doi: 10.1093/nar/gkz1001. PMID: 31701148; PMCID: PMC7145627.
3. Yu CP, Lin JJ, Li WH. Positional distribution of transcription factor binding sites in Arabidopsis thaliana. Sci Rep. 2016 Apr 27;6:25164. doi: 10.1038/srep25164. PMID: 27117388; PMCID: PMC4846880.



Symposium

Integrative structural modeling in the era of big data and artificial intelligence

Integrative Modelling of Macromolecular complexes

Riccardo PELLARIN[1]

[1] Institut Pasteur, Structural Bioinformatics Unit, Department of Structural Biology and Chemistry, CNRS UMR 3528, C3BI USR 3756, Paris, France.

Corresponding Author: Riccardo.pellarin@pasteur.fr

Integrative Modelling of Macromolecular complexes

Riccardo Pellarin

Institut Pasteur, Structural Bioinformatics Unit, Department of Structural Biology and Chemistry, CNRS UMR 3528, C3BI USR 3756, Paris, France.

riccardo.pellarin@pasteur.fr

Due to the limitation of individual techniques such as X-ray crystallography or cryo Electron Microscopy (cryoEM), studies of large macromolecular assemblies have often been tackled by using integrative structural biology approaches. The integrative structural determination uses as much of the relevant biochemical and biophysical data about a macromolecular complex as possible to generate three-dimensional structures, and exploits the mutual synergy and consistency of the datasets in such a way that the resulting model is more informative than the models generated by each individual dataset [1]. Integration of Chemical cross-linking combined with Mass-Spectrometry (XL-MS) and Electron Microscopy (EM) is a powerful strategy to determine the architecture of macromolecular complexes, especially when combined with X-ray crystallography of protein domains or homology modeling. The two methods provide orthogonal structural information. XL-MS probes the proximity of residues, peptides or domains in macromolecular complexes [3,4]. EM instead allows visualization of entire particles such as cellular components and macromolecular complexes in a form of 2D images or 3D density maps [2]. We recently developed a Bayesian approach to model the structure of a macromolecular system by optimally combining cryo-EM data with and XL-MS. We use Bayesian inference to determine the optimal weight of cryoEM data in integrative structural modeling. The approach models the structure of the system while simultaneously and automatically quantifying the level of noise in the data. By accounting for both data noise and correlation, this approach enables an effective use of cryoEM density maps in integrative structural modeling.

References

1. Kim, S. J. *et al.* Integrative structure and functional anatomy of a nuclear pore complex. *Nature* **555**, 475 (2018).
2. Bonomi M, Hanot S, Greenberg CH, Sali A, Nilges M, Vendruscolo M, Pellarin R. Bayesian weighing of electron cryo-microscopy data for integrative structural modeling. *Structure* (2018).
3. Robinson, P. J. *et al.* Structure of a Complete Mediator-RNA Polymerase II Pre-Initiation Complex. *Cell* (2016). doi:10.1016/j.cell.2016.08.050
4. Shi, Y. *et al.* A strategy for dissecting the architectures of native macromolecular assemblies. *Nat. Methods* (2015). doi:10.1038/nmeth.3617

Dynamic Integrative Modelling of Molecular Machines

Ezgi KARACA[1, 2]

[1] Izmir Biomedicine and Genome Center, Balcova, 35340, Izmir, Turkey
[2] Dokuz Eylul University, Health Campus, 35330, Izmir, Turkey

Corresponding Author: ezgi.karaca@ibg.edu.tr

Adding a structural dimension to the ever-accumulating omics data presents a grand challenge to the structural biology community. Integrative modelling predicts protein assemblies under the guidance of experimental data to alleviate this challenge. During the last decade, we have observed a substantial improvement in the field of integrative modelling. Though, the field is still being challenged by numerous large, heterogeneous, and dynamic machineries. During my talk, I will present our recent efforts in modelling the structure of such an assembly. In this work, upon joining forces of molecular dynamics and integrative modelling, we explore how a specific class of transcription factor, Sox, can recognize its cognate sequence on a compact nucleosome. Our approach proposes that the productive binding of Sox transcription factor depends on the localization of its cognate sequence on the nucleosome. It also reveals that the position-dependency emanates from the differential histone-DNA interactions encoded at distinct nucleosomal positions. These striking findings came as an outcome of multiple simulation cycles, which I will discuss in detail during my presentation.

Entering the post-protein structure prediction era

Sergei GRUDININ
Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, LJK, Grenoble, 38000, FRANCE
sergei.grudinin@univ-grenoble-alpes.fr

The potential of deep learning has been recognized in structural bioinformatics for already some time, and became indisputable after the CASP13 (Critical Assessment of Structure Prediction) community-wide experiment in 2018. In CASP14, held in 2020, deep learning has boosted the field to unexpected levels reaching near-experimental accuracy. Its results demonstrate dramatic improvement in computing the three-dimensional structure of proteins from amino acid sequence, with many models rivalling experimental structures. This success comes from advances transferred from several machine-learning areas, including computer vision and natural language processing. At the same time, the community has developed methods specifically designed to deal with protein sequences and structures, and their representations. Novel emerging approaches include (i) geometric learning, i.e. learning on non-regular representations such as graphs, 3D Voronoi tessellations, and point clouds; (ii) pre-trained protein language models leveraging attention; (iii) equivariant architectures preserving the symmetry of 3D space; (iv) use of big data, e.g. large meta-genome databases; (v) combining protein representations; (vi) and finally truly end-to-end architectures, i.e. single differentiable models starting from a sequence and returning a 3D structure. These observations suggest that deep learning approaches will also be effective for a range of related structural biology applications that I will discuss in this talk.Pages must NOT be numbered. Final pagination will be set by the editors of the proceedings.

The list of references is headed *References*, it should be placed at the end of your contribution. It should be in

Cryo-EM studies of continuous conformational variability of biomolecules *in vitro* and *in situ* based on image analysis, simulation and deep learning

Mohamad HARASTANI, Ilyes HAMITOUCHE and Slavica JONIC

IMPMC - UMR 7590 CNRS, MNHN, Sorbonne Université, 4 place Jussieu, 75005, Paris, France

Corresponding Author: Slavica.Jonic@upmc.fr

1. Introduction

Biomolecular complexes adopt continuous conformational changes to accomplish various biological functions. The determination of the full conformational landscape from single-particle cryo electron microscopy (cryo-EM) images of purified complexes (*in vitro*) is challenging but can provide insights into their working mechanisms. The cryo electron microscope can also be used for structural determination of macromolecules in cells (*in situ*), by cryo-electron tomography (cryo-ET). The potential of cryo-ET to provide macromolecular dynamics information is still largely unexploited. Conventional subtomogram analysis methods propose discrete rather than continuous solutions, via subtomogram classification and class averaging. We present two methods, for *in vitro* cryo-EM (HEMNMA) and *in situ* cryo-ET (HEMNMA-3D) analysis of continuous conformational variability, both based on normal mode analysis (NMA). Also, we show that such hybrid approaches can be combined with deep learning to speed up the data analysis.

2. Methods

HEMNMA is a single-particle cryo-EM method for analyzing continuous conformational variability of macromolecules, introduced in 2014 [1]. Its software is part of the open-source ContinuousFlex plugin of Scipion 2 and 3 [2]. HEMNMA combines image analysis, NMA, and dimension reduction to visualize the full conformational landscape in a low-dimensional (usually 2D or 3D) space, and allows obtaining 3D reconstructions and movies of molecular motions along desired directions in this space. **HEMNMA-3D** [3] is a method for continuous conformational variability analysis of macromolecules in cryo-ET 3D subtomograms. It combines subtomogram analysis, NMA, and dimension reduction to visualize the full conformational landscape in a low-dimensional space, and allows obtaining subtomogram averages and movies of molecular motions along desired directions in this space. HEMNMA-3D software is part of the ContinuousFlex plugin of Scipion 3. Both methods use a flexible 3D-reference (atomic structure or a density map) to match the conformation, orientation, and position of the molecule in each image (HEMNMA) or subtomogram (HEMNMA-3D), through elastic and rigid-body alignments, where the conformational parameters are the amplitudes of normal modes. Recently, we have combined HEMNMA with deep learning for faster conformational space determination [4]. In this approach, a deep learning network predicts the amplitudes of normal modes from a large set of images, based on the normal-mode amplitudes previously estimated by HEMNMA from a small set of images. This approach will later be extended to HEMNMA-3D.

Acknowledgements

We acknowledge the support of the ANR (ANR-19-CE11-0008-01; ANR-20-CE11-0020-03) and the GENCI (A0100710998, A0070710998, AP010712190, AD011012188).

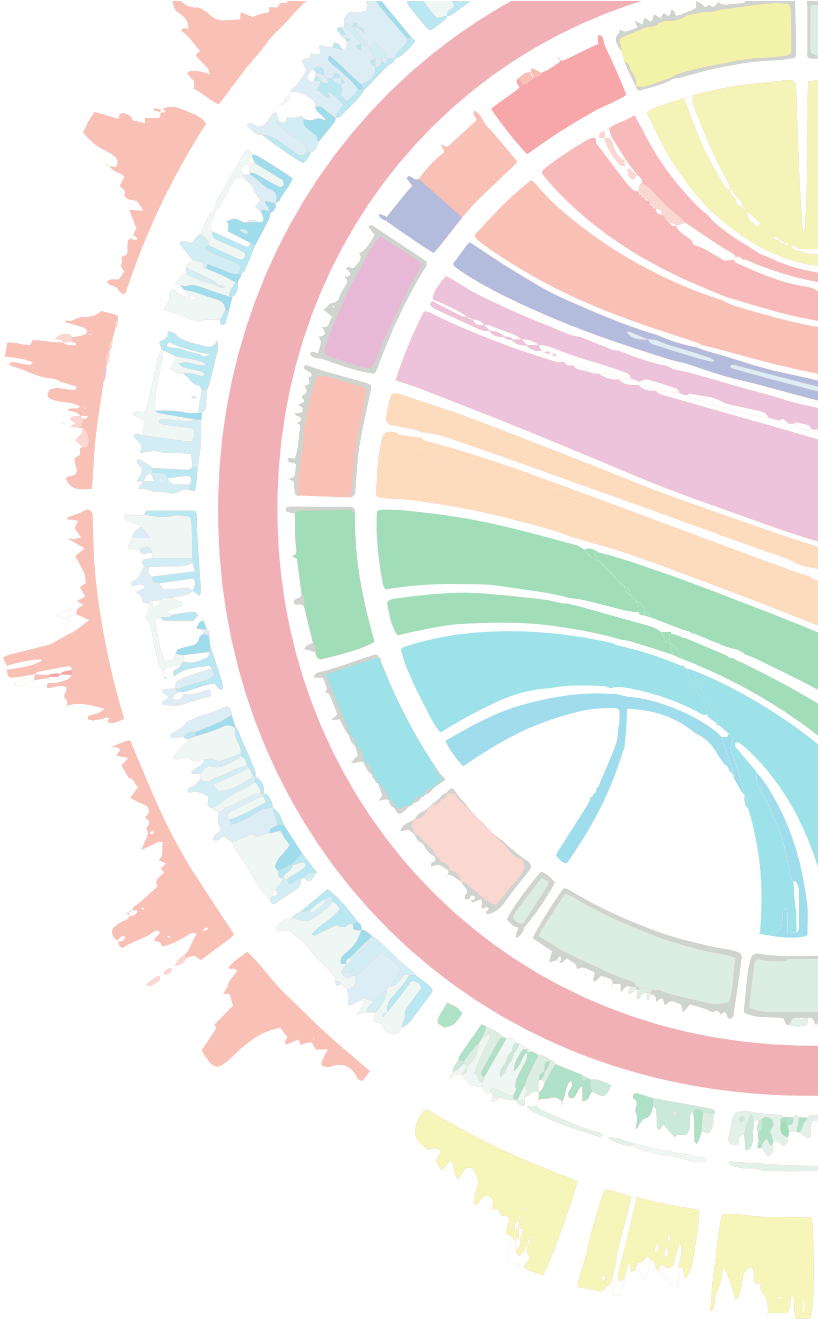
References

[1] Jin Q, Sorzano CO, de la Rosa-Trevin JM, Bilbao-Castro JR, Nunez-Ramirez R, Llorca O, Tama F, and Jonic S. Iterative Elastic 3D-to-2D Alignment Method Using Normal Modes for Studying Structural Dynamics of Large Macromolecular Complexes. Structure 22: 496-506, 2014.

[2] Harastani M, Sorzano COS, and Jonic S. Hybrid Electron Microscopy Normal Mode Analysis with Scipion. Protein Sci 29**:** 223-236, 2020.

[3] Harastani M, Eltsov M, Leforestier A, Jonic S. HEMNMA-3D: Cryo Electron Tomography Method Based on Normal Mode Analysis to Study Continuous Conformational Variability of Macromolecular Complexes. Front. Mol. Biosci, 2021, in press.

[4] Hamitouche I and Jonic S. Deep learning of elastic 3D shapes for cryo electron microscopy analysis of continuous conformational changes of biomolecules. *In Proc 29th European Signal Processing Conference, EUSIPCO 2021*, accepted.



Sponsors

Le Collecteur analyseur de données du Plan France médecine génomique

Yves VANDENBROUCK
Plan France Médecine Génomique 2025, 8 rue de la croix Jarry, 75013, Paris, France
yves.vandenbrouck@cea.fr

Introducing Illumina® Connected Analytics, a flexible platform for analysing, aggregating, and exploring multi-omic data sets.

David TOWNLEY

Illumina Centre,19 Granta Park, Cambridge, CB21 6DF, United Kingdom
dtownley@illumina.com

Introduction

Authors names: *Patel*

The soaring volume of data generated by NGS and other omics technologies presents both opportunities and challenges. Scaling up computing infrastructure to address the increasing number of petabyte-scale omics datasets is costly in both hardware and human resources. The vital maintenance of security, privacy, and compliance is a non-trivial task which requires substantial investment and regular audits, while discrete silos of data inhibit collaboration and easy data sharing, slowing the pace of innovation. Building and deploying new workflows can involve navigating many systems, tools, and technologies, often leading to inefficiencies. As a result, successfully deriving novel insights from these data sets can be prohibitive.
Illumina Connected Analytics (ICA) is designed to eliminate many of these data challenges - empowering users to do more with their data. This talk will focus on the ICA cloud-based solution to managing population scale data sets and workloads without limitations. Data and tools can be accessed using the graphical user interface (GUI), or programmatically via the APIs and command-line interface (CLI). ICA breaks down data silos by fostering global collaboration and sharing of data, tools, and workflows amongst users. ICA takes care of the details so users can maintain a secure, compliant, and private data environment with minimal effort.
Key features of ICA include audit trails/logs, access controls, and multi-factor authentication. Users can streamline development of new pipelines with support for custom, shared, and pre-packaged tools, including Illumina's DRAGEN bioinformatics pipelines. ICA frees up bioinformaticians and data scientists to focus more time and energy on building new tools to explore the data, and less time on routine, cumbersome tasks which can be automated or made accessible for end users.

Acknowledgements

Jay Patel, Illumina 5200 Illumina Way, San Diego, CA 92122 U.S.A.

References

1. Illumina DRAGEN Bio-IT Platform Variant calling and secondary genomic analysis. Illumina website. www.illumina.com/products/by-type/informatics-products/dragen-bio-it-platform.html.Accessed October 22, 2020.
2. Enabling responsible genomic data sharing for the benefit of human health. Global Alliance for Genomics and Health website. www.ga4gh.org. Accessed October 22, 2020.
3. Scikit-learn machine learning in Python.scikit-learn website.Scikit-learn.org/stable/. Accessed January 11, 2021
4. General Data Protection Regulation (GDPR) Compliance Guide- lines. GDPR website.gdpr.eu. Accessed January 11, 2021

Genomics in Genopole : A second wind

Christophe LANNEAU

GIP GENOPOLE, Genopole Campus-1, 20 rue Henri Desbruères, F-91030, EVRY, France
christophe.lanneau@genopole.fr

Genopole, the cradle of genomics in France, is still involving in this discipline but the field has changed. The current challenges in genomics have shifted to computational challenges. To keep pace with this evolution, Genopole is shifting its strengths and missions from genomics to computational genomics, with the ambition to build a new branch at a national level.
Genopole's ambition is based on the biocluster model, which brings together the strengths of the business world, academic research and training on a single site. The Genopole ecosystem covers all aspects from fundamental genomics (genomics, epigenomics, etc.) to medical, industrial (metabolic genomics, synthetic biology, etc.) and environmental (metagenomics, etc.) applications. The French computational genomics community needs to organize and structure itself in order to take advantage of the potential of genomics in all application areas, to stay on the international race and to build its own pipeline in terms of genomic data processing. Genopole is moving step by step to this direction in conjunction with national and European partners.

Index