JOBIM 2020

Montpellier, 30 juin - 3 juillet

Long papers

Paper 1

Long-read error correction: a survey and qualitative comparison

Pierre Morisse[1], Thierry Lecroq[2] and Arnaud Lefebvre[2]
[1] Normandie Université, UNIROUEN, INSA Rouen, LITIS, 76000 Rouen, France
[2] Normandie Univ, UNIROUEN, LITIS, 76000 Rouen, France

Corresponding author: pierre.morisse2@univ-rouen.fr
Extended preprint: https://doi.org/10.1101/2020.03.06.977975

Abstract *Third generation sequencing technologies Pacific Biosciences and Oxford Nanopore Technologies were respectively made available in 2011 and 2014. In contrast with second generation sequencing technologies such as Illumina, these new technologies allow the sequencing of long reads of tens to hundreds of kbps. These so-called long reads are particularly promising, and are especially expected to solve various problems such as contig and haplotype assembly or scaffolding, for instance. However, these reads are also much more error prone than second generation reads, and display error rates reaching 10 to 30%, depending on the sequencing technology and to the version of the chemistry. Moreover, these errors are mainly composed of insertions and deletions, whereas most errors are substitutions in Illumina reads. As a result, long reads require efficient error correction, and a plethora of error correction tools, directly targeted at these reads, were developed in the past nine years. These methods can adopt a hybrid approach, using complementary short reads to perform correction, or a self-correction approach, only making use of the information contained in the long reads sequences. Both these approaches make use of various strategies such as multiple sequence alignment, de Bruijn graphs, hidden Markov models, or even combine different strategies. In this paper, we describe a complete survey of long read error correction, reviewing all the different methodologies and tools existing up to date, for both hybrid and self-correction. Moreover, the long reads characteristics, such as sequencing depth, length, error rate, or even sequencing technology, can have an impact on how well a given tool or strategy performs, and can thus drastically reduce the correction quality. We thus also present an in-depth benchmark of the available long read error correction tools, on a wide variety of datasets, composed of both simulated and real data, with various error rates, coverages, and read lengths, ranging from small bacterial to large mammal genomes.*

Keywords long reads, error correction, hybrid correction, self-correction

1 Introduction

Since their inception in 2011 and 2014, third generation sequencing technologies Pacific Biosciences (PacBio) and Oxford Nanopore Technologies (ONT) became widely used and allowed the sequencing of massive amounts of data. These technologies distinguish themselves from second generation sequencing technologies, such as Illumina, by the fact that they allow to produce much longer reads, reaching lengths of tens of kbps on average, and up to 1 million bps [1]. Thanks to their length, these so-called long reads are expected to solve various problems, such as contig and haplotype assembly of large and complex organisms, scaffolding, or even structural variant calling, for instance. These reads are however extremely noisy, and display error rates of 10 to 30%, while second generation short reads usually reach error rates of around 1%. Moreover, long reads errors are mainly composed of insertions and deletions, whereas short reads mainly contain substitutions. As a result, in addition to a higher error rate, the error profiles of the long reads are also much more complex than the error profiles of the short reads. In addition, ONT reads also suffer from bias in homopolymer regions, and thus tend to contain systematic errors in such regions, when they reach more than 6 bps. As a consequence, error correction is often used as a first step in projects dealing with long reads. Since the error profiles and error rates of the long reads are much different than those of the short reads, this necessity led to new algorithmic developments, specifically targeted at these long reads.

Two major ways of approaching long read correction were thus developed. The first one, hybrid

correction, makes use of additional short reads data to perform the correction. The second one, self-correction, on the contrary, attempts to correct long reads solely based on the information contained in their sequences. Both these approaches rely on various strategies, such as multiple sequence alignment, de Bruijn graphs, or hidden Markov models, for instance. Since 2012, 29 different long read correction tools were thus developed.

1.1 Contribution

In this paper, we propose a description of the state-of-the-art of long read correction. In particular, we draw a summary of every single approach described in the literature, both for hybrid correction and for self-correction. In addition, we also dress a list of all the available methods, and briefly describe the strategy they rely on. We thus propose the most complete survey on long read correction up to date.

Additionally, long reads characteristics, such as the sequencing depth, the length, the error rate, and the sequencing technology, can impact how well a given tool or strategy performs. As a result, a given tool performing the best on a given dataset does not mean that this same tool will perform the best on other datasets, especially if their characteristics fluctuate from one another. As a result, we also present a benchmark of available long read correction tools, on a wide variety of datasets with diverse characteristics. In particular, we assess both simulated and real data, and rely on datasets having varying read lengths, error rates, and sequencing depths, ranging from smaller bacterial to large mammal genomes.

2 State-of-the-art

As mentioned in Section 1, the literature describes two main approaches to tackle long read error correction. On the one hand, hybrid correction makes use of complementary, high quality, short reads to perform correction. On the other hand, self-correction attempts to correct the long reads solely using the information contained in their sequences.

One of the major interests of hybrid correction is that error correction is mainly guided by the short reads data. As a result, the sequencing depth of the long reads has no impact on this strategy whatsoever. As a result, datasets composed of a very low coverage of long reads can still be efficiently corrected using a hybrid approach, as long as the sequencing depth of the short reads remains sufficient, *i.e.* around 50x.

Contrariwise, self-correction is purely based on the information contained in the long reads. As a result, deeper long reads coverages are usually required, and self-correction can thus prove to be inefficient when dealing with datasets displaying low coverages. The required sequencing depth to allow for an efficient self-correction is however reasonable, as it has been shown that from a coverage of 30x, self-correction methods are able to provide satisfying results [2].

We present the state-of-the-art of available long read error correction methods. More particularly, we describe the various methodologies adopted by the different tools, and list the tools relying on each methodology, both for hybrid and self-correction. Details about performances, both in terms of resource consumption and quality of the results, are however not discussed here. Experimental results of a subset of the available correction tools, on various datasets displaying diverse characteristics, are presented in Section 3. A summary of the available hybrid correction tools is given in Table 1. A summary of the available self-correction tools is given in Table 2.

2.1 Hybrid correction

Hybrid correction was the fist approach to be described in the literature. This strategy is based on a set of long reads and a set of short reads, both sequenced for the same individual. It aims to use the high quality information contained in the short reads to enhance the quality of the long reads. As first long read sequencing experiments displayed high error rates ($> 15\%$ on average), most methods relied on this additional use of short reads data. Four different hybrid correction approaches thus exist:

1. Alignment of short reads to the long reads;
2. Alignment of contigs and long reads;
3. Use of de Bruijn graphs;

4. Use of Hidden Markov Models.

We describe each approach more in details, and list the related tools, in the following subsections.

2.1.1 Short reads alignment

This approach was the first long read error correction approach described in the literature. It consists of two distinct steps. First, the short reads are aligned to the long reads. This step allows to cover each long read with a subset of related short reads. This subset can then be used to compute a high quality consensus sequence, which can, in turn, be used as the correction of the original long read. The different methods adopting this approach mainly vary by the alignment methods they use, and also by the algorithmic choices made during the consensus sequences computation. PBcR / PacBioToCA [3], LSC [4], Proovread [5], Nanocorr [6], LSCplus [7], CoLoRMap [8], and HECIL [9] are all based on this approach.

2.1.2 Contigs alignment

Given their length, short reads can be difficult to align to repeated regions, or to extremely noisy regions of the long reads. This approach aims to address this issue by first assembling the short reads. Indeed, the contigs obtained after assembling the short reads are much longer than the original short reads. As a result, they can cover the repeated or highly erroneous regions of the long reads much more efficiently, by using the context of the adjacent regions during the alignment. In the same fashion as the short reads alignment strategy described in Section 2.1.1, the contigs aligned with the long reads can then be used to compute high quality consensus sequences, and thus correct the long reads they are associated to. Once again, the different methods adopting this strategy vary by the alignment methods they use, and by the algorithmic choices made during consensus computation. ECTools [10], HALC [11], and MiRCA [12] adopt this methodology.

2.1.3 De Bruijn graphs

Another alternative to the alignment of short reads to the long reads is the direct use of a de Bruijn graph, built from the short reads k-mers. This approach aims to avoid the explicit step of short reads assembly altogether, contrary to the fmethods mentioned in Section 2.1.2, and instead directly use the graph to correct the long reads. The graph is first built from the solid k-mers of the short reads (*i.e.* k-mers appearing more frequently than a given threshold). The long reads can then be anchored to the graph according to their k-mers. Finally, the graph can be traversed in order to find paths, and link anchored regions of the long reads together, and thus correct erroneous, unanchored, regions. Methods adopting this approach vary by the way they represent the graph, but also by the way they anchor the long reads to the graph, and by the way they correct unanchored regions. LoRDEC [13], Jabba [14], FMLRC [15], and ParLECH [16] rely on this strategy.

2.1.4 Hidden Markov models

Hidden Markov models, used for short read error correction, were also adopted for the error correction of long reads. To this aim, models are first initialized in order to represent the original long reads. A subset of short reads is then assigned to each long read, by alignment. Each subset of short reads can then be used to train the model it is associated to. Finally, the trained models can be used to compute consensus sequences, and thus correct the long reads they represent. Hercules [17] is based on this approach.

2.1.5 Combination of strategies

Other methods combine different of the aforementioned strategies, in order to balance their advantages and drawbacks. For instance, NaS [18] combines a first step of short reads alignment to a second step of short reads recruitment and assembly in order to correct the long reads. HG-CoLoR [19] is also based on a first step relying on short reads alignment, but then makes use of a variable order de Bruijn graph (*i.e.* a single data structure containing all the de Bruijn graphs between k and K) in order to correct regions of the long reads that were not covered by the original alignments.

2.2 Self-correction

Self-correction aims to avoid the use of short reads data altogether, and to correct long reads solely based on the information contained in their sequences. Third generation sequencing technologies indeed evolve fast, and now allow the sequencing of long reads reaching error rates of 10-12%. As a result, correction is still required to properly deal with errors, but self-correction has recently undergone important developments. Two different self-correction approaches thus exist:

1. Multiple sequence alignment;
2. Use of de Bruijn graphs.

We describe each approach more in details, and list the related tools, in the following subsections.

2.2.1 Multiple sequence alignment

This approach is highly similar to the short reads alignment approach for hybrid correction, described in Section 2, and to the contigs alignment approach described in Section 2.1.2. It is thus composed of a first step of overlaps computation between the long reads, and of a second step of consensus computation from the overlaps. The overlaps computation can be performed either via a mapping strategy, which only provides the positions of the similar regions of the long reads, or via alignment, which provides the positions of the similar regions, as well as their actual base-to-base correspondence in terms of matches, mismatches, insertions and deletions. For the consensus computation step, a directed acyclic graph (DAG) is usually built in order to summarize the alignments, and extract a consensus sequence. Methods adopting this strategy thus vary by their overlapping strategy, but also by their algorithmic choices during the consensus computation. PBDAGCon (the correction module used in the HGAP assembler) [20], PBcR-BLASR [21], Sprai [4] [22], PBcR-MHAP [23], FalconSense (the correction module used in the assembler Falcon) [24], Sparc [25], the correction module used in the assembler Canu [26], MECAT [27] and FLAS [28] rely on this approach.

2.2.2 De Bruijn graphs

This approach is similar to the hybrid correction approach using de Bruijn graphs, mentioned in Section 2.1.3. In a first step, the graph is built from the long reads k-mers, and in a second step, the graph is traversed in order to find paths allowing to correct unanchored regions of the long reads. The main difference with the hybrid approach comes from the fact that here the graph is only constructed from the solid k-mers from the long reads. The methods adopting this approach mainly differ by the scale at which the graph is built. On the one hand, it can be built globally, by studying the frequency of all the k-mers appearing in the reads. On the other hand, it can be built locally, by first computing overlaps between the long reads, in order to define small similar regions of the long reads, and then building small, local graphs at the scale of these regions. LoRMA [29] and Daccord [30] are based on this approach.

4. http://zombie.cb.k.u-tokyo.ac.jp/sprai/index.html

2.2.3 Combination of strategies

As for hybrid correction, some methods also rely on combinations of the two previously described strategies. For instance, CONSENT [31] relies on both multiple sequence alignment and de Bruijn graphs. It first computes overlaps between the long reads, using a mapping approach. Small, similar regions of the long reads (called windows) are then defined from these overlaps. Windows are then processed in two different steps. First, a multiple sequence alignment strategy is used in order to compute a consensus sequence for each given window. Once a consensus sequence is computed for a given window, it further goes through a second correction step, in which a local de Bruijn graph is built and traversed in order to further polish remaining errors.

2.3 Summary

In this section, we draw a summary of the available hybrid and self-correction methods. For each method, we recall the main strategy or strategies it relies on, and the sequencing technologies it has been validated on. Hybrid correction tools are summarized in Table 1. Self-correction tools are summarized in Table 2.

Method	Approach	Release	Validated on
PBcR	SR alignment	2012	PacBio
LSC	SR alignment	2012	PacBio
ECTools	Contigs alignment	2014	PacBio
LoRDEC	DBG	2014	PacBio
Proovread	SR alignment	2014	PacBio
Nanocorr	SR alignment	2015	ONT
NaS	SR alignment	2015	ONT
CoLoRMap	SR alignment	2016	PacBio
Jabba	DBG	2016	PacBio
LSCplus	SR alignment	2016	PacBio
HALC	Contigs alignment	2017	PacBio
HECIL	SR alignment	2018	PacBio
Hercules	Modèles de Markov cachés	2018	PacBio
FMLRC	DBG	2018	PacBio
HG-CoLoR	SR alignment + DBG	2018	PacBio + ONT
MiRCA	Contigs alignment	2018	ONT
ParLECH	DBG	2019	PacBio

Tab. 1. List of long read hybrid correction tools.

Method	Approach	Release	Validated on
PBcR-BLASR	MSA	2013	PacBio
PBDAGCon	MSA	2013	PacBio
Sprai	MSA	2014	PacBio
PBcR-MHAP	MSA	2015	PacBio
FalconSense	MSA	2016	PacBio
LoRMA	DBG	2016	PacBio
Sparc	MSA	2016	PacBio + ONT
Canu	MSA	2017	PacBio + ONT
Daccord	DBG	2017	PacBio + ONT
MECAT	MSA	2017	PacBio + ONT
CONSENT	MSA + DBG	2019	PacBio + ONT
FLAS	MSA	2019	PacBio

Tab. 2. List of long read self-correction tools.

3 Qualitative comparison

In this section, we study how the characteristics of the datasets impact the quality of the aforementioned error correction methods. However, we exclude the following tools, since they could not install or could not be run: FalconSense, HECIL, LSCplus, MiRCA, PBcR, PBcR-BLASR, PBcR-MHAP, PBDAGCon, Sparc, and Sprai. We also exclude the following tools, for performance reasons: ECTools, Hercules, LSC, Nanocorr, and NaS.

3.1 Datasets

We study the performances of the different tools on a set of six different datasets: one from *Acinetobacter baylyi*, three from *Saccharomyces cerevisiae* and two from *Caenorhabditis elegans*. A summary of these datasets is given in Table 3. We only showcase results on these datasets for place sake, but actually performed a much more in-depth benchmark, on a total of 20 datasets. Complete results of this benchmark are available in the extended bioRxiv preprint, available at: https://doi.org/10.1101/2020.03.06.977975.

3.2 Results

To evaluate the quality of the correction provided by each tool, we used ELECTOR [32], a software specially developed for large scale error correction tools benchmark. The results of our experiments are summarized in Table 4. For place sake, we only provide the number of corrected bases, and the error rate of the reads after correction, as well as the time and memory consumption of each tool.

However, as previously mentioned, the actual benchmark we performed provides a large number of additional metrics. These results nonetheless illustrate how the different characteristics impact the quality of the correction, and the performances of each tool.

Dataset	Number of reads	Error rate	Coverage	Number of bases
Simulated PacBio data				
S. cerevisiae 30x	45,198	12.28	30x	371 Mbp
C. elegans 30x	366,416	12.28	30x	3,006 Mbp
S. cerevisiae 60x	90,397	12.28	60x	742 Mbp
C. elegans 60x	732,832	12.28	60x	6,011 Mbp
Real ONT data				
A. baylyi	89,011	29.91	106x	381 Mbp
S. cerevisiae real	205,923	44.51	95x	1,173 Mbp

Tab. 3. Characteristics of the datasets used during the experiments.

Tool	Metric	*S. cerevisiae 30x*	*C. elegans 30x*	*S. cerevisiae 60x*	*C. elegans 60x*	*A. baylyi*	*S. cerevisiae real*
CoLoRMap	Number of bases (Mbp)	343	1,198	664	-	141	165
	Error rate (%)	0.3183	0.8955	0.6143	-	0.4921	0.3042
	Runtime	4 h 36 min	150 h 21 min	8 h 08 min	-	3 h 41 min	10 h 44 min
	Memory (MB)	14,243	32,267	24,375	-	13,028	18,241
FMLRC	Number of bases (Mbp)	348	2,821	695	5,652	391	1,185
	Error rate (%)	0.2447	1.4161	0.2469	1.4213	0.3221	3.2836
	Runtime	1 h 59 min	11 h 55 min	3 h 57 min	23 h 25 min	2 h 01 min	6 h 15 min
	Memory (MB)	892	7,937	4 h 25 min	7,937	449	876
HALC	Number of bases (Mbp)	348	2,819	694	5,649	190	255
	Error rate (%)	0.3611	1.0897	0.3648	1.0880	0.1655	0.7067
	Runtime	1 h 53 min	9 h 30 min	4 h 25 min	19 h 10 min	47 h 41 min	2 h 56 min
	Memory (MB)	1,892	2,853	2,487	5,716	10,577	2,329
HG-CoLoR	Number of bases (Mbp)	347	2,795	690	-	285	512
	Error rate (%)	0.5115	1.1664	0.5995	-	0.0240	0.2824
	Runtime	7 h 20 min	108 h 26 min	12 h 23 min	-	1 h 34 min	8 h 51 min
	Memory (MB)	3,656	27,212	7,297	-	3,750	11,575
Jabba	Number of bases (Mbp)	340	2,464	679	4,935	179	243
	Error rate (%)	0.1067	0.2319	0.1040	0.2312	0.0774	0.1111
	Runtime	5 min	43 min	5 min	49 min	2 min	7 min
	Memory (MB)	1,215	13,362	1,215	13,360	1,217	1,217
LoRDEC	Number of bases (Mbp)	348	2,824	696	5,657	175	221
	Error rate (%)	0.3990	1.2710	0.3948	1.2731	0.0552	1.1832
	Runtime	35 min	11 h 30 min	1 h 09 min	23 h 30 min	16 min	1 h 09 min
	Memory (MB)	799	2,320	794	2,332	436	797
Proovread	Number of bases (Mbp)	342	2,704	971	-	156	160
	Error rate (%)	0.2365	0.4325	0.2568	-	0.0314	0.1021
	Runtime	5 h 37 min	85 h 23 min	11 h 51 min	-	3 h 25 min	13 h 42 min
	Memory (MB)	16,777	29,934	23,591	-	10,618	8,709
Canu	Number of bases (Mbp)	226	2,773	599	5,112	81	-
	Error rate (%)	1.1052	0.5008	0.7919	0.7934	5.4081	-
	Runtime	29 min	9 h 09 min	1 h 11 min	9 h 30 min	31 min	-
	Memory (MB)	3,681	6,921	3,710	7,050	3,015	-
CONSENT	Number of bases (Mdp)	344	2,787	688	5,586	183	179
	Error rate (%)	0.4258	0.6720	0.2812	0.3806	8.0530	23.2735
	Runtime	47 min	7 h 55 min	1 h 49 min	19 h 13 min	48 min	40 min
	Memory (MB)	5,514	16,772	11,335	15,607	5,150	14,663
Daccord	Number of bases (Mbp)	348	-	695	-	175	-
	Error rate (%)	0.1259	-	0.0400	-	6.7454	-
	Runtime	1 h 19 min	-	2 h 26 min	-	43 min	-
	Memory (MB)	31,798	-	32,190	-	25,801	-
FLAS	Number of bases (Mbp)	344	2,729	689	5,584	165	221
	Error rate (%)	0.3272	0.7613	0.2034	0.3997	8.3926	22.8287
	Runtime	29 min	3 h 07 min	1 h 30 min	10 h 45 min	32 min	39 min
	Memory (MB)	2,935	10,565	4,984	13,682	3,015	7,398
LoRMA	Number of bases (Mbp)	14	33	443	781	76	11
	Error rate (%)	2.1640	3.6960	0.2225	0.6446	1.9290	4.7390
	Runtime	46 min	8 h 19 min	5 h 25 min	31 h 04 min	29 min	1 h 35 min
	Memory (MB)	31,899	31,827	31,828	32,104	31,575	1,505
MECAT	Number of bases (Mbp)	285	2,084	616	4,938	154	84
	Error rate (%)	0.3040	0.3908	0.2088	0.2675	8.5324	19.9237
	Runtime	5 min	48 min	16 min	2 h 43 min	23 min	14 min
	Memory (MB)	2,907	10,535	4,954	10,563	2,978	7,374

Tab. 4. Results of the different tools on the studied datasets. CoLoRMap, HG-CoLoR, and Proovread were not run on the *C. elegans* 60x dataset, due to their runtimes being too large. Daccord could not be run on the two *C. elegans* datasets, and on the *S. cerevisiae* real dataset as it consumed a large amount of memory, and could not even be run on a cluster node with 128 GB of RAM. Canu reported an error on the *S. cerevisiae* real dataset.

4 Conclusion

In this paper, we presented the state-of-the-art of long read error correction, tackling both hybrid and self-correction. For each approach, we described the different existing methodologies, and listed all tools available at the moment. Four different approaches thus exist for hybrid correction: short reads alignment, contigs alignment, use of de Bruijn graphs and use of hidden Markov models. For self-correction, two main methodologies exist: multiple sequence alignment and use of de Bruijn graphs. As of today, a total of 29 different methods exist for performing long read error correction.

We also showcased how the long reads datasets characteristics can impact the quality of the correction. In particular, our experiments show that self-correction performs better than hybrid correction as the sequencing depth grows. Oppositely, given high error rates, hybrid correction tends to perform the best, even when the sequencing depth is high. In addition, our experiments also underline the fact that self-correction tends to perform better when the complexity of the sequenced organism grows.

Further work shall focus on a more in-depth description of each available tool, to give the reader a better understanding of the algorithmic differences exiting between tools adopting the same approaches. In addition, more datasets could also be studied, in order to provide better guidelines as to which tool to choose according to the datasets characteristics.

Acknowledgements

Part of this work was performed using computing resources of CRIANN (Normandy, France), project 2017020.

References

[1] Miten Jain, Sergey Koren, Karen H Miga, Josh Quick, Arthur C Rand, Thomas A Sasani, John R Tyson, Andrew D Beggs, Alexander T Dilthey, Ian T Fiddes, et al. Nanopore sequencing and assembly of a human genome with ultra-long reads. *Nature biotechnology*, 36(4):338, 2018.

[2] Fritz J Sedlazeck, Philipp Rescheneder, Moritz Smolka, Han Fang, Maria Nattestad, Arndt Von Haeseler, and Michael C Schatz. Accurate detection of complex structural variations using single-molecule sequencing. *Nature Methods*, 15(6):461–468, 2018.

[3] Sergey Koren, Michael C Schatz, Brian P Walenz, Jeffrey Martin, Jason T Howard, Ganeshkumar Ganapathy, Zhong Wang, David A Rasko, W Richard McCombie, Erich D Jarvis, and Adam M Phillippy. Hybrid error correction and de novo assembly of single-molecule sequencing reads. *Nature biotechnology*, 30(7):693–700, 2012.

[4] Kin Fai Au, Jason G. Underwood, Lawrence Lee, and Wing Hung Wong. Improving PacBio Long Read Accuracy by Short Read Alignment. *PLoS ONE*, 7(10):1–8, 2012.

[5] Thomas Hackl, Rainer Hedrich, Jörg Schultz, and Frank Förster. Proovread: Large-scale high-accuracy PacBio correction through iterative short read consensus. *Bioinformatics*, 30(21):3004–3011, 2014.

[6] Sara Goodwin, James Gurtowski, Scott Ethe-Sayers, Panchajanya Deshpande, Michael C Schatz, and W Richard Mccombie. Oxford Nanopore sequencing, hybrid error correction, and de novo assembly of a eukaryotic genome. *Genome Research*, 25:1750–1756, 2015.

[7] Ruifeng Hu, Guibo Sun, and Xiaobo Sun. LSCplus: a fast solution for improving long read accuracy by short read alignment. *BMC Bioinformatics*, 17(1):451, 2016.

[8] Ehsan Haghshenas, Faraz Hach, S Cenk Sahinalp, and Cedric Chauve. CoLoRMap: Correcting Long Reads by Mapping short reads. *Bioinformatics*, 32:i545–i551, 2016.

[9] Olivia Choudhury, Ankush Chakrabarty, and Scott J. Emrich. HECIL: A hybrid error correction algorithm for long reads with iterative learning. *Scientific Reports*, 8(1):1–9, 2018.

[10] Hayan Lee, James Gurtowski, Shinjae Yoo, Shoshana Marcus, W. Richard McCombie, and Michael Schatz. Error correction and assembly complexity of single molecule sequencing reads. *bioRxiv*, page 006395, 2014.

[11] Ergude Bao and Lingxiao Lan. HALC: High throughput algorithm for long read error correction. *BMC Bioinformatics*, 18:204, 2017.

[12] Mehdi Kchouk and Mourad Elloumi. An Error Correction and DeNovo Assembly Approach for Nanopore Reads Using Short Reads. *Current Bioinformatics*, 13(3):241–252, 2018.

[13] Leena Salmela and Eric Rivals. LoRDEC: Accurate and efficient long read error correction. *Bioinformatics*, 30:3506–3514, 2014.

[14] Giles Miclotte, Mahdi Heydari, Piet Demeester, Stephane Rombauts, Yves Van de Peer, Pieter Audenaert, and Jan Fostier. Jabba: hybrid error correction for long sequencing reads. *Algorithms for Molecular Biology*, 11:10, 2016.

Paper 1

[15] Jeremy R. Wang, James Holt, Leonard McMillan, and Corbin D. Jones. FMLRC: Hybrid long read error correction using an FM-index. *BMC Bioinformatics*, 19(1):1–11, 2018.

[16] Arghya Kusum Das, Sayan Goswami, Kisung Lee, and Seung Jong Park. A hybrid and scalable error correction algorithm for indel and substitution errors of long reads. *BMC Genomics*, 20(Suppl 11):1–15, 2019.

[17] Can Firtina, Ziv Bar-joseph, Can Alkan, and A Ercument Cicek. Hercules: a profile HMM-based hybrid error correction algorithm for long reads. *Nucleic Acids Research*, 46(21), 2018.

[18] Mohammed-Amin Madoui, Stefan Engelen, Corinne Cruaud, Caroline Belser, Laurie Bertrand, Adriana Alberti, Arnaud Lemainque, Patrick Wincker, and Jean-Marc Aury. Genome assembly using Nanopore-guided long and error-free DNA reads. *BMC Genomics*, 16:327, 2015.

[19] Pierre Morisse, Thierry Lecroq, and Arnaud Lefebvre. Hybrid correction of highly noisy long reads using a variable-order de Bruijn graph. *Bioinformatics*, 34(24):4213–4222, 06 2018.

[20] Chen-Shan Chin, David H Alexander, Patrick Marks, Aaron A Klammer, James Drake, Cheryl Heiner, Alicia Clum, Alex Copeland, John Huddleston, Evan E Eichler, Stephen W Turner, and Jonas Korlach. Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data. *Nature Methods*, 10:563–569, 2013.

[21] Sergey Koren, Gregory P Harhay, Timothy P L Smith, James L Bono, Dayna M Harhay, Scott D Mcvey, Diana Radune, Nicholas H Bergman, and Adam M Phillippy. Reducing assembly complexity of microbial genomes with single-molecule sequencing. *Genome Biology*, 14(9):R101, sep 2013.

[22] Kazuyoshi Gotoh, Teruo Yasunaga, Takamasa Imai, Daisuke Motooka, Kazutoshi Yoshitake, Toshihiro Horii, Mari Miyamoto, Masahiro Kasahara, Tetsuya Iida, Kazuharu Arakawa, Shota Nakamura, and Naohisa Goto. Performance comparison of second- and third-generation sequencers using a bacterial genome with two chromosomes. *BMC Genomics*, 15(1):699, 2014.

[23] Konstantin Berlin, Sergey Koren, Chen-Shan Chin, James P Drake, Jane M Landolin, and Adam M Phillippy. Assembling large genomes with single-molecule sequencing and locality-sensitive hashing. *Nature biotechnology*, 33:623–630, 2015.

[24] 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, Grant R. Cramer, Massimo Delledonne, Chongyuan Luo, Joseph R. Ecker, Dario Cantu, David R. Rank, and Michael C. Schatz. Phased diploid genome assembly with single-molecule real-time sequencing. *Nature Methods*, 13(12):1050–1054, 2016.

[25] Chengxi Ye and Zhanshan (Sam) Ma. Sparc: a sparsity-based consensus algorithm for long erroneous sequencing reads. *PeerJ*, 4:e2016, 2016.

[26] Sergey Koren, Brian P Walenz, Konstantin Berlin, Jason R Miller, Nicholas H Bergman, and Adam M Phillippy. Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation. *Genome Research*, 27:722–736, 2017.

[27] Chuan Le Xiao, Ying Chen, Shang Qian Xie, Kai Ning Chen, Yan Wang, Yue Han, Feng Luo, and Zhi Xie. MECAT: Fast mapping, error correction, and de novo assembly for single-molecule sequencing reads. *Nature Methods*, 14(11):1072–1074, 2017.

[28] Ergude Bao, Fei Xie, Changjin Song, and Dandan Song. FLAS: fast and high-throughput algorithm for PacBio long-read self-correction. *Bioinformatics*, 2019.

[29] Leena Salmela, Riku Walve, Eric Rivals, and Esko Ukkonen. Accurate selfcorrection of errors in long reads using de Bruijn graphs. *Bioinformatics*, 33:799–806, 2017.

[30] German Tischler and Eugene W Myers. Non Hybrid Long Read Consensus Using Local De Bruijn Graph Assembly. *bioRxiv, doi: https://doi.org/10.1101/106252*, 2017.

[31] Pierre Morisse, Camille Marchet, Antoine Limasset, Thierry Lecroq, and Arnaud Lefebvre. Consent: Scalable self-correction of long reads with multiple sequence alignment. *RECOMB-Seq*, 2019.

[32] Camille Marchet, Pierre Morisse, Lolita Lecompte, Arnaud Lefebvre, Thierry Lecroq, Pierre Peterlongo, and Antoine Limasset. ELECTOR: evaluator for long reads correction methods. *NAR Genomics and Bioinformatics*, 2(1), 11 2020. lqz015.

Paper 2

UMI-VarCal: a new UMI-based variant caller that efficiently improves low-frequency variant detection in paired-end sequencing NGS libraries

Vincent SATER [1,2,3], Pierre-Julien VIAILLY [2,3], Thierry LECROQ [1], Élise PRIEUR-GASTON [1], Élodie BOHERS [2,3], Mathieu VIENNOT [2,3], Philippe RUMINY [2,3], Hélène DAUCHEL [1], Pierre VERA [1,2], 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
Original article is published at https://doi.org/10.1093/bioinformatics/btaa053
A preprint full text is available at https://doi.org/10.1101/77

Abstract

Due to recent advances in the field of oncology, and especially the increased use of liquid biopsy to monitor the tumor burden in the blood, the rise of new variant calling algorithms or strategies adapted to the low frequency variant detection has become a must. Because of PCR enrichment and sequencing technologies limitations, artifactual variants (sequencing and DNA polymerase errors) are also introduced at low frequencies making the distinction between real variants and artifactual ones a true challenge. However, the recent use of Unique Molecular Identifiers (UMI) in targeted sequencing protocols has offered a trustworthy approach to accurately call low frequency variants.

Here, we present UMI-VarCal, a new UMI-based variant caller with remarkably higher specificity compared to raw-reads-based variant callers. Although our variant caller is far from being the only one that uses UMI information to call variants, UMI-VarCal stands out from the crowd by not relying on SAMtools to do its pileup. Instead, thanks to an innovative homemade pileup algorithm specifically designed to treat the UMI tags present in the reads, our variant caller surpasses the other variant callers (OutLyzer [1], DeepSNVMiner [2], SiNVICT [3]) in terms of specificity. Furthermore, being developed with performance in mind, our tool is considerably more efficient than the other approaches in terms of execution time and memory consumption.

We illustrate the results obtained using UMI-VarCal through the sequencing of 3 samples from patients suffering from lymphoma and 2 simulated samples (at different depths) in which we inserted a known set of variants. We demonstrate that UMI-VarCal can detect variants with frequencies as low as 0.3% and filter out false positives resulting in a sensitivity that outmatches other variant callers.

References

[1] Muller E, Goardon N, Brault B, et al. OutLyzer: software for extracting low-allele-frequency tumor mutations from sequencing background noise in clinical practice. *Oncotarget*. 2016;7(48):79485–79493. doi:10.18632/oncotarget.13103

[2] Andrews TD, Jeelall Y, Talaulikar D, Goodnow CC, Field MA. DeepSNVMiner: a sequence analysis tool to detect emergent, rare mutations in subsets of cell populations. *PeerJ*. 2016;4:e2074. Published 2016 May 24. doi:10.7717/peerj.2074

[3] Can Kockan, Faraz Hach, Iman Sarrafi, Robert H Bell, et al. SiNVICT: ultra-sensitive detection of single nucleotide variants and indels in circulating tumour DNA. Bioinformatics, Volume 33, Issue 1:26-34, 2017

Paper 3

Local conformations in ordered and intrinsically disordered proteins

Alexandre G. DE BREVERN[1]

[1] INSERM UMR_S 1134, Univ Paris, INTS, 6 rue Alexandre Cabanel, 75015, Paris, France

Corresponding Author: alexandre.debrevern@univ-paris-diderot.fr

Narwani TJ *et al.* (2019). Discrete analyses of protein dynamics, J Biomol Struct Dyn. 2019, 12:1-15. https://doi.org/10.1080/07391102.2019.1650112.

Vattekatte AM *et al.* (2020). A structural entropy index to analyse local conformations in intrinsically disordered proteins, J Struct Biol. 2020 in press. https://doi.org/10.1016/j.jsb.2020.107464.

Abstract *Protein structures are highly dynamic macromolecules. This dynamics is often analyzed with a limited number of proteins. In our study, molecular dynamics (MDs) simulations were performed on a large set of 169 representative protein domains. To investigate protein flexibility, classical approaches such as RMSf or solvent accessibility were used, but also innovative approaches such as local entropy.*

At first, classical secondary structures were explored. Concerning the helical structures, only 76.4% of the residues associated to α-helices retain the conformation; this tendency drops to 40.5% for 3_{10}-helices and near zero for π-helices. However, this last impressive non-stability is entirely dependent on the assignment approach. Indeed, with the most recent DSSP version, these results are totally scrambled, the π–helices showed behaviors equivalent to 3_{10}-helix [1].

The rigidity of β-sheet was confirmed, but we also show its capacity to transform into turns. Finally, while the dynamics between turns (with hydrogen bond) and bends (without hydrogen bond) have some strong similarities, they also showed differences as turns convert easily to helical structures while bends prefer the extended conformations.

Analyses were similarly performed using a structural alphabet [2], namely the Protein Blocks (PBs) [3]. For half of the PBs, to be buried or exposed does not change at all its dynamics. The majority of PBs remain as their original conformation, or at least with a high frequency. Few PBs have a higher tendency to be more flexible. The intriguing fact is that the change from a PB to another one does not correspond to a simple geometrical evolution. It is more frequent to go to an unexpected PB than an expected one.

To go further, a dataset of disorder protein ensembles was analyzed with the PB. Using a PB derived entropy index, we quantify, for the first time, continuum from rigidity to flexibility and finally disorder. We also highlight non-disordered regions in the ensemble of disordered proteins.

These studies show the complex nature of protein dynamics and the value of their analysis at a local level. In addition, they show the possibility of performing these analyzes on both ordered and disordered proteins.

Keywords. Structural alphabet, entropy, molecular dynamics, flexibility, secondary structure.

References

1. Tarun J. Narwani, Pierrick Craveur, Nicolas K Shinada, Hubert Santuz, Joseph Rebehmed, Catherine Etchebest, and Alexandre G. de Brevern. Dynamics and deformability of α-, 3_{10}- and π-helices. *Archives of Biological Science*s, 70(1):21-31, 2018.
2. Bernard Offmann, Manoj Tyagi, and Alexandre G. de Brevern.. Local Protein Structures. *Current Bioinformatics*, 3:165-202, 2007.
3. Alexandre G. de Brevern, Catherine Etchebest, and Serge Hazout. Bayesian probabilistic approach for predicting backbone structures in terms of protein blocks. *Proteins*, 41(3):271-87, 2000.

Paper 17

Ensembl 2020, data growth - Species Quick Release Processing

Marc Chakiachvili [1], Fergal Martin[1], Steve Trevanion[1], Anne Parker[1], Thomas Maurel[1], James Allen[1], Luca Da Rin Fioretto[1], Vinay Kaikala[1], and Andrew D Yates[1]

[1] European Molecular Biology Laboratory, European Bioinformatics Institute, Wellcome Genome Campus, Hinxton, Cambridge, CB10 1SD, United Kingdom

Corresponding Author: mchakiachvili@ebi.ac.uk

Url: 2020.ensembl.org

Abstract: *Ensembl[1] (https://www.ensembl.org) and Ensembl Genomes[2] (https://ensemblgenomers.org) are systems for generating and distributing genome annotation such as genes, variation, regulation and comparative genomics across a large taxonomic space. The Ensembl annotation pipeline is capable of integrating experimental and reference data from multiple providers into a single integrated resource. Both software and data are made available without restriction via our websites, online tools platform and programmatic interfaces (available under an Apache 2.0 license) four times a year. Historically Ensembl release cycles last about three months. The increasing amount of data managed by our different teams has led to an increasing strain on the release cycle, regarding prospects in terms of number of species due in the near future. Since Ensembl's first release, we have constantly improved our processes to keep up with data growth. This is getting even more important regarding the growth of large scale biodiversity sequencing projects all around the world. We currently define our release process as fully integrated (FI) processing in the sense that every single data we provide is available with all available related data computed before release (Variation, Regulation and Comparatives Genomics). To address the latest increases in data volume, we plan to turn our release into a partial integration (PI) process. Under PI there would be the concept of a minimum data release. Hence new and updated data release would not require the full set of related data, allowing us to release on a much faster basis, we are expecting to release on a two week base, allowing updates with new data available as soon as they are available.*

Keywords: Ensembl, EnsemblGenomes, Sequencing Data, Ensembl 2020 WebSite, Darwin Tree of Life, Vertebrates Genomes Project.

Introduction

Historically Ensembl release cycles last about three months for all our available resources. The process starts with the production of species assembly databases for new and updated data available and ends with the release of data via various servers around the world. Comparative genomics analysis is performed on each species, providing cross-species resources and analysis, both at gene and sequence level to produce fully reconciled phylogenies of genes (both for protein coding and non-coding genes), ortholog and paralog prediction. For species with variation (substitutions, insertions, deletions and structural events) and regulation (DNase-seq, FAIRE-seq and ChIP-seq) data we create dedicated Variation [3] , Regulation [4] builds. The expectation that these downstream data types are made available alongside our gene sets at release is called full integration (FI). The increasing amount of data managed by our different analysis methods has led to increasing pressure on Ensembl's release cycle, and these numbers will grow dramatically in the near-future with projects such as the Darwin Tree of Life (DToL) [5] and Vertebrates Genomes Project (VGP) [6] .

Current release processing (FI)

Currently there is an expectation that Ensembl will analyse and release all suitable data for an assembly (full integration) and available to access via our APIs, databases and FTP site for it to be available in Ensembl.

Here is a simplified *Ensembl release process*[7] workflow:

- Assemblies: New/updated assemblies from sequencing projects around the world are submitted to archives and then have genes annotated using computational methods based on experimental data. New species are added frequently every release cycle, and existing species may receive updates held in databases. These databases are then handed over to other parts of the Ensembl project to be further processed.
- Variation/Regulation/Comparative analysis data: Once a genome has been annotated, QC'd and released internally, other analysis methods are free to process all available genomes in order to create additional value on data sets. These range from our comparative genomics methods to our variation and regulation data builds
- Data transformation: We then process all available data to produce a comprehensive set of flat file serialisations (GTF, Genbank, FASTA, amongst others) to be delivered on our FTP site.
- Web publication: Once finished data is published via our website. This involves the handover of all known data sets and the transformation of a subset of data into web optimised formats.

Release: When release is ready, an archive is created from previously published data and new dataset is made available. Since data accrues as time passes, so does the load on a release resulting in longer release cycles. As volumes of genomes increase, the FI model means that achieving four public releases a year is difficult to achieve and must be revised to cope with growing amounts of data.

Expected Quick Release Process

Under the PI there would be the concept of a minimum data release. The most natural minimum data release would be an assembly and an annotated gene set, however conceivably the minimum possible data release is simply an unannotated assembly against which analysis tools such as BLAST can be run. PI is based on the idea that we can update data sets linked assemblies much more dynamically with downstream analyses being placed in the next available release window. Under this strategy we would by default only display assemblies with full integration to users, but would allow advanced users access to partially processed data as soon as it becomes available. This could mean access to gene sets without compara data.

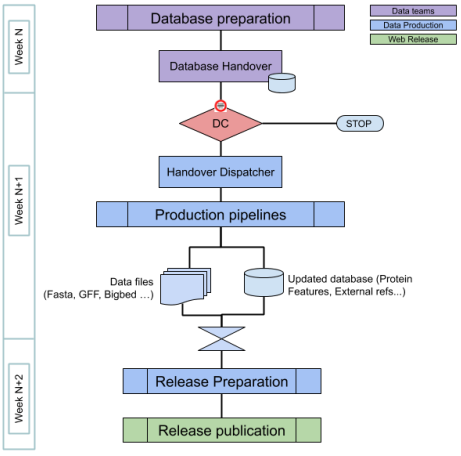


Fig 1. - Ensembl Quick Release: Workflow overview, timeline and teams involved. Data Teams are Genebuild, Variation, Regulation and Comparative Genomics.

Future prospects

The Quick Release Processing (QRP) release procedure, in its first incarnation, will provide access to genomes across the tree of life from the wealth of genome sequencing projects now available. QRP is

intended to bridge the gap between our current infrastructure and our future infrastructure. A preview of this future infrastructure is available from http://2020.ensembl.org. Our intention is for this new infrastructure to use QRP's processing systems to create denormalised data representations suitable for consumption through our new programmatic and visual interfaces. As such, Ensembl will continue to support rapid release of emerging genomes to researchers with minimal delay through a system ensuring both consistency of annotation and correctness of data.

Fundings

Wellcome Trust [WT108749/Z/15/Z]; National Human Genome Research Institute [U41HG007823, 2U41HG007234]. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health; Biotechnology and Biological Sciences Research Council [BB/N019563/1, BB/M011615/1]; Open Targets; Wellcome Trust [WT104947/Z/14/Z, WT200990/Z/16/Z, WT201535/Z/16/Z, WT108749/Z/15/A, WT212925/Z/18/Z]; ELIXIR: the research infrastructure for life-science data; This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 733161 (MultipleMS).; 'Save the Tasmanian Devil Program'; European Molecular Biology Laboratory. Funding for open access charge: Wellcome Trust [WT108749/Z/15/Z].

Conflict of interest statement. Paul Flicek is a member of the Scientific Advisory Boards of Fabric Genomics, Inc. and Eagle Genomics, Ltd.

References

1. Andrew D Yates, Premanand Achuthan, Wasiu Akanni, James Allen, Jamie Allen, Jorge Alvarez-Jarreta, M Ridwan Amode, Irina M Armean, Andrey G Azov, Ruth Bennett, Jyothish Bhai, Konstantinos Billis, Sanjay Boddu, José Carlos Marugán, Carla Cummins, Claire Davidson, Kamalkumar Dodiya, Reham Fatima, Astrid Gall, Carlos Garcia Giron, Laurent Gil, Tiago Grego, Leanne Haggerty, Erin Haskell, Thibaut Hourlier, Osagie G Izuogu, Sophie H Janacek, Thomas Juettemann, Mike Kay, Ilias Lavidas, Tuan Le, Diana Lemos, Jose Gonzalez Martinez, Thomas Maurel, Mark McDowall, Aoife McMahon, Shamika Mohanan, Benjamin Moore, Michael Nuhn, Denye N Oheh, Anne Parker, Andrew Parton, Mateus Patricio, Manoj Pandian Sakthivel, Ahamed Imran Abdul Salam, Bianca M Schmitt, Helen Schuilenburg, Dan Sheppard, Mira Sycheva, Marek Szuba, Kieron Taylor, Anja Thormann, Glen Threadgold, Alessandro Vullo, Brandon Walts, Andrea Winterbottom, Amonida Zadissa, Marc Chakiachvili, Bethany Flint, Adam Frankish, Sarah E Hunt, Garth IIsley, Myrto Kostadima, Nick Langridge, Jane E Loveland, Fergal J Martin, Joannella Morales, Jonathan M Mudge, Matthieu Muffato, Emily Perry, Magali Ruffier, Stephen J Trevanion, Fiona Cunningham, Kevin L Howe, Daniel R Zerbino, Paul Flicek, Ensembl 2020, Nucleic Acids Research, Volume 48, Issue D1, 08 January 2020, Pages D682–D688, https://doi.org/10.1093/nar/gkz966
2. Kevin L Howe, Bruno Contreras-Moreira, Nishadi De Silva, Gareth Maslen, Wasiu Akanni, James Allen, Jorge Alvarez-Jarreta, Matthieu Barba, Dan M Bolser, Lahcen Cambell, Manuel Carbajo, Marc Chakiachvili, Mikkel Christensen, Carla Cummins, Alayne Cuzick, Paul Davis, Silvie Fexova, Astrid Gall, Nancy George, Laurent Gil, Parul Gupta, Kim E Hammond-Kosack, Erin Haskell, Sarah E Hunt, Pankaj Jaiswal, Sophie H Janacek, Paul J Kersey, Nick Langridge, Uma Maheswari, Thomas Maurel, Mark D McDowall, Ben Moore, Matthieu Muffato, Guy Naamati, Sushma Naithani, Andrew Olson, Irene Papatheodorou, Mateus Patricio, Michael Paulini, Helder Pedro, Emily Perry, Justin Preece, Marc Rosello, Matthew Russell, Vasily Sitnik, Daniel M Staines, Joshua Stein, Marcela K Tello-Ruiz, Stephen J Trevanion, Martin Urban, Sharon Wei, Doreen Ware, Gary Williams, Andrew D Yates, Paul Flicek, Ensembl Genomes 2020—enabling non-vertebrate genomic research, *Nucleic Acids Research*, Volume 48, Issue D1, 08 January 2020, Pages D689–D695, https://doi.org/10.1093/nar/gkz890
3. Hunt S.E., McLaren W., Gil L., Thormann A., Schuilenburg H., Sheppard D., Parton A., Armean I.M., Trevanion S.J., Flicek P. et al.. *Ensembl variation resources. Database J. Biol. Databases Curation. 2018*; https://doi.org/10.1093/database/bay119
4. Zerbino D.R., Wilder S.P., Johnson N., Juettemann T., Flicek P.R. *The Ensembl regulatory build. Genome Biol. 2015*; 16 :56.
5. The Darwin Tree of Life project: https://www.darwintreeoflife.org/
6. The Vertebrate Genomes Project: https://vertebrategenomesproject.org/
7. The Ensembl Release Cycle: https://www.ensembl.org/info/about/release_cycle.html

Paper 18

Automated Inference of Gene Regulatory Networks Using Explicit Regulatory Modules

Clémence Réda[1] and Bartek Wilczyński[2]
[1] Université Paris Diderot, 5, rue Thomas Mann, 75013, Paris, France
[2] Faculty of Mathematics, Informatics and Mechanics, ulica Stefana Banacha 2, 02 – 097, Warsaw, Poland

Corresponding author: clemence.reda@inserm.fr

Reference paper: **Réda and Wilczyński (2020) Automated inference of gene regulatory networks using explicit regulatory modules.** ***Journal of Theoretical Biology.*** https://doi.org/10.1016/j.jtbi.2019.110091

Gene regulatory networks are a popular tool for modelling important biological phenomena. Efficient identification of the causal connections between genes, their products and regulating transcription factors, is key to understanding how defects in their function may trigger diseases. Adding more biologically-motivated topological constraints on the network might lead to better results in network inference. Moreover, in recent years, we have seen great improvements in mapping of specific binding sites of many transcription factors to distinct regulatory regions. Recent gene regulatory network models use binding measurements in addition to gene expression data from perturbation experiments; but usually only to define gene-to-gene interactions, ignoring regulatory module structure, which might be key to a better understanding of the studied network dynamics. Eventually, current huge amount of transcriptomic data, and exploration of all possible cis-regulatory arrangements which can lead to the same transcriptomic response, makes manual model building, from literature, both tedious and time-consuming [1].

In our paper, we suggest a generic method to explicitly specify possible cis-regulatory connections in a gene regulatory network, based on transcription factor binding evidence. We have implemented our method using the formalism of Boolean networks. Our networks explicitly define cis-regulatory regions as additional nodes in the network, and further constraint the topology of the network using transcription factor bindings to cis-regulatory elements. Previous Boolean networks can be turned ("expanded") into such networks in a simple, automated way. Automatic network inference can then be performed on networks with putative regulatory interactions, using expression data, in order to find regulatory functions and edges which are consistent with wet-lab results. Infered networks can then be inspected in *in silico* simulations.

We use our new modelling framework in order to design a pipeline which automatically enumerates all biological scenarii (as "cis-regulatory" Boolean models) that can explain the experimental data provided. We suppose that the possibly multiple results obtained for a given dataset can be interpreted as different transcription factor binding site arrangements, and that the modular structure of these solution models allows to better understand the regulatory phenomena at play. We have tested our method on two previously published regular Boolean models [2,1], and were able to observe that redundant biological interactions can effectively be modelled using cis-regulatory interactions, and that such cis-regulatory networks preserve stable states of the initial Boolean models.

This work is a proof-of-concept that current qualitative modelling frameworks can benefit from topological biological knowledge. The fully automated method for model identification has been implemented in Python, and the expansion algorithm in R. The method resorts to the Z3 Satisfiability Modulo Theories (SMT) solver [3], and is similar to the RE:IN application [4].

References

[1] Collombet and van Oevelen *et al.* Logical modeling of lymphoid and myeloid cell specification and transdifferentiation. *Proceedings of the National Academy of Sciences*, page 201610622, 2017.

[2] Dunn and Martello *et al.* Defining an essential transcription factor program for naive pluripotency. *Science*, 344(6188):1156–1160, 2014.

[3] Yordanov and Wintersteiger *et al.* Z34bio: An smt-based framework for analyzing biological computation. *Proceedings of SMT*, 13, 2013.

[4] Yordanov and Dunn *et al.* A method to identify and analyze biological programs through automated reasoning. *NPJ systems biology and applications*, 2:16010, 2016.

Guidelines for cell-type heterogeneity quantification based on a comparative analysis of reference-free DNA methylation deconvolution software

Clémentine Decamps[1], Yuna Blum[2] and Magali Richard[1]

[1] Laboratory TIMC-IMAG, UMR 5525, Univ. Grenoble Alpes, 38700, Grenoble, France
[2] Programme Cartes d'Identité des Tumeurs (CIT), Ligue Nationale Contre le Cancer, 75014, Paris, France

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

Paper Reference: **Decamps *et al.* (2020) Guidelines for cell-type heterogeneity quantification based on a comparative analysis of reference-free DNA methylation deconvolution software, BMC Bioinformatics 21, 16 (2020).**
https://doi.org/10.1186/s12859-019-3307-2

1 Background

Each tumor is constituted of different cell types, in different proportions. This *cell-type heterogeneity* should be considered in cancer studies as it plays a significant role in tumor progression and response to chemotherapy [1]. A cost-effective way to infer the cell-type composition is to rely on computational deconvolution methods to obtain an individual profile from the global DNA methylation of surgical specimens. Recently, several "reference-free" algorithms have been proposed to estimate tumor cell-type heterogeneity from bulk DNA methylation samples [2,3,4], but a comparative evaluation of the performance of these methods is still lacking.

2 Results

First, we used simulations to evaluate several computational pipelines based on the software packages RefFreeEWAS [2], MeDeCom [3] and EDec [4]. We identified that *accounting for confounders* and *feature selection* of more informative probes decrease very significantly the deconvolution error. The choice of the *number of estimated cell types* was also highlighted as a critical step, and we recommended the Cattell's rule based on the scree plot to determine it. Once the pre-processing steps achieved, the three deconvolution methods provided comparable results.

Then, we compared the algorithms' performance depending on simulations parameters, such as the inter-sample variation of cell-type proportions or the number of samples. Based on all these results, we developed a benchmark pipeline for the inference of cell-type proportions and implemented it in the *R package medepir*.

Finally, we applied this pipeline on the lung cancer DNA methylation data of The Cancer Genome Atlas, and observed that the immune cell proportions we obtained are similar to those estimated by the reference-based EpiDISH algorithm [5] or by the ESTIMATE algorithm [6] using RNA-seq profile.

References

[1] Ash A Alizadeh, *et al.*, Toward understanding and exploiting tumor heterogeneity. *Nat Med*. 21:846–53, 2015.
[2] Eugene A Houseman, *et al.*, Reference-free deconvolution of DNA methylation data and mediation by cell composition effects. *BMC Bioinformatics*. 17:259, 2016.
[3] Pavlo Lutsik, *et al.*, MeDeCom: discovery and quantification of latent components of heterogeneous methylomes. *Genome Biol. BioMed Central*. 18:55, 2017.
[4] Vitor Onuchic, *et al.*, Epigenomic Deconvolution of breast tumors reveals metabolic coupling between constituent cell types. *Cell Rep*. 17:2075–86, 2016.
[5] Shijie C Zheng, *et al.*, A novel cell-type deconvolution algorithm reveals substantial contamination by immune cells in saliva, buccal and cervix. *Epigenomics*. 10:925–40, 2018.
[6] Kosuke Yoshihara, *et al*., Inferring tumour purity and stromal and immune cell admixture from expression data. *Nat Commun*. 4:2612, 2013.

Paper 65

Chromosight: a computer vision based program for pattern recognition in chromosome contact maps

Cyril Matthey-Doret[1,2], Lyam Baudry[1,2], Axel Breuer[1,3], Rémi Montagne[1], Nadège Guiglielmoni[1,4], Vittore Scolari[1], Etienne Jean[1], Arnaud Campeas[3], Philippe-Henri Chanut[3], Edgar Oriol[3], Adrien Meot[3], Laurent Politis[3], Antoine Vigouroux[5], Pierrick Moreau[1], Romain Koszul[1] and Axel Cournac[1]

[1] Institut Pasteur, Unité Régulation Spatiale des Génomes, UMR 3525, CNRS, Paris, F-75015, France
[2] Sorbonne Université, Collège Doctoral, F-75005, Paris, France
[3] ENGIE Global Energy Management, 92930, Paris, France
[4] Service Evolution Biologique et Ecologie, Avenue F.D. Roosevelt 50, 1050, Brussels, Belgium
[5] Synthetic Biology Laboratory, Institut Pasteur, 75015, Paris, France

Corresponding author: rkoszul@pasteur.fr,acournac@pasteur.fr

Abstract *Chromosomes of all species studied so far display a variety of higher-order organizational features such as domains, loops, or compartments. Many of these structures have been characterized from the genome-wide contact maps generated by chromosome conformation capture approaches (Hi-C, ChIA-PET,...). Indeed, DNA 3D structures translate as distinct patterns visible on these maps. We developed* Chromosight*, an algorithm based on computer vision approaches that automatically detect and quantify any type of pattern in contact data. Chromosight detects 3 times as many patterns as existing programs, while being faster and fit to any genome, including small, compact ones. Chromosight is user-friendly and can be extended to user-provided patterns. We validated the program by applying it to a variety of chromosomal structures found in mammals. Code and documentation: https://github.com/koszullab/chromosight*

Keywords Domain borders, Genomics, Loops, Hi-C, Detection

Introduction

Proximity ligation derivatives of the chromosome conformation capture (3C) approach [1] such as Hi-C [2] have unveiled a wide variety of chromatin 3D arrangements of potential interest regarding chromosome metabolic processes. Indeed, these approaches reveal the average contact frequencies between DNA segments within a genome, computed over hundreds of thousands of cells. These frequencies reflect the relative spatial distances separating these regions. In all species studied so far, chromosomes are sub-divided into sub-Mb domains. In mammals, topologically associating domains (TADs) are relatively stable self-interacting regions formed and maintained by the action of the structural maintenance of chromosomes (SMC) protein complex cohesin [3,4,5,6]. TADs have been proposed to emerge from a loop extrusion mechanism, in which cohesins would enlarge DNA loops between two roadblocks along the chromosomes. These roadblocks are formed by the CCCTC-binding factor (CTCF) [7]., enriched at TADs borders. A number of experimental and computational studies suggest that TADs may serve as scaffolds for gene regulation [8] Chromatin loops connecting distant loci across the genome (from a few kb to several Mb) are also common features of chromosome architecture and have been detected by Hi-C along yeast metaphase [9] and mammalian interphase chromosomes [10]. In mammals, these loops frequently bridge CTCF-binding sites, at the extremities of TADs, and are dependent on cohesin. The regulation of cohesin-dependent loops appears to be conserved from yeast to mammals, suggesting a ubiquitous mechanism that evolved to promote different functions [9].

Most structural features can be identified by eye on a Hi-C contact map. This identification is sometimes carried out manually, which may prove unwieldy or impractical for large or noisy datasets. Several methods have been developed to identify specifically looping interactions in Hi-C contact maps. For instance, diffHiC [11] looks for contact enrichments between pairs of loci. Other tools explicitly look for loop patterns using tailored rules: HiCExplorer [12] computes statistical distributions from the Hi-C contacts and looks for groups of outlier pixels forming neighbourhoods, while tools like HOMER [13] or HiCCUPS [10] compare the intensity of each pixel with a surrounding region. However, most of

these tools remain perfectible. First, they were developed to investigate loops in humans, *e.g.* discrete dots positioned at relatively large distances from the main diagonal of Hi-C contact maps, and very different from loops found so far in the smaller, more compact genomes of bacteria and fungi. Second, they miss many loops otherwise clearly visible by eye and hence suffer from a low detection rate. Most of these tools are also computationally intensive and require either a dedicated GPU (HiCCUPS) or a long run-time (*e.g.* HOMER). Some groups have recently started to use kernel convolutions to tackle the latter limitation. Notably, the cooltools suite (https://github.com/mirnylab/cooltools) has implemented a "dot finder" algorithm that uses the same surrounding regions method as HiCCUPS, but uses kernel convolution instead of explicit comparisons to speed up operations.

Here we introduce *Chromosight*, a program that automatically detects generic patterns in chromosome contact maps, with a specific focus on chromosomal loops and domain borders. Chromosight is a user-friendly python package, with minimal installation requirements. It can be applied to any contact map, independently of species, protocol or genome size. The source code is available on github `https://www.github.com/koszullab/chromosight` and our implementation is readily available on PyPi and bioconda as a standalone package. We benchmarked its precision and recall rate on simulated datasets and compared it to existing algorithms, showing it outperforms all other available programs. It is also markedly faster than most of these. Importantly, it works well on any genome and any pattern. The approach can easily be extended to user-defined structures visible on a contact map, such as cohesin injection points, or centromere clustering.

Results

Presentation of the algorithm

Chromosight takes a single, whole-genome contact map as an input in `cool` or `bedgraph2d` format, and starts by pre-processing each chromosome's submatrix to enhance local variations in the signal (Fig. 1a, methods). Intra-chromosomal contacts above a user-defined distance are discarded to constrain the analysis to relevant scales and improve performance. The core task of the Chromosight's algorithm consists in detecting a given template (e.g. loop or border kernel) within an image (i.e. the Hi-C matrix). This task is known as template matching and has been commonplace for a long time in the computer vision community [14]. Like most template matching procedures, Chromosight proceeds in 2 steps: 1) a correlation step where each sub-image is correlated to the template and 2) a selection step where sub-images with highest correlation values are labelled as template representations.

Convolution algorithms are often used in computer vision where images are typically dense. Hi-C contact maps, on the other hand, are extremely sparse. Chromosight's convolution algorithm is therefore designed to be fast and memory efficient on sparse matrices. For selecting contiguous regions of high correlation values, Chromosight uses connected component labelling (CCL). By converting the thresholded correlation map into a sparse adjacency graph, Chromosight can take advantage of an existing CCL implementation optimized for graphs to minimize both running time and memory usage.

Unlike other tools which rely on tailored scoring methods for each pattern type (*e.g.* Arrowhead for TAD detection [10]), Chromosight uses a single algorithm to detect built-in (loops, borders, hairpins...) or user-defined patterns. Regardless of the pattern, each detected instance is associated with a score (Pearson correlation), facilitating the interpretation of the result.

We assessed the performances of Chromosight for loop detection by benchmarking it against 4 existing programs on synthetic Hi-C data (Methods). We found that Chromosight has comparable precision (proportion of false positive calls) to state-of-the-art algorithms while having much higher recall rates (higher proportion of true positives detected) (Fig. 1b). Moreover, Chromosight's speed is comparable to the fastest tools available. For instance, on a machine with a 12 threads Intel i7-8700k CPU at 4.7GHz, Chromosight took 5 minutes and 6.5GB of RAM to perform loop detection at up to 20Mbp interaction distance on a human matrix with 249M contacts at a resolution of 10kb [15].

Chromosight can run either in the aforementioned detection mode, or in quantification mode (*quantify*). Chromosight *quantify* takes an input set of coordinates and returns their Pearson coefficients with a desired kernel. We used *quantify* to precisely measure the spatial scales at which cohesin loops

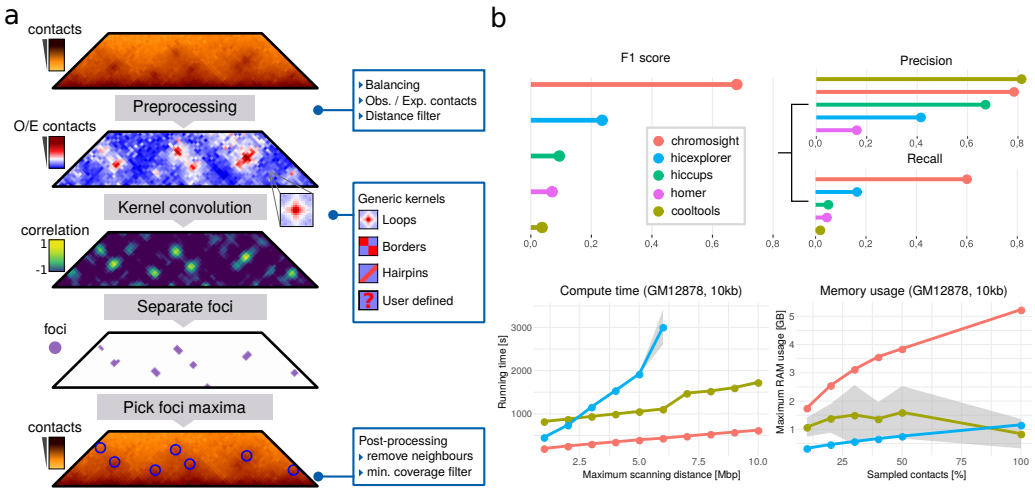


Fig. 1. Chromosight algorithm workflow and benchmark. a, Matrix preprocessing involves normalization balancing followed by the computation of observed / expected contacts. Only contacts between bins separated by a user-defined maximum distance are considered. The preprocessed matrix is then convolved with a kernel representing the pattern of interest. For each pixel of the matrix, a Pearson correlation coefficient is computed between the kernel and the surrounding window. A threshold is applied on the coefficients and a connected component labelling algorithm is used to separate groups of pixels (*i.e.* foci) with high correlation values (Methods). For each focus, the coordinate with the highest correlation value is used as the pattern coordinate. Coordinates located in poorly covered regions are discarded (Methods). **b,** Comparison of Chromosight with different loop callers. Top: F1 score, Precision and Recall score assessed on labelled synthetic Hi-C data (Methods). Higher is better. Bottom: performance of the different algorithms. Run-time and memory usage according to maximum scanning distance and the amount of downsampled contact events, respectively. The performance benchmark was run 5 times on data from human lymphoblastoid cell line (GM12878) Hi-C maps [15]. Means and standard deviations (grey areas) are plotted.

act. This allows us to compute what a loop "spectra", *i.e.* loop scores for pairs of cohesin peaks separated by increasing genomic distances (see Methods).

Detection and quantification in mammals

We applied Chromosight on published human and mouse Hi-C data generated in different laboratories. First, we searched for loops, borders, and hairpins in genome-wide contact maps generated from lymphoblastoids (GM12878) [15] (**Fig 2a**).

Loop detection yielded more than 18,000 occurrences (**Fig 2b**). The majority of the loop basis ($\simeq$55%, $p < 10^{-16}$, Fisher test)) fall into loci enriched in cohesin subunit Rad21, as expected[16]. Multiple loops often originate from a same basis, reflecting either an heterogeneity of structures in the population, the formation of rosette-like structures, or both.

We then compared loop scores between wild type (WT) and mutant Hela cells where cohesin was depleted [16]. Loop detection in WT Hela cells yields similar results as with GM12878 cells, with 15,600 loops. The loop scores of these WT borders in cohesin-depleted contact maps show the disappearance of the loop signal in the absence of cohesin (**Fig 2c**). This analysis confirms that the loops identified by Chromosight in WT contact maps are indeed biologically relevant structures, and not unwanted signal.

To measure more precisely the spatial distributions of cohesin loops, we computed the loop spectrum on pairs of cohesin peaks (**Fig 2d**) using Hi-C data of Hela synchronized cells released from mitosis into G1 [17]. At the beginning of the kinetics, the spectrum is flat with no significant loop scores. As cells progress through mitosis and re-enter G1, a loop signal clearly emerges with a peak

at 130 kb, a distance similar in other cell types (data not shown). The loop spectra show a secondary, weaker peak at 260 kb, suggesting a structural model with regular loop structures.

The 11,389 borders detected by Chromosight appeared enriched in CTCF deposition sites ($\simeq$40%, $P < 10^{-16}$, Fisher test) [18], showing that the detection of domain borders based on pattern matching is also relevant to capture biological features already identified with algorithms based on other approaches (ex: segmentation [19]).

Finally, Chromosight detected 1,700 hairpin like patterns (**Fig 2b**) in GM12878. Interestingly, the chromosome coordinates for this detected group are enriched in NIPBL (2 fold effect, $p < 10^{-16}$, Fisher test), a cohesin loading factor. The hairpin-like structures detected by Chromosight could therefore be interpreted as injection points for cohesin, an hypothesis of potential interest regarding the regulation of genome organization.

We analyzed Hi-C data from cells depleted of NIPBL to test for its implication in the detected hairpin patterns [20]. The hairpin patterns almost disappear in this mutant (average hairpin scores decreased from 0.40 to 0.03, $P < 10^{-16}$) (**Fig 2e**).

Discussion

Chromosight is a fast program that detects any type of pattern in chromosome contact maps for any genome. We have shown that it outperforms all other programs at reliably detecting a large number of DNA loops. In addition, it allows the user to search for any type of pattern, and additional structures could easily be added such as stripes, or patterns corresponding to large-scale structural variants (*e.g.* inversions, translocations). Chromosight could also be used to facilitate the detection of genomic misassemblies from the Hi-C signal, to help their correction and polishing [21].

Chromosight's execution time (a few minutes for human datasets) as well as its compatibility with widespread contact data formats (`cool` and `bedgraph2`) allows the exploratory analysis of large amounts of contact data. It also successfully identifies DNA loops in compact genomes such as yeast or *B. subtilis* (not shown). We envision that, as more species are investigated through Hi-C, and data resolution increases, new spatial structures will be unveiled. The user-friendly, flexible approach of Chromosight makes it a versatile tool that can easily be adapted and applied to different types of experimental data and provides a computational and statistical framework for the discovery of new principles governing chromosome architecture.

Methods

Simulation of Hi-C matrices

Simulated matrices were generated using a bootstrap strategy based on Hi-C data from mitotic *S. cerevisiae* [9] at 2kb resolution. Three main features were extracted from the yeast contact data: the probability of contact as a function of the genomic distance ($P(s)$), the positions of borders detected by HicSeg [19] and positions of loops detected manually on chromosome 5. Positions from loops and borders were then aggregated into pileups of 17x17 pixels. We generated 2000 simulated matrices of 289x289 pixels. A first probability map of the same dimension is generated by making a diagonal gradient from $P(s)$ representing the polymer matrix. For each of the 2000 generated matrices, two additional probability maps are generated. The first by placing several occurrences of the border pileup on the diagonal, where the distance between borders follows a normal distribution fitted on the experimental coordinates. The second probability map is generated by adding the loop kernel 2-100 pixels away from the diagonal with the constraint that it must be aligned vertically and horizontally with border coordinates. For each generated matrix, the product of the P(s), borders and loops probability maps is then computed and used as a probability law to sample contact positions while keeping the same number of reads as the experimental map.

Benchmarking

To benchmark precision, recall and F1 score, the simulated Hi-C dataset with known loop coordinates were used. Each algorithm was run with a range of 60-180 parameter combinations on the

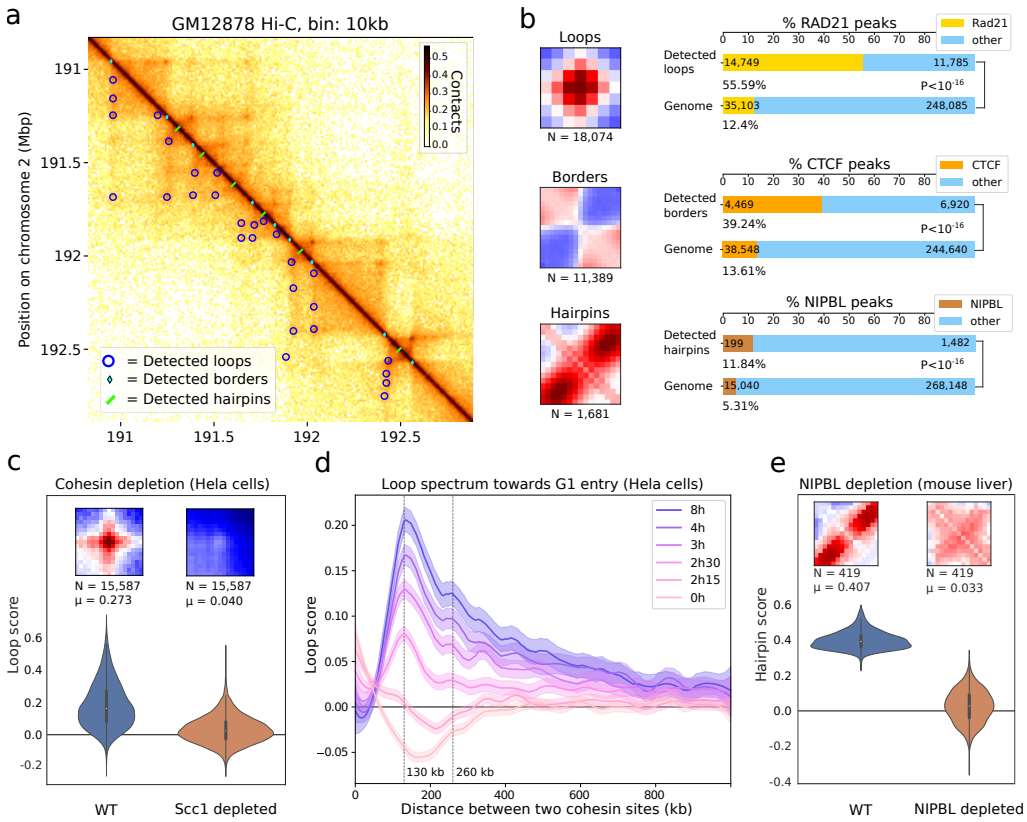


Fig. 2. Chromosight on mammalian Hi-C contact maps. a, Magnification of human chromosome 2 contact map (bin: 10 kb, [15]). The positions of the loops, borders and hairpins detected by Chromosight are indicated. **b,** Pileup (*i.e.* element-wise median) plots of windows centered on the detected loops, borders and hairpins. N: number of occurrences. Bar plots on the right panel show the enrichment in Rad21, CTCF, and NIPBL at the coordinates of loops' bases, borders, and hairpin's bases, respectively. **c,** Comparison of loop score distributions in WT (*Homo sapiens*) and in mutant cells depleted in Scc1 [16] for loops detected in WT condition. Associated pileup plots of windows centered on detected loops in WT condition. μ: mean of Pearson coefficients. **d,** Loop spectra of the loop scores for pairs of Rad21 deposition sites separated by increasing distances, at different time points during release from mitosis into G1 [17]. **e,** Comparison of hairpin score distributions in WT (*Mus musculus*, liver cells) and in mutant cells depleted in NIPBL [20] for hairpins detected in the WT condition. Associated pileup plots of windows centered on detected hairpins in WT condition.

2000 simulated matrices and F1 score was calculated on the ensemble of results for each parameter combination separately. For each software, scores used in the final benchmark (Fig. 1) are those from the parameter combination that yielded the highest F1 score.

For the performance benchmark, HiCCUPS and HOMER were excluded. The former because it runs on GPU, and the latter because it uses genomic alignments as input and is much slower. The dataset used is a published high coverage Hi-C library [15] from human lymphoblastoid cell lines (GM12878). To compare RAM usage across programs, this dataset was downsampled at 10, 20, 30, 40 and 50% contacts and the maximum scanning distance was set to 2Mbp. To compare CPU time, all programs were run on the full dataset, at different maximum scanning distances, with a minimum scanning distance of 0 and all other parameters left to default. All programs were run on a single thread, on a Intel(R) Core(TM) i7-8700K CPU at 3.70GHz with 32GB of available RAM.

Preprocessing of Hi-C matrices

Prior to detection, Chromosight balances the whole genome matrix using the ICE algorithm [22] to account for Hi-C associated biases. For each intrachromosomal matrix, the observed/expected contact ratios are then computed by dividing each pixel by the mean of its diagonal. This erases the diagonal gradient due to the power-law relationship between genomic distance and contact probability, thus emphasizing local variations.

Calculation of Pearson coefficients

The contact map can be considered an image IMG_{CONT} where the intensity of each pixel $\text{IMG}_{\text{CONT}}[i, j]$ represents the contact probability between loci i and j of the chromosome. In that context, each pattern of interest can be considered a template image IMG_{TMP} with M_{TMP} rows and N_{TMP} columns.

The correlation operation consists in sliding the template (IMG_{TMP}) over the image (IMG_{CONT}) and measuring, for each template position, the similarity between the template and its overlap in the image. We used the Pearson correlation coefficient as a the measure of similarity between the two images. The output of this matching procedure is an image of correlation coefficients IMG_{CORR} such that

$$\text{IMG}_{\text{CORR}}[i, j] = Corr\left(\text{IMG}_{\text{CONT}}[i : i + M_{\text{TMP}}, j : j + N_{\text{TMP}}],\ \text{IMG}_{\text{TMP}}\right) \tag{1}$$

where the correlation operator $Corr(\cdot, \cdot)$ is defined as

$$Corr\left(\text{IMG}_{\text{X}}, \text{IMG}_{\text{Y}}\right) = \frac{cov(\text{IMG}_{\text{X}}, \text{IMG}_{\text{Y}})}{std(\text{IMG}_{\text{X}}) \cdot std(\text{IMG}_{\text{Y}})} \tag{2}$$

$$= \frac{\sum_i \sum_j (\text{IMG}_{\text{X}}(i, j) - \overline{\text{IMG}_{\text{X}}}) \cdot (\text{IMG}_{\text{Y}}(i, j) - \overline{\text{IMG}_{\text{Y}}})}{\sqrt{\sum_i \sum_j (\text{IMG}_{\text{X}}(i, j) - \overline{\text{IMG}_{\text{X}}})^2} \cdot \sqrt{\sum_i \sum_j (\text{IMG}_{\text{Y}}(i, j) - \overline{\text{IMG}_{\text{Y}}})^2}} \tag{3}$$

where $\overline{\text{IMG}} = \frac{1}{M \cdot N} \sum_i \sum_j \text{IMG}(i, j)$.

Separation of high-correlation foci

Selection is done by localizing specific local maxima within IMG_{CORR}. We proceeded as follows: first, we discard all points (i, j) where $\text{IMG}_{\text{CORR}}[i, j] < \tau_{\text{CORR}}$. An adjacency graph A_{dxd} is then generated from the d remaining points. The value of $A[i, j]$ is a boolean indicating the (4-way) adjacency status between the i^{th} and j^{th} nonzero pixels. The scipy implementation of the CCL algorithm for sparse graphs is then used on D to label the different contiguous foci of nonzero pixels. Foci with less than two pixels are discarded. For each focus, the pixel with the highest coefficient is determined as the pattern coordinate.

Patterns are then filtered out if they overlap too many empty pixels or are too close from another detected pattern. The remaining candidates in IMG_{CORR} are scanned by decreasing order of magnitude:

every time a candidate is appended to the list of selected local maxima, all its neighboring candidates are discarded. The proportion of empty pixels allowed and the minimum separation between two patterns are also user defined parameters.

Author contributions

All authors contributed to the design of the algorithm. CMD, AB, LB, AC implemented it. CMD, NG, RM compared Chromosight to other programs. LB and AC designed the data simulations. CMD, PM, RK and AC analysed biological data and interpret results. CMD, RK and AC wrote the manuscript.

Competing financial interests

The authors declare no competing financial interests.

Acknowledgements

This work was initiated as a hackathon challenge regrouping Institut Pasteur scientists and ENGIE engineers. We thank all the people that made this event possible, especially Anne-Gaëlle Coutris, Romain Tchertchian and Olivier Gascuel. Frédéric Beckouët and all the members of Spatial Regulation of Genomes unit are thanked for stimulating discussions and feedbacks. AB works within the framework of a "Mécénat Compétence" contract of the company ENGIE. CMD is supported by the Pasteur - Paris University (PPU) International PhD Program. NG is supported by the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement 764840. This research is supported by funding from Agence Nationale de la Recherche (ANR JCJC 2019, "Apollo") to AC and funding to RK from the European Research Council under the Horizon 2020 Program (ERC grant agreement 260822).

References

[1] Job Dekker, Karsten Rippe, Martijn Dekker, and Nancy Kleckner. Capturing chromosome conformation. *Science*, 295:1306–1311, 2002.

[2] Erez Lieberman-Aiden, Nynke L. van Berkum, Louise Williams, Maxim Imakaev, Tobias Ragoczy, Agnes Telling, Ido Amit, Bryan R. Lajoie, Peter J. Sabo, Michael O. Dorschner, Richard Sandstrom, Bradley Bernstein, M. A. Bender, Mark Groudine, Andreas Gnirke, John Stamatoyannopoulos, Leonid A. Mirny, Eric S. Lander, and Job Dekker. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science*, 326(5950):289–293, Oct 2009.

[3] Suhas S P Rao, Su-Chen Huang, Brian Glenn St Hilaire, Jesse M Engreitz, Elizabeth M Perez, Kyong-Rim Kieffer-Kwon, Adrian L Sanborn, Sarah E Johnstone, Gavin D Bascom, Ivan D Bochkov, Xingfan Huang, Muhammad S Shamim, Jaeweon Shin, Douglass Turner, Ziyi Ye, Arina D Omer, James T Robinson, Tamar Schlick, Bradley E Bernstein, Rafael Casellas, Eric S Lander, and Erez Lieberman Aiden. Cohesin loss eliminates all loop domains. *Cell*, 171(2):305–320.e24, October 2017.

[4] Elphège P Nora, Bryan R Lajoie, Edda G Schulz, Luca Giorgetti, Ikuhiro Okamoto, Nicolas Servant, Tristan Piolot, Nynke L van Berkum, Johannes Meisig, John Sedat, Joost Gribnau, Emmanuel Barillot, Nils Blüthgen, Job Dekker, and Edith Heard. Spatial partitioning of the regulatory landscape of the x-inactivation centre. *Nature*, 485(7398):381–5, April 2012.

[5] Wouter de Laat and Denis Duboule. Topology of mammalian developmental enhancers and their regulatory landscapes. *Nature*, 502(7472):499–506, October 2013.

[6] Jesse R. Dixon, Siddarth Selvaraj, Feng Yue, Audrey Kim, Yan Li, Yin Shen, Ming Hu, Jun S. Liu, and Bing Ren. Topological domains in mammalian genomes identified by analysis of chromatin interactions. *Nature*, 485(7398):376–380, May 2012.

[7] Elzo de Wit, Erica S M Vos, Sjoerd J B Holwerda, Christian Valdes-Quezada, Marjon J A M Verstegen, Hans Teunissen, Erik Splinter, Patrick J Wijchers, Peter H L Krijger, and Wouter de Laat. Ctcf binding polarity determines chromatin looping. *Mol. Cell*, 60(4):676–84, November 2015.

[8] Daniel M. Ibrahim and Stefan Mundlos. Three-dimensional chromatin in disease: What holds us together and what drives us apart? *Current Opinion in Cell Biology*, 64:1 – 9, 2020.

[9] Lise Dauban, Rémi Montagne, Agnès Thierry, Luciana Lazar-Stefanita, Nathalie Bastié, Olivier Gadal, Axel Cournac, Romain Koszul, and Frédéric Beckouët. Regulation of cohesin-mediated chromosome folding by eco1 and other partners. *Mol. Cell*, January 2020.

Paper 65

[10] Suhas S P Rao, Miriam H Huntley, Neva C Durand, Elena K Stamenova, Ivan D Bochkov, James T Robinson, Adrian L Sanborn, Ido Machol, Arina D Omer, Eric S Lander, and Erez Lieberman Aiden. A 3d map of the human genome at kilobase resolution reveals principles of chromatin looping. *Cell*, 159(7):1665–80, December 2014.

[11] Aaron T L Lun and Gordon K Smyth. diffhic: a bioconductor package to detect differential genomic interactions in hi-c data. *BMC Bioinformatics*, 16:258, August 2015.

[12] Fidel Ramírez, Vivek Bhardwaj, Laura Arrigoni, Kin Chung Lam, Björn A Grüning, José Villaveces, Bianca Habermann, Asifa Akhtar, and Thomas Manke. High-resolution tads reveal dna sequences underlying genome organization in flies. *Nat Commun*, 9(1):189, 01 2018.

[13] Sven Heinz, Lorane Texari, Michael G B Hayes, Matthew Urbanowski, Max W Chang, Ninvita Givarkes, Alexander Rialdi, Kris M White, Randy A Albrecht, Lars Pache, Ivan Marazzi, Adolfo García-Sastre, Megan L Shaw, and Christopher Benner. Transcription elongation can affect genome 3d structure. *Cell*, 174(6):1522–1536.e22, 09 2018.

[14] Roberto Brunelli. *Template Matching Techniques in Computer Vision: Theory and Practice*. Wiley Publishing, 2009.

[15] Jay Ghurye, Arang Rhie, Brian P Walenz, Anthony Schmitt, Siddarth Selvaraj, Mihai Pop, Adam M Phillippy, and Sergey Koren. Integrating hi-c links with assembly graphs for chromosome-scale assembly. *PLoS Comput. Biol.*, 15(8):e1007273, 08 2019.

[16] Gordana Wutz, Csilla Várnai, Kota Nagasaka, David A Cisneros, Roman R Stocsits, Wen Tang, Stefan Schoenfelder, Gregor Jessberger, Matthias Muhar, M Julius Hossain, Nike Walther, Birgit Koch, Moritz Kueblbeck, Jan Ellenberg, Johannes Zuber, Peter Fraser, and Jan-Michael Peters. Topologically associating domains and chromatin loops depend on cohesin and are regulated by ctcf, wapl, and pds5 proteins. *EMBO J.*, 36(24):3573–3599, 12 2017.

[17] Kristin Abramo, Anne-Laure Valton, Sergey V Venev, Hakan Ozadam, A Nicole Fox, and Job Dekker. A chromosome folding intermediate at the condensin-to-cohesin transition during telophase. *Nat. Cell Biol.*, 21(11):1393–1402, 11 2019.

[18] The ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489(7414):57–74, 2012.

[19] Celine Lévy-Leduc, Maud Delattre, Tristan Mary-Huard, and Stephane Robin. Two-dimensional segmentation for analyzing hi-c data. *Bioinformatics*, 30(17):i386–i392, 2014.

[20] Wibke Schwarzer, Nezar Abdennur, Anton Goloborodko, Aleksandra Pekowska, Geoffrey Fudenberg, Yann Loe-Mie, Nuno A Fonseca, Wolfgang Huber, Christian H Haering, Leonid Mirny, and Francois Spitz. Two independent modes of chromatin organization revealed by cohesin removal. *Nature*, 551(7678):51–56, 11 2017.

[21] Hervé Marie-Nelly, Martial Marbouty, Axel Cournac, Jean-François Flot, Gianni Liti, Dante Poggi Parodi, Sylvie Syan, Nancy Guillén, Antoine Margeot, Christophe Zimmer, and Romain Koszul. High-quality genome (re)assembly using chromosomal contact data. *Nat Commun*, 5:5695, 2014.

[22] Maxim Imakaev, Geoffrey Fudenberg, Rachel Patton McCord, Natalia Naumova, Anton Goloborodko, Bryan R Lajoie, Job Dekker, and Leonid A Mirny. Iterative correction of hi-c data reveals hallmarks of chromosome organization. *Nature methods*, 9(10):999–1003, 2012.

Paper 66

Bringing ABC inference to the machine learning realm : AbcRanger, an optimized random forests library for ABC

François-David Collin[1], Arnaud Estoup[2], Jean-Michel Marin[1] and Louis Raynal[1]
[1] Université de Montpellier, CNRS, IMAG UMR 5149, Montpellier, France
[2] CBGP, INRA, CIRAD, IRD, Montpellier SupAgro, Univ. Montpellier, Montpellier, France

Corresponding author: Francois-David.Collin@umontpellier.fr@umontpellier.fr

Abstract *The AbcRanger library provides methodologies for model choice and parameter estimation based on fast and scalable* Random Forests*, tuned to handle large and/or high dimensional datasets. The library, initially intended for the population genetics ABC framework DIYABC, has been generalized to any ABC reference table generator.*

At first, computational issues were encountered with the reference ABC-Random Forest. Those issues have been diagnosed by us as friction between "strict" Machine Learning setup and ABC context, and this incited us to modify the C++ implementation of state-of-the-art random forests, ranger, to tailor it for ABC needs: potentially "deep" decision trees are not stored in memory anymore, but are processed by batches in parallel.

We focused on memory and thread scalability, ease of use (minimal hyperparameter set). R and python interfaces are provided.

Keywords Approximate Bayesian Computation, Random Forests, Model Choice, Parameter Estimation, C++, Python, R

1 Introduction : challenges for ABC from Population Genetics

In the context of recent advances in population genetics the *number of simulated data* in a ABC context could reach over the hundred of thousands (10^{e5}) mark. Similarly, with the advent of multi-population summary statistics in this domain (see [1]) the number of summary statistics computed by ABC (as covariables) could range from several hundred to tens of thousands (scenario with several populations and combinatorial "explosion" of multi-population statistics). Moreover, not all summary statistics are relevant, and traditional variable selection methods still have to be tuned for each case in an *ad hoc* manner. From both row and column inflation point of view, classical methods for ABC (k-nn and local methods) doesn't cope very well with this situation.

[2] and [3] proposed a novel approach, coined as *ABC-random forest* or *ABC-RF*, which relies on *Random Forests* to provide tractable and efficient methodologies, for both model choice and parameter estimation.

2 First building block : ABC simulations to generate the Random Forest training database

In a Bayesian context, when the likelihood function is too complex or untractable, several *likelihood-free* methods are available to approximate it, including Approximate Bayesian Computation (ABC) [4]. Given an observed data, the basic idea of ABC is to approximate the likelihood of a parametrized model with selected simulations, by comparing the observed data and simulated ones via computed *summary statistics*. The table of summary statistics for simulated data is called *the reference table* (see fig. 1). It corresponds to the so called "training dataset" in Machine Learning terminology.

2.1 ABC-RF posterior methodologies

2.1.1 Model Choice Given an observed data, and several (parametrized) models, the purpose is to estimate the best model to fit our data. A reference table combining summary statistics of simulated samples (particles) is generated from each model (models are sampled according a prior distribution, e.g. by penalizing the model complexity). A *Model Choice* methodology is an inference method which takes this reference table, the observed data and *infers* the best fitted model for this data, along with

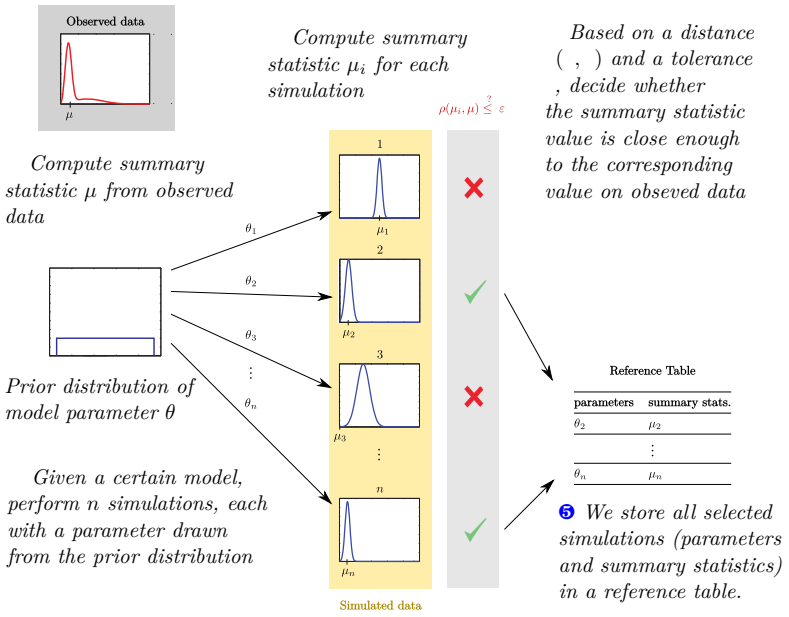


Fig. 1. ABC simulations to generate the Random Forest training database

an estimated posterior probability (the probability of the model knowing the observed data), which assesses the fitness of the predicted model.

2.1.2 Parameter Estimation Given an observed data and one parametrized model, the purpose is to infer one or several parameters for this model given the observed data. An ABC reference table is generated from the model. The *Parameter estimation* methodology is an inference method which takes this reference table, the observed data and *infers* one or several parameters, along with the usual Bayesian decorum : posterior distribution, quantiles and so on.

2.1.3 General workflow A sensible workflow is to first choose a model and then infer its parameters (see fig. 2).

3 Second build block : Random Forests

Enter the *Supervised Machine Learning* (SML) realm [5]: at the beginning lies a list a pair of input data/output data $\{x_i, y_i\}$ from and Y domains, called a *training dataset*. The objective is to *learn* the best function $f_\theta(x)$ parametrized by $\theta \in \Theta$ so that a *scalar loss function* $L : Y \times Y \quad \mathbb{R}$ is minimized on the Θ domain :

$$f_\theta = \underset{\theta}{\operatorname{argmin}}\ L(f(x_i), y_i)$$

Random Forests are based on CART, *Classification and Regression Trees*, an algorithm developed by [6].

3.1 CART

A CART is a *supervised machine learning algorithm* which essentially performs, recursively, a partitioning of the predictor space into disjoint subspaces. A prediction value is assigned to each of those subspaces (or *Leaves*). Once the partitioning is done, the result is a binary tree which could predict outcomes from an input data, either classes or continuous values, by *routing* the data to a *leaf*, whose assigned value will be used then as prediction (see fig. 3).

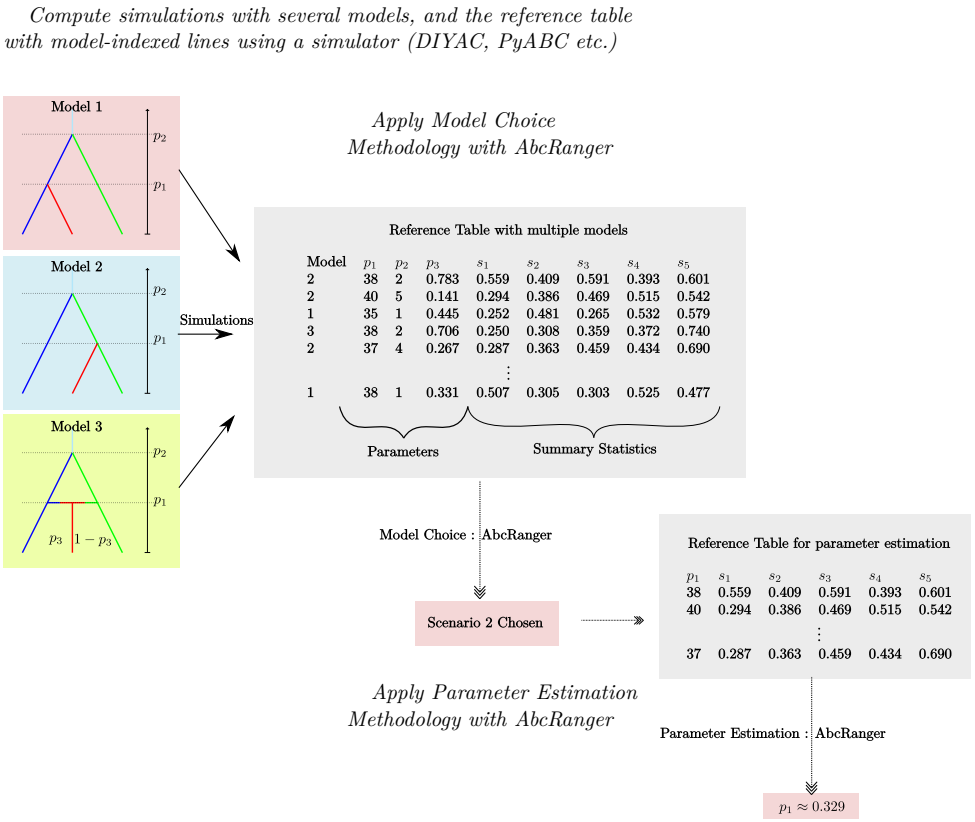


Fig. 2. Workflow with AbcRanger

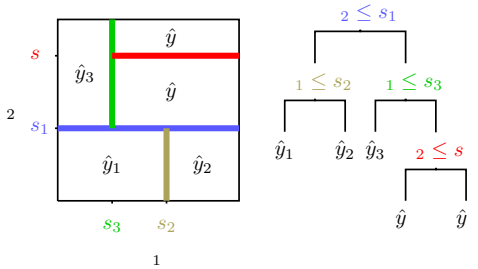


Fig. 3. An example of CART and the associated partition of the two dimensional predictor space. Each splitting condition takes the form $\leq s$ and the prediction at a leaf is denoted $\hat{y}_\ell$.

3.2 Random Forests

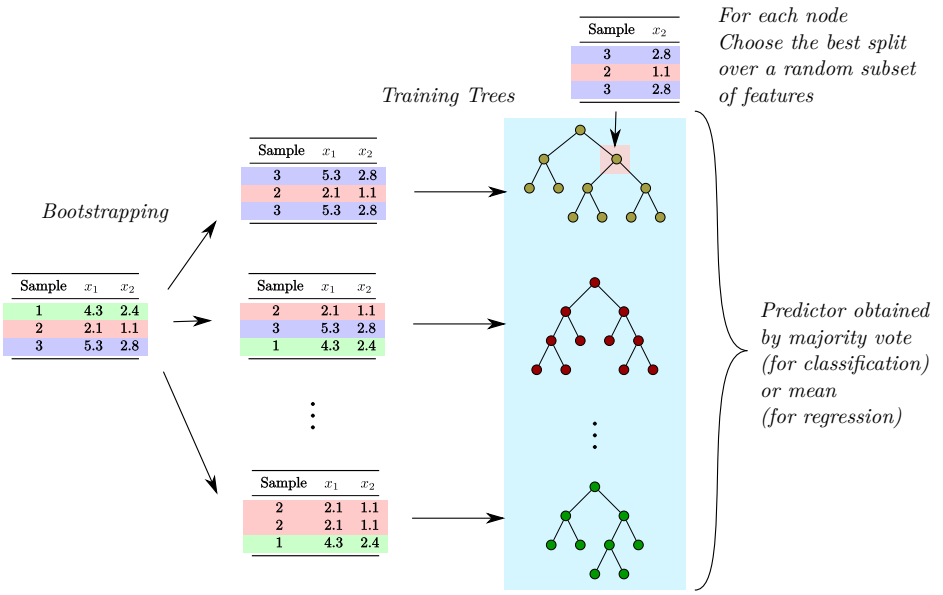


Fig. 4. Random Forest

Random Forests [7] are a three pronged extension of CART (see fig. 4). First it is an **Ensemble method** which trains a *set* of CART (not just one) and predict the outcome with the majority vote (resp. mean) of this set of trained trees for classification (resp. regression) target. Second, **bootstrapping** is applied before each tree training, i.e. training data is random sampled (with replacement). And last but not least, in a growing tree, at each node, the best split is computed on a **random subset of the features**. Those three extensions have multiple benefits; the main ones are lower variance compared to a single CART tree, due to the ensemble method, and *unbiasedness*, because of the de-correlation of the trees induced by both bootstrapping and features random sampling. Other advantages are : robustness to noise, variable importance for (almost) free, integrated cross-validation procedure (out-of-bag samples, no need to get a validation dataset), easy parallelization, very good scaling properties (both in rows and columns axes), and provides both classification and regression target.

3.3 ABC Random Forest

A reference implementation of the *ABC-Random Forest* setup is given by *abcrf* [8]. We provide here a brief description of ABC Random Forest methodologies for model choice and parameter estimation.

3.3.1 Model Choice Model Choice methodology in ABC-RF is two staged. *First a classification random forest is trained* with the models (classes) as target. The trained random forest model is evaluated on the observed data, getting votes and the best model to fit. *Second*, using the obtained random forest from the first stage, each sample from the training dataset is labeled classified/misclassified with the *out-of-bag prediction* and finally as numerical 0 or 1 for a new target. Then, a *new regression random forest* is trained on the training dataset, but this time with this new target (as continuous, non-categorical one for regression). And finally a prediction on the observed data is evaluated with the obtained random forest, and this predicted value (between 0 and 1) is a viable estimator for the *posterior probability of the chosen model*.

3.3.2 Parameter Estimation In ABC Random Forest setup, parameter estimation is limited to one parameter at a time. Choosing a parameter θ to estimate, a regression RF is trained on a reference

table generated only with the corresponding model and with the θ parameter values as target, forming the training dataset. Once trained, the regression RF is evaluated on the observed data and several outcomes are obtained, like an estimation of θ, variance, and quantiles with the help of **quantile regression forests** [9]. It is worth noting that Quantile Forests are not new forests per se but an – integrated – method to compute weights distribution of the samples, knowing an observed (or out-of-bag) data. This distribution is then used to compute quantiles, for example. Finally a set of both prior et posterior estimators is inferred from the RF predictions, for example a prior (resp. posterior) *pdf*, obtainable via standard kernel density estimation (resp. standard weighted density estimation).

3.4 Linear augmentations

As stated in [2] (resp. [3]), for Model choice (resp. parameter estimation), there is the option – enabled by default – to add linear combinations covariables to the existing summary statistics in the reference table via *Linear Discriminant Analysis* (resp. *Partial Least Squares*) [5]. By refining the "square" partitioning of the trees, this sensibly improves the prediction accuracy of Random Forests outcomes, .

3.5 Computational limitations with ABC Random Forests reference implementation

Faced with training dataset including 100 000 lines and more than 10 000 summary statistics, *abcrf* has been found growing trees over one gigabyte of memory size each. So, as typical random forests are made of 500 or 1000 trees for prediction performance, even with state of the art RF packages like [10], memory constraints are preventing completion of the training.

This issue has a longer reach than an simple implementation issue and exhibits a fundamental mismatch of objectives between "classical" supervised machine learning setup and ABC posterior methodologies. Indeed, within "pure" SML, a model (like a Random Forest) is first trained, and then used to make predictions on a potentially endless source of new data; the whole model is stored by training and loaded in memory each time for prediction purpose. However, within the ABC inference context, the SML model is only needed for specific predictions directly on one or several observed data sample(s) and out-of-bag samples. Moreover, the corresponding trained Random Forest is coupled to the generated reference table (aka the training dataset), and is by no mean meant to generalize to new data (other reference tables), let alone other model and relevant observed data: in fact storing the forest is useless. Those remarks established the need of an adaptation of random forest algorithm for ABC.

4 New implementation of Random Forest and ABC Random Forest

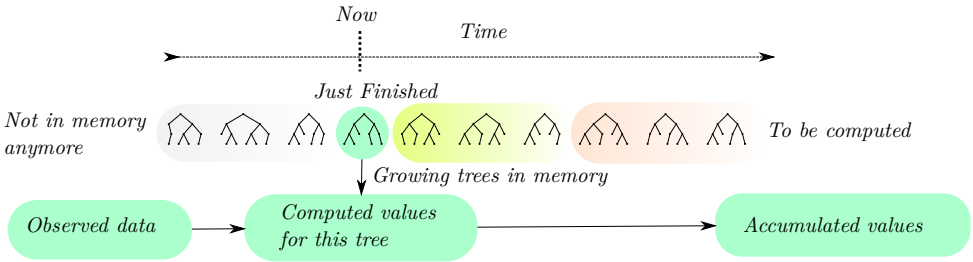


Fig. 5. Window of growing trees

Based on our own version of the core RF (written in C++) from the ranger package [10], our new implementation of Random Forest for ABC, *AbcRanger*, solves the memory constraint issue related to the deep trees. Leveraging the cumulative nature of the ensemble method, Random Forest computations are now done in a *joint grow/predict phase* for each tree, and then optimized in order to grow a limited batch of trees in memory. As illustrated by fig. 5), this means that grow/predict computations for each tree is executed in a sequential –- i.e. batch-wise – order: as now tree growing and predictions are computed in a single pass, predictions and posteriors are then stored/accumulated and each tree is finally discarded, freeing the system memory for next growing trees. The trees of

the currently processed batch are still computed in parallel to leverage nowadays ubiquitous multicore architectures.

Although this doesn't precludes the in-memory storage of the entire training dataset at once, this way of processing avoids the in-memory storage of the whole forest at no performance cost. In a very constrained memory environment, one should just have to lower the number of computing threads to keep the memory of a training batch in check. A special care has also been given to the Meinshausen's quantiles computations, completely parallelized and typically unnoticeable on multicore systems. Another advantage over *abcrf* package: methodologies are now pure C++. So, it is relatively easy to provide wrappers/interfaces to other languages than R, like Python, with the added guarantee that no copy of the reference table happens between the core C++ layer handling the methodologies, and the interfaced language providing the reference table.

4.1 A toy example application with the ELFI python package

The *ELFI* python package [11] provides a popular and flexible ABC framework, meant to integrate complex ABC and inferences pipelines. Inspired by the $Ma(2)$ toy example used by original ABC authors in [4], we used a more general $Ma(q)$ example for model choice and parameter estimation, fixing $q = 10$ in the following.

$MA(q)$ is a time series model defined by :

$$x_t = \mu + \epsilon_t - \sum_{i=1}^{q} \; _i\epsilon_{t-i}$$

For identifiability purposes the parameters should verify the following condition, roots of

$$(\;\;) = 1 - \sum_{i=1}^{q} \;\; _i \;\; ^i$$

should be strictly outside the (complex) unity disc, and this is our main prior constraint. Prior for θ_q is also sampled from an uniform distribution.

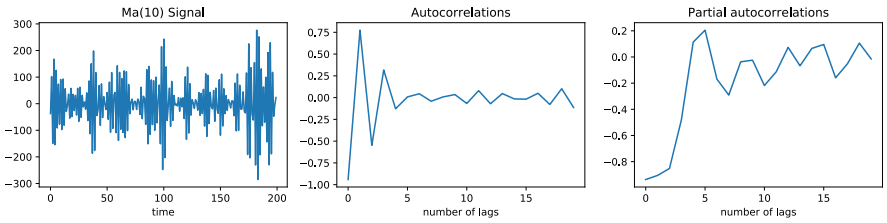


Fig. 6. Example of an $MA(10)$ model

From the generated examples of $Ma(10)$ on a 200-length signal, sampling the prior θ_{10} uniformly in the $[1, 2]$ interval, the usual row of (partial) autocorrelation features seems to be nonconclusive (see fig. 6) to discriminate between, for example $Ma(8)$, $Ma(10)$ or $Ma(12)$.

4.1.1 Model choice: (10) vs "all" ($6 \leq q \leq 16$) An ABC pipeline has been configured with *elfi*, choosing the default sampler, without rejection (option `quantile` fixed to 1). For 100 trials, priors for $Ma(10)$ are sampled and observation generated, and models to choose are from $Ma(q)$ with $6 \leq q \leq 16$. On fig. 7, the performance of the ABC-RF setup is illustrated.

Also, features coming from LDA linear augmentation are discriminative, see fig. 8 for one particular inference.

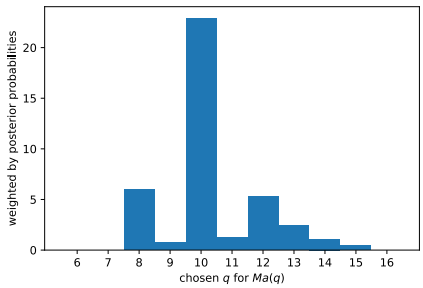


Fig. 7. Model Choice weighted histogram of inferred models: 100 $Ma(10)$ models are tried with ABC simulations followed by RF Model Choice inference (Signal length : 200 points, reftables : 2000 particles each).

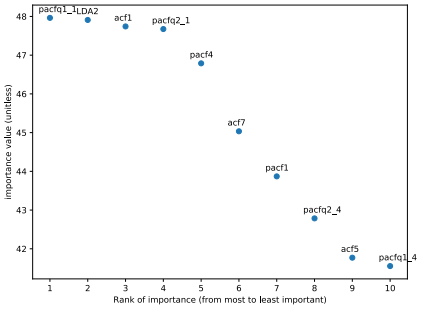


Fig. 8. 10 most ranked summary statistics, sorted by permutation importance. acf_i, *(resp.* $pacf_i$, $pacf1_i$, $pacf2_i$*)* are i-lagged autocorrelations (resp. partial autocorrelations, 0.05 and 0.95 corresponding quantiles).

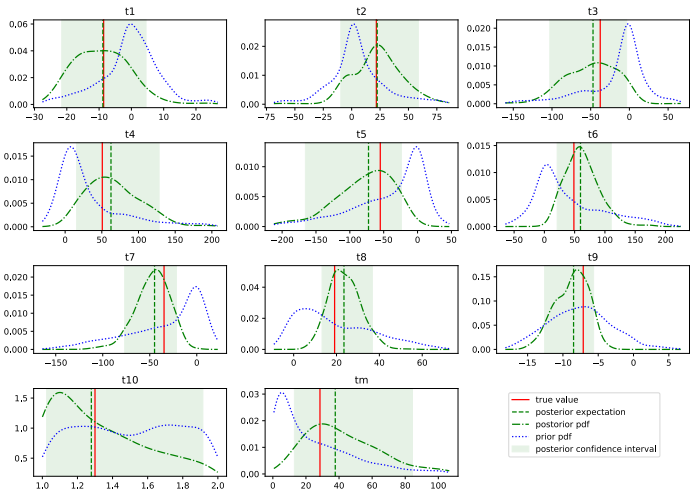


Fig. 9. Inferred posterior distributions of a $MA(10)$ model

4.1.2 Parameter Estimation For parameter estimation one $Ma(10)$ is sampled and observed, and then all parameters are inferred individually with ABC-RF methodology (whith the help of *AbcRanger* python wrapper). Results are illustrated in fig. 9. All parameters of the model were nicely estimated, and the posterior/prior distributions clearly discriminated.

5 Conclusions and perspectives

ABC-RF posterior methodologies are a clean and efficient integration of SML techniques in a model-based approach, although the main objective is not the raw predictive power *per se* like in a pure machine learning perspective, but easy to get, accurate and interpretable posteriors.

Many ideas emphasized in both posterior methodologies from [2] and [3] have strong connections with *Generalized Random Forests* framework [12] we would like to explore in order to extend our developments to other fields than population genetics.

Moreover, we intend to pursue the algorithm adaptation of Random Forests for ABC even further, at the tree level: for a growing tree, only encountered leaves should be stored for point estimates and final moments. Thus, the memory footprint of the trees becomes negligible, and their growing could finally be parallelized at full scale.

Finally, by nature of Breiman's *CART*, the computational bottleneck for random forests lies in the greedy, local split procedure at each node. To alleviate this, they are promising optimizations coming from the *Gradient Boosted Trees* community [13] and also some inspired by the *Deep Learning* one like [14].

References

[1] Valentin Hivert, Raphaël Leblois, Eric J Petit, Mathieu Gautier, and Renaud Vitalis. Measuring genetic differentiation from pool-seq data. *Genetics*, 210(1):315–330, 2018.

[2] Pierre Pudlo, Jean-Michel Marin, Arnaud Estoup, Jean-Marie Cornuet, Mathieu Gautier, and Christian P Robert. Reliable abc model choice via random forests. *Bioinformatics*, 32(6):859–866, 2015.

[3] Louis Raynal, Jean-Michel Marin, Pierre Pudlo, Mathieu Ribatet, Christian P Robert, and Arnaud Estoup. ABC random forests for Bayesian parameter inference. *Bioinformatics*, 35(10):1720–1728, 10 2018.

[4] Jean-Michel Marin, Pierre Pudlo, Christian P Robert, and Robin J Ryder. Approximate bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, 2012.

[5] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, NY, USA:, 2001.

[6] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis, 1984.

[7] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[8] Jean-Michel Marin, Louis Raynal, Pierre Pudlo, Christian P. Robert, and Arnaud Estoup. *abcrf: Approximate Bayesian Computation via Random Forests*, 2019. R package version 1.8.1.

[9] Nicolai Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7(Jun):983–999, 2006.

[10] Marvin N Wright and Andreas Ziegler. Ranger: a fast implementation of random forests for high dimensional data in c++ and r. *arXiv preprint arXiv:1508.04409*, 2015.

[11] Jarno Lintusaari, Henri Vuollekoski, Antti Kangasrääsiö, Kusti Skytén, Marko Järvenpää, Pekka Marttinen, Michael U. Gutmann, Aki Vehtari, Jukka Corander, and Samuel Kaski. Elfi: Engine for likelihood-free inference. *Journal of Machine Learning Research*, 19(16):1–7, 2018.

[12] Susan Athey, Julie Tibshirani, and Stefan Wager. Generalized random forests. *The Annals of Statistics*, 47(2):1148–1178, Apr 2019.

[13] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in neural information processing systems*, pages 3146–3154, 2017.

[14] Peter Kontschieder, Madalina Fiterau, Antonio Criminisi, and Samuel Rota Bulo. Deep neural decision forests. In *Proceedings of the IEEE international conference on computer vision*, pages 1467–1475, 2015.

Paper 68

On the accuracy in high dimensional linear models and its application to genomic selection

Charles-Elie Rabier[1,2], Brigitte Mangin[3] and Simona Grusea[4]

[1] Institut des Sciences de l'Evolution (ISEM), Université de Montpellier, CNRS, IRD, EPHE, Montpellier, France

[2] Institut Alexander Grothendieck Montpellier Institute (IMAG), Université de Montpellier, CNRS, France

[3] Laboratoire des Intéractions Plantes Microorganismes (LIPM), Université de Toulouse, INRA, CNRS, Castanet-Tolosan, France

[4] Institut de Mathématiques de Toulouse (IMT), Université de Toulouse, INSA de Toulouse, France

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

Reference paper: **Rabier *et al.* (2019). On the accuracy in high dimensional linear models and its application to genomic selection. *Scandinavian Journal of Statistics, 46(1).***
https://onlinelibrary.wiley.com/doi/full/10.1111/sjos.12352

Keywords: Genomic Selection, Quantitative Trait Locus, Prediction Accuracy, High Dimension, Rice data

For many years, geneticists focused on linkage analysis (LA) in order to detect on a given chromosome a Quantitative Trait Locus, so-called QTL: a QTL is a section of the DNA that contains one or more genes influencing a quantitative trait which is able to be measured. In this context, the most popular statistical method was Interval Mapping (Lander and Botstein, 1989). It consists in performing statistical tests along the genome. Using the information brought by genetic markers, the presence of a QTL is tested at every location in the genome. Later, geneticists moved on to genome-wide association studies (GWAS). In contrast to LA, GWAS are based on unrelated individuals and as a result, larger sample sizes can be considered. GWAS enabled the discovery of many SNP-trait associations in humans (e.g. age-related macular degeneration, Fritsche et al., 2016, autisum spectrum disorder, Connolly et al., 2017). However, both approaches (LA and GWAS) suffered from the fact that they were unable to detect QTLs with very small effects. Recall that most traits of interest are governed by a large number of small-effect QTLs (Goddard and Hayes, 2009, Buckler et al., 2009). It turns out that predictions based on selected SNPs could not be considered as reliable.

Today, Genomic Selection (GS), motivated by the seminal paper of Hayes et al. (2001), is an extremely popular technique in genetics. It consists in predicting breeding values of selection candidates using a large number of genetic markers, thanks to the recent progress in molecular biology. The goal is not to detect QTLs anymore, but to predict the future phenotype of young candidates as soon as their DNA has been collected. GS relies on the expectation that each QTL will be highly correlated with at least one marker (Schulz-Streeck et al., 2012). GS was first applied to animal breeding (see Hayes et al, 2009) and GS is nowadays extensively investigated in plants. For instance, we can mention studies on apple (Muranty et al. (2015)), eucalyptus (Tan et al. (2017)), japanese pears (Minamikawa et al. (2018)), strawberry (Gezan et al. (2017)), banana (Nyine et al. (2018)) and coffea (Ferrao et al. (2018)).

In GS, the quality of the prediction is evaluated according to some accuracy criteria, i.e. the correlation between predicted and true values. This criteria is a key element in genetics: it plays a role in the rate of genetic gain. Indeed, the accuracy is one component present in the breeders equation (see for instance Lynch and Walsh, 1998). One of the most popular methods, for prediction of breeding values, is Ridge regression. In genetics, this regression model, initially proposed by Hayes et al. (2001) and Whittaker et al. (2000), is called random regression best linear unbiased predictor (RRBLUP) or genomic best linear unbiased predictor (GBLUP). We focus here on some predictive aspects of Ridge regression and present theoretical results regarding the accuracy criteria. We show the influence of the singular values, the regularization parameter, and the projection of the signal on the space spanned by the rows of the design matrix. On simulated data, proxies built on our theoretical results outperformed existing proxies in GS, built on Daetwyler et al. (2008)'s seminal formula. Next, we will discuss on how to improve the prediction, using a "modified" predictor derived from Ridge regression. Finally, a real data analysis is proposed; it relies on the paper of Spindel et al. (2015) dealing with GS in rice.

A Fast Likelihood Method to Reconstruct and Visualize Ancestral Scenarios

Sohta A. Ishikawa,[†,1,2,3] Anna Zhukova,[†,1] Wataru Iwasaki,[2] and Olivier Gascuel[*,1]

[1]Unité Bioinformatique Evolutive, Institut Pasteur, C3BI USR 3756 IP & CNRS, Paris, France

[2]Department of Biological Sciences, The University of Tokyo, Tokyo, Japan

[3]Evolutionary Genomics of RNA Viruses, Virology Department, Institut Pasteur, Paris, France

***Corresponding author:** E-mail: olivier.gascuel@pasteur.fr.

[†]These authors contributed equally to this work.

Associate editor: Tal Pupko

Abstract

The reconstruction of ancestral scenarios is widely used to study the evolution of characters along phylogenetic trees. One commonly uses the marginal posterior probabilities of the character states, or the joint reconstruction of the most likely scenario. However, marginal reconstructions provide users with state probabilities, which are difficult to interpret and visualize, whereas joint reconstructions select a unique state for every tree node and thus do not reflect the uncertainty of inferences.

We propose a simple and fast approach, which is in between these two extremes. We use decision-theory concepts (namely, the Brier score) to associate each node in the tree to a set of likely states. A unique state is predicted in tree regions with low uncertainty, whereas several states are predicted in uncertain regions, typically around the tree root. To visualize the results, we cluster the neighboring nodes associated with the same states and use graph visualization tools. The method is implemented in the PastML program and web server.

The results on simulated data demonstrate the accuracy and robustness of the approach. PastML was applied to the phylogeography of Dengue serotype 2 (DENV2), and the evolution of drug resistances in a large HIV data set. These analyses took a few minutes and provided convincing results. PastML retrieved the main transmission routes of human DENV2 and showed the uncertainty of the human-sylvatic DENV2 geographic origin. With HIV, the results show that resistance mutations mostly emerge independently under treatment pressure, but resistance clusters are found, corresponding to transmissions among untreated patients.

Key words: **phylogenetics, ancestral character reconstruction, maximum likelihood, marginal and joint posterior probabilities, maximum a posteriori, Brier scoring rule, simulations, Dengue, HIV, phylogeography, drug resistance mutations.**

Introduction

A central issue in biology is to recover and understand the evolutionary history of biological entities. These may be of different nature and scale, ranging from DNA and protein sequences to communities, going through biological systems, organs, strains, individuals, species, and populations. The characteristics and evolution of these objects are measured using a variety of "characters," including molecular properties (e.g., Werner et al. 2014; Bickelmann et al. 2015; Busch et al. 2016), gene contents of genomes (e.g., Iwasaki and Takagi 2007), morphological and phenotypic characteristics (e.g., Endress and Doyle 2009; Marazzi et al. 2012; Beaulieu et al. 2013; Sauquet et al. 2017), ecological traits (e.g., Maor et al. 2017), and geographic locations (e.g., Arbogast 2001; Wallace et al. 2007; Lemey et al. 2009, 2014; Edwards et al. 2011; Dudas et al. 2017; Magee et al. 2017). Ancestral character reconstruction (ACR) is central in all these studies to trace the origin and evolution of the character of interest. ACR relies first on the inference of phylogenetic relationships among the studied objects, that is, a phylogenetic tree, typically inferred from DNA or protein sequences. The character state is generally known for all (most) tips of the tree (some methods can accommodate for unknown or ambiguous state values). ACR is commonly used to reconstruct ancestral sequences corresponding to specific tree nodes (typically the tree root). ACR is also used to determine how the character of interest has changed on the tree from the root to the tips over evolutionary time, by assigning the most likely ancestral character states to every internal node. This global reconstruction over the whole tree describes the evolutionary history of the character and is commonly called an "ancestral scenario," which is the focus of this article. Several approaches have been proposed for ACR so far, including parsimony (Swofford and Maddison 1987), maximum likelihood (ML; Pagel 1999; Pupko et al. 2000; Felsenstein 2004; Ree and Smith 2008), and Bayesian methods (Huelsenbeck and Bollback 2001; Pagel et al. 2004).

© The Author(s) 2019. Published by Oxford University Press on behalf of the Society for Molecular Biology and Evolution. This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited. For commercial re-use, please contact journals.permissions@oup.com

Open Access

Downloaded from https://academic.oup.com/mbe/article-abstract/36/9/2069/5498 Article Pasteur user on 09 March 2020

Paper 90

A new mechanism of glucose transport by GluT1 discovered in molecular dynamics simulations

Tatiana Galochkina[1,2], Matthieu Ng Fuk Chong[1,2] and Catherine Etchebest[1,2]
[1] Université de Paris, Biologie Intégrée du Globule Rouge, UMR_S 1134, BIGR, INSERM, Paris, France
[2] Laboratoire d'Excellence GR-Ex, Paris, France

Corresponding author: tatiana.galochkina@u-paris.fr

Reference paper: **Galochkina *et al.* (2019) New insights into GluT1 mechanics during glucose transfer, *Scientific reports*, 9, 998, 2019.** https://doi.org/10.1038/s41598-018-37367-z

Glucose is an essential source of energy for the mammalian cells. Its transport to erythrocytes and endothelial cells of the blood-brain barrier occurs as the result of the facilitative diffusion governed by the human glucose transporter type 1 (GluT1). GluT1 deficiency and inactivating mutations are associated with the severe central nervous system dysfunction (de Vivo disease). Understanding the role of GluT1 point mutations as well as modulation of GluT1 activity requires detailed description of GluT1 mechanics during glucose transport, which is the main subject of our study.

GluT1 belongs to the Major Facilitator Superfamily (MFS) of membrane transporters. According to the generally accepted hypothesis on the alternating access mechanism, glucose transport by GluT1 appears through a cycle of major conformational changes: the protein would adopt outward facing (open to the extracellular medium) conformation for the ligand uptake, then switch to the inward facing (open to cytoplasm) state for the ligand release [1] and go back to the outward facing state to accept a new ligand molecule. These key conformational transitions can be clearly distinguished for different X-ray structures of MFS proteins using a principle component analysis on the atom coordinates of their common transmembrane part. Thus, the plane formed by the two first principal components (PC plane) is a valuable tool to characterize any conformation of the family members.

For today, GluT1 was resolved only in the inward facing state in the presence of detergent [2,3]. In the current work we have explored GluT1 conformational space by running long molecular dynamics (MD) simulations in membrane environment. We have projected the obtained conformations on the PC plane and demonstrated that human GluT1 transporter adopts conformations distinctly separated from those of GluT1 bacterial homologs, which means that it can potentially follow a different mechanism of solute transport. We have further verified this hypothesis by running GluT1 MD simulations in presence of glucose. According to our results, glucose transfer can occur without any prominent transition of GluT1 conformation suggested by the alternating access mechanism hypothesis. In our simulations, it is driven by the side chain translocation and minor rearrangement of helical segments. We identify the main binding sites occupied by glucose molecule during its diffusion through the protein cavity and explore its kinetic properties of transfer. The obtained model also allows us to investigate the impact of point mutations on GluT1 mechanics and explain their role in glucose transport inhibition. In conclusion, the current study clearly revisits the alternating access mechanism and brings new insights to better understanding of GluT1 mechanics during glucose transit.

Acknowledgements

The authors thank CINES (Centre Informatique National de l'Enseignement Supérieur) for providing computational resources (project: A0050710622).

References

[1] Dong Deng, Pengcheng Sun, Chuangye Yan, Meng Ke, Xin Jiang, Lei Xiong, Wenlin Ren, Kunio Hirata, Masaki Yamamoto, Shilong Fan, and Nieng Yan. Molecular basis of ligand recognition and transport by glucose transporters. *Nature*, 526(7573):391–396, 2015.

[2] Dong Deng, Chao Xu, Pengcheng Sun, Jianping Wu, Chuangye Yan, Mingxu Hu, and Nieng Yan. Crystal structure of the human glucose transporter GLUT1. *Nature*, 510(7503):121–125, 2014.

[3] K. Kapoor, J. S. Finer-Moore, B. P. Pedersen, L. Caboni, A. Waight, R. C. Hillig, P. Bringmann, I. Heisler, T. Müller, H. Siebeneicher, and R. M. Stroud. Mechanism of inhibition of human glucose transporter GLUT1 is conserved between cytochalasin B and phenylalanine amides. *Proc. Natl. Acad. Sci.*, 113(17):4711–4716, 2016.

Paper 110

Reproducibility first: mining CLIP-seq data to understand the Exon Junction Complex

Toni Paternina[1], Auguste Genovesio[1] and Hervé Le Hir[1]

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

Corresponding Author: paternin@biologie.ens.fr

Abstract *The Exon Junction Complex (EJC) plays a central role in post-transcriptional gene expression regulation. To characterize the binding landscape of the EJC, we applied Cross-linking Immunoprecipitation and sequencing (CLIP-seq) protocols. However, low coverage and reproducibility rate limited the analysis of individual binding sites obtained with currently available tools. Here, we present an exon-level detection strategy that focuses on the specificity of EJC binding signal. We obtained statistically significant higher reproducibility rates both at the gene and exon levels. The study of robustly detected and undetected exons confirmed the inherent sequence bias of cross-linking. However, our data suggests that sequence bias alone cannot explain robustly unloaded exons. We present a highly specific strategy that mines EJC data to yield a reliable list of binding sites. This opens the door to study the link between EJC binding and gene topology features, and to perform comparative studies to elucidate the underlying mechanism of EJC deposition.*

Keywords RNA-binding proteins, CLIP-seq, NGS, reproducibility.

1 Introduction

RNA binding proteins (RBPs) play central roles in post-transcriptional gene expression regulation (PTGR). In eukaryotes, they are key elements in pre-mRNA splicing, and mRNA nuclear export, localization, storage, translation and degradation[1]. The Exon junction Complex (EJC) is an important node of the PTGR network. It is deposited around 24 nucleotides upstream of exon junctions by the spliceosome and accompanies transcripts at different stages of their life[2,3]. Although many of its fundamental roles have been described, a high-resolution, transcriptome-wide map of the EJC deposition sites is still lacking. This is crucial to explain its implication in developments and tissue-specific diseases.

Cross-linking and Immunoprecipitation (CLIP) coupled with high-throughput sequencing (CLIP-seq) aims to identify RBP targets through purification and identification of the RNA fragments they bind. The study of EJC CLIP data of HeLa cells, published in 2012 by our lab, suggested: a) a deposition rate of 80% of exons, directly correlated to transcript abundance, which implies that not all exon junctions of a gene are loaded, and b) a 50% rate of deposition in non-canonical binding sites, which means away from the canonical 24 nucleotides upstream the exon junction[4]. However, this study presented limitations in binding site resolution, and lacked technical replicates.

Since this 2012 publication, several improvements and variations of the CLIP protocol have appeared during the decade. Using single nucleotide CLIP techniques, we obtained EJC libraries that show a sharp enrichment 27 nucleotides upstream of the exon junction[5] (Fig. 1). Here, we develop a data analysis strategy that makes use of this high resolution to yield reproducible EJC binding sites. Ultimately, this work will help us gain insight into the EJC binding site landscape.

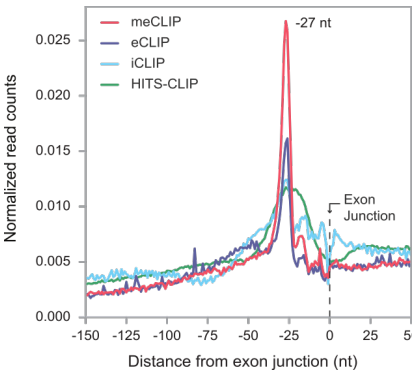


Fig 1: Meta-exon plot comparing the signal enrichment of EJC libraries obtained with different CLIP protocols.

2 Results

We aim to use EJC CLIP data to obtain a transcriptome-wide map of binding sites. However, data analysis of EJC CLIP libraries is challenging. Particularly, we struggled with low reproducibility of binding sites when applying the high resolution peak caller PureCLIP[6]. Moreover, while reproducible results are crucial to discern specific signal from random noise, reproducibility of CLIP data is often bypassed or treated as a secondary question in the literature. We thus developed an alternative strategy with focus on maximizing the reproducibility of our results.

2.1 Data description

Pseudo-replicates to overcome low coverage

To study the EJC binding site landscape at high resolution, we applied single-nucleotide protocols to obtain EJC CLIP libraries. Firstly, our lab generated eight meCLIP[5] libraries to quantify the percentage of read-through events that misplace the protein cross-linking site. However, these libraries were not perfect technical replicates and did not have sufficient coverage. Thus we decided to merge them to constitute two meCLIP pseudo-replicates: meCLIP-1 and meCLIP-2.

Monitoring PCR duplication yields better libraries

We incorporated Preseq[7] in our data pre-processing pipeline to estimate library complexity from sequencing pre-runs. This allowed us to obtain two actual EJC eCLIP replicates (eCLIP1 and eCLIP2), with lower PCR duplication rates and higher coverage. We sequenced the new libraries twice at separate times. Additionally, we generated two input control libraries (cross-linked RNA-fragments prior to immunoprecipitation), and added to the analyses two RNA-seq libraries published by our lab[8]. The table below summarizes the data:

Library	Protocol	Date	Reads in coding exons (no PCR duplicates)
meCLIP-1	meCLIP	10/2016	638636
meCLIP-2	meCLIP	10/2016	600934
eCLIP1-1	eCLIP	10/2019	2031071
eCLIP2-1	eCLIP	10/2019	8201373
eCLIP1-2	eCLIP	11/2019	2014508
eCLIP2-2	eCLIP	11/2019	8253491
input-1	eCLIP	11/2019	1182290
input-2	eCLIP	11/2019	2006825
RNA-1	RNA-seq	11/2014	45278330
RNA-2	RNA-seq	11/2014	43735087

We performed uniform sub-sampling of eCLIP2 to obtain data sets with the exact same number of reads as eCLIP1, in order to correct the difference between eCLIP1 and eCLIP2 and yield comparable results.

2.2 Finding signal enrichment at different levels

Peaks: high number, low reproducibility

Initially, we studied reproducibility of EJC binding sites using peaks detected by publicly available single-nucleotide peak caller PureCLIP. Although the number of peaks per replicate was of several thousands, we found that only 18% of these were common to both replicates. From these, even a smaller fraction corresponded to peaks in the *canonical region* (around 27 nucleotides upstream the exon junction). We concluded that the results obtained from our data were not very reproducible, despite the sound algorithm behind PureCLIP.

These results revealed a contradiction in our data: a high specificity of aggregated data (shown as a sharp enrichment in the meta-exon profile), but noisy and non-reproducible individual binding sites. To study the reproducibility of our data at different levels, we established two EJC signal enrichment scores. The first one measures EJC enrichment at the exon level, while the second measures deposition rate at the gene level.

Exon and gene scores: EJC Enrichment Score (EES) and Loaded Fraction (LF)

To obtain the most EJC-specific signal, we decided to focus on the canonical region of the exons. For this, we considered a 10-nucleotide window from the 22nd to the 32nd position upstream the exon junction. Similarly, we defined a non-canonical region as a window from the 5th to the 15th position. The EJC Enrichment Score (EES) is the ratio between the number of canonical reads over non-canonical reads. We designated exons with EES > 2 as enriched in EJC signal. Once we obtained enriched exons, we computed the EJC Loaded Fraction (LF) per gene, which corresponds to the number of enriched exons divided by the total number of exons of the longest isoform of a gene. We applied the same strategy on input controls and RNA-seq libraries.

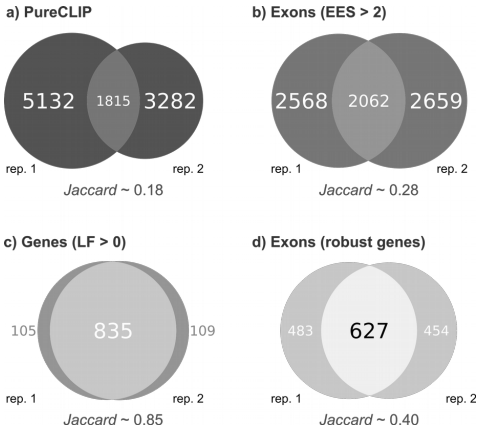


Fig 2: Venn diagrams at different levels of comparison between CLIP replicates: a) common PureCLIP peaks, b) common enriched exons (EES > 2), c) common detected genes (with at least 1 enriched exon, LF > 0), d) common detected exons in robust genes (genes with similar LF values in CLIP replicates).

Next, we compared the percentage of overlap at three different detection levels: individual PureCLIP peaks, enriched exons (EES > 2), and detected genes (LF > 0) (Fig. 2). We found that the gene level is the most reproducible, with approximately 85% of common genes (Fig. 2c). This shows that this strategy yields highly reproducible results from single-nucleotide EJC data, which was not the case when applying currently available methods.

More common exons in genes with reproducible LF values

To dig deeper into the reproducibility of LF values, we computed pairwise similarity ratios among CLIP libraries. Then, we selected genes with ratios between 0.66 and 1.5 in all pairwise comparisons and designated them as robust (N = 149).

Next we computed exon-level Jaccard indexes within robust genes. We found that the proportion of common enriched exons is higher (~40%) than prior to robust gene selection (Fig. 2b-c). Thus, reproducible LF values correlate with higher exon-level reproducibility. By prioritizing reproducibility and specificity, we established a reliable EJC positive control.

2.3 CLIP-detected genes are longer and more abundant than all expressed genes

Next, we aimed to characterize the population of robust genes. We obtained transcript abundance (RPKM) values form the RNA-seq data, and plotted their distribution in all expressed genes, in all CLIP detected genes (LF > 0), and in robust genes (Fig. 3a). We observe that detected and robust gene abundance is slightly skewed towards higher values compared to expressed genes. Then, we studied the spliced transcript size distribution, and found that detected and robust genes were consistently longer than all expressed genes (Fig. 3b). We analyzed the number of exons per gene and the length of individual exons, and found that detected and robust genes had more exons than expressed genes, while their median exon length is comparable (Fig. 3c-d). This shows that the difference in transcript length is due to a higher exon number rather than a higher exon size.

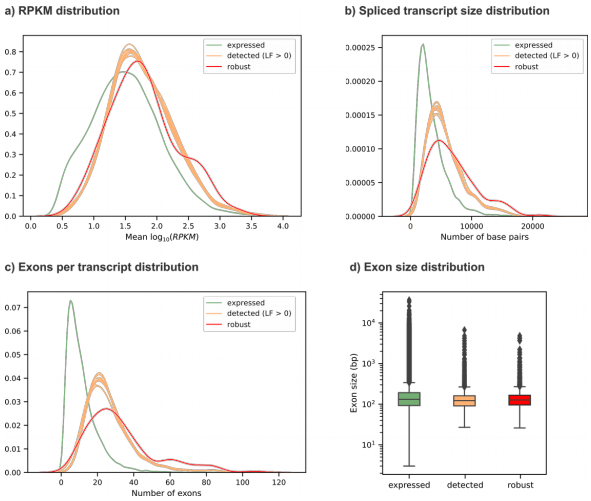


Fig 3: Distribution of gene features comparing all expressed genes, CLIP detected genes (LF > 0), and robust genes. a) The distribution of transcript abundance (log transformed RPKM values). b) The distribution of spliced transcript sizes (sum of exon sizes without introns). c) The distribution of number of exons per gene (using the longest isoform form genome annotation). d) The distribution of individual exon sizes.

Interestingly, transcript abundance and exons per gene distributions do not reveal striking differences between detected and robust genes. This suggests that our strategy selects robust genes from the pool of CLIP-detected genes rather than favoring particular genes. On the other hand, the differences with expressed genes suggests that our strategy finds reliable CLIP signal in more abundant and longer genes.

2.4 Studying sequence bias to determine robustly loaded and robustly unloaded exons

One of the questions we aim to answer is whether the EJC is systematically deposited in all exons of a transcript. The distribution LF values of CLIP detected genes suggest that on average around 20% of exons are loaded, and that most genes have between ~12% and ~25% of loaded exons (Fig, 4a). To study EJC loading within the robust gene population, we first determined the statistical significance of the observed exon Jaccard values. We shuffled the position of enriched exons within each gene, then we computed the Jaccard index of the shuffled configuration. We found that all observed Jaccard values fell outside the null distributions obtained from exon shuffling (Fig. 4b, all p-values equal to zero). This proves that the configuration of loaded/unloaded exons in a gene is not explained by random detection. We can thus affirm that a) robustly detected exons are likely to be loaded with EJC, and b) that robustly undetected exons are likely to be unloaded.

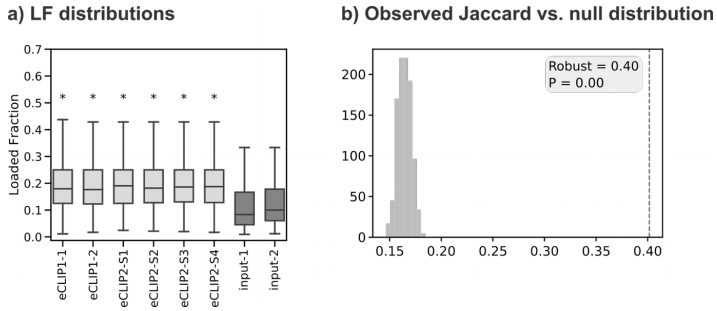


Fig 4: a) Distribution of Loaded Fraction values in all CLIP-detected genes in eCLIP data sets and input controls. Here we show sub-samplings of eCLIP2 (marked with an S); * $P < 0.05$, Mann-Whitney test. b) Distribution of Jaccard values in the shuffled exon configuration; Jaccard values between eCLIP1-1 and eCLIP2-S1 are shown; dashed lines correspond to observed values in robust genes; exact observed values and their corresponding p-values are shown. c) P-value matrix for the observed Jaccard index tested against shuffled distributions across all CLIP data set comparisons.

It is known that protein-RNA cross-linking is biased towards uracil bases[9]. To study the extent of the sequence bias in our data, we analyzed the nucleotide composition of the canonical region of robust gene exons. We defined four classes of exons: loaded (detected in all CLIP data sets), likely loaded (detected between 5 and 7 times across data sets), not robust (detected between 1 and 4 times), and unloaded (detected 0 times). We also analyzed the canonical region of expressed exons as a background reference. We found that robustly loaded exons had a higher thymine (T) content (uracil, U, in RNA) than exons in the other classes (Fig. 5). Interestingly, in a window going from the 2nd to the 9th positions in the canonical region, we observe an enrichment of T in the loaded exons, whereas in unloaded exons we observe a depletion.

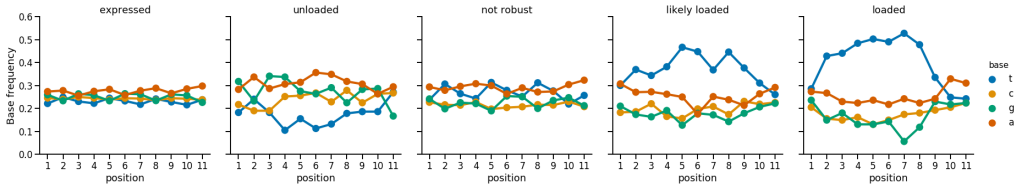


Fig 5: Nucleotide occurrence in the canonical region of expressed exons (from RNA-seq detected genes); robustly unloaded exons (detected 0 times across CLIP replicates); not robust exons (detected 1 to 4 times in CLIP replicates); likely loaded (detected 5 to 7 times in CLIP replicates), and robustly loaded (detected in all CLIP replicates). The occurrence of each nucleotide per position was obtained using the convert-matrix program of RSAT[10].

This observation reveals that the robustness of loaded exon detection is highly influenced by the uracil content in the canonical region. We created two classes of exons based on T content: high-T (exons with more than 3 Ts, or with Ts in the 2-9 window), and low-T (exons with less than 3 Ts, and no Ts in the 2-9 window). We set up a contingency table of EJC loading (loaded/unloaded) and T-content (high-T/low-T), to study the extent of the sequence bias:

Loading / T content	High T	Low T
Unloaded	199	59
Loaded	159	2

We observed that 99% of loaded exons belong to the high-T class, whereas around 77% of unloaded exons do. This observation confirms a significant impact of T-content on robust loaded exon detection (Chi-square test $P < 0.01$). However, we should note that a big fraction of unloaded exons belong to the high-T fraction. Thus, we hypothesize that for many unloaded exons, a low T content does not suffice to explain their robust lack of detection.

3 Conclusions

1. We set up an EJC CLIP analysis pipeline focused on the specificity of the EJC signal. With this strategy, we have overcome the lowly reproducible results obtained with currently available tools.
2. By focusing on reproducibility, we selected a highly specific EJC- positive population of robustly detected genes. We found a significant and reproducible exon-level configuration of loaded and unloaded exons.
3. Although robust detection of loaded exons is highly biased by T-content, our data suggests that sequence content does not fully explain the lack of exon detection. Further work requires establishing a statistical framework to compute the probability of observing T-enriched unloaded exons and test this hypothesis.
4. Finally, our study on specific EJC signal reveals that its deposition occurs in a smaller number of junctions than previous estimations. Future comparative studies of robustly loaded exons will elucidate the underlying mechanisms of EJC deposition on transcripts.

References

1. Gerstberger, S., Hafner, M. & Tuschl, T. A census of human RNA-binding proteins. *Nat. Rev. Genet.* **15**, 829–845 (2014).
2. Le Hir, H., Saulière, J. & Wang, Z. The exon junction complex as a node of post-transcriptional networks. *Nat. Rev. Mol. Cell Biol.* **17**, 41–54 (2015).
3. Boehm, V. & Gehring, N. H. Exon Junction Complexes: Supervising the Gene Expression Assembly Line. *Trends Genet.* **32**, 724–735 (2016).
4. Saulière, J. *et al.* CLIP-seq of eIF4AIII reveals transcriptome-wide mapping of the human exon junction complex. TL - 19. *Nat. Struct. Mol. Biol.* **19 VN-r**, 1124–1131 (2012).
5. Hocq, R., Paternina, J., Alasseur, Q., Genovesio, A. & Le Hir, H. Monitored eCLIP: high accuracy mapping of RNA-protein interactions. *Nucleic Acids Res.* (2018) doi:10.1093/nar/gky858.
6. Krakau, S. & Richard, H. PureCLIP : capturing target-specific protein-RNA interaction footprints from single-nucleotide CLIP-seq data Supplementary material 1 Data.
7. Daley, T. & Smith, A. D. Predicting the molecular complexity of sequencing libraries. *Nat. Methods* **10**, 325–327 (2013).
8. Wang, Z., Murigneux, V. & Le Hir, H. Transcriptome-wide modulation of splicing by the exon junction complex. *Genome Biol.* **15**, 551 (2014).
9. Zhang, C. & Darnell, R. B. Mapping in vivo protein-RNA interactions at single-nucleotide resolution from HITS-CLIP data. *Nat. Biotechnol.* **29**, 607–614 (2011).
10. Nguyen, N. T. T. *et al.* RSAT 2018: regulatory sequence analysis tools 20th anniversary. *Nucleic Acids Res.* **46**, W209–W214 (2018).

Paper 119

Inferring ligand-receptor interactions from single-cell and bulk transcriptomic data

Simon Cabello-Aguilar[1,2,3], Mélissa Alamé[1,2,3] and Jacques Colinge[1,2,3]

[1] Institut de Recherche en Cancérologie de Montpellier, Inserm U1194, 34298, Montpellier, France

[2] University of Montpellier, 34000, Montpellier, France

[3] Institut régional du Cancer de Montpellier, 34298, Montpellier, France

Corresponding Author: jacques.colinge@inserm.fr

Paper References: **Alame *et al.* The molecular landscape and microenvironment of salivary duct carcinoma reveal new therapeutic opportunities, Theranostics, 2020, 10(10): 4383-94. https://doi:10.7150/thno.42986. Cabello-Aguilar *et al.* SingleCellSignalR: Inference of intercellular networks from single-cell transcriptomics, Nucleic Acids Research, 2020, in press. https://doi:10.1093/nar/gkaa183**

1. LR*db* and single-cell transcriptomes

Single-cell transcriptomics offers unprecedented opportunities to infer the ligand-receptor interactions underlying cellular networks. We introduce a new, curated ligand-receptor database (LR*db*) and a novel regularized score to perform such inferences. For the first time, we try to assess the confidence in predicted ligand-receptor interactions and show that our regularized score outperforms other scoring schemes while controlling false positives. LR*db* and the scoring system are implemented in SingleCellSignalR, an open-access R package accessible to entry-level users and available from GitHub (Bioconductor integration pending). Inference results come in a variety of tabular and graphical formats such as a network view integrating all the intercellular interactions, complemented by the capability to explore how signaling downstream receptors enters each cell population intracellular pathways. Among various examples, we show how the ability to control false positives might unravel peculiar communication structures in tissues, e.g., mouse epidermis.

2. Bulk transcriptomes

The contribution of the tumor microenvironment (TME) to tumor progression and therapy resistance is substantial in most tumors. Immunotherapies have revolutionized the treatment of cancer, and antibodies targeting immune checkpoints or ligands thereof, *e.g.*, PD-1/PD-L1 or CTLA-4, have demonstrated clinical benefit. Such therapies disrupt TME ligand-receptor interactions.

We developed an algorithm to infer ligand-receptor interactions taking place in the TME from bulk transcriptomes. This algorithm integrated LR*db* with Reactome pathways and was applied to salivary duct carcinoma (SDC), a rare and aggressive cancer. We uncovered 179 high confidence interactions, 72 of which were correlated with the immune system infiltrate present in the TME. We validated three interactions by immunofluorescence and digital imaging, and discussed further targetable interactions based on the literature available for other tumors.

The Alame *et al*. paper provides the first description of the genomics of SDCs and shows the existence of two groups of tumors: immune-infiltrated and immune-poor. Besides other considerations about SDCs, the exploitation of LR*db* and the development of a first bulk algorithm to infer cellular interactions enabled us to propose novel options to treat immune-infiltrated SDCs.

3. Oral presentation

We will discuss the importance of the microenvironment, and the principles behind LR*db* construction and ligand-receptor scoring. Examples from the two articles as well as work in progress will illustrate the concrete application of ligand-receptor inference.

Paper 121

ComPotts: Optimal alignment of coevolutionary models for protein sequences

Hugo Talibart[1] and François Coste[1]
Univ Rennes, Inria, CNRS, IRISA, Campus de Beaulieu, 35042, Rennes, France

Corresponding author: hugo.talibart@irisa.fr

Abstract *To assign structural and functional annotations to the ever increasing amount of sequenced proteins, the main approach relies on sequence-based homology search methods, e.g. BLAST or the current state-of-the-art methods based on profile Hidden Markov Models (pHMMs), which rely on significant alignments of query sequences to annotated proteins or protein families. While powerful, these approaches do not take coevolution between residues into account. Taking advantage of recent advances in the field of contact prediction, we propose here to represent proteins by Potts models, which model direct couplings between positions in addition to positional composition. Due to the presence of non-local dependencies, aligning two Potts models is computationally hard. To tackle this task, we introduce an Integer Linear Programming formulation of the problem and present ComPotts, an implementation able to compute the optimal alignment of two Potts models representing proteins in tractable time. A first experimentation on 59 low sequence identity pairwise alignments, extracted from 3 reference alignments from sisyphus and BaliBase3 databases, shows that ComPotts finds better alignments than the other tested methods in the majority of these cases.*

Keywords Protein, sequence alignment, coevolution, Direct Coupling Analysis

1 Introduction

Thanks to sequencing technologies, the number of available protein sequences has considerably increased in the past years, but their functional and structural annotation remains a bottleneck. This task is thus classically performed *in silico* by scoring the alignment of new sequences to well-annotated homologs. One of the best-known method is BLAST[1], which performs pairwise sequence alignments. The main tools for homology search use now Profile Hidden Markov Models (pHMMs), which model position-specific composition, insertion and deletion probabilities of families of homologous proteins. Two well-known software packages using pHMMs are widely used today: HMMER[2] aligns sequences to pHMMs and HH-suite[3] takes it further by aligning pHMMs to pHMMs.

Despite their solid performance, pHMMs are innerly limited by their positional nature. Yet, it is well-known that residues that are distant in the sequence can interact and co-evolve, e.g. due to their spatial proximity, resulting in correlated positions (see for instance [4]).

There have been a few attempts to make use of long-distance information. Menke, Berger and Cowen introduced a Markov Random Field (MRF) approach where MRFs generalize pHMMs by allowing dependencies between paired residues in β-strands to recognize proteins that fold into β-structural motifs[5]. Their MRFs are trained on multiple structure alignments. Simplified models[6] and heuristics[7] have been proposed to speed up the process. While these methods outperform HMMER[2] in propeller fold prediction, they are limited to sequence-MRF alignment on β-strand motifs with available structures. Xu et al.[8] proposed a more general method, MRFalign, which performs MRF-MRF alignments using probabilities estimated by neural networks from amino acid frequencies and mutual information. Unlike SMURF, MRFalign allows dependencies between all positions and MRFs are built on multiple sequence alignments. MRFalign showed better alignment precision and recall than HHalign and HMMER on a dataset of 60K non-redundant SCOP70 protein pairs with less than 40% identity with respect to reference structural alignments made by DeepAlign[9], showing the potential of using long-distance information in protein sequence alignment.

Meanwhile, another type of MRF led to a breakthrough in the field of contact prediction[10]: the Potts model. This model was brought forward by Direct Coupling Analysis[11], a statistical method to extract direct correlations from multiple sequence alignments. Once inferred on a MSA, a Potts model's nodes represent positional conservation, and its edges represent direct couplings between positions in the MSA. Unlike mutual information which also captures indirect correlations between positions, Potts models are global models capturing the collective effects of entire networks of correlations through their coupling parameters[12], thus tackling indirect effects and making them a relevant means of predicting interactions between residues. Beyond contact prediction, the positional and the direct coupling information captured by Potts model's parameters might also be valuable in the context of protein homology search. The idea of using Potts models for this purpose was proposed last year at the same workshop by Muntoni and Weigt[13], who propose to align sequences to Potts models, and by us[14] with the introduction of ComPotts, our method to align Potts models to Potts models.

In this paper, we fully describe ComPotts and focus on its performances in terms of alignment quality. In the following sections, we explain our choices for Potts model inference and we describe our method for aligning them, which builds on the work of Wohlers, Andonov, Malod-Dognin and Klau[15,16,17] to propose an Integer Linear Programming formulation for this problem, with an adequate scoring function. We assess the quality of ComPotts' alignments with respect to 59 reference pairwise alignments extracted from sisyphus[18] and BaliBase3[19] databases. On these first experiments, computation time was tractable and ComPotts found better alignments than its main competitors: BLAST, HHalign (which is HHblits' alignment method) and MRFalign.

2 Methods

In this section, we describe our approach to align two Potts models. We start with a short summary of Potts models notations and then we explain the choices we made for the inference of Potts models. Then, we introduce our formulation of the alignment problem as an Integer Linear Programming problem, using notations from [20].

2.1 Inference of Potts models

Potts models are discrete instances of pairwise Markov Random Fields which originate from statistical physics. They generalize Ising models by describing interacting spins on a crystalline lattice with a finite alphabet. In the paper introducing Direct Coupling Analysis[11], Weigt et al. came up with the idea of applying them to proteins: inferred on a multiple sequence alignment, a Potts Model could then be used to predict contacts between residues.

A Potts model on protein sequences can be defined as follows:

Let S be a multiple sequence alignment (MSA) of length L over an alphabet Σ of length q (here we use the amino acid alphabet, which is of length $q = 20$). A Potts model for S is a statistical model defining a probability distribution over the set Σ^L of all sequences of length L which complies to the maximum-entropy principle and whose single and double marginal probabilities are the empirical frequencies of the MSA. Formally, denoting $f_i(a)$ the frequency of letter a at position i in the MSA S and $f_{ij}(a, b)$ the frequency of a and b together at positions i and j in S, a Potts model for S satisfies:

$$\forall i = 1, \cdots, L, \sum_{x \in \Sigma^L :\, x_i = a} \mathbb{P}(x_1, \cdots, x_L) = f_i(a)$$

$$\forall i = 1, \cdots, L, \forall j = 1, \cdots, L, \sum_{x \in \Sigma^L :\, x_i = a, x_j = b} \mathbb{P}(x_1, \cdots, x_L) = f_{ij}(a, b)$$

and defines a Boltzmann distribution on Σ^L with:

$$\mathbb{P}(x_1, \cdots, x_L | v, w) = \frac{1}{Z} \exp\left(\sum_{i=1}^{L} v_i(x_i) + \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} w_{ij}(x_i, x_j) \right)$$

where:

— Z is a normalization constant : $Z = \sum_{y \in \Sigma^L} \exp\left(\sum_{i=1}^{L} v_i(y_i) + \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} w_{ij}(y_i, y_j)\right)$

— $\{v_i\}_{i=1,\cdots,L}$ are positional parameters termed *fields*. Each v_i is a real vector of length q where $v_i(a)$ is a weight related to the propensity of letter a to be found at position i.

— $\{w_{ij}\}_{i,j=1,\cdots,L}$ are *pairwise couplings*. Each w_{ij} is a $q \times q$ real weight matrix where $w_{ij}(a, b)$ quantifies the tendency of the letters a and b to co-occur at positions i and j.

The value $\mathcal{H}(x) = -\left(\sum_{i=1}^{L} v_i(x_i) + \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} w_{ij}(x_i, x_j)\right)$ is called Hamiltonian.

In theory, one could infer a Potts model from a MSA S by likelihood maximization, i.e. by finding the positional parameters v and coupling parameters w that maximize $\mathbb{P}(S|v, w)$. In practice, however, this would require the computation of the normalization constant Z at each step, which is computationally intractable. Among the several approximate inference methods that have been proposed [21,22,23,24,12], we opted for pseudo-likelihood maximization since it was proven to be a consistent estimator in the limit of infinite data [25,26] within reasonable time. Furthermore, since our goal is to align Potts models, we need the inferrence to be geared towards similar models for similar MSAs, which is not what inference methods were initially designed for. In an effort towards inferring canonical Potts models, we chose to use CCMpredPy[27], a recent Python-based version of CCMpred[28] which, instead of using the standard L_2 regularization prior $R(v, w) = \lambda_v \|v\|_2^2 + \lambda_w \|w\|_2^2$, uses a smarter prior on v: $R(v, w) = \lambda_v \|v - v^*\|_2^2 + \lambda_w \|w\|_2^2$ where v^* obeys $\frac{\exp(v_i^*(a))}{\sum_{b=1}^{q} \exp(v_i^*(b))} = f_i(a)$ which yields the correct probability model if no columns are coupled, i.e. $\mathbb{P}(x|v, w) = \prod_{i=1}^{L} \mathbb{P}(x_i)$. Our intuition is that positional parameters should explain the MSA as much as possible and only necessary couplings should be added.

2.2 Alignment of Potts models

We introduce here our method for aligning two Potts models. We start by describing the function we designed to score a given alignment, then we add the constraints that make the alignment proper by embedding it in an Integer Linear Programming formulation, following Wohlers et al.[17], allowing us to use their efficient solver for the optimization.

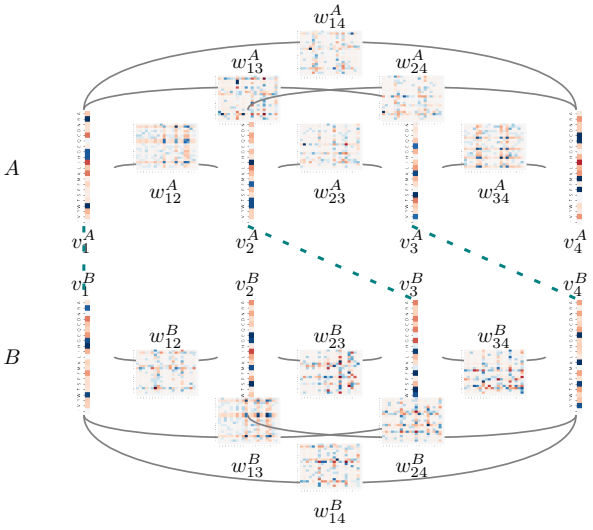


Fig. 1. Illustration of the alignment of two Potts models A and B.

2.2.1 Scoring an alignment We want the alignment score of two Potts models A and B to maximize the similarity between aligned fields and aligned couplings.

Formally, we want to find the binary variables x_{ik} and y_{ikjl}, where $x_{ik} = 1$ iff node i of Potts model A is aligned with node k of Potts Model B and $y_{ikjl} = 1$ iff edge (i, j) of Potts model A is aligned with edge (k, l) of Potts model B, such that: $\sum_{i=1}^{L_A}\sum_{k=1}^{L_B} s_v(v_i^A, v_k^B)x_{ik} + \sum_{i=1}^{L_A-1}\sum_{j=i+1}^{L_A}\sum_{k=1}^{L_B-1}\sum_{l=k+1}^{L_B} s_w(w_{ij}^A, w_{kl}^B)y_{ikjl}$ is maximized, where $s_v(v_i^A, v_k^B)$ and $s_w(w_{ij}^A, w_{kl}^B)$ are similarity scores between positional parameters v_i^A and v_k^B and coupling parameters w_{ij}^A and w_{kl}^B.

To score the similarity $s_v(v_i^A, v_k^B)$ between positional parameters v_i^A and v_k^B we use the scalar product :

$$s_v(v_i^A, v_k^B) = \langle v_i^A, v_k^B \rangle = \sum_{a=1}^{q} v_i^A(a)v_k^B(a)$$

And to score the similarity $s_w(w_{ij}^A, w_{kl}^B)$ between coupling parameters w_{ij}^A and w_{kl}^B we use the Frobenius inner product, which is the natural extension of scalar product to matrices :

$$s_w(w_{ij}^A, w_{kl}^B) = \langle w_{ij}^A, w_{kl}^B \rangle = \sum_{a=1}^{q}\sum_{b=1}^{q} w_{ij}^A(a,b)w_{kl}^B(a,b)$$

This scoring function can be seen as a natural extension of the opposite of the Hamiltonian of a sequence x, since $-\mathcal{H}(x|v,w) = \sum_{i=1}^{L} v_i(x_i) + \sum_{i=1}^{L-1}\sum_{j=i+1}^{L} w_{ij}(x_i, x_j) = \sum_{i=1}^{L}\langle v_i, e_{x_i}\rangle + \sum_{i=1}^{L-1}\sum_{j=i+1}^{L}\langle w_{ij}, e_{x_i x_j}\rangle$

where :
- e_{x_i} is the vector defined by $\forall a \in [1..q], e_{x_i}(a) = \delta(a, x_i)$
- $e_{x_i x_j}$ is the matrix defined by $\forall (a,b) \in [1..q]^2, e_{x_i x_j}(a,b) = \delta(a, x_i)\delta(b, x_j)$

2.2.2 Optimizing the score with respect to constraints Naturally, the scoring function should be maximized with respect to constraints ensuring that the alignment is proper. In that perspective, we build on the work of Wohlers et al.[17], initially dedicated to protein structure alignment, to propose an Integer Linear Programming formulation for the Potts model alignment problem.

We remind first the constraints for a proper alignment following [20].

First, we need the definition of *alignment graph*. For A and B two Potts Models of lengths L_A and L_B, the alignment graph G is a $L_A \times L_B$ grid graph where rows (from bottom to top) represent the nodes of A and columns (from left to right) represent the nodes of B. A node $i.k$ in the alignment graph indicates the alignment of node i from Potts model A and node k from Potts model B.

Every proper alignment of two Potts model is described by a *strictly increasing path* in this alignment graph, which is defined as a subset $\{i_1.k_1, i_2.k_2, \cdots, i_n.k_n\}$ of alignment graph nodes that can be ordered such that each node is strictly larger than the previous one, i.e. $i_1 < i_2 < \cdots < i_n$ and $k_1 < k_2 < \cdots < k_n$.

To specify the constraints of the ILP, they defined sets of mutually contradicting nodes, called *decreasing paths*. A decreasing path is a set $\{i_1.k_1, i_2.k_2, \cdots, i_n.k_n\}$ of alignment graph nodes for which $i_1 \geq i_2 \geq \cdots \geq i_n$ and $k_1 \leq k_2 \leq \cdots \leq k_n$ holds. The set of all decreasing paths is denoted $\mathcal{C}$.

We also give notations for the left and right neighborhood of a node : let $i.k$ be a node in the alignment graph and $V_{i.k}^+$ (resp. $V_{i.k}^-$) denote the set of couples that are strictly larger (resp. smaller) than $i.k$, e.g. $V_{i.k}^+ = \{(j,l) \mid (j > i) \wedge (l > k)\}$ and $V_{i.k}^- = \{(j,l) \mid (j < i) \wedge (l < k)\}$ and let $\mathcal{C}_{i.k}^+$ (resp. $\mathcal{C}_{i.k}^-$) denote the set of all decreasing paths in $V_{i.k}^+$ (resp. $V_{i.k}^-$).

Given the notations above, with A (resp. B) a Potts Model of length L_A (resp. L_B) with parameters v^A and w^A (resp v^B and w^B), aligning A and B can be formulated as the following Integer Linear Programming problem:

$$\max_{x,y} \quad \sum_{i=1}^{L_A}\sum_{k=1}^{L_B} s_v(v_i^A, v_k^B)x_{ik} + \sum_{i=1}^{L_A-1}\sum_{j=i+1}^{L_A}\sum_{k=1}^{L_B-1}\sum_{l=k+1}^{L_B} s_w(w_{ij}^A, w_{kl}^B)y_{ikjl} \tag{1}$$

$$\text{s.t.} \quad x_{ik} \geq \sum_{j.l\in C} y_{ikjl} \quad \forall C \in \mathcal{C}^+_{i.k}, i \in [1..L_A-1], k \in [1..L_B-1] \tag{2}$$

$$x_{ik} \geq \sum_{j.l\in C} y_{jlik} \quad \forall C \in \mathcal{C}^-_{i.k}, i \in [2..L_A], k \in [2..L_B] \tag{3}$$

$$x_{ik} \leq 1 + \sum_{j.l\in C} (y_{ikjl} - x_{jl}) \quad \forall C \in \mathcal{C}^+_{i.k}, i \in [1..L_A-1], k \in [1..L_B-1] \tag{4}$$

$$\sum_{i.k\in C} x_{ik} \leq 1 \quad \forall C \in \mathcal{C} \tag{5}$$

$$y \geq 0 \tag{6}$$

$$x \text{ binary} \tag{7}$$

As in [17], an affine gap cost function can be added to the score function to account for insertions and deletions in the sequences.

3 Results

We implemented this ILP formulation in a program, ComPotts, embedding the solver from [17]. We assessed the performances of ComPotts in terms of alignment precision and recall with respect to a set of 59 pairwise reference alignments. For each sequence, a Potts model was inferred on a multiple sequence alignment of close homologs retrieved by HHblits.

3.1 Data

We extracted 59 reference pairwise sequence alignments from 3 reference multiple sequence alignments from sisyphus[18] and BaliBase3[19] with a particularly low sequence identity. To focus on sequences with coevolution information, we considered only sequences with at least 1000 close homologs (see next section). We also discarded sequences with more than 200 amino acids for memory considerations with respect to CCMpredPy. Reference alignments are summed up in table 3.1

Alignment ID	Database	% identity	Selected sequences
AL00049879	sisyphus	11.7	1g6gA, 1gxcA, 1lgqA, 1mzkA, 1r21A, uhtA, 1ujxA, 1wlnA, 2cswA, 2fezA, 2g1lA
AL00055723	sisyphus	6	1tu1A, 1v2bB
BB11022	BB3	11.3	1au7, 1neq, 1r69

Tab. 1. Reference multiple alignments used in our experiment and selected sequences extracted.

3.2 Alignment experiment

3.2.1 Potts model inference For each sequence, we built a MSA of its close homologs by running HHblits[3] v3.0.3 on Uniclust30[29] (version 08/2018) with parameters recommended in [30] for CCMpred: `-maxfilt 100000 -realign_max 100000 -all -B 100000 -Z 100000 -n 3 -e 0.001` which we then filtered at 80% identity using HHfilter, and took the first 1000 sequences. If the MSA had less than 1000 sequences we removed it from the experiment. This MSA was then used to train a Potts model with CCMpredPy using default parameters except for the w regularization factor coefficient (we set it to 30, which we empirically found to result in $\|v\|_2^2 \simeq \frac{1}{2}\|w\|_2^2$, in other words making v and w scores commensurable) and the uniform pseudo-counts count on the v (we set it to 1000 to have as many pseudo-counts as the number of sequences in the alignment in order to enhance stronger conservation signal and limit excessive similarity scores due to missing the same rare residues).

3.2.2 Potts model alignment We ran ComPotts with a gap open cost of 8 and no gap extend cost, which we found empirically to yield the best alignments in previous experiments. To speed up the computations, we decided to stop the optimization when $\frac{2(UB-LB)}{s(A,A)+s(B,B)} < 0.005$ with UB and LB

the current upper and lower bounds of the Lagrangian optimization, since we realized in preliminary experiments that in practice it gave the same alignments as the optimal ones in significantly less time.

3.3 Alignment quality assessment

We compared each resulting alignment with the reference pairwise alignment extracted from the multiple sequence alignment, considering the alignment precision with the Modeler score[31] $\frac{\#\ correctly\ aligned\ columns}{\#\ columns\ in\ test\ alignment}$ and the alignment recall with the TC score[32] $\frac{\#\ correctly\ aligned\ columns}{\#\ columns\ in\ ref\ alignment}$, computed using Edgar's qscore program[33] v2.1.

To compare our results, we ran HHalign v3.0.3 to align HMMs built on the MSAs outputted by HHblits, MRFalign v0.90 to align MRFs it built from the sequences, both with default options, and BLASTp v2.9.0+ without E-value cutoff. As a control, we also ran Matt v1.00 on the corresponding PDB structures. Results are summarized in figure 2. Note that Matt failed to run 3 of the alignments.

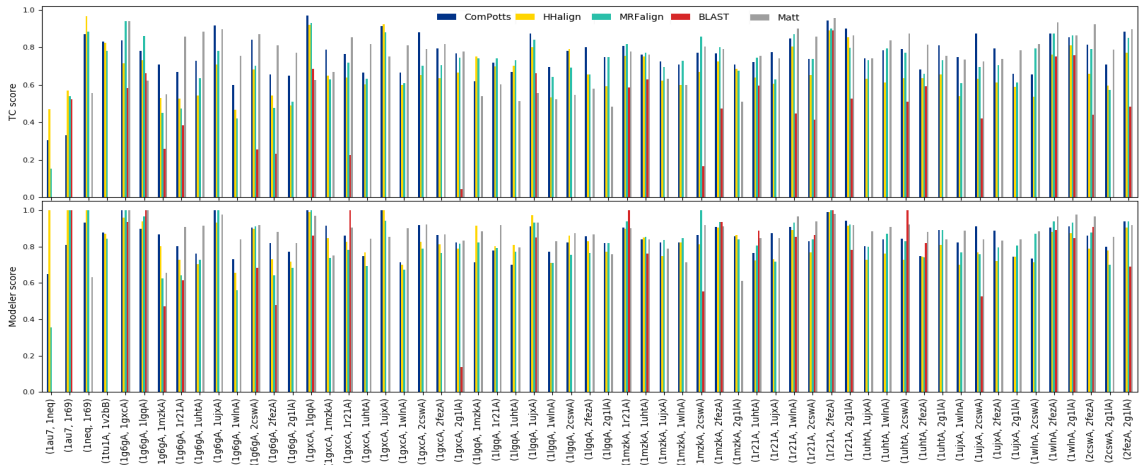


Fig. 2. TC score (alignment recall) and Modeler score (alignment precision) for all 59 alignments.

BLAST is unquestionably outperformed by all other tools on this set. 10 out of the 59 sequence pairs could not be aligned (not hit was found) and, on 22 of the alignments it performed, BLAST had both a recall (TC score) and a precision (Modeler score) of 0. Its average TC score is 0.2694 and its average Modeler score is 0.4357, which is about half the average scores of the other methods. BLAST has a better precision on some alignments, most of the time because its alignments are smaller, which results in a rather low recall, except for some alignments which seem to be quite easy for everyone, such as 1r21A and 2fezA.

All methods seem to struggle with the alignment of 1au7 and 1neq: HHalign's precision skyrockets to 1.0, but at the cost of a recall of 0.47, while ComPotts and MRFalign yield their worst scores, with respective recalls of 0.31 and 0.65 and respective precisions of 0.15 and 0.35.

On average, ComPotts' alignments have a better recall than all compared tools including Matt with 0.758, versus 0.670 for HHalign, 0.713 for MRFalign, and 0.749 for Matt, outperforming HHalign most of the time – in 52 out of the 59 alignments – and MRFalign in 39 alignments out of the 59, while still having a slightly better precision than all other sequence-based tools with 0.847 while HHalign's is 0.826 and MRFalign's is 0.822, outperforming HHalign in 46 alignments out of the 59 and MRFalign in 30 alignments. Matt has the best precision on average with 0.872. Overall, ComPotts has an average $F1$ score ($\frac{2\times precision\times recall}{precision+recall}$) of 0.800, versus 0.740 for HHalign and 0.763 for MRFalign, yielding better alignments than HHalign in 52 cases and better than MRFalign in 39 cases. For as-yet-unknown reasons though, our scores for the alignment of 1au7 and 1r69 are remarkably lower than our competitors.

3.4 Computation time considerations

We examined the computation times of ComPotts, HHalign and MRFalign, considering only the time they took to align the models and not the time needed to build the models. Not surprisingly, ComPotts is significantly slower than HHalign and MRFalign. This is explained by the fact that HHalign only performs 1D alignment, and MRFalign uses a heuristic to compute the alignment, whereas ComPotts uses an exact algorithm. Aligning two sequences took between 37 seconds (for two models with 75 and 63 positions) and 14.49 minutes (for two models with 144 and 151 positions), with an average of 3.33 minutes on a Debian9 virtual machine with 4 vCPUs and 8GB of RAM, whereas HHalign yields a solution in less than 4 seconds and MRFalign in less than 0.20 seconds. It is worth noting that, although the computation time is significantly higher than its competitors, the solver yields an exact solution in tractable time, even though this problem is NP-complete[34]. In this experiment, the computation time seems to be dominated by the computation of all the s_w scores, which is quadratic in the number of pairs of edges.

4 Conclusion

We described ComPotts, our ILP-based method for Potts model-Potts model alignment which can yield the exact solution in tractable time. We reported encouraging results on first experiments where ComPotts often yields better alignments than its two main competitors, HHalign and MRFalign, with respect to a set of 59 low sequence identity reference pairwise alignments. These initial results suggest that direct coupling information can improve protein sequence alignment and might improve sequence-based homology search as well. We still have to see whether the score yielded by ComPotts has more discriminatory power than other methods and enables to better distinguish homologous from non-homologous proteins.

Acknowledgements

HT is supported by a PhD grant from *Ministère de l'Enseignement Supérieur et de la Recherche* (MESR). We would like to warmly thank Inken Wohlers for providing us with her code, and Mathilde Carpentier for providing a selection of difficult reference alignments and helpful scripts for alignment assessment.

References

[1] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.

[2] Sean R. Eddy. Profile hidden markov models. *Bioinformatics (Oxford, England)*, 14(9):755–763, 1998.

[3] Martin Steinegger, Markus Meier, Milot Mirdita, Harald Voehringer, Stephan J Haunsberger, and Johannes Soeding. Hh-suite3 for fast remote homology detection and deep protein annotation. *bioRxiv*, page 560029, 2019.

[4] Michael Socolich, Steve W Lockless, William P Russ, Heather Lee, Kevin H Gardner, and Rama Ranganathan. Evolutionary information for specifying a protein fold. *Nature*, 437(7058):512, 2005.

[5] Matt Menke, Bonnie Berger, and Lenore Cowen. Markov random fields reveal an n-terminal double beta-propeller motif as part of a bacterial hybrid two-component sensor system. *Proceedings of the National Academy of Sciences*, 107(9):4069–4074, 2010.

[6] Noah M Daniels, Raghavendra Hosur, Bonnie Berger, and Lenore J Cowen. Smurflite: combining simplified markov random fields with simulated evolution improves remote homology detection for beta-structural proteins into the twilight zone. *Bioinformatics*, 28(9):1216–1222, 2012.

[7] Noah M Daniels, Andrew Gallant, Norman Ramsey, and Lenore J Cowen. Mrfy: remote homology detection for beta-structural proteins using markov random fields and stochastic search. *IEEE/ACM transactions on computational biology and bioinformatics*, 12(1):4–16, 2014.

[8] Jianzhu Ma, Sheng Wang, Zhiyong Wang, and Jinbo Xu. Mrfalign: protein homology detection through alignment of markov random fields. *PLoS computational biology*, 10(3):e1003500, 2014.

[9] Sheng Wang, Jianzhu Ma, Jian Peng, and Jinbo Xu. Protein structure alignment beyond spatial proximity. *Scientific reports*, 3:1448, 2013.

[10] Bohdan Monastyrskyy, Daniel D'Andrea, Krzysztof Fidelis, Anna Tramontano, and Andriy Kryshtafovych. New encouraging developments in contact prediction: Assessment of the casp 11 results. *Proteins: Structure, Function, and Bioinformatics*, 84:131–144, 2016.

[11] Martin Weigt, Robert A White, Hendrik Szurmant, James A Hoch, and Terence Hwa. Identification of direct residue contacts in protein–protein interaction by message passing. *Proceedings of the National Academy of Sciences*, 106(1):67–72, 2009.

[12] Matteo Figliuzzi, Pierre Barrat-Charlaix, and Martin Weigt. How pairwise coevolutionary models capture the collective residue variability in proteins? *Molecular biology and evolution*, 35(4):1018–1027, 2018.

[13] Anna Paola Muntoni, Andrea Pagnani, Martin Weigt, and Francesco Zamponi. Using direct coupling analysis for the protein sequences alignment problem. In *CECAM 2019 - workshop on Co-evolutionary methods for the prediction and design of protein structure and interactions*, 2019.

[14] Hugo Talibart and François Coste. Using residues coevolution to search for protein homologs through alignment of potts models. In *CECAM 2019 - workshop on Co-evolutionary methods for the prediction and design of protein structure and interactions*, 2019.

[15] Rumen Andonov, Noël Malod-Dognin, and Nicola Yanev. Maximum contact map overlap revisited. *Journal of Computational Biology*, 18(1):27–41, 2011.

[16] Inken Wohlers, Rumen Andonov, and Gunnar W Klau. Algorithm engineering for optimal alignment of protein structure distance matrices. *Optimization Letters*, 5(3):421–433, 2011.

[17] Inken Wohlers, Rumen Andonov, and Gunnar W Klau. Dalix: optimal dali protein structure alignment. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 10(1):26–36, 2012.

[18] Antonina Andreeva, Andreas Prlić, Tim JP Hubbard, and Alexey G Murzin. Sisyphus—structural alignments for proteins with non-trivial relationships. *Nucleic acids research*, 35(suppl_1):D253–D259, 2007.

[19] Julie D Thompson, Patrice Koehl, Raymond Ripp, and Olivier Poch. Balibase 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins: Structure, Function, and Bioinformatics*, 61(1):127–136, 2005.

[20] Inken Wohlers. *Exact Algorithms For Pairwise Protein Structure Alignment*. PhD thesis, Vrije Universiteit, 01 2012.

[21] Faruck Morcos, Andrea Pagnani, Bryan Lunt, Arianna Bertolino, Debora S Marks, Chris Sander, Riccardo Zecchina, José N Onuchic, Terence Hwa, and Martin Weigt. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–E1301, 2011.

[22] Carlo Baldassi, Marco Zamparo, Christoph Feinauer, Andrea Procaccini, Riccardo Zecchina, Martin Weigt, and Andrea Pagnani. Fast and accurate multivariate gaussian modeling of protein families: predicting residue contacts and protein-interaction partners. *PloS one*, 9(3):e92721, 2014.

[23] Magnus Ekeberg, Cecilia Lövkvist, Yueheng Lan, Martin Weigt, and Erik Aurell. Improved contact prediction in proteins: using pseudolikelihoods to infer potts models. *Physical Review E*, 87(1):012707, 2013.

[24] John P Barton, Eleonora De Leonardis, Alice Coucke, and Simona Cocco. Ace: adaptive cluster expansion for maximum entropy graphical model inference. *Bioinformatics*, 32(20):3089–3097, 2016.

[25] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[26] Julian Besag. Statistical analysis of non-lattice data. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 24(3):179–195, 1975.

[27] Susann Vorberg. *Bayesian Statistical Approach for Protein Residue-Residue Contact Prediction*. PhD thesis, Ludwig-Maximilians-Universität, 2017.

[28] Stefan Seemayer, Markus Gruber, and Johannes Söding. Ccmpred—fast and precise prediction of protein residue–residue contacts from correlated mutations. *Bioinformatics*, 30(21):3128–3130, 2014.

[29] Milot Mirdita, Lars von den Driesch, Clovis Galiez, Maria J Martin, Johannes Söding, and Martin Steinegger. Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic acids research*, 45(D1):D170–D176, 2017.

[30] Stefan Seemayer. Github ccmpred - frequently asked questions (faq). https://github.com/soedinglab/CCMpred/wiki/FAQ.

[31] J Michael Sauder, Jonathan W Arthur, and Roland L Dunbrack Jr. Large-scale comparison of protein sequence alignment algorithms with structure alignments. *Proteins: Structure, Function, and Bioinformatics*, 40(1):6–22, 2000.

[32] Julie D. Thompson, Frédéric Plewniak, and Olivier Poch. Balibase: a benchmark alignment database for the evaluation of multiple alignment programs. *Bioinformatics (Oxford, England)*, 15(1):87–88, 1999.

[33] Robert C. Edgar. Qscore. http://www.drive5.com/qscore/.

[34] Richard H Lathrop. The protein threading problem with sequence amino acid interaction preferences is np-complete. *Protein Engineering, Design and Selection*, 7(9):1059–1068, 1994.

Paper 124

nf-core, a community effort for collaborative, peer-reviewed analysis pipelines

Philip A. Ewels[1], Alexander Peltzer[2], Sven Fillinger[2], Harshil Patel[3], Johannes Alneberg[4], Andreas Wilm[5], Maxime U. Garcia[6], Paolo Di Tommaso[7,8] and Sven Nahnsen[2]

[1] Science for Life Laboratory (SciLifeLab), Department of Biochemistry and Biophysics, Stockholm University, Stockholm, Sweden

[2] Quantitative Biology Center (QBiC), University of Tübingen, Tübingen, Germany

[3] Bioinformatics and Biostatistics, The Francis Crick Institute, London, UK

[4] Science for Life Laboratory (SciLifeLab), School of Engineering Sciences in Chemistry, Biotechnology and Health, Department of Gene Technology, Royal Institute of Technology, Stockholm, Sweden

[5] Computational

[6] Systems Biology, Genome Institute of Singapore, Singapore, Singapore

[7] Department of Oncology-Pathology, Karolinska Institutet, Stockholm, Sweden

[8] Centre for Genomic Regulation (CRG), The Barcelona Institute for Science and Technology, Barcelona, Spain

[9] Universitat Pompeu Fabra (UPF), Barcelona, Spain

Corresponding author: maxime.garcia@scilifelab.se

Reference paper: **Ewels, P.A., Peltzer, A., Fillinger, S. *et al.* (2020) The nf-core framework for community-curated bioinformatics pipelines. *Nat Biotechnol.*** https://dx.doi.org/10.1038/s41587-020-0439-x

The standardization, portability and reproducibility of analysis pipelines are key issues within the bioinformatics community. Most bioinformatics pipelines are designed for use on-premises; as a result, the associated software dependencies and execution logic are likely to be tightly coupled with proprietary computing environments. This can make it difficult or even impossible for others to reproduce the ensuing results, which is a fundamental requirement for the validation of scientific findings.

Here, we introduce the nf-core framework as a means for the development of collaborative, peer-reviewed, best-practice analysis pipelines. All nf-core pipelines are written in Nextflow and so inherit the ability to be executed on most computational infrastructures, as well as having native support for container technologies such as Docker and Singularity. The nf-core community has developed a suite of tools that automate pipeline creation, testing, deployment and synchronization. Our goal is to provide a framework for high-quality bioinformatics pipelines that can be used across all institutions and research facilities.

As the usage of workflow management tools spreads, an increasing number of tertiary tools are tying into the ecosystem. The nf-core analysis pipelines are at the forefront of this, collaborating with initiatives such as bio.tools and the GA4GH-compliant Dockstore, as well as having plans to work together with the Biocontainers project to further simplify software packaging. The primary portal to the nf-core community is its website https://nf-co.re, which lists available analysis pipelines, user- and developer-centric documentation, and tutorials, as well as usage and contributor statistics.

All source code the nf-core framework and all nf-core pipelines is publicly available on GitHub under the nf-core organization https://github.com/nf-core/ and released under the MIT license. Where applicable, Zenodo DOIs are available on the respective pipeline repositories.

We hope to welcome more contributors and pipelines to the nf-core community to build on the solid foundation that has already been established.

Paper 135

Towards a better understanding of the low discovery rate of short-read based insertion variant callers

Wesley Delage[1], Julien Thevenon[2] and Claire Lemaitre[1]
[1] Univ Rennes, CNRS, Inria, IRISA - UMR 6074, F-35000, Rennes, France
[2] Unité de Génétique Clinique,Pôle Couple Enfant, CHU de Grenoble Site Nord-Hôpital Couple-Enfant, 38043, Grenoble, France

Corresponding author: wesley.delage@inria.fr

Abstract *Since 2009, numerous tools have been developed to detect structural variants using short read technologies. Insertions are one of the hardest type to discover and are drastically underrepresented in gold standard variant call sets. The advent of long read technologies has completely changed the situation. In 2019, two independent cross technologies studies have published the most complete variant call sets with sequence resolved insertions in human individuals. Among the reported insertions, only 17% could be discovered with short-read based tools. In this work, we performed an in-depth analysis on one of these unprecedented insertion call sets, in order to investigate the causes of such failures. We have first established a precise classification of insertion variants according to three different layers of characterization: the nature of the inserted sequence, the genomics context of the insertion site and the breakpoints junction complexity. Because these levels are intertwined, we used simulations to characterize the impact of each complexity factor. Most reported insertions exhibited characteristics that may interfere with their discovery: 56% were tandem repeat expansions, 25% contained homology larger than 20 bp within their breakpoints junctions and 64% were located in simple repeats. Consequently, the recall of short-read based variant callers was significantly lower for such insertions (6% vs 48% for mobile element and novel insertions). Simulations showed that the most impacting factor on the discovery rate was the insertion type rather than the genomics context, and that the different factors of insertion complexities were handled differently depending on the chosen tool.*

Keywords short reads, variant calling, structural variants, insertions

1 Introduction

Structural variants (SVs) are defined as a fragment of DNA of at least 50 bp that differs between two individuals[1]. SV are categorized by type : deletion (DEL) for a loss of a fragment, insertion (INS) for a gain of a fragment, inversion for a reversion of a fragment (INV) and translocation (TRANS) for moving a fragment to another position in the genome. Such variations in the genome sequence may have important functional impacts on the organism and SVs are commonly associated to human genetic diseases or disorders [2].

The classical approach to call SVs from Whole Genome sequencing (WGS) with short reads relies on a first mapping step to a reference genome. Then SV callers look for atypical mapping signals, such as discordant read pairs, clipped reads or abnormal read depth, to identify putative SV breakpoints along the reference genome [3,4]. More than 70 SV callers have been developed up to date and several benchmarks have highlighted the low level of agreement between the different methods, demonstrating that SV detection using short reads sequencing remains challenging [5]. Indeed the size of the reads is small compared to the target event size and the detection is mainly based on alignments which may produce artefacts[6]. In particular, insertions are one of the most difficult SV types to call. Because the inserted sequence is absent from the reference genome, or at least at the given locus of insertion, calling such variants and resolving the exact inserted sequence require trickier approaches such as de novo or local assembly [7,8]. This increased difficulty is well exemplified by the dramatic under-representation of such SV type in usual reference databases or standard variant call sets such as dbVar.

Recently, the commercialization of novel long reads technologies has completely changed the situation, and insertion variants are finally being discovered and referenced in human populations[9]. Thanks to several international efforts, some gold standard call sets have been produced in 2019, referencing tens of thousands of insertions in a given human individual [10,11]. Among the reported insertions by Chaisson et al, a great majority (83 %) could not be discovered by any of the tested short-read based tools. This result of discovery rate below 17 % is drastically different from the announced performances of insertion callers when evaluated on simulated datasets [12]. Indeed, Chaisson et al showed that 59 % of insertion variants are found in a tandem repeat context, highlighting the fact that most real insertion variants in human individuals are probably not "simple" sequences inserted in "easy" genomic contexts. However, their analysis went no further in order to precisely identify the actual features of insertion events that make them so difficult to be discovered by short read data.

In this work, we performed an in-depth analysis of this unprecedented insertion call set, in order to investigate the causes of short read based caller failures. We have first established a precise classification of insertion variants according to three different layers of characterization: the nature of the inserted sequence, the genomic context of the insertion site and the breakpoint junction complexity. Because these levels are intertwined, we used simulations to characterize the impact of each complexity factor on the discovery rate of several SV callers, accounting for the different types of methodological approaches.

2 Results

2.1 In-depth analysis of an exhaustive insertion variant call set

In this work, we first aimed at precisely characterizing an exhaustive set of insertion variants present in a given human individual. We based our study on a recently published SV call set published by Chaisson and colleagues in 2019[10]. Using an extensive sequencing dataset, combining several different sequencing technologies and methodological approaches (short, linked and long reads, mapping-based and assembly-based SV calling), three human trios were thoroughly analysed to establish exhaustive and gold standard SV call sets. We focused our study on the individual NA19240, son of the so-called Yoruban (YRI) Nigerian trio, whose SV call set contains 15,693 insertions greater than 50 bp.

We have established a precise classification of these insertion variants according to three different layers of characterization: the nature of the inserted sequence, the genomic context of the insertion site and the breakpoint junction complexity.

Insertion sub-types. Insertion variants can be classified in different sub-types according to the nature of the inserted sequence. Whereas only 3 insertion categories were distinguished in the original publication, namely tandem repeats, mobile element (ME) insertions and complex ones for all the other types, we chose to refine this classification in six insertion sub-types, illustrated in Figure 1. A classical subdivision consists in opposing novel sequence to duplicative insertions. In the first case, the inserted sequence is completely absent in the reference genome, whereas in the second, the inserted sequence has one or several homologous copies elsewhere in the genome. Among duplications, *mobile element* insertions are a very specific sub-type and are defined based on the homology of the inserted sequence with an already known mobile element. Then, several sub-types of duplicative insertions are then defined according to the location or amount of the inserted sequence copies in the reference genome. We therefore distinguish *tandem duplications*, for which at least one copy of the inserted sequence is adjacent to the insertion site, from *dispersed duplications*, for which its copies can be located anywhere else in the genome. Among tandem duplications, we defined a specific sub-type called *tandem repeats*, where the inserted sequence itself is composed of multiple tandem repetitions of a seed motif. Mobile elements (ME) are characterized by very high copy numbers in the genome (typically greater than 500), other dispersed duplication types were then required to have a copy number lower than 50, in order not to be confounded with potential MEs. Finally, a sub-type of dispersed duplications is a segmental duplication, that must be larger than 1kb and share more than 90 % of sequence identity with at least one copy, following previous definitions [2].

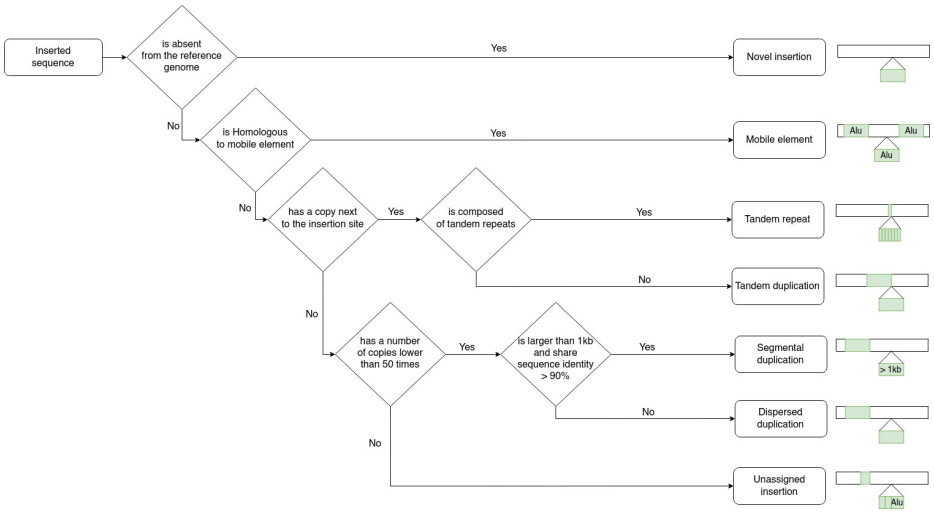


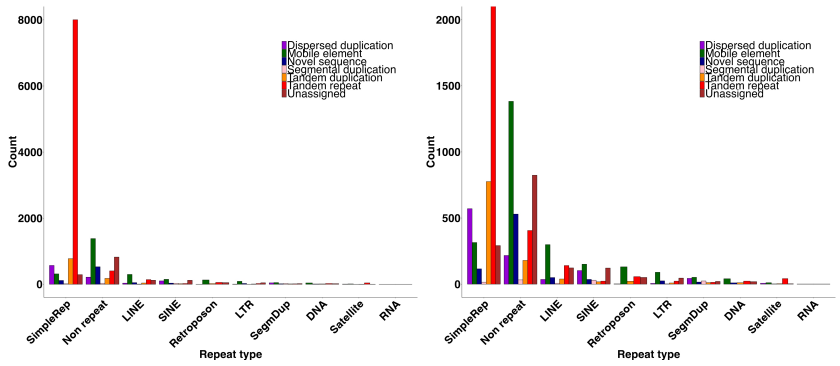
Fig. 1. Decision tree used to classify insertion variants in six insertion sub-types.

In order to classify the insertion call set, all inserted sequences were aligned against the human reference genome, a mobile element database and were scanned for tandem repeats (see Material and Methods). We used a minimal sequence coverage threshold to annotate each insertion to an insertion sub-type according to the decision tree described in Figure 1. We set the threshold to 80% for our analysis to ensure a good compromise between specificity and quantity of annotated insertions in all sub-types. For instance, an insertion is classified as a novel sequence insertion if more than 80 % of its inserted sequence is not covered by any alignment with the reference genome nor with the ME reference sequences, nor contains tandem repeats. Insertions that does not meet the 80 % coverage requirement to be annotated as one of the previous sub-types are qualified as *unassigned* insertions.

With a threshold set at 80%, 90% of insertions could be assigned to a given type. Among the 15,693 insertions, 56% were annotated as tandem repeats, 16 % as mobile elements, 7 % as tandem duplications, 5 % as novel sequences, 6 % as dispersed duplications and 1% as segmental duplications (Figure 2). Compared to the classification of Chaisson et al, the proportions of tandem repeats (57% vs 56%) and mobile elements (23% vs 16%) were very similar. The difference in mobile element proportions represent mainly insertions that are unassigned in our annotation, suggesting that our classification is more conservative. Interestingly, 77 % of their complex insertions were more precisely classified in one of our six sub-types, with mainly 3 sub-types being roughly equally involved: novel sequences, tandem and dispersed duplications. Short read based SV callers used in the original study were able to detect 17 % of these insertions, mainly represented by MEs. This short-read recall was highly variable with respect to the insertion type: ME and novel sequence insertions showed the best recalls (49 and 45 % respectively), whereas other types were all below 11 %. In particular, tandem repeats appeared to be a very hard insertion type to discover (recall of 5 %), although it represents most of the insertion variation in a human genome.

Characterization of insertion locations in the genome. We then characterized the insertions based on the genomic context of their insertion site. We investigated in particular genomic features that can make read mapping and SV calling difficult, such as the repetitive content. A strong over-representation was found in regions annotated as simple repeats, with 64% of the insertions located in these regions that only represent 1.2 % of the genome. As expected, 93 % of tandem repeats were found in simple repeat regions, revealing expansions of already known sites to be highly repeated. We also observed most of the duplications, tandem (72 %) or dispersed (58 %) in these regions. Conversely, 68 % of novel sequence insertions and 56 % of mobile element insertions were located in non repeated regions (Figure 2). We did not find a higher rate of insertions among exon, intron or intergenic regions compared to their distribution along the genome. Compared to GC content

variation along the genome, insertions showed an under-representation in regions with GC content lower than 41% (20 % vs 29% of the genome content) and an over-representation in regions with GC content higher than 46% (17 % vs 7% of the genome content). Novel sequence and mobile element insertions showed to be located in lower GC content regions (median lower than 40 %) than tandem and dispersed duplications, and tandem repeats (median greater than 43%).



(a) (b)

Fig. 2. Dispersion of insertion sub-types according to the repeat content of their insertion site. (a) overview of the dispersion, (b) zoom-in for class counts below 2,000.

Junctional homology. Junctional homology is defined as a DNA sequence that has two identical or nearly identical copies at the junctions of the two genomic segments involved in the rearrangement, when the sequence is short (<70 bp) this is often called a micro-homology [13]. These homologies and micro-homologies have been found involved in several molecular mechanisms generating rearrangements (NAHR for homologies, and MMEJ or MMBIR for microhomologies) [14,15]. In the case of an insertion, a junctional homology is a sequence segment at the left (resp. right) side of the insertion site which is nearly identical to the end (resp. beginning) of the inserted sequence. From a detection point of view, these homologies can have an impact on SV calling performance, since the concerned region at the inserted site is no longer specific to the reference allele and it is no longer possible to identify the exact location of the insertion site. Therefore, we systematically compared the insertion site junction sequences with the inserted sequence extremities to identify stretches of identical or nearly identical sequences. Half of the insertions contained junctional homologies larger than 5 bp, and still 25 % larger than 20 bp, mainly represented by dispersed duplications. The size distribution of the homologies varied between insertion types, novel sequences had small microhomologies (median of 5bp), mobile elements a medium size (median of 15 bp) and dispersed duplications showed a higher homology size (median of 86 bp). Interestingly, insertions called by long reads only had larger junctional homologies than insertions that could be discovered by short reads also (median size of 64 bp vs 12 bp resp.), pointing towards junctional homologies being a potential difficulty factor for short-read based callers.

2.2 Using simulations to investigate the factors impacting the insertion calling recall

In real insertion call sets, most of the previously identified factors impacting SV discovery are correlated. In order to quantify the impact of each factor independently, we produced various simulated datasets of 2x150 bp reads at 40x coverage, containing each 200 homozygous insertion variants on the human chromosome 3. As a baseline, we simulated 250 bp novel sequences taken from yeast exonic sequences inserted inside human exons. This is meant to represent the easiest type of insertions to detect, where inserted sequences contain very few repeats and are novel in the genome, the genomic context of insertion is also simple and repeat-free, and breakpoint junctions do not have any homology. Then, we considered 3 scenarii of simulations, where only one of the three factors, among insertion type (complexity of the inserted sequence), insertion site location and homology at the breakpoints, is changed at a time with respect to the baseline simulation. Four insertion variant callers were evaluated

on these datasets. They were chosen according to their good performances in recent benchmarks [5] and to maximise the methodological diversity. GRIDSS[8], Manta[12] and SVaba[4] are based on a first mapping step to the reference genome, contrary to MindTheGap[7] which uses solely an assembly data structure (the De Bruijn graph).

Discovery rates for all four methods are presented for the different simulated datasets in Table 1. On the baseline simulation, all tools had a close to perfect discovery rate. However, it should be noted that the tools were evaluated solely on their ability to detect an insertion event at a given site regardless of the predicted genotype and the resolution of the inserted sequence. As a matter of fact, only MindTheGap was able to return sequence resolved insertions. The other tools returned either only the insertion site or the insertion site with a partial inserted sequence.

Impact of the insertion type. When simulating various insertion types, GRIDSS was the only tool whose discovery rate was not impacted. Manta could not find any dispersed duplications and very few mobile elements, MindTheGap was unable to detect any type of tandem duplications and SVaba was not able to detect any tandem repeat with a small motif and almost half of the mobile element insertions (Table 1).

Impact of microhomology. Concerning junctional homology, GRIDSS and SVaba were both the less impacted tools. Only the scenario with 50 bp size microhomology impacted them, reducing by 30 to 40 % their discovery rate. Manta discovery rate decreased with the size of microhomology, starting at 50 bp size, reaching 0 % with 150 bp homologies. MindTheGap was the most impacted by microhomology, being unable to detect insertions with microhomology at any tested size.

		Recall (insertion site only)			
		GRIDSS	**Manta**	**MindTheGap**	**SVaba**
Baseline simulation: 250 bp novel sequences in exons		100	95	99	97
# False positive		33	0	14	184
Scenario 1 Insertion type	Dispersed duplication	97	0	97	91
	Tandem duplication	98	98	0	100
	Mobile element	100	5	70	58
	Tandem repeat (6 bp pattern)	100	92	0	0
	Tandem repeat (25 bp pattern)	100	71	5	99
# False positive		33-533	1-22	14-20	6-592
Scenario 2 Microhomology	20 bp	100	99	0	96
	50 bp	70	45	0	59
	100 bp	100	14	0	100
	150 bp	100	0	0	100
# False positive		33-200	2-56	15	2-595
Scenario 3 Genomic location	Low GC content	84	100	72	99
	medium GC content	85	100	69	99
	high GC content	86	100	75	99
	Non repeat	83	99	76	99
	Simple repeat	86	100	71	98
	SINE	86	100	53	99
	LINE	82	99	91	100
	Real locations	84	80	38	71
# False positive		106-144	3-9	16-21	6-25
Scenario 4: real insertions at real locations		45	35	6	44
# False positive		513	107	19	523

Tab. 1. Discovery rate of several short-read insertion callers according to different simulation scenarii. Cells of the table are colored according to the variation of the recall value of the given tool with respect to the recall obtained with the baseline simulation (first line, colored in blue): cells in red show a loss of recall >10%, cells in grey show no difference compared to baseline recall at +/- 10%. For each scenario, the last line indicates the range of the number of false positive predictions.

Impact of the genomic location. Concerning the impact of the genomic context of insertion, the tools showed two distinct behaviors. On the one hand, Manta and SVaba were not affected by

the repeat or GC content of the regions hosting the insertion site. On the other hand, both GRIDSS and MindTheGap showed a loss of recall even in repeat-free and medium GC contexts with respect to exonic locations simulated in the baseline simulation. Interestingly, when using the locations of the real insertions of NA19240 to simulate simple insertions, all tools underwent a loss in their recall compared to the same inserted sequences but in exonic locations in the baseline simulation.

Finally, when simulating the real insertions at their real location as described in the NA19240 variant calling file for the chromosome 3, the discovery rate of all tools dropped to less than 45 %, reaching for many tools their lowest values among the different simulated datasets. This suggested that several levels of difficulties might be combined in real insertions. GRIDSS reached the largest discovery rate (45 %), but it produced the largest amount of false positive discoveries. Surprisingly, the amount of false positives was not constant for most tools, it increased when the simulated insertions are less well discovered or with particular insertion types.

3 Discussion

We have presented here one of the most detailed and comprehensive analyses of factors impacting the detection of insertion variants in the human genome with short read re-sequencing data. This could be possible thanks to the publication of an exceptional SV call set by Chaisson et al[10]. Not only, this catalog of insertion variants is considered as the most exhaustive for a given human individual, but this is also the first set with sequence-resolved events for any size and type of insertions. This resolution of sequence enabled us to propose a refined classification of insertion variants and to quantify the presence of sequence homologies at the breakpoint junctions. Our results showed a strong over-representation of insertion types and contexts towards the most difficult ones to detect with short-read data, for instance tandem repeats inserted in simple repeat contexts. Moreover, most insertions and even the simplest types, such as novel sequence insertions, showed junctional homologies of substantial size that affect SV calling with short reads.

Our simulation protocol enabled to study each difficulty factor independently and highlighted the larger impact of insertion type compared to insertion location. However, all studied factors taken independently could not explain the whole loss of discovery rate and there is probably an important synergetic effect of combining in a single insertion event several of the studied factors. Surprisingly, the different evaluated tools showed very contrasted sensitivities to the different simulated difficulties. This result suggests that combining the calls of several SV callers could improve substantially the overall discovery rate. Currently, Structural Variation studies are based on intersection selections of a combination of SV callers, selecting only the calls that are discovered concordantly between different tools to increase the precision. [5]. Our results suggest an utterly different way of combining tools by taking a careful union of calls based for instance on the type or location of insertions. The main shortcoming of this strategy would then be to control the false positive rate. Our results on simulated data showed that except for MindTheGap, short-read based tools can not provide sequence-resolved variants. We argue that systematically assembling the inserted sequence, such as what is performed with MindTheGap using the whole read set instead of a sub-sample, could help in controlling the false discovery rate.

4 Methods

Data origin. The SV call set of individual NA12940 was downloaded from the following link: `ftp://ftp.ncbi.nlm.nih.gov/pub/dbVar/data/Homo_sapiens/by_study/genotype/nstd152/NA19240.BIP-unified.vcf.gz`. Out of the 17,026 described insertions, only insertions that were sequence resolved (ie. with an inserted sequence entirely defined) and that were also present in at least one of the parent were kept, resulting in a set of 15,693 insertions. The human reference genome version for this study was Hg38.

Insertion annotation. TandemRepeatFinder (TRF) was used to annotate tandem repeats within each inserted sequence [16]. Recommended parameters were used, except for the maximum expected TR length (-l) which was set to 6 millions. In order to annotate Mobile Elements (MEs) in inserted sequences, we used one of the annotation tools of RepeatMasker, namely dfam [17]. Each

inserted sequence was scanned by dfam with the standard hmm profile database of human MEs provided by the tool. For the detection of dispersed duplications and the occurrence count of their copies in the reference genome, each inserted sequence was locally aligned against the human genome using Blat with default parameters [18]. Only the alignments with at least 80 % identity were kept. For the detection of tandem duplications, both breakpoint junction sequences were aligned against the inserted sequence using Blat.

Junctional homology detection. From the previous obtained alignments between the breakpoint junctions and the inserted sequence, only the alignments with at least 90 % identity and occurring as close as 10 bp from extremities of the inserted sequence and from the insertion site were kept. Only alignments between the left (resp. right) side of the insertion site and the end (resp. beginning) of the inserted sequence were kept. In case of multiple candidates hits at one side of the junction, the one located at the closest position from extremities was kept. If homologies were found at both sides of the junction, the homology size was obtained by summing both alignment sizes.

Genomic context characterization. To study the genomic context of insertions, we used the repeat content annotations of RepeatMasker from the UCSC genome browser for the Hg38 genome and the gene annotations from the Gencode v32. To study the GC content, we segmented the genome into isochores with isoSegmenter [19], giving the following five families of isochores: <37 %, 37-41%, 41-46%, 46-53% and >53% GC content.

Simulations. 18 sequencing datasets were simulated to characterize the impact of potential difficulties for variant calling. Each dataset was obtained by altering the human chromosome 3 with 200 insertions. Reads were generated using ART with the following parameters : 2x150 bp reads, at 40 X coverage, with insert size of 300 bp on average and 20 bp standard deviation [20].
Baseline simulation. We simulated 250 bp novel sequence insertions located in exons without any microhomology at the breakpoint junctions. Novel sequences were extracted from random exonic regions of the *Saccharomyces cerevisae* genome.
Scenario 1: Insertion type impact. Insertion locations were identical to the baseline simulation, but the 250 bp inserted sequences were alternatively replaced by dispersed duplications, tandem repeats, tandem duplications or mobile elements. Two types of tandem repeats were simulated, with a pattern size of 6 bp or 25 bp, the pattern originating from the left breakpoint junction. 200 Alu mobile element sequences with a size ranging between 200 and 300 bp were randomly extracted from the human genome based on the RepeatMasker annotation. Tandem duplications were generated by duplicating the 250 bp left breakpoint sequence. The inserted sequences of simulated dispersed duplications were extracted from exons of the chromosome 3.
Scenario 2 : Microhomology impact. The 250 bp insertion sequences produced in the baseline simulation were altered with microhomology. To simulate microhomologies, we replaced the X first bases of each insertion with the same size sequence originating from the right breakpoint sequence. We simulated four microhomology sizes : 20, 50, 100 and 150 bp.
Scenario 3 : Location impact. The 250 bp insertions from the baseline simulation were inserted in specific genomic contexts : either inside different types of mobile elements, namely SINEs and LINEs, in simple repeats or in non-repeated regions with different GC contents. We defined three families of GC content : low (<41%), middle (41-46%) and high (>46 %).
Scenario 5 : Real insertions at real locations. The 889 insertions located on the chromosome 3 from the NA19240 call set were simulated as described in the vcf file.

Insertion calling. Simulated reads were aligned with bwa against the hg38 reference genome, and read duplicates were marked. GRIDSS v2.8.0, Manta v1.6.0, MindTheGap v2.2.1 and SVaba v1.1.0 were all run using recommended, or otherwise default, parameters [8,12,7,4]. Only "PASS" insertions, that were larger than 50 bp, were kept for the recall calculation. Since most of the tools are not able to output sequence resolved variants, the discovery rate was assessed solely based on the insertion site location prediction with a 10 bp margin around the expected location (after left-normalization).

Acknowledgements

We acknowledge the GenOuest bioinformatics core facility (https://www.genouest.org) for providing the computing infrastructure.

References

[1] Monya Baker. Structural variation: the genome's hidden architecture. *Nature methods*, 9(2):133–137, 2012.

[2] Lars Feuk, Andrew R Carson, and Stephen W Scherer. Structural variation in the human genome. *Nature Reviews Genetics*, 7(2):85–97, 2006.

[3] Mehdi Pirooznia, Melissa Kramer, Jennifer Parla, Fernando S Goes, James B Potash, W Richard McCombie, and Peter P Zandi. Validation and assessment of variant calling pipelines for next-generation sequencing. *Human genomics*, 8(1):14, 2014.

[4] Jeremiah A. Wala, Pratiti Bandopadhayay, Noah Greenwald, Ryan O Rourke, Ted Sharpe, Chip Stewart, Steve Schumacher, Yilong Li, Joachim Weischenfeldt, Xiaotong Yao, Chad Nusbaum, Peter Campbell, Gad Getz, Matthew Meyerson, Cheng-Zhong Zhang, Marcin Imielinski, and Rameen Beroukhim. SvABA: genome-wide detection of structural variants and indels by local assembly. *Genome Research*, mar 2018.

[5] Shunichi Kosugi, Yukihide Momozawa, Xiaoxi Liu, Chikashi Terao, Michiaki Kubo, and Yoichiro Kamatani. Comprehensive evaluation of structural variation detection algorithms for whole genome sequencing. *Genome Biology*, 20(1), jun 2019.

[6] Irina Abnizova, Rene te Boekhorst, and Y Orlov. Computational errors and biases of short read next generation sequencing. *J Proteomics Bioinform*, 10(1):1–17, 2017.

[7] Guillaume Rizk, Anaïs Gouin, Rayan Chikhi, and Claire Lemaitre. Mindthegap : integrated detection and assembly of short and long insertions. *Bioinformatics*, 30(24):3451–3457, Dec 2014.

[8] Daniel L. Cameron, Jan Schröder, Jocelyn Sietsma Penington, Hongdo Do, Ramyar Molania, Alexander Dobrovic, Terence P. Speed, and Anthony T. Papenfuss. Gridss: sensitive and specific genomic rearrangement detection using positional de bruijn graph assembly. *Genome Research*, 2017.

[9] Fritz J Sedlazeck, Philipp Rescheneder, Moritz Smolka, Han Fang, Maria Nattestad, Arndt von Haeseler, and Michael C Schatz. Accurate detection of complex structural variations using single-molecule sequencing. *Nature methods*, 15(6):461–468, 2018.

[10] Mark J.P. Chaisson, Ashley D. Sanders, ..., Tobias Marschall, Jan Korbel, Evan E. Eichler, and Charles Lee. Multi-platform discovery of haplotype-resolved structural variation in human genomes. *Nature Communications*, 10:1784, April 2019.

[11] Justin M. Zook, Nancy F. Hansen, ..., Mark JP Chaisson, Noah Spies, Fritz J. Sedlazeck, Marc Salit, and the Genome in a Bottle Consortium. A robust benchmark for germline structural variant detection. *bioRxiv*, jun 2019.

[12] Xiaoyu Chen, Ole Schulz-Trieglaff, Richard Shaw, Bret Barnes, Felix Schlesinger, Morten Källberg, Anthony J Cox, Semyon Kruglyak, and Christopher T Saunders. Manta: rapid detection of structural variants and indels for germline and cancer sequencing applications. *Bioinformatics (Oxford, England)*, 32:1220–1222, April 2016.

[13] Diego Ottaviani, Magdalena LeCain, and Denise Sheer. The role of microhomology in genomic structural variation. *Trends in Genetics*, 30(3):85–94, 2014.

[14] Donald F Conrad, Christine Bird, Ben Blackburne, Sarah Lindsay, Lira Mamanova, Charles Lee, Daniel J Turner, and Matthew E Hurles. Mutation spectrum revealed by breakpoint sequencing of human germline cnvs. *Nature genetics*, 42(5):385, 2010.

[15] Jeffrey M Kidd, Tina Graves, Tera L Newman, Robert Fulton, Hillary S Hayden, Maika Malig, Joelle Kallicki, Rajinder Kaul, Richard K Wilson, and Evan E Eichler. A human genome structural variation sequencing resource reveals insights into mutational mechanisms. *Cell*, 143(5):837–847, 2010.

[16] Gary Benson. Tandem repeats finder: a program to analyze dna sequences. *Nucleic acids research*, 27(2):573–580, 1999.

[17] Robert Hubley, Robert D Finn, Jody Clements, Sean R Eddy, Thomas A Jones, Weidong Bao, Arian FA Smit, and Travis J Wheeler. The dfam database of repetitive dna families. *Nucleic acids research*, 44(D1):D81–D89, 2016.

[18] W James Kent. Blat—the blast-like alignment tool. *Genome research*, 12(4):656–664, 2002.

[19] Paolo Cozzi, Luciano Milanesi, and Giorgio Bernardi. Segmenting the human genome into isochores. *Evolutionary Bioinformatics*, 11:EBO–S27693, 2015.

[20] Weichun Huang, Leping Li, Jason R Myers, and Gabor T Marth. Art: a next-generation sequencing read simulator. *Bioinformatics*, 28(4):593–594, 2012.

Paper 138

Template for JOBIM 2020 Highlight
Formalizing and Enriching Phenotype Signatures Using Boolean Networks

Méline Wery[1,2], Olivier Dameron[1], Jacques Nicolas[1], Elisabeth Remy[2] and Anne Siegel[1]
[1] Univ Rennes, Inria, CNRS, IRISA F-35000 Rennes, France
[2] SANOFI R&D, Translational Sciences, Chilly Mazarin, 91385, France
[3] Aix Marseille Univ, CNRS, Centrale Marseille, I2M, Marseille, France

Corresponding author: anne.siegel@irisa.fr

Reference paper: **Wery *et al.* (2019) Formalizing and enriching phenotype signatures using Boolean Networks. *Journal of theoretical biology.*** https://doi.org/10.1016/j.jtbi.2019.01.015

A biological signature can be defined as a set of biological features that characterize a phenotype or a population. To identify this signature, we analyze observational data, *i.e.* gene expression, protein quantification. However, without a prior knowledge network, this approach does not take into account the regulation dependencies.

In our study, we used two Boolean regulatory networks that represent the differenciation of T helper lymphocytes (Th) [1,2]. Each node is a gene and each edge characterizes an activation or an inhibition. The dynamic simulation of those models lead to the generation of steady states and attractors. The classification of those steady states allows to predict in which sub-types of Th (canonical phenotypes), the system will differenciate in, according to the environmental inputs.

We developed a method that automatically classifies steady states based on given signatures using Formal Concept Analysis (FCA), a symbolic bi-clustering technic [3,4]. FCA generated a lattice structure describing the associations between elements in the signature and steady states of the Boolean network. We defined the concept of a signature in a Boolean network and of a phenotype with the FCA.

We first validated our method on the smallest network [1]. We classified the steady states according to the three given phenotypes that are generated with two simulations (with or without IL12). Because the number of steady states increases with the network size and the number of simulation conditions, we then evaluated our approach with a larger network [2]. The analysis of this lattice lead to the same classification than the manual classification performed in [2]. Moreover, we enriched the biological signatures according to the regulation dependencies. The enriched signatures characterize variant phenotypes which are sub-types of the canonical phenotypes. When variants are shared by several canonical sub-types, they are called hybrid phenotypes. Compared to the initial results, we identified a new hybrid phenotype using extended simulation conditions.

References

[1] Elisabeth Remy, Paul Ruet, Luis Mendoza, Denis Thieffry, and Claudine Chaouiya. From Logical Regulatory Graphs to Standard Petri Nets: Dynamical Roles and Functionality of Feedback Circuits. In Corrado Priami, Anna Ingólfsdóttir, Bud Mishra, and Hanne Riis Nielson, editors, *Transactions on Computational Systems Biology VII*, pages 56–72. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[2] Aurélien Naldi, Jorge Carneiro, Claudine Chaouiya, and Denis Thieffry. Diversity and plasticity of Th cell types predicted from regulatory network modelling. *PLoS Computational Biology*, 6(9), 2010.

[3] Bernhard. Ganter and Rudolf. Wille. *Formal concept analysis : mathematical foundations*. Springer, 1999.

[4] Bernhard. Ganter, Gerd Stumme, and Rudolf. Wille. *Formal concept analysis : foundations and applications*. Springer, 2005.

Paper 139

Fostering Open Science and FAIR practices among the IFB infrastructures : the OpenLink and maDMP4LS projects

Olivier COLLIN[1] , Julien SEILER[2] and Laurent BOURI[2]

[1] Univ Rennes, Inria, CNRS, IRISA, 3500 Rennes, France

[2] IGBMC, 1 Rue Laurent Fries, 67400, Illkirch-Graffenstaden, France

Corresponding Author: julien.seiler@igbmc.fr

Abstract *The IFB and its partners are the bearers of numerous initiatives to accompany biologists through the new challenges of open science. This federation leads project to build automated process around the Data Management Plan (DMP). DMP is a static document, initiated at the beginning of a project, keeping tracks of his different stages. This central role enforce the necessity of making the DMP active and machine actionable. The goals of maDMP4LS and OpenLink projects, both starting in early 2020 (the 1st of april for maDMP4LS and the 1st of february for openlink), are to set up dashboards and automatic procedures to support researchers in data management and guide them towards the adoption of a FAIR approach.*

Keywords Data Management plan, Open Science, FAIR.

1. Introduction

The French Bioinformatics Institute (IFB), with its two computing infrastructures (NNCR-cluster and NNCR-cloud) and its 30 member core facilities, is an essential structure for Life Sciences, providing a production, analysis and management environment for the biology and medical biology communities.

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, IFB and its partners have launched two projects that have been selected during the "ANR Flash Données Ouvertes": OpenLink and maDMP4LS. These two projects, starting in early 2020 for 18 months, rely on the infrastructure federation supported by the IFB for their development.

2. Federation and mutualisation of infrastructure

Since February 2018, the IFB has adopted a new organisational mode aimed at federating all its infrastructures. Thus, a collaborative network called the National Network of Computing Resources

(NNCR) brings together engineers and bioinformaticians with the desire to co-construct a computing environment for biology.

This federation aims to be non-intrusive, leaving each partner infrastructure free to adopt the solutions, methodologies or organizational structure proposed by the IFB according to their needs.

This federation makes it possible today to co-construct the tools and solutions that will enable the adoption of the good practices of the FAIR principles and to accompany biologists towards open sciences. Actors of this federation are now leading innovation projects to build automated process around the Data Management plan (DMP). Taking advantage of digital technologies, automatization of data management, and data transfer between data production sites and bioinformatics facilities, it becomes conceivable to build an integrated system where DMP and tools interact to create a data continuum for Life Sciences.

The DMP, the founding element of data managementA key tool in the adoption of FAIR practice and principles is the Data Management Plan (DMP). It helps reflecting, anticipating and recording the decisions made regarding the different issues of scientific output production and management. Initiated at the beginning of a project, it should be updated throughout the project and further after, and thus provide a dynamic inventory of the outputs of a project and include some information regarding their provenance and accessibility.

Data Management Plan (DMP) is a static document, often created to initiate a new project, as a part of research practice, in the form of a list of questions and structured answers. Considering that DMP is meant to "provide a dynamic index that articulates the relevant information relating to a project and linkages with its various FAIR components"[2], this gives it a central role in the information exchange and in the coordination of data management implementation and enforces the necessity of rendering DMP active and machine actionable.

3. maDMP4LS : machine actionable DMP for Life Sciences

In partnership with the DMP-OPIDoR team (INIST - CNRS), IFB has launched the maDMP4LS project that will produce a new machine-actionable version of DMP-OPIDoR (maOPIDoR ?). In is first version, this new tool will help to configure the computing environments according to the information contained in the DMP.

This tool will be based on the common DMP model that is being produced by the RDA DMP Common Standard working group. The common model will be extended to meet the needs of DMP OPIDoR user group and IFB. The RDA DMP Common Standard is a minimum set of universal terms which ensure basic interoperability between systems producing or consuming machine-actionable data management plans.

The DMP information will be used on computational infrastructures to help managing the scientific data: linking data with scientific projects, determining storage requirements, defining access policies, assessing the fate of the data, etc. This will be achieved with an API to retrieve DMP information for the automated processes (projects storage space management) on the research environments.

In a second time, information pertaining to the various processes applied to the data will be pushed to the machine-actionable DMP.

4. 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.

The preliminary work of the OpenLink project has made it possible to create a matrix describing the data that is automatically accessible for numerous tools that support research projects (data management plan manager, electronic laboratory notebook, data visualization platform, analysis tools, data storage service, institutional repositories).

This matrix describes a fertile ground for the implementation of "machine actionable" tools to support researchers from the definition of their data management plan to the publication of data and results in open science by removing the many technical barriers related to the interoperability of tools.

Transversal metadata described in this matrix 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.

5. Conclusion

The aim of the maDMP4LS and OpenLink projects is to set up dashboards and automatic procedures to support researchers in data management and guide them towards the adoption of a FAIR approach compatible with the commitments made by the Ministry of Research in favour of open science. The FAIR approach has to be initiated from the very beginning of each research project through the elaboration of a Data Management Plan.

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

Paper 159

Designing a statistical and user friendly application to analyse cell imaging

Rachel Torchet[1], Stevenn Volant[1], Pascal Campagne[1], Maxime Mistretta[2] and Guilia Manina[2]

[1]Hub de Bioinformatique et Biostatistique – Département Biologie Computationnelle, Institut Pasteur, USR 3756 CNRS, Paris, France

[2]Microbial Individuality and Infection,Department Of Cell Biology & Infection,Paris, France

Corresponding Author: rachel.torchet@pasteur.fr

Paper Reference: **Manina *et al.* (2019) Preexisting variation in DNA damage response predicts the fate of single mycobacteria under stress, The EMBO Journal, 2019, https://doi.org/10.15252/embj.2019101876**

Clonal microbial populations are inherently heterogeneous, and such a phenotypic diversity is often considered an adaptation strategy. In clinical infections, like in mycobacterium tuberculosis case, phenotypic diversity has been found to be associated with drug tolerance, which may favour genetic resistance evolution. In order to study phenotypic variation in bacteria, a microfluidic system is being developed. The purpose is to track single cells by live-cell fluorescence imaging and to carry out a screening at the single-cell level. A direct application is to screen molecules that homogenize bacterial phenotypes, in order to enhance the effectiveness of standard treatments.

With a total of almost 15000 pictures taken during a single experiment, the task of image processing is way too huge to be manually handled on a daily basis. Analyzing all those pictures by hand could be painful and time consuming. To palliate the issue, we propose a semi-automated approach that combines both image processing and statistical analysis of extracted data through a wizard-based application to alleviate the users'workload.

Throughout the process of development we adopted a User-Centered Design (UCD) approach, where the needs of users are primarily considered from start. At early stages, the focus is on understanding users behavior, needs, and goals. During the user research phase, shadowing workshops and interviews are organized both to identify the different pain points in the user journey and find out a few opportunities to ease the process. We also benchmarked existing cell-imaging softwares used by scientists like ImageJ or Fiji to create a baseline for a better understanding of the current user experience. In our application, the succession of screens guides the user from metadata up to results visualization through different steps.

(1) In the first steps, the user provides information about its experiment and imports the microscope output *.dv* files into the application. The tedious task of extracting, naming and organizing files according to the corresponding experimental treatment is then automatically executed. A key issue in image processing is to detect colonies of bacteria embedded in noisy background. (2) The next step therefore consists in building a mask of the colonies using the control fluorescent reporter: a binary image of the mask is thus created at time point by thresholding the kernel density. Further filtering and cleaning are required to remove irrelevant scories and patches in the mask object by using both heuristic criteria (distance to the center, patches on border, size of the patches…) and a shape complexity index. The remaining colony mask is then refined by trimming pixels at its edges . (3) Colonies that do not exhibit an exponential growth are discarded and outlier time-points within sets of observations are identified with a robust linear regression. (4) After automated filtering the user is invited to manually control the filtering among pictures that should be kept or discarded. (5) Robust statistical parameters (such as mean, variance,etc.) are then computed based on the posterior probability of a two-components mixture gaussian model and interactive visualizations of the results are provided.

Considering the tremendous number of pictures to process and analyze we invested our efforts and imagination to design an interface requesting minimal participation from users that reduce significantly errors and time consumption for biologists.

Paper 165

What's new on IFB NNCR Cluster(s) ?

David Benaben[1,2], Nicole Charrière[3], David Christiany[3], François Gerbes[3], Jean-Christophe Haessig[4], Didier Laborie[5], Gildas Le Corguillé[6*], Olivier Sallou[7], Julien Seiler[4*] and Guillaume Seith[4]

[1] CBiB, Université de Bordeaux, 142 rue Léo Saignat, 33076 Bordeaux, France

[2] INRAE, UMR 1332, Biologie du Fruit et Pathologie, CS20032 Villenave d'Ornon, France

[3] IFB/Institut Français de Bioinformatique, CNRS UMS 3601, IFB-Core, Génoscope, 91057, ÉVRY, France

[4] IGBMC, 1 rue Laurent Fries, 67404, Illkirch, France

[5] GenoToul-Bioinfo, INRAE, 24 chemin de Borde-Rouge, Auzeville, 31326 Castenet-Tolosan, France

[6] Sorbonne Université/CNRS, FR2424, ABiMS, Station Biologique, 29680, Roscoff, France

[7] IRISA/Université Rennes 1, 263 Avenue Général Leclerc, 35000 Rennes, France

* Corresponding Author: lecorguille@sb-roscoff.fr, julien.seiler@igbmc.fr

https://www.france-bioinformatique.fr/en/cluster

Since November 2018, the IFB (Institut Français de Bioinformatique) has deployed, in addition to the Cloud infrastructure, a central HPC (High Performance Computer) computing resource: the IFB Core Cluster. This resource, hosted at the IDRIS datacenter, offers a capacity of 4000 cores (HT - Hyper Thread) and 1 Po of storage. It is implemented and operated by a collective of about ten regional platform engineers from the entire network of IFB platforms ("mutualised task force") who dedicate a percentage of their time to the development of this common project.

The IFB federation NNCR Cluster

The IFB Core Cluster, a Core resource, has always been designed to take part in a set, the NNCR for National Network of Computational Resources. Indeed, 6 open infrastructures have been implemented on the regional platforms for many years now. The initial wish was to build a "federation" of clusters. And even if this term "federation" can be confusing, we imagined to set up a certain (à la carte) harmonization of practices, technologies ... to ultimately offer users a unified experience from one cluster to another. The other interest is the sharing of administration recipes, user documentation in order to reduce the individual maintenance costs.

Thus, since the end of 2019, a certain number of advances are to be noted:

- From a single git repository (gitlab.com/ifb-elixirfr/cluster/tools), the installation of tools (Conda packages and Singularity images) is done simultaneously on the IFB Core Cluster and on the ABiMS and IGBMC/BISTRO platforms.
- The *My* account manager, developed by the GenOuest platform, has been ported to the IFB Core Cluster and should be implemented on the ABiMS platform in the near future.
- Several Ansible roles developed by the TaskForce are now played on several IFB infrastructures. In particular, the Ansible role for the deployment of the Slurm Scheduler developed by the IGBMC platform for the IFB Core Cluster allowed the implementation of a new Slurm cluster on the ABiMS platform in just 2 days.

usegalaxy.fr

Since the beginning of 2020, the IFB Core Cluster Task Force proposes a national Galaxy instance usegalaxy.fr in the line of usegalaxy.* (org, eu, org.au, ...). This instance, like the Core Cluster, is intended to be federative. Several managers of local instances have already decided to migrate their users to the national instance. usegalaxy.fr also offers "subdomains" which allow to propose subsets of tools and a dedicated homepage around a theme or a project. Thus, workflow4metabolomics.usegalaxy.fr and

proteore.usegalaxy.fr have joined this instance. The goal is once again to pool efforts to offer a service of the highest quality, while maintaining existing thematic identities.
Access is finally facilitated by 3 connection modes: anonymous mode, self-registration and authentication via the ELIXIR AAI authentication portal (eduGAIN, ORCID, Google or LinkedIn). The instance already provides more than 400 tools.
The instance itself is like the IFB Core Cluster and usegalaxy.eu fully managed via Ansible and Continuous Integration (CI) processes. Thus the contributions to the project can be done in an open and secure way via gitlab.com/ifb-elixirfr/usegalaxy-fr: instance tuning, installation of banks and tools... To guarantee a quality of service, tests on a set of workflows provided by the Galaxy Training Network are launched periodically.

Finally, it is interesting to note that all these developments have in common the fact that they were born within the platforms, were then refactorized, generalized and deployed on the IFB Core Cluster to finally be redistributed to other regional platforms.

Paper 170

Inferring biochemical reactions and metabolite structures using a molecular transformation approach

Arnaud BELCOUR[1], Jacques NICOLAS[1], Anne SIEGEL[1] and Gabriel MARKOV[2]

[1]Univ Rennes 1, Inria, CNRS, Irisa, 35052, Rennes, France

[2] CNRS - Sorbonne Université - Integrative Biology of Marine Models (UMR8227) - Station Biologique de Roscoff, Place Georges Teissier, 29680, Roscoff, France

Corresponding Author: gabriel.markov@sb-roscoff.fr

Paper Reference: **Belcour *et al.* (2020) Inferring Biochemical Reactions and Metabolite Structures to Understand Metabolic Pathway Drift, iScience 23, 2020, 100849.**
https://doi.org/10.1016/j.isci.2020.100849

Integrating large-scale mass spectrometry data into genome-scale metabolic networks is challenged by knowledge gaps, even in the most studied model organisms [1]. In emerging model organisms, that constitute the major part of biodiversity, the issue is even more acute, due to structural variation within biochemical pathways during evolution [2]. Therefore, specific bioinformatic tools are necessary to infer experimentally testable biochemical reactions and metabolic structures. We developped such an approach, abstracting molecular transformations from known biochemical reactions. Then, those molecular transformations were used either for connecting known metabolites to partially known pathways, or to infer new metabolite structures corresponding to unannotated metabolites with a known mass-to-charge ratio. As a proof of concept, we implemented this approach into the pathmodel program [3], using data from two pathways in a model red algal, and got inferences that are consistent with experimental data. Specifically, one of the two metabolite structures predicted from known mass-to-charge ratios was identical to a molecule recently identified in other red algae [4]. Generalizing our approach to scale it up will necessitate further optimization of atom mapping procedures to enable comparability of identical molecular transformations carried on different molecule families, and in line with this, further refining of the Enzyme Commission classification [5].

Acknowledgements

This research received funding from the French Government via the National Research Agency investment expenditure program IDEALG (ANR–10–BTBR–04) and from Région Bretagne via the grant « SAD 2016 – METALG (9673) »

References

[1] Clément Frainay, Emma L. Schymanski, Steffen Neumann, Benjamin Merlet, Reza M. Salek, Fabien Jourdan and Oscar Yanes. Mind the Gap: Mapping Mass Spectral Databases in Genome-Scale Metabolic Networks Reveals Poorly Covered Areas. *Metabolites,* 8, 2018.

[2] Eric S. Haag and John R. True. Developmental System Drift *in* Nuno de la Rosa, L. & Müller, G. *(Eds.) Evolutionary Developmental Biology: A Reference Guide, Springer International Publishing,* pages 1-12, 2018.

[3] https://github.com/pathmodel

[4] Maria Orfanoudaki, Anja Hartmann, Ulf Karsten, and Markus Ganzera. Chemical profiling of mycosporine- like amino acids in twenty- three red algal species. *Journal of Phycology*, 55: 393-403, 2019.

[5] Andrew G. McDonald and Keith F. Tipton. Fifty-five years of enzyme classification: advances and difficulties *The FEBS journal,* 281, pages 583-592, 2014.

Paper 180

Overcoming uncollapsed haplotypes in long-read assemblies of non-model organisms

Nadège Guiglielmoni[1], Antoine Houtain[2], Alessandro Derzelle[2], Karine van Doninck[2] and Jean-François Flot[1,3]

[1] Service Evolution Biologique et Ecologie, Université libre de Bruxelles, 1050 Brussels, Belgium
[2] Laboratoire d'Ecologie et Génétique Evolutive, Université de Namur, 5000 Namur, Belgium
[3] Interuniversity Institute of Bioinformatics in Brussels - (IB)^2, 1050 Brussels, Belgium

Corresponding author: nadege.guiglielmoni@ulb.be

Abstract *Third-generation sequencing, also called long-read sequencing, has revolutionized genome assembly: as PacBio and Nanopore technologies have become more accessible in technicity and in cost (with decreasing error rates and increasing read lengths), long-read assemblers have flourished and are starting to deliver chromosome-level assemblies. However, an independent, comparative assessment of the performance of these programs on a common, real-life dataset is still lacking.*

To fill this gap, we tested the efficiency of long-read assemblers on the genome of the rotifer Adineta vaga, a non-model organism for which both PacBio and Nanopore reads were available. Although all the assemblers included in our benchmark aimed to produce a haploid genome assembly with collapsed haplotypes, we observed strikingly different behaviors of these assemblers on highly heterozygous regions: allelic regions that were most divergent were sometimes not merged, resulting in variable amounts of duplicated regions. We identified three strategies to alleviate this problem: setting a read-length threshold to filter out shorter reads; choosing an assembler less prone to retaining uncollapsed haplotypes; and post-processing the assembled set of contigs using a downstream tool to remove uncollapsed haplotypes. These three strategies are not mutually exclusive and, when combined, generate haploid assemblies with genome sizes, coverage distributions, and k-mer completeness matching expectations.

Keywords genome assembly, long reads, benchmark, heterozygosity

Introduction

With the advent of long-read sequencing, high-quality assemblies are now commonly achieved on all types of organisms. The competition between the two main long-read sequencing companies, Pacific Biosciences (PacBio) and Oxford Nanopore Technologies (Nanopore), has prompted an increase in output as well as a decrease in cost, making their technologies more accessible to research teams and more applicable to challenging genomes. The main advantage of long reads over short reads (such as those generated by Illumina sequencing platforms) is their typical length that averages around 10 kilobases (kb) [1]. Their length facilitates genome assembly into contigs and scaffolds as they can span repetitive regions [2]; they can also be used to resolve haplotypes [3].

However, long reads have a much higher error rate than Illumina data, and these errors are mainly insertions and deletions for long reads vs. substitutions for Illumina reads. PacBio data have a random error pattern that can be compensated with high coverage: as a result, reading the same DNA regions over and over several times can be used to generate a consensus with an accuracy close to 99%, in a process dubbed Circular Consensus Sequencing (CSS) and marketed as PacBio HiFi (standing for "high-fidelity")[4]. Nanopore reads, on the other hand, have systematic errors in homopolymeric regions and are thus often combined with Illumina sequencing to correct the errors still present in contigs, in a process called "polishing" [5][6]. Nanopore reads keep getting longer, with runs attaining N50s over 100 kilobases (kb) and longest reads spanning over 1 Megabase (Mb) [7][8].

This progress has prompted the development of many programs to produce *de novo* assemblies from long reads, all of which follow the Overlap Layout Consensus (OLC) paradigm [9]. Briefly, OLC methods start by building an overlap graph (the "O" step), then simplify it and clean it by applying

various heuristics (the "L" step), and finally compute the consensus sequence of each contig (the "C" step). Some long-read assemblers follow strictly this paradigm, such as Flye [10], Ra [11], Raven [11] (a further development of Ra by the same author), Shasta [12] and wtdbg2 [13]; whereas others such as Canu [14] and NextDeNovo add a preliminary correction step based on an all-versus-all alignment of the reads.

Long-read assemblers were recently benchmarked on real and simulated PacBio and Nanopore bacterial datasets [15], and all assemblers tested proved their efficiency at reconstructing full microbial genomes within one hour and with a low RAM usage. The Flye publication [10] provides an evaluation of Canu, Flye, Ra and wtdbg2 on several eukaryotic genomes. However, more complete evaluations of these tools on their ability to provide structurally correct (i.e., without artefactually duplicated genome regions) haploid assemblies from non-model diploid organisms are still lacking. To fill this gap, we present here a quantitative and qualitative assessment of seven long-read assemblers on a relatively small eukaryotic genome, *Adineta vaga*, for which a short-read, fragmented assembly was published some years ago [16]. As with most non-model organisms, *Adineta vaga*'s genome presents a mid-range heterozygosity of ca. 2% with a mix of highly heterozygous and low-heterozygosity regions, making such genome more challenging to assemble than those of model organisms that have often a very low level of polymorphism [17]. Assemblies were evaluated with several measures to assess for contiguity, quality and proper haplotype collapsing:

- assembly size: the sum of the lengths of all the contigs in the assemblies;
- N50: the largest contig length for which 50% of the assembly size occur in fragments equal or greater in length;
- BUSCO score: the number of features from a set of orthologs retrieved completely in the assembly, in single-copy or duplicated;
- k-mer completeness: the percentage of k-length words frequently observed in a low error-rate set of reads that are present in the assembly;
- coverage: the number of reads covering a given position in a contig (calculated after mapping reads on the assembly).

Results

Assemblies of PacBio reads

The PacBio assemblies have variable lengths, ranging from 89 Mb (Shasta, reads > 15 kb) to 169 Mb (Canu, all reads). Larger assemblies also present higher numbers of duplicated BUSCOs as well as higher k-mer completeness and bimodal coverage distributions (Figure 1). N50s range from 301 kb (Canu, all reads) to 12 Mb (NextDeNovo, all reads).

Canu produced the largest assemblies, between 147 Mb and 169 Mb, as well as the highest number of duplicated BUSCOs. Besides, its k-mer completeness is systematically higher than the expected value of 50% for a haploid assembly of a diploid genome. Canu's assembly coverage distributions exhibit two peaks around 100X and 210X. The half-coverage peak represents allelic regions that have not been collapsed, resulting in artefactually duplicated regions in the assembly.

Flye, NextDeNovo, Raven and Shasta assemblies also have two peaks in their coverage distribution, although the 100X peak is smaller than with Canu, and a third low-coverage peak for Raven and Shasta. Flye assemblies all have similar lengths, BUSCO score, k-mer completeness and coverage distribution regardless of the read-length threshold used; however, their N50 decreases with increasing read-length threshold. NextDeNovo, Raven and Shasta, on the other hand, produced shorter assemblies when smaller reads were removed. The 100X peak is absent in NextDeNovo and Shasta assemblies when selecting reads superior to 15 kb. While the N50 of NextDeNovo assemblies decreases with read selection, it remains close to other assemblers. The quality of Shasta assemblies is greatly diminished, but this is likely due to a lower reads coverage.

Ra and wtdbg2 both produced assemblies with very small 100X peaks. The sizes of the Ra assemblies decreased when using higher read-length threshold, with low-coverage contigs disappearing from the assembly in a fashion similar to Raven; their contig N50, however, remained fairly constant.

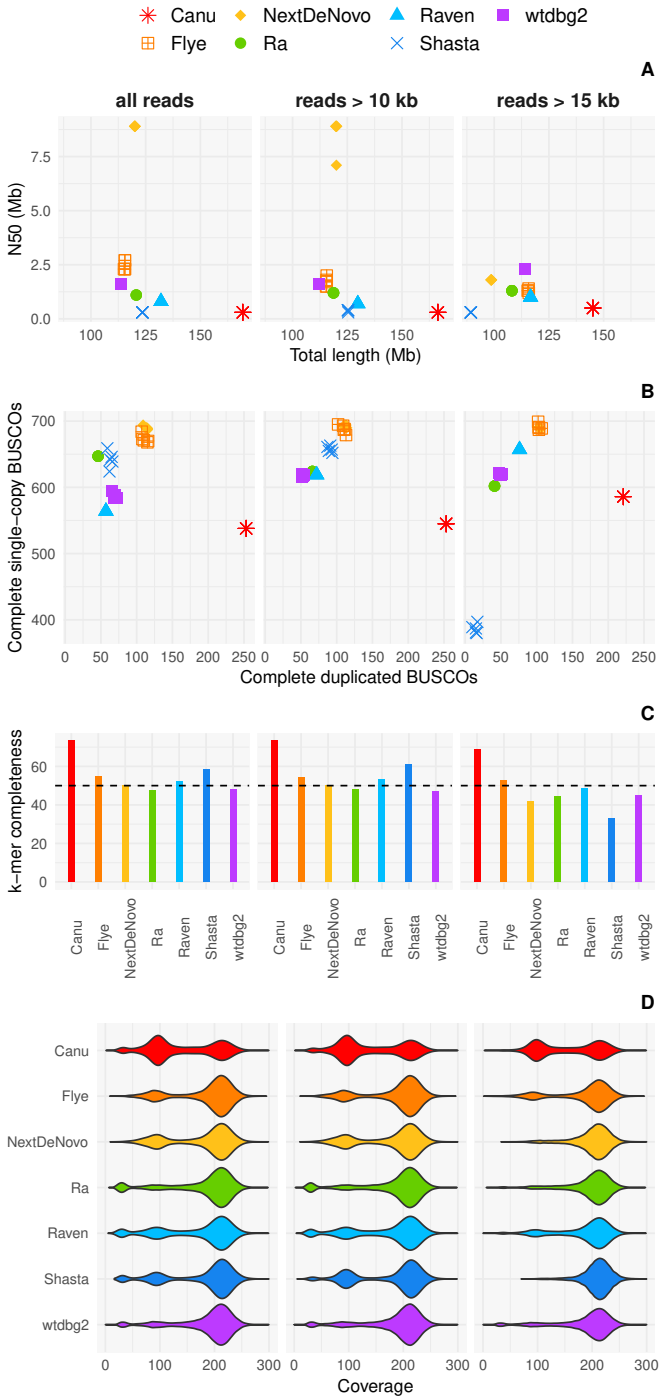


Fig. 1. Statistics of the PacBio assemblies obtained, with A) N50 plotted against total assembly length, B) number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, C) mean *k*-mer completeness and D) long-reads coverage distribution of the contigs of one replicate assembly for each program. In A and B, each program is represented by five points that correspond to five replicate runs.

By contrast, assembly size did not vary for wtdbg2 when using higher read-length threshold but contig N50 increased.

Assemblies of Nanopore reads

Similarly to the PacBio assemblies, Nanopore assembly sizes range from 93 Mb (Ra, reads > 40 kb) to 154 Mb (Canu, reads > 10 kb) (Figure 2). Except when a stringent read-length threshold is applied, all assemblies exceed the expected size of 102.3 Mb. Nanopore assemblies generally achieve a much higher contiguity than PacBio assemblies: while PacBio assemblies yield a highest N50 of about 2.5 Mb, with the exception of NextDeNovo assemblies, Nanopore assemblies reach 12.5 Mb (wtdbg2, reads > 30 kb). The number of complete single-copy BUSCOs is lower in Nanopore assemblies (up to 600) than in PacBio assemblies (up to 700), and the number of duplicate complete BUSCOs is also much smaller (up to 50 with Nanopore vs. up to 250 with PacBio). These lower BUSCO scores, together with the lower *k*-mer completeness of Nanopore assemblies, likely result from the non-random error pattern of Nanopore reads that produces errors (mostly indels) in the consensus sequences produced by the assemblers.

As for PacBio reads, Canu assemblies of Nanopore reads are oversized and the coverage distribution shows two distinct peaks around 75X and 160X, indicating that many haplotypes have not been collapsed and remain present in two copies in the assemblies. Flye, NextDeNovo, Raven and Shasta also present two peaks, and Raven and Shasta have a third peak corresponding to low-coverage contigs. Although Raven's N50 increases when shorter reads were filtered out, this is not the case for Flye, NextDeNovo and Shasta. Raven assemblies improve also with increasing read-length threshold: low-coverage contigs disappear, assembly length becomes closer to expectations, and BUSCO score increases. Likewise, Ra assemblies improved with read selection: when using only reads longer than 30 kb, the low-coverage and half-coverage regions disappeared completely. However, when keeping only reads over 40 kb, the assembly N50 decreased and genome size dropped under the expected value.

Purging duplicated regions

As our long-read assemblies contained various amount of uncollapsed haplotypes, resulting in assemblies larger than expected, we further tested the possibility of improving these assemblies by collapsing haplotypes *a posteriori* with purge_haplotigs [18]. This tool relies on coverage distribution, that proved in our analysis a key aspect to identify uncollapsed haplotypes. We tested this program on assemblies produced with all reads, as they were systematically oversized, and on only one replicate run for each assembler tested (as the replicates showed minimal differences).

After purging duplicated regions, all assemblies improved except for Flye and NextDeNovo that did not change much (Figure 3): N50s remained stable, but genome sizes became closer to the expected values. *k*-mer completeness also decreased and became closer to the expected 50%, except for the Canu assembly of PacBio reads. None of the purged assemblies exhibited any low-coverage contigs, and half-coverage peaks were reduced. Ra, Raven and wtdbg2 assemblies after haplotype purging were similar to those obtained with a high read-length threshold, although low-coverage contigs were better removed from the wtdbg2 assembly by purge_haplotigs.

Computational performance

As computational resources can be a limiting factor in genome assembly, we provide CPU time and RAM measurements for all the assemblers, except Canu and NextDeNovo that required significantly higher resources and were therefore run on a computer cluster (Figure 4). The assembler with the smallest resource consumption was wtdbg2: it required the lowest amount of RAM, ran the fastest on PacBio reads and was also time-efficient on Nanopore reads. As would be expected, using a read-length threshold improved RAM usage and CPU time, with the exception of Flye that still required a high amount of RAM for Nanopore reads. Shasta ran fast but required the largest amount of RAM for the full Nanopore dataset. Ra and Raven used only a limited amount of RAM, with a peak usage at 50 GB.

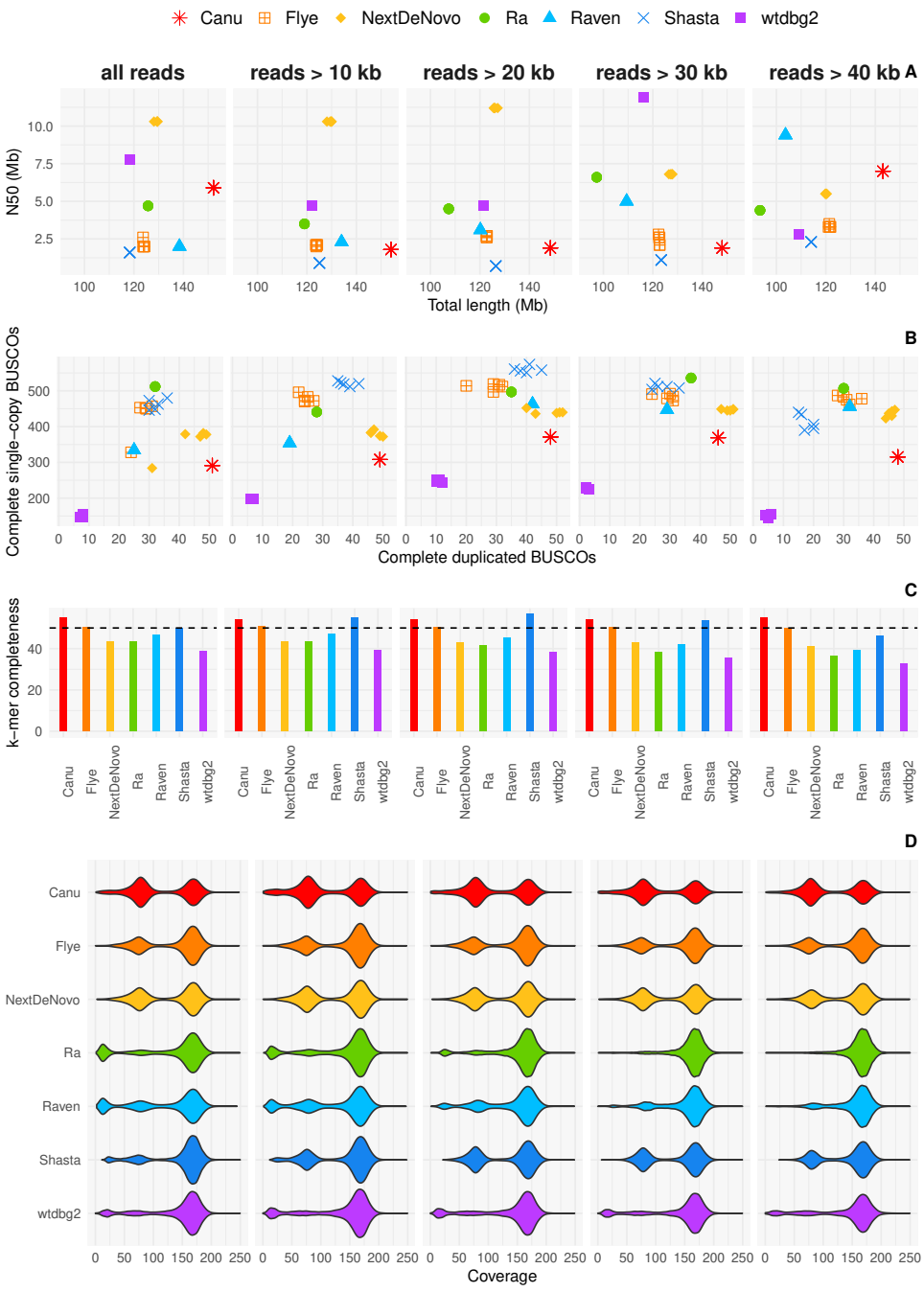


Fig. 2. Statistics of Nanopore assemblies, with with A) N50 plotted against total assembly length, B) number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, C) mean *k*-mer completeness and D) long-reads coverage distribution of the contigs of one replicate assembly for each program. In A and B, each program is represented by five points that correspond to five replicate runs.

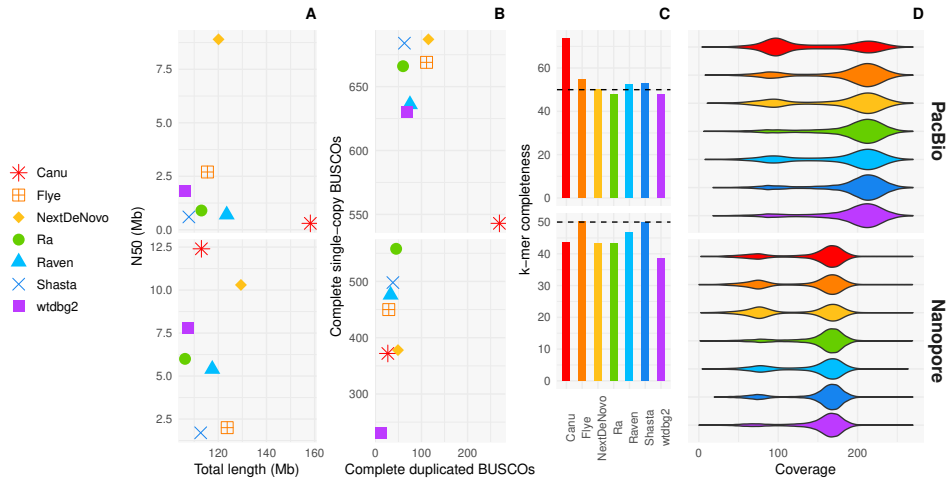


Fig. 3. Statistics of PacBio and Nanopore assemblies after purge_haplotigs, with A) N50 plotted against total assembly length, B) number of complete single-copy BUSCOs plotted against number of complete duplicated BUSCOs, C) mean *k*-mer completeness and D) long-reads coverage distribution of the contigs.

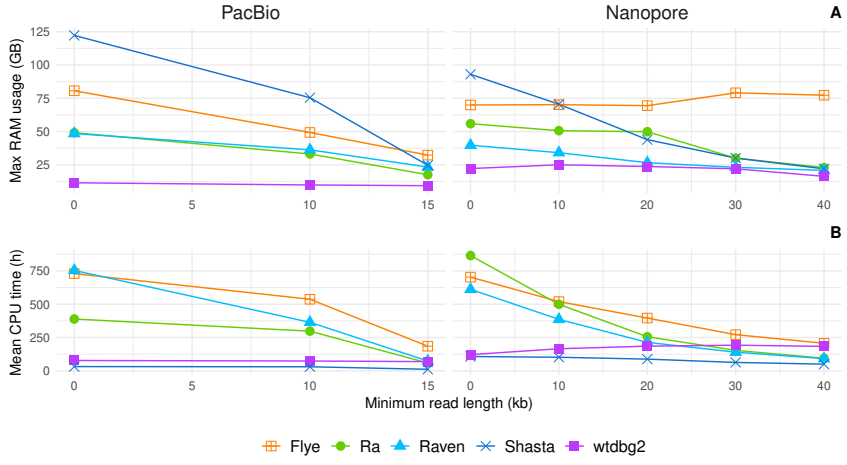


Fig. 4. Computational resources (RAM and CPU time) used by the assemblers with the full Nanopore and PacBio datasets and the subsets. Canu and NextDeNovo are not included as they were run on a cluster.

Discussion

While PacBio assemblies were superior in terms of BUSCO scores and *k*-mer completeness, the contiguity of Nanopore assemblies was far greater for most assemblers. These results are coherent with the characteristics of these reads. An important finding is that keeping only reads longer than a given threshold improves in many cases the quality of haploid assemblies.

We found that Canu poorly collapses allelic regions and yields oversized assemblies. The program does not seem adequate to solve highly divergent regions on its own, but haplotype collapsing was improved on a Nanopore assembly combined with purge_haplotigs. Flye assemblies also exhibited uncollapsed haplotypes; selecting the longest reads did not help and neither did purge_haplotigs. Still, Flye exhibited both good contiguity and good quality. These two assemblers are likely better designed for separating haplotypes. wtdbg2 performed well on PacBio data, but less on Nanopore reads. This program did not seem to have difficulty with heterozygous regions but was rather affected by low-coverage contigs. Read selection on size did not significantly improve the assemblies, but purge_haplotigs removed low-coverage contigs, therefore improving the output. Although Shasta was less good in collapsing divergent haplotypes (and neither read selection nor purge_haplotype helped with that) and did not achieve particularly high contiguity, its *k*-mer completeness and BUSCO scores were very good. NextDeNovo produced highly contiguous assemblies, but with poorly collapsed haplotypes. This flaw was improved however on PacBio assemblies when selecting the longest reads.

Ra and Raven performed better on size-selected reads, which led to smaller genome sizes closer to expectation. With these assemblers, purge_haplotigs was also efficient at purging uncollapsed haplotypes. While Ra and Raven both achieved convincing contiguity and quality, Ra proved more efficient at producing a haploid assembly. Both Ra and wtdbg2 stood out among all the assemblers as the ones less prone to retain uncollapsed haplotypes.

We believe that this benchmark will help researchers working on non-model organisms select a long-read sequencing technology and an assembly method suitable for their project, and will also help them better understand the resulting assemblies.

Material & Methods

The genome size of *Adineta vaga* was estimated using KAT [19] on an Illumina dataset of 25 millions paired-end 250 basepairs (bp) reads. The diploid size was estimated to 204.6 Mb, therefore a haploid assembly should have a length around 102.3 Mb.

Canu, Flye, NextDeNovo, Ra, Raven, Shasta and wtdbg2 were tested on two *Adineta vaga* long-read datasets: PacBio reads totalling 23.5 Gb with a N50 of 11.6 kb; and Nanopore reads totalling 17.5 Gb with a N50 of 18.8 kb (after trimming using Porechop, `github.com/rrwick/Porechop`). All assemblers were used with default parameters, except for Shasta for which the minimum read length was set to zero (instead of the default 10 kb setting) and parameters recommended on the github repository were used for PacBio assemblies. When assemblers required an estimated size, the value 100 Mb was provided. PacBio assemblies were run on all reads, on reads > 10 kb (14.4 Gb) and on reads > 15 kb (4.7 Gb). Nanopore assemblies were run on all reads, on reads > 10 kb (13,3 Gb), on reads > 20 kb (8.3 Gb), on reads > 30 kb (5.7 Gb) and on reads > 40 kb (4.10 Gb). To test for reproducibility, all assemblers were run five times.

To run purge_haplotigs, reads were mapped to contigs using minimap2 [20] and we then computed coverage histograms with `purge_haplotigs hist`, that we used to set low, mid and high cutoffs; these values were then used by `purge_haplotigs cov` to detect suspect contigs. Finally, we ran `purge_haplotigs purge` to eliminate duplicated regions.

To evaluate the assemblies, we ran BUSCO 4 [21] against metazoa odb10 (954 features) without the parameter –long. We ran `KAT comp` [19] to calculate *k*-mer completeness by reference to the same Illumina 2*250 bp dataset used to estimate the genome size. To compute coverage, long reads were mapped on one replicate assembly per assembler using minimap2 and the coverage was computed with tinycov, available at `github.com/cmdoret/tinycov`, with a window size of 20 kb.

For Flye, Ra, Raven, Shasta and wtdbg2, maximal RAM usage and mean CPU time were measured using the command `time` with 14 threads on a computer with an i9-9900X 3.5 Ghz processor and 128 GB RAM. Canu and NextDeNovo were run on different machines as the compute time was too long (Canu) or the RAM usage was too high (NextDeNovo).

Acknowledgements

Part of this analysis was performed on computing clusters of the Leibniz-Rechenzentrum (LRZ) and the Consortium des Équipements de Calcul Intensif (CéCI). This project is funded by the Horizon 2020 research and innovation program of the European Union under the Marie Skłodowska-Curie grant agreement 764840 (ITN IGNITE, www.itn-ignite.eu).

References

[1] Fritz J. Sedlazeck, Hayan Lee, Charlotte A. Darby, and Michael C. Schatz. Piercing the dark matter: bioinformatics of long-range sequencing and mapping. *Nature Reviews Genetics*, 19(6):329–346, 2018.

[2] Martin O. Pollard, Deepti Gurdasani, Alexander J. Mentzer, Tarryn Porter, and Manjinder S. Sandhu. Long reads: their purpose and place. *Human Molecular Genetics*, 27(R2):R234–R241, 2018.

[3] Murray Patterson et al. WhatsHap: weighted haplotype assembly for future-generation sequencing reads. *Journal of Computational Biology*, 22(6):498–509, 2015.

[4] Aaron M. Wenger et al. Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome. *Nature Biotechnology*, 37(10):1155–1162, 2019.

[5] Ritu Kundu, Joshua Casey, and Wing-kin Sung. HYPO: super fast & accurate polisher for long read assemblies. *bioRxiv*, 2019.

[6] Aleksey V. Zimin and Steven L. Salzberg. The genome polishing tool POLCA makes fast and accurate corrections in genome assemblies. *bioRxiv*, 2019.

[7] Miten Jain et al. Nanopore sequencing and assembly of a human genome with ultra-long reads. *Nature Biotechnology*, 36(4):338–345, 2018.

[8] Karen H. Miga et al. Telomere-to-telomere assembly of a complete human X chromosome. *bioRxiv*, 2019.

[9] Jason R. Miller, Sergey Koren, and Granger Sutton. Assembly algorithms for next-generation sequencing data. *Genomics*, 95(6):315–327, 2010.

[10] 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.

[11] Robert Vaser and Mile Šikić. Yet another de novo genome assembler. *International Symposium on Image and Signal Processing and Analysis, ISPA*, pages 147–151, 2019.

[12] Kishwar Shafin et al. Efficient de novo assembly of eleven human genomes using PromethION sequencing and a novel nanopore toolkit. *bioRxiv*, 2019.

[13] Jue Ruan and Heng Li. Fast and accurate long-read assembly with wtdbg2. *Nature Methods*, 17(2):155–158, 2020.

[14] Sergey Koren et al. Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation. *Genome Research*, 25(2):1–11, 2017.

[15] Ryan R. Wick and Kathryn E. Holt. Benchmarking of long-read assemblers for prokaryote whole genome sequencing. *F1000Research*, 8:2138, 2019.

[16] Jean-François Flot et al. Genomic evidence for ameiotic evolution in the bdelloid rotifer *Adineta vaga*. *Nature*, 500(7463):453–457, 2013.

[17] Ellen M. Leffler et al. Revisiting an old riddle: what determines genetic diversity levels within species? *PLoS Biology*, 10(9):e1001388, 2012.

[18] Michael J Roach, Simon A Schmidt, and Anthony R Borneman. Purge Haplotigs: allelic contig reassignment for third-gen diploid genome assemblies. *BMC Bioinformatics*, 19(1):1–10, 2018.

[19] Daniel Mapleson, Gonzalo Garcia Accinelli, George Kettleborough, Jonathan Wright, and Bernardo J. Clavijo. KAT: a K-mer analysis toolkit to quality control NGS datasets and genome assemblies. *Bioinformatics*, 33(4):574–576, 2016.

[20] Heng Li. Sequence analysis Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*, 34(18):3094–3100, 2018.

[21] 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.

Paper 186

Quantifying transmission dynamics of acute hepatitis C virus infections in a heterogeneous population using sequence data

Gonché Danesh[1], Victor Virlogeux[2], Christophe Ramière[3], Caroline Charre[3], Laurent Cotte[4] and Samuel Alizon[1]

[1] MIVEGEC (UMR CNRS 5290, IRD, UM), Montpellier, France
[2] Clinical Research Center, Croix-Rousse Hospital, Hospices Civils de Lyon, France
[3] Virology Laboratory, Croix-Rousse Hospital, Hospices Civils de Lyon, France
[4] Infectious Diseases Department, Croix-Rousse Hospital, Hospices Civils de Lyon, France

Corresponding author: gonche.danesh@ird.fr

Abstract *Opioid substitution and syringes exchange programs have drastically reduced hepatitis C virus (HCV) spread in France but HCV sexual transmission in men having sex with men (MSM) has recently arisen as a significant public health concern. The fact that the virus is transmitting in a heterogeneous population, with 'new' and 'classical' hosts, makes prevalence and incidence rates poorly informative. However, additional insights can be gained by analyzing virus phylogenies inferred from dated genetic sequence data. Here, using a phylodynamics approach based on Approximate Bayesian Computation, we estimate key epidemiological parameters of an ongoing HCV epidemic in MSM in Lyon (France). We show that this new epidemics is largely independent from the 'classical' HCV epidemics and that its doubling time is one order of magnitude lower (51 days versus 1.75 years). These results have practical implications for HCV control and illustrate the additional information provided by virus genomics in public health.*

Keywords Phylodynamics, Epidemiology, Hepatitis C, Doubling time, Modelling

Background

It is estimated that 71 million people worldwide suffer from chronic hepatitis C virus (HCV) infections [1,2]. The World Health Organisation (WHO) and several countries have issued recommendations towards the 'elimination' of this virus, which they define as an 80% reduction in new chronic infections and a 65% decline in liver mortality by 2030 [2]. HIV-HCV coinfected patients are targeted with priority because of the shared transmission routes between the two viruses [3] and because of the increased virulence of HCV in coinfections [4–6]. Successful harm reduction interventions, such as needle-syringe exchange and opiate substitution programs, as well as a high level of enrolment into care of HIV-infected patients, have led to a drastic drop in the prevalence of active HCV infections in HIV-HCV coinfected patients in several European countries during the recent years [7–10]. Unfortunately, this elimination goal is challenged by the emergence of HCV sexual transmission, especially among men having sex with men (MSM). This trend is reported to be driven by unprotected sex, drug use in the context of sex ('chemsex'), and potentially traumatic practices such as fisting [11–13]. In area of Lyon (France), HCV incidence has been shown to increase concomitantly with a shift in the profile of infected hosts [14]. Understanding and quantifying this recent increase is the main goal of this study.

Several modeling studies have highlighted the difficulty to control the spread of HCV infections in HIV-infected MSM in the absence of harm reduction interventions [12, 15]. Furthermore, we recently described the spread of HCV from HIV-infected to HIV-negative MSM, using HIV pre-exposure prophylaxis (PrEP) or not, through shared high-risk practices [14]. More generally, an alarming incidence of acute HCV infections in both HIV-infected and PrEP-using MSM was reported in France in 2016-2017 [13]. Additionally, while PrEP-using MSM are regularly screened for HCV, those who are HIV-negative and do not use PrEP may remain undiagnosed and untreated for years. In general, we know little about the population size and practices of HIV-negative MSM who do not use PrEP. All these epidemiological events could jeopardize the goal of HCV elimination by creating a large pool of infected and undiagnosed patients, which could fuel new infections in intersecting populations. Furthermore, the epidemiological dynamics of HCV infection have mostly been studied in intravenous

drug users (IDU) [16–19] and in the general population [20,21]. Results from these populations are not easily transferable to other populations, which calls for a better understanding of the epidemiological characteristics of HCV sexual transmission in MSM.

Given the lack of knowledge about the focal population driving the increase in HCV incidence, we analyse virus sequence data with phylodynamics methods. This research field has been blooming over the last decade and hypothesizes that the way rapidly evolving viruses spread leaves 'footprints' in their genomes [22–24]. By combining mathematical modelling, statistical analyses and phylogenies of infections, where each leaf corresponds to the virus sequence isolated from a patient, current methods can infer key parameters of viral epidemics. This framework has been successfully applied to other HCV epidemics [25–28], but the ongoing one in Lyon is challenging to analyze because the focal population is heterogeneous, with 'classical' hosts (typically HIV-negative patients infected through nosocomial transmission or with a history of opioid intravenous drug use or blood transfusion) and 'new' hosts (both HIV-infected and HIV-negative MSM, detected during or shortly after acute HCV infection phase, potentially using recreational drugs such as cocaine or cathinones). Our phylodynamics analysis relies on an Approximate Bayesian Computation (ABC, [29]) framework that was recently developed and validated [30].

Assuming an epidemiological model with two host types, 'classical' and 'new' (see the Methods), we use dated virus sequences to estimate the date of onset of the HCV epidemics in 'classical' and 'new' hosts, the level of mixing between hosts types, and, for each host type, the duration of the infectious period and the effective reproduction ratio (i.e. the number of secondary infections, [31]). We find that the doubling time of the epidemics is one order of magnitude lower in 'new' than in 'classical' hosts, therefore emphasising the urgent need for public health action.

Results

The phylogeny inferred from the dated virus sequences shows that 'new' hosts (in red) tend to be grouped in clades (Figure 1). This pattern suggests a high degree of assortativity in the epidemics (i.e. hosts tends to infect hosts from the same type). The ABC phylodynamics approach allows us to go beyond a visual description and to quantify several epidemiological parameters.

As for any Bayesian inference method, we need to assume a prior distribution for each parameter. These priors, shown in grey in Figure 2, are voluntarily designed to be large and uniformly distributed so as to be as little informative as possible. We also assume the date of the 'new' hosts epidemics to be posterior to 1997 based on epidemiological data.

The inference method converges towards posterior distributions for each parameter, which are shown in red in Figure 2. The estimate for the origin of the epidemic in 'classical' hosts is $t_0 = 1977$ $[1966; 1981]$ (numbers in brackets indicate the 95% Highest Posterior Density, or HPD). For the 'new' host type, we estimate the epidemic to have started in $t_2 = 2003$ $[2000; 2005]$.

We find the level of assortativity between host types to be high for 'classical' ($a_1 = 0.97$ $[0.91; 0.99]$) as well as for 'new' hosts ($a_2 = 0.88$ $[0.70; 0.99]$). Therefore, hosts mainly infect hosts from the same type and this effect seems even more pronounced for 'classical' hosts.

The phylodynamics approach also allows us to infer the duration of the infectious period for each host type. Assuming that this parameter does not vary over time, we estimate it to be 1.2 years $[0.40; 7.69]$ for 'classical' hosts (parameter $1/\gamma_1$) and 0.4 years $[0.25; 0.78]$ for 'new' hosts (parameter $1/\gamma_2$).

Regarding effective reproduction numbers, i.e. the number of secondary infections caused by a given host over its infectious period, we estimate that of 'classical' hosts to have decreased from $R_0^{(1),t_1} = 3.29$ $[1.2; 6.63]$ to $R_0^{(1),t_2} = 1.47$ $[0.37; 2.67]$ after the introduction of the third generation HCV test in 1997 (paramater t_1). The inference on the differential transmission parameter indicates that HCV transmission rate is $\nu = 7.97$ $[6.01; 9.90]$ times greater from 'new' hosts than from 'classical' hosts. By combining these results (see the Methods), we estimate the effective reproduction number in 'new' hosts to be $R_0^{(2),t_3} = 2.9$ $[0.81; 6.26]$.

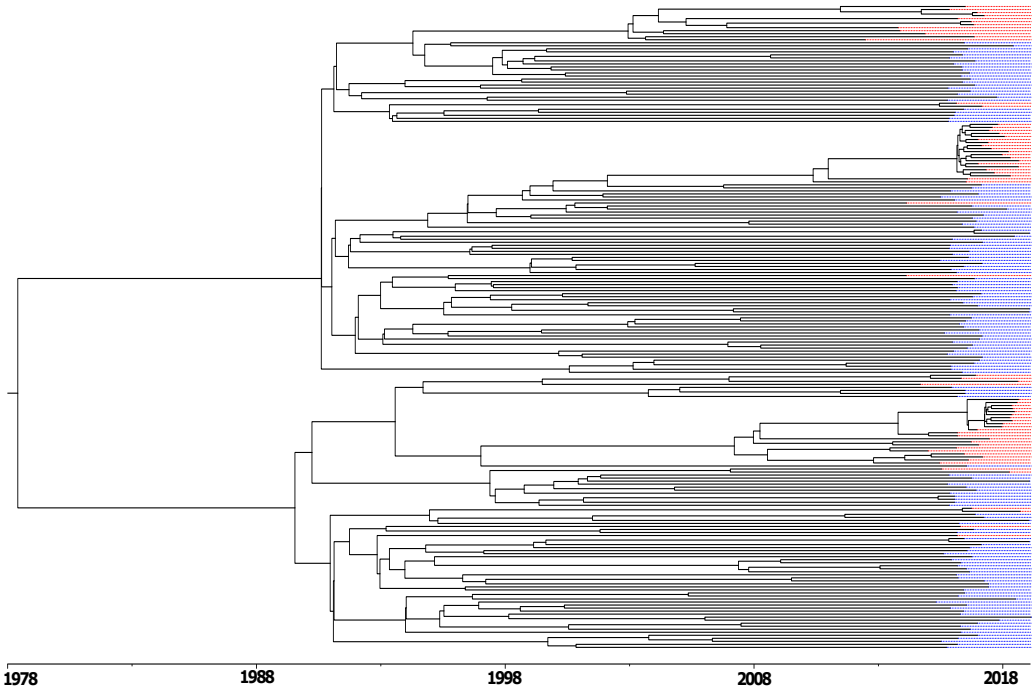


Fig. 1. Phylogeny of HCV infections in the area of Lyon (France). The phylogeny present 213 leaves where 145 of them are associated to the 'classical' hosts and 68 of them to the 'new' hosts. 'Classical' hosts are in blue and 'new' hosts are in red. Sampling events correspond to the end of black branches. The phylogeny was estimated and time-scaled using Bayesian inference (Beast2). See the Methods for additional details.

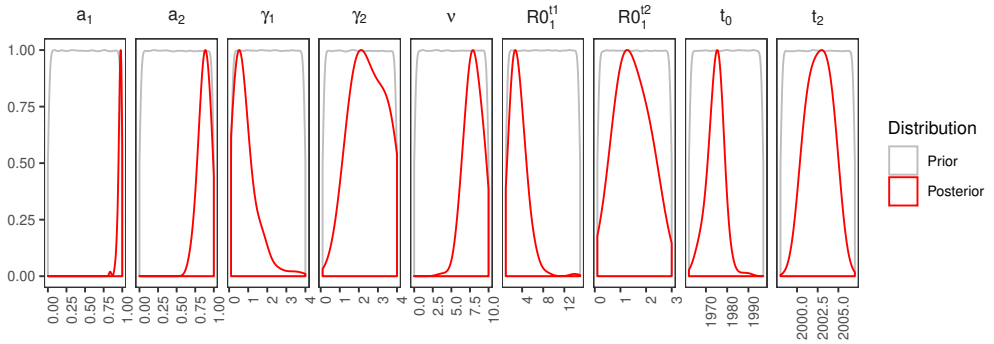


Fig. 2. Parameter prior and posterior distributions. Prior distributions are in grey and posterior distributions inferred by ABC are in red. The thinner the posterior distribution, the more accurate the inference.

To better apprehend the differences between the two host types, we compute the epidemic doubling time (t_D), which is the time for an infected population to double in size. t_D is computed for each type of host, assuming complete assortativity (see the Methods). We find that for the 'classical' hosts, before 1997 $t_D^{(1),t1} \approx 8$ months ($[0.1; 2.63]$ years). After 1997, the pace decreases with a doubling time of $t_D^{(1),t2} \approx 1.75$ years ($[0; 28.55]$ years). For the epidemics in the 'new' hosts, we estimate that $t_D^{(2),t3} \approx 51$ days ($[0; 2.73]$ years).

Discussion

Over the last years, the area of Lyon (France) witnessed an increase in HCV incidence both in HIV-positive and HIV-negative populations of men having sex with men (MSM) [14]. This increase appears to be driven by sexual transmission and echoes similar trends in Amsterdam [32] and in Switzerland [33]. A quantitative analysis of the epidemic is necessary to optimise public health interventions. Unfortunately, this is challenging because the monitoring of the population at risk is limited and because classical tools in quantitative epidemiology, especially incidence time series, are poorly informative with such a heterogeneous population. To circumvent this problem, we used HCV sequence data, which we analysed using phylodynamics. In order to account for host heterogeneity, we extended and validated an existing Approximate Bayesian Computation framework [30].

From a public health point of view, our results have two major implications. First, we find a strong degree of assortativity in both 'classical' and 'new' host populations. The virus phylogeny does hint at this result (Figure 1) but the ABC approach allows us to quantify the pattern and to show that assortativity may be higher for 'classical' hosts. The second main result has to do with the striking difference in doubling times. Indeed, the current spread of the epidemics in 'new' hosts appears to be at least comparable to the spread in the 'classical' hosts in the early 1990s before the advent of the third generation tests. That the duration of the infectious period in 'new' hosts is in the same order of magnitude as the time until treatment suggests that the majority of the transmission events may be occurring during the acute phase. This underlines the necessity to act rapidly upon detection, for instance by emphasising the importance of protection measures such as condom use and by initiating treatment even during the acute phase [34]. A better understanding of the underlying contact networks could provide additional information regarding the structure of the epidemics and, with that respect, next generation sequence data could be particularly informative [35–37].

Some potential limitations of the study are related to the sampling scheme, the assessment of the host type, and the transmission model. Regarding the sampling, the proportion of infected 'new' host that are sampled is unknown but could be high. For the 'classical' hosts, we selected a representative subset of the patients detected in the area but this sampling is likely to be low. However, the effect of underestimating sampling for the new epidemics would be to underestimate its spread, which is already faster than the classical epidemics. In general, implementing a more realistic sampling scheme in the model would be possible but it would require a more detailed model and more data to avoid identifiability issues. Regarding assignment of hosts to one of the two types, this was performed by clinicians independently of the sequence data. The main criterion used was the infection stage (acute or chronic), which was complemented by other epidemiological criteria (history of intravenous drug use, blood transfusion, HIV status). Finally, the 'classical' and the 'new' epidemics appear to be spreading on contact networks with different structures. However, such differences are beyond the level of details of the birth-death model we use here, and would require a larger dataset for them to be inferred.

In order to test whether the infection stage (acute vs. chronic) can explain the data better than the existence of two host types, we developed an alternative model where all infected hosts first go through an acute phase before recovering or progressing to the chronic phase. As for the model with two host types, we used 3 time intervals. Interestingly, it was almost impossible to simulate phylogenies with this model, most likely because of its intrinsic constrains on assortativity (both acute and chronic infections always generate new acute infections).

To our knowledge, few attempts have been made in phylodynamics to tackle the issue of host population heterogeneity. In 2018, a study used the structured coalescent model to investigate the importance of accounting for so-called 'superspreaders' in the recent ebola epidemics in West Africa [38]. The same year, another study used the birth-death model to study the effect of drug resistance mutations on the R_0 of HIV strains [39]. Both of these are implemented in Beast2. However,the birth-death model is unlikely to be directly applicable to our HCV epidemics because it links the two epidemics via mutation (a host of type A becomes a host of type B), whereas in our case the linking is done via transmission (a host of type A infects a host of type B).

Overall, we show that our ABC approach, which we validated for simple epidemiological models such as Susceptible-Infected-Recovered [30], can be applied to more elaborate models that current phylodynamics methods have difficulties to capture. Further increasing the level of details in the model may require to increase the number of simulations but also to introduce new summary statistics. Another promising perspective would be to combine sequence and incidence data. Although this could not be done here due to the limited sampling, such data integration can readily be done with regression-ABC.

Material and methods

Epidemiological data

The Dat'AIDS cohort is a collaborative network of 23 French HIV treatment centers covering approximately 25% of HIV-infected patients followed in France (Clinicaltrials.gov ref NCT02898987). The epidemiology of HCV infection in the cohort has been extensively described from 2000 to 2016 [40–42]. The incidence of acute HCV infection has been estimated among HIV-infected MSM between 2012 and 2016, among HIV-negative MSM enrolled in PrEP between in 2016-2017 [13] and among HIV-infected and HIV-negative MSMs from 2014 to 2017 [14]. A réécrire pour ne citer que les données de séquences que nous utilisons (voire un autre article si on en a besoin pour le labeling)

HCV sequence data

We included HCV molecular sequences of all MSM patients diagnosed with acute HCV genotype 1a infection at the Infectious Disease Department of the Hospices Civils de Lyon, France, and for whom NS5B sequencing was performed between January 2014 and December 2017 ($N = 68$). HCV genotype 1a isolated from $N = 145$ non-MSM, HIV-negative, male patients of similar age were analysed by NS5B sequencing at the same time for phylogenetic analysis. This study was conducted in accordance with French ethics regulations. All patients gave their written informed consent to allow the use of their personal clinical data. The study was approved by the Ethics Committee of Hospices Civils de Lyon.

HCV testing and sequencing

HCV RNA was detected and quantified using the Abbott RealTime HCV assay (Abbott Molecular, Rungis, France). The NS5B fragment of HCV was amplified between nucleotides 8256 and 8644 by RT-PCR as previously described and sequenced using the Sanger method. Electrophoresis and data collection were performed on a GenomeLab^{TM} GeXP Genetic Analyzer (Beckman Coulter). Consensus sequences were assembled and analysed using the GenomeLab^{TM} sequence analysis software. The genotype of each sample was determined by comparing its sequence with HCV reference sequences obtained from GenBank.

Nucleotide accession numbers

All HCV NS5B sequences isolated in MSM and non-MSM patients reported in this study were submitted to the GenBank database. The list of Genbank accession numbers for all sequences is provided in Appendix.

Dated viral phylogeny

To infer the time-scaled viral phylogeny from the alignment we used a Bayesian Skyline model in BEAST v2.4.8 [43]. The general time reversible (GTR) nucleotide substitution model was used with a strict clock rate fixed at 10^{-3} based on data from Ref. [44] and a gamma distribution with four substitution rate categories. The MCMC was run for 100 million iterations and samples were saved every 5,000 iterations. We selected the maximum clade credibility using TreeAnnotator BEAST2 package. The date of the last common ancestor was estimated to be 1977.67 with a 95% Highest Posterior Density (HPD) of $[1960.475; 1995.957]$.

Tab. 1. Prior distributions for the birth-death model parameters over the three time intervals. t_0 is the date of origin of the epidemics in the studied area, t_1 is the date of introduction of 3rd generation HCV tests, t_2 is the date of emergence of the epidemic in 'new' hosts and t_f is the time of the most recent sampled sequence.

Interval	γ_i	ν	$R_0^{(1)}$	a_i
$[t_0, t_1]$	Unif(0.1, 4)	0	Unif(0.9, 15)	Unif(0, 1)
$[t_1, t_2]$			Unif(0.1, 3)	
$[t_2, t_3]$		Unif(0, 10)		

Epidemiological model and simulations

We assume a Birth-Death model with two hosts types with 'classical' hosts (numbered 1) and new hosts (numbered 2). This model is described by the following system of ordinary differential equations (ODEs):

$$\frac{dI_1}{dt} = a_1\beta I_1 + (1-a_2)\nu\beta I_2 - \gamma_1 I_1 \tag{1a}$$

$$\frac{dI_2}{dt} = a_2\beta\nu I_2 + (1-a_1)\beta I_1 - \gamma_2 I_2 \tag{1b}$$

In the model, transmission events are possible within each type of hosts and between the two types of hosts at a transmission rate β. Parameter ν corresponds to the transmission rate differential between classical and new hosts. Individuals can be 'removed' at a rate γ_1 from an infectious compartment (I_1 or I_2) via infection clearance, host death or change in host behaviour (e.g. condom use). The assortativity between host types, which can be seen as the percentage of transmissions that occur with hosts from the same type, is captured by parameter a_i.

The effective reproduction number (denoted R_0) is the number of secondary cases caused by an infectious individual in a fully susceptible host population [31]. We seek to infer the R_0 from the classical epidemic, denoted $R_0^{(1)}$ and defined by $R_0^{(1)} = \beta/\gamma_1$, as well as the R_0 of the new epidemic, denoted $R_0^{(2)}$ and defined by $R_0^{(2)} = \nu\beta/\gamma_2 = \nu R_0^{(1)}\gamma_1/\gamma_2$.

The doubling time of an epidemics (t_D) corresponds to the time required for the number of infected hosts to double in size. It is usually estimated in the early stage of an epidemics, when epidemic growth can assumed to be exponential. To calculate it, we assume perfect assortativity ($a_1 = a_2 = 1$) and approximate the initial exponential growth rate by $\beta - \gamma_1$ for 'classical' hosts and $\nu\beta - \gamma_2$ for 'new' hosts. Following [45], we obtain $t_D^{(1)} = \ln(2)/(\beta - \gamma_1)$ and $t_D^{(2)} = \ln(2)/(\nu\beta - \gamma_2)$.

We consider three time intervals. During the first interval $[t_0, t_1]$, t_0 being the year of the origin of the epidemic in the area of Lyon, we assume that only classical hosts are present. For t_0, we use the lower and upper bounds of the 95% HPD of the inferred date of the last common ancester by Beast as the lower and upper bound of a uniform prior. The second interval $[t_1, t_2]$, begins in $t_1 = 1997.3$ with the introduction of the third generation HCV tests, which we assume to have affected $R_0^{(1)}$ through the decrease of the transmission rate β. Finally, the 'new' hosts appear during the last interval $[t_2, t_f]$, where t_2, which we infer, is the date of origin of the second outbreak. The final time (t_f) is set by the most recent sampling date in our dataset (2018.39). The prior distributions used are summarized in Table 1 and shown in Figure 2.

To simulate phylogenies, we use a simulator implemented in `R` via the `Rcpp` package. This is done in a two-step procedure. First, epidemiological trajectories are simulated using the compartmental model in equation 1 and Gillespie's stochastic event-driven simulation algorithm [46]. The number of individuals in each compartment and the reactions occurring through the simulations of trajectories, such as recovery or transmission events, are recorded. Using the target phylogeny, we know when sampling events occur. For each simulation, each sampling date is randomly associated to a host compartment using the observed fraction of each infection type (here 68% of the dates associated with 'classical' hosts type and 32% with 'new' hosts). Once the sampling dates are added to the trajectories,

we move to the second step, which involves simulating the phylogeny. This step starts from the last sampling date and follows the epidemiological trajectory through a coalescent process, that is backward-in-time. Each backward step in the trajectory can induce a tree modification: a sampling event leads to a labelled leaf in the phylogeny, a transmission event can lead to the coalescence of two sampled lineages or to no modification of the phylogeny (if one of the lineages is not sampled).

We implicitly assume that the sampling rate is low, which is consistent with the limited number of sequences in the dataset. We also assume that the virus can still be transmitted after sampling.

We simulate $71,000$ phylogenies from known parameter sets drawn in the prior distributions shown in Table 1. These are used to perform the rejection step and build the regression model in the Approximate Bayesian Computation (ABC) inference.

ABC inference

Summary statistics Phylogenies are rich objects and to compare them we break them into summary statistics. These are chosen to capture the epidemiological information of interest. In particular, following an earlier study, we use summary statistics from branch lengths, tree topology, and lineage-through-time (LTT) [30].

We also compute new summary statistics to extract information regarding the heterogeneity of the population, the assortativity, and the difference between the two R_0. To do so, we annotate each internal node by associating it with a probability to be in a particular state (here the host type, 'classical' or 'new'). We assume that this probability is given by the ratio

$$P(Y) = \frac{\text{number of leaves labelled } Y}{\text{number of descendent leaves}} \tag{2}$$

where Y is a state (or host type). Each node is therefore annotated with n ratios, n being the number of possible states. Since in our case $n = 2$, we only follow one of the labels and use the mean and the variance of the distribution of the ratios (one for each node) as summary statistics.

In a phylogeny, cherries are pairs of leaves that are adjacent to a common ancestor. There are $n(n+1)/2$ categories of cherries. Here, we compute the proportion of homogeneous cherries for each label and the proportion of heterogeneous cherries. We also consider pitchforks, which we define as a cherry and a leaf adjacent to a common ancestor, and introduce three categories: homogeneous pitchforks, pitchforks whose cherries are homogeneous for a label and whose leaf is labelled with another trait, and pitchforks whose cherries are heterogeneous.

The Lineage-Through-Time (LTT) plot displays the number of lineages of a phylogeny over time. In this plot, the number of lineages is incremented by one every time there is a new branch in the phylogeny, and is decreased by one every time there is a new leaf in the phylogeny. We use the ratios defined for each internal node to build a LTT for each label type, which we refer to as 'LTT label plot'. After each branching event in phylogeny, we increment the number of lineages by the value of the ratio of the internal node for the given label. This number of lineages is decreased by one every time there is a leaf in the phylogeny. In the end, we obtain $n = 2$ LTT label plots.

Finally, for each label, we compute some of our branch lengths summary statistics on homogeneous clades and heterogeneous clades present in the phylogeny. Homogeneous clades are defined by their root having a ratio of 1 for one type of label and their size being greater than $N_{\min}$. For heterogeneous clades, we keep the size criterion and impose that the ratio is smaller than 1 but greater than a threshold ϵ. After preliminary analyses, we set $N_{\min} = 4$ leaves and $\epsilon = 0.7$. We therefore obtain a set of homogeneous clades and a set of heterogeneous clades, the branch lengths of which we pool into two sets to compute the summary statistics of heterogeneous and homogeneous clades. Note that we always select the largest clade, for both homogeneous and heterogeneous cases, to avoid redundancy.

Regression-ABC We first measure multicollinearity between summary statistics using variance inflation factors (VIF). Each summary statistic is kept if its VIF value is lower than 10. This stepwise VIF test leads to the selection of 88 summary statistics out of 234.

We then use the **abc** function from the `abc R` package to infer posterior distributions generated using only the rejection step. Finally, we perform linear adjustment using an elastic net regression.

The **abc** function performs a classical one-step rejection algorithm [29] using a tolerance parameter P_δ, which represents a percentile of the simulations that are close to the target. To compute the distance between a simulation and the target, we use the Euclidian distance between normalized simulated vector of summary statistics and the normalized target vector.

Prior to linear adjustment, the `abc` function performs smooth weighting using an Epanechnikov kernel [29]. Then, using the `glmnet` package in `R`, we implement an elastic-net (EN) adjustment, which balances the Ridge and the LASSO regression penalties [47].

In the end, we obtain posterior distributions for t_0, t_2, a_1, a_2, ν, γ_1, γ_2, $R_0^{(1),t_1}$ and $R_0^{(1),t_2}$ using our ABC-EN regression model with $P_\delta = 0.1$.

Parametric bootstrap and cross validation Our parametric bootstrap validation consists in simulating $5,000$ additional phylogenies from parameter sets drawn in posterior distributions. We then compute summary statistics and perform a principal component analysis (PCA) on the vectors of summary statistics for the simulated and for the target data. If the posterior distribution is informative, we expect the target data to be similar to the simulated phylogenies. On the contrary, if the posterior distribution can generate phylogenies with a variety of shapes, the target data can be outside the cloud of simulated phylogenies in the PCA.

In order to assess the robustness of our ABC-EN method to infer epidemiological parameters of our BD model, we also perform a 'leave-one-out' cross-validation as in [30]. This consists in inferring posterior distributions of the parameters from one simulated phylogeny, assumed to be the target phylogeny, using the ABC-EN method with the remaining $60,999$ simulated phylogenies. We run the cross-validation 100 times with 100 different target phylogenies. We consider three parameter distributions θ: the prior distribution, the prior distribution reduced by the feasibility of the simulations and the ABC inferred posterior distribution. For each of these parameter distributions, we measure the median and compute, for each simulation scenario, the mean relative error (MRE) such as:

$$MRE = \frac{1}{100} \sum_{i=1}^{100} \mid \frac{\theta_i}{\Theta} - 1 \mid \tag{3}$$

where Θ is the true value.

Acknowledgements

We thank Jūlija Pečerska for her help with Beast2. GD is funded by the Fondation pour la Recherche Médicale (FRM grant number ECO20170637560). GD and SA acknowledge further support from the CNRS, the IRD and the itrop HPC (South Green Platform) at IRD montpellier, which provided HPC resources that contributed to the results reported here (https://bioinfo.ird.fr/).

References

[1] Jane P. Messina, Isla Humphreys, Abraham Flaxman, Anthony Brown, Graham S. Cooke, Oliver G. Pybus, and Eleanor Barnes. Global distribution and prevalence of hepatitis C virus genotypes. *Hepatology*, pages 77–87, 2015.

[2] European Union HCV Collaborators. Hepatitis C virus prevalence and level of intervention required to achieve the WHO targets for elimination in the European Union by 2030: a modelling study. *Lancet Gastroenterol Hepatol*, 2(5):325–336, 2017.

[3] Miriam J Alter. Epidemiology of viral hepatitis and HIV co-infection. *J. Hepatol.*, 44(S1):S6–9, 2006.

[4] E. Rosenthal, D. Salmon-Céron, C. Lewden, V. Bouteloup, G. Pialoux, F. Bonnet, M. Karmochkine, T. May, M. François, C. Burty, E. Jougla, D. Costagliola, P. Morlat, G. Chêne, and P. Cacoub. Liver-related deaths in HIV-infected patients between 1995 and 2005 in the French GERMIVIC Joint Study Group Network (Mortavic 2005 Study in collaboration with the Mortalité 2005 survey, ANRS EN19)*. *HIV Medicine*, 10(5):282–289, 2009.

[5] Helen Kovari, Bruno Ledergerber, Matthias Cavassini, Juan Ambrosioni, Andrea Bregenzer, Marcel Stöckle, Enos Bernasconi, Roger Kouyos, Rainer Weber, and Andri Rauch. High hepatic and extrahepatic mortality and low treatment uptake in HCV-coinfected persons in the Swiss HIV cohort study between 2001 and 2013. *Journal of Hepatology*, 63(3):573–580, September 2015.

[6] Marina B. Klein, Keri N. Althoff, Yuezhou Jing, Bryan Lau, Mari Kitahata, Vincent Lo Re, Gregory D. Kirk, Mark Hull, H. Nina Kim, Giada Sebastiani, Erica E. M. Moodie, Michael J. Silverberg, Timothy R. Sterling, Jennifer E. Thorne, Angela Cescon, Sonia Napravnik, Joe Eron, M. John Gill, Amy Justice, Marion G. Peters, James J. Goedert, Angel Mayor, Chloe L. Thio, Edward R. Cachay, Richard Moore, Gregory D. Kirk, Constance A. Benson, Ronald J. Bosch, Stephen Boswell, Kenneth H. Mayer, Chris Grasso, Robert S. Hogg, P. Richard Harrigan, Julio S. G. Montaner, Angela Cescon, Hasina Samji, John T. Brooks, Kate Buchacz, Kelly A. Gebo, Richard D. Moore, Richard D. Moore, Benigno Rodriguez, Michael A. Horberg, Michael J. Silverberg, Jennifer E. Thorne, James J. Goedert, Lisa P. Jacobsonc, Gypsyamber D'Souza, Marina B. Klein, Sean B. Rourke, Ann N. Burchell, Anita R. Rachlis, Robert F. Hunter-Mellado, Angel M. Mayor, M. John Gill, Steven G. Deeks, Jeffrey N. Martin, Pragna Patel, John T. Brooks, Michael S. Saag, Michael J. Mugavero, James Willig, Joseph J. Eron, Sonia Napravnik, Mari M. Kitahata, Heidi M. Crane, H. Nina Kim, Daniel R. Drozd, Timothy R. Sterling, David Haas, Sally Bebawy, Megan Turner, Amy C. Justice, Robert Dubrow, David Fiellin, Stephen J. Gange, Kathryn Anastos, Richard D. Moore, Michael S. Saag, Stephen J. Gange, Mari M. Kitahata, Keri N. Althoff, Rosemary G. McKaig, Amy C. Justice, Aimee M. Freeman, Richard D. Moore, Aimee M. Freeman, Carol Lent, Mari M. Kitahata, Stephen E. Van Rompaey, Heidi M. Crane, Daniel R. Drozd, Liz Morton, Justin McReynolds, William B. Lober, Stephen J. Gange, Keri N. Althoff, Alison G. Abraham, Bryan Lau, Jinbing Zhang, Jerry Jing, Elizabeth Golub, Shari Modur, Cherise Wong, Brenna Hogan, Weiqun Tong, and Bin Liu. Risk of End-Stage Liver Disease in HIV-Viral Hepatitis Coinfected Persons in North America From the Early to Modern Antiretroviral Therapy Eras. *Clin Infect Dis*, 63(9):1160–1167, November 2016.

[7] Pierre Pradat, Pascal Pugliese, Isabelle Poizot-Martin, Marc-Antoine Valantin, Lise Cuzin, Jacques Reynes, Eric Billaud, Thomas Huleux, Firouze Bani-Sadr, David Rey, Anne Frésard, Christine Jacomet, Claudine Duvivier, Antoine Cheret, Laurent Hustache-Mathieu, Bruno Hoen, André Cabié, Laurent Cotte, L. Cotte, C. Chidiac, T. Ferry, F. Ader, F. Biron, A. Boibieux, P. Miailhes, T. Perpoint, I. Schlienger, J. Lippmann, E. Braun, J. Koffi, C. Longuet, V. Guéripel, C. Augustin-Normand, C. Brochier, S. Degroodt, P. Pugliese, C. Ceppi, E. Cua, J. Cottalorda, J. Courjon, P. Dellamonica, E. Demonchy, A. De Monte, J. Durant, C. Etienne, S. Ferrando, J. G. Fuzibet, R. Garraffo, A. Joulie, K. Risso, V. Mondain, A. Naqvi, N. Oran, I. Perbost, S. Pillet, B. Prouvost-Keller, S. Wehrlen-Pugliese, E. Rosenthal, S. Sausse, V. Rio, P. M. Roger, S. Brégigeon, O. Faucher, V. Obry-Roguet, M. Orticoni, M. J. Soavi, P. Geneau de Lamarlière, H. Laroche, E. Ressiot, M. Carta, M. J. Ducassou, I. Jacquet, S. Gallie, A. Galinier, A. S. Ritleng, A. Ivanova, C. Blanco-Betancourt, C. Lions, C. Debreux, V. Obry-Roguet, I. Poizot-Martin, R. Agher, C. Katlama, M. A. Valantin, C. Duvivier, O. Lortholary, F. Lanternier, C. Charlier, C. Rouzaud, C. Aguilar, B. Henry, D. Lebeaux, G. Cessot, A. Gergely, P. H. Consigny, F. Touam, C. Louisin, M. Alvarez, N. Biezunski, L. Cuzin, A. Debard, P. Delobel, C. Delpierre, C. Fourcade, B. Marchou, G. Martin-Blondel, M. Porte, M. Mularczyk, D. Garipuy, K. Saune, I. Lepain, M. Marcel, E. Puntis, N. Atoui, M. L. Casanova, V. Faucherre, J. M. Jacquet, V. Le Moing, A. Makinson, C. Merle De Boever, A. Montoya-Ferrer, C. Psomas, J. Reynes, F. Raffi, C. Allavena, E. Billaud, C. Biron, B. Bonnet, S. Bouchez, D. Boutoille, C. Brunet, T. Jovelin, N. Hall, C. Bernaud, P. Morineau, V. Reliquet, O. Aubry, P. Point, M. Besnier, L. Larmet, H. Hüe, S. Pineau, E. André-Garnier, A. Rodallec, Ph. Choisy, S. Vandame, Th. Huleux, F. Ajana, I. Alcaraz, V. Baclet, T. H. Huleux, H. Melliez, N. Viget, M. Valette, E. Aissi, Ch. Allienne, A. Meybeck, B. Riff, F. Bani-Sadr, C. Rouger, J. L. Berger, Y. N'Guyen, D. Lambert, I. Kmiec, M. Hentzien, D. Lebrun, C. Migault, D. Rey, M. L. Batard, C. Bernard-Henry, C. Cheneau, E. de Mautort, P. Fischer, M. Partisani, M. Priester, F. Lucht, A. Frésard, E. Botelho-Nevers, A. Gagneux-Brunon, C. Cazorla, C. Guglielminotti, F. Daoud, M. F. Lutz, C. Jacomet, H. Laurichesse, O. Lesens, M. Vidal, N. Mrozek, V. Corbin, C. Aumeran, O. Baud, S. Casanova, D. Coban, L. Hustache-Mathieu, M. C. Thiebaut-Drobacheff, A. Foltzer, V. Gendrin, F. Bozon, C. Chirouze, S. Abel, A. Cabié, R. Césaire, G. Dos Santos, L. Fagour, F. Najioullah, M. Ouka, S. Pierre-François, M. Pircher, B. Rozé, B. Hoen, R. Ouissa, and I. Lamaury. Direct-acting antiviral treatment against hepatitis C virus infection in HIV-Infected patients – "En route for eradication"? *Journal of Infection*, 75(3):234–241, September 2017.

[8] Charles Béguelin, Annatina Suter, Enos Bernasconi, Jan Fehr, Helen Kovari, Heiner C. Bucher, Marcel Stoeckle, Mathias Cavassini, Mathieu Rougemont, Patrick Schmid, Gilles Wandeler, and Andri Rauch. Trends in HCV treatment uptake, efficacy and impact on liver fibrosis in the Swiss HIV Cohort Study. *Liver International*, 38(3):424–431, 2018.

[9] Juan Berenguer, Inmaculada Jarrín, Leire Pérez-Latorre, Víctor Hontañón, María J. Vivancos, Jordi Navarro, María J. Téllez, Josep M. Guardiola, José A. Iribarren, Antonio Rivero-Juárez, Manuel Márquez, Arturo Artero, Luis Morano, Ignacio Santos, Javier Moreno, María C. Fariñas, María J. Galindo, María A.

Hernando, Marta Montero, Carmen Cifuentes, Pere Domingo, José Sanz, Lourdes Domíngez, Oscar L. Ferrero, Belén De la Fuente, Carmen Rodríguez, Sergio Reus, José Hernández-Quero, Gabriel Gaspar, Laura Pérez-Martínez, Coral García, Lluis Force, Sergio Veloso, Juan E. Losa, Josep Vilaró, Enrique Bernal, Sari Arponen, Amat J. Ortí, Ángel Chocarro, Ramón Teira, Gerardo Alonso, Rafael Silvariño, Ana Vegas, Paloma Geijo, Josep Bisbe, Herminia Esteban, and Juan González-García. Human Immunodeficiency Virus/Hepatits C Virus Coinfection in Spain: Elimination Is Feasible, but the Burden of Residual Cirrhosis Will Be Significant. *Open Forum Infect Dis*, 5(1), January 2018.

[10] Anne Boerekamps, Guido E. van den Berk, Fanny N. Lauw, Eliane M. Leyten, Marjo E. van Kasteren, Arne van Eeden, Dirk Posthouwer, Mark A. Claassen, Anton S. Dofferhoff, Dominique W. M. Verhagen, Wouter F. Bierman, Kamilla D. Lettinga, Frank P. Kroon, Corine E. Delsing, Paul H. Groeneveld, Robert Soetekouw, Edgar J. Peters, Sebastiaan J. Hullegie, Stephanie Popping, David A. M. C. van de Vijver, Charles A. Boucher, Joop E. Arends, and Bart J. Rijnders. Declining Hepatitis C Virus (HCV) Incidence in Dutch Human Immunodeficiency Virus-Positive Men Who Have Sex With Men After Unrestricted Access to HCV Therapy. *Clin Infect Dis*, 66(9):1360–1365, April 2018.

[11] Thijis van de Laar, Oliver Pybus, Sylvia Bruisten, David Brown, Mark Nelson, Sanjay Bhagani, Martin Vogel, Alex Baumgarten, Marie-Laure Chaix, Martin Fisher, Hannelore Gőtz, Gail V. Matthews, Stefan Neifer, Peter White, William Rawlinson, Stanislav Pol, Jurgen Rockstroh, Roel Coutinho, Greg J. Dore, Geoffrey M. Dusheiko, and M. Danta. Evidence of a Large, International Network of HCV Transmission in HIV-Positive Men Who Have Sex With Men. *Gastroenterology*, 136(5):1609–1617, May 2009.

[12] Luisa Salazar-Vizcaya, Roger D. Kouyos, Cindy Zahnd, Gilles Wandeler, Manuel Battegay, Katharine Elizabeth Anna Darling, Enos Bernasconi, Alexandra Calmy, Pietro Vernazza, Hansjakob Furrer, Matthias Egger, Olivia Keiser, and Andri Rauch. Hepatitis C virus transmission among human immunodeficiency virus-infected men who have sex with men: Modeling the effect of behavioral and treatment interventions. *Hepatology*, 64(6):1856–1869, 2016.

[13] Pierre Pradat, Thomas Huleux, François Raffi, Pierre Delobel, Marc-Antoine Valantin, Isabelle Poizot-Martin, Pascal Pugliese, Jacques Reynes, David Rey, Bruno Hoen, André Cabie, Firouzé Bani-Sadr, Antoine Cheret, Claudine Duvivier, Christine Jacomet, Anne Fresard, Laurent Hustache-Mathieu, Laurent Cotte, and the Dat'AIDS study Group. Incidence of new hepatitis C virus infection is still increasing in French MSM living with HIV. *AIDS*, 32(8):1077, May 2018.

[14] Christophe Ramière, Caroline Charre, Patrick Miailhes, François Bailly, Sylvie Radenne, Anne-Claire Uhres, Corinne Brochier, Matthieu Godinot, Pierre Chiarello, Pierre Pradat, Laurent Cotte, Marie Astrie, Claude Augustin-Normand, Bailly François, François Biron, André Boibieux, Corinne Brochier, Evelyne Braun, Florence Brunel, Caroline Charre, Pierre Chiarello, Christian Chidiac, Laurent Cotte, Tristan Ferry, Matthieu Godinot, Olivier Guillaud, Joseph Koffi, Jean-Michel Livrozet, Djamila Makhloufi, Patrick Miailhes, Thomas Perpoint, Pierre Pradat, Sylvie Radenne, Christophe Ramière, Isabelle Schlienger, Caroline Scholtes, Isabelle Schuffenecker, Jean-Claude Tardy, Mary-Anne Trabaud, and Anne-Claire Uhres. Patterns of Hepatitis C Virus Transmission in Human Immunodeficiency Virus (HIV)–infected and HIV-negative Men Who Have Sex With Men. *Clin Infect Dis*, 2019.

[15] Victor Virlogeux, Fabien Zoulim, Pascal Pugliese, Isabelle Poizot-Martin, Marc-Antoine Valantin, Lise Cuzin, Jacques Reynes, Eric Billaud, Thomas Huleux, Firouze Bani-Sadr, David Rey, Anne Frésard, Christine Jacomet, Claudine Duvivier, Antoine Cheret, Laurent Hustache-Mathieu, Bruno Hoen, André Cabié, Laurent Cotte, and the Dat'AIDS Study Group. Modeling HIV-HCV coinfection epidemiology in the direct-acting antiviral era: the road to elimination. *BMC Medicine*, 15(1):217, December 2017.

[16] Oliver G. Pybus, Alexandra Cochrane, Edward C. Holmes, and Peter Simmonds. The hepatitis C virus epidemic among injecting drug users. *Infection, Genetics and Evolution*, 5(2):131–139, March 2005.

[17] M. J. Sweeting, D. De Angelis, M. Hickman, and A. E. Ades. Estimating hepatitis C prevalence in England and Wales by synthesizing evidence from multiple data sources. Assessing data conflict and model fit. *Biostatistics*, 9(4):715–734, October 2008.

[18] Jisoo A. Kwon, Jenny Iversen, Lisa Maher, Matthew G. Law, and David P. Wilson. The Impact of Needle and Syringe Programs on HIV and HCV Transmissions in Injecting Drug Users in Australia: A Model-Based Analysis. *JAIDS Journal of Acquired Immune Deficiency Syndromes*, 51(4):462, August 2009.

[19] Ashley B Pitcher, Annick Borquez, Britt Skaathun, and Natasha K Martin. Mathematical modeling of hepatitis c virus (HCV) prevention among people who inject drugs: A review of the literature and insights for elimination strategies. *Journal of Theoretical Biology*, November 2018.

[20] Romulus Breban, Naglaa Arafa, Sandrine Leroy, Aya Mostafa, Iman Bakr, Laura Tondeur, Mohamed Abdel-Hamid, Wahid Doss, Gamal Esmat, Mostafa K Mohamed, and Arnaud Fontanet. Effect of preventive and curative interventions on hepatitis C virus transmission in Egypt (ANRS 1211): a modelling study. *The Lancet Global Health*, 2(9):e541–e549, September 2014.

[21] Alastair Heffernan, Graham S Cooke, Shevanthi Nayagam, Mark Thursz, and Timothy B Hallett. Scaling up prevention and treatment towards the elimination of hepatitis C: a global mathematical model. *The Lancet*, 393(10178):1319–1329, March 2019.

[22] B. T. Grenfell, O. G. Pybus, J. R. Gog, J. L. Wood, J. M. Daly, J. A. Mumford, and E. C. Holmes. Unifying the epidemiological and evolutionary dynamics of pathogens. *Science*, 303(5656):327–32, 2004.

[23] Erik M Volz, Katia Koelle, and Trevor Bedford. Viral phylodynamics. *PLoS Comput Biol*, 9(3):e1002947, 2013.

[24] Simon DW Frost, Oliver G. Pybus, Julia R. Gog, Cecile Viboud, Sebastian Bonhoeffer, and Trevor Bedford. Eight challenges in phylodynamic inference. *Epidemics*, 10:88–92, 2015.

[25] O G Pybus, M A Charleston, S Gupta, A Rambaut, E C Holmes, and P H Harvey. The epidemic behavior of the hepatitis C virus. *Science*, 292(5525):2323–5, 2001.

[26] Gkikas Magiorkinis, Emmanouil Magiorkinis, Dimitrios Paraskevis, Simon Y. W. Ho, Beth Shapiro, Oliver G. Pybus, Jean-Pierre Allain, and Angelos Hatzakis. The Global Spread of Hepatitis C Virus 1a and 1b: A Phylodynamic and Phylogeographic Analysis. *PLOS Medicine*, 6(12):e1000198, December 2009.

[27] Tanja Stadler, Denise Kühnert, Sebastian Bonhoeffer, and Alexei J Drummond. Birth-death skyline plot reveals temporal changes of epidemic spread in HIV and hepatitis C virus (HCV). *Proc Natl Acad Sci USA*, 110(1):228–33, 2013.

[28] Jeffrey B Joy, Rosemary M McCloskey, Thuy Nguyen, Richard H Liang, Yury Khudyakov, Andrea Olmstead, Mel Krajden, John W Ward, P Richard Harrigan, Julio S G Montaner, and Art F Y Poon. The spread of hepatitis C virus genotype 1a in North America: a retrospective phylogenetic study. *The Lancet Infectious Diseases*, 16(6):698–702, June 2016.

[29] Mark A. Beaumont, Wenyang Zhang, and David J. Balding. Approximate Bayesian Computation in Population Genetics. *Genetics*, 162(4):2025–2035, December 2002.

[30] Emma Saulnier, Olivier Gascuel, and Samuel Alizon. Inferring epidemiological parameters from phylogenies using regression-ABC: A comparative study. *PLOS Computational Biology*, 13(3):e1005416, March 2017.

[31] R. M. Anderson and R. M. May. *Infectious Diseases of Humans. Dynamics and Control*. Oxford University Press, Oxford, 1991.

[32] Thijs J. W. van de Laar, Akke K. van der Bij, Maria Prins, Sylvia M. Bruisten, Kees Brinkman, Thomas A. Ruys, Jan T. M. van der Meer, Henry J. C. de Vries, Jan-Willem Mulder, Michiel van Agtmael, Suzanne Jurriaans, Katja C. Wolthers, and Roel A. Coutinho. Increase in HCV Incidence among Men Who Have Sex with Men in Amsterdam Most Likely Caused by Sexual Transmission. *J Infect Dis*, 196(2):230–238, July 2007.

[33] Gilles Wandeler, Thomas Gsponer, Andrea Bregenzer, Huldrych F. Günthard, Olivier Clerc, Alexandra Calmy, Marcel Stöckle, Enos Bernasconi, Hansjakob Furrer, and Andri Rauch. Hepatitis C Virus Infections in the Swiss HIV Cohort Study: A Rapidly Evolving Epidemic. *Clin Infect Dis*, 55(10):1408–1416, November 2012.

[34] AASLD/IDSA HCV Guidance Panel. Hepatitis C guidance: AASLD-IDSA recommendations for testing, managing, and treating adults infected with hepatitis C virus. *Hepatology*, 62(3):932–954, September 2015.

[35] Ethan O. Romero-Severson, Ingo Bulla, and Thomas Leitner. Phylogenetically resolving epidemiologic linkage. *PNAS*, 113(10):2690–2695, March 2016.

[36] Colin J. Worby, Marc Lipsitch, and William P. Hanage. Shared Genomic Variants: Identification of Transmission Routes Using Pathogen Deep-Sequence Data. *Am J Epidemiol*, 186(10):1209–1216, November 2017.

[37] Chris Wymant, Matthew Hall, Oliver Ratmann, David Bonsall, Tanya Golubchik, Mariateresa de Cesare, Astrid Gall, Marion Cornelissen, Christophe Fraser, and STOP-HCV Consortium, The Maela Pneumococcal Collaboration, and The BEEHIVE Collaboration. PHYLOSCANNER: Inferring Transmission from Within- and Between-Host Pathogen Genetic Diversity. *Mol Biol Evol*, 35(3):719–733, 2018.

[38] Erik M. Volz and Igor Siveroni. Bayesian phylodynamic inference with complex models. *PLOS Computational Biology*, 14(11):e1006546, November 2018.

[39] Denise Kühnert, Roger Kouyos, George Shirreff, Jūlija Pečerska, Alexandra U. Scherrer, Jürg Böni, Sabine Yerly, Thomas Klimkait, Vincent Aubert, Huldrych F. Günthard, Tanja Stadler, Sebastian Bonhoeffer, and the Swiss HIV Cohort Study. Quantifying the fitness cost of HIV-1 drug resistance mutations through phylodynamics. *PLOS Pathogens*, 14(2):e1006895, February 2018.

[40] P. Pradat, E. Caillat-Vallet, F. Sahajian, F. Bailly, G. Excler, M. Sepetjan, C. Trépo, and J. Fabry. Prevalence of hepatitis C infection among general practice patients in the Lyon area, France. *Eur J Epidemiol*, 17(1):47–51, January 2001.

[41] Jr A. D'Oliveira, N. Voirin, R. Allard, D. Peyramond, C. Chidiac, J. L. Touraine, J. Fabry, C. Trepo, and P. Vanhems. Prevalence and sexual risk of hepatitis C virus infection when human immunodeficiency virus was acquired through sexual intercourse among patients of the Lyon University Hospitals, France, 1992-2002. *J Viral Hepat*, 12(3):330–332, May 2005.

[42] F. Sahajian, F. Bailly, P. Vanhems, B. Fantino, C. Vannier-Nitenberg, J. Fabry, and C. Trepo. A randomized trial of viral hepatitis prevention among underprivileged people in the Lyon area of France. *J Public Health (Oxf)*, 33(2):182–192, June 2011.

[43] Remco Bouckaert, Joseph Heled, Denise Kühnert, Tim Vaughan, Chieh-Hsi Wu, Dong Xie, Marc A. Suchard, Andrew Rambaut, and Alexei J. Drummond. BEAST 2: A Software Platform for Bayesian Evolutionary Analysis. *PLoS Computational Biology*, 10(4):e1003537, April 2014.

[44] Rebecca R Gray, Joe Parker, Philippe Lemey, Marco Salemi, Aris Katzourakis, and Oliver G Pybus. The mode and tempo of hepatitis C virus evolution within and among hosts. *BMC Evol Biol*, 11:131, 2011.

[45] Jacco Wallinga and Marc Lipsitch. How generation intervals shape the relationship between growth rates and reproductive numbers. *Proc. R. Soc. Lond. B*, 274:599–604, 2007.

[46] Daniel Thomas Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22(4):403–434, 1976.

[47] Hui Zou and Trevor Hastie. Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 67(2):301–320, 2005.

Paper 193

Docking of RNA Hairpin on Protein Using a Fragment-Based Method

Antoine Moniot[1], Rohit Roy[1,2], Yann Guermeur[1] and Isaure Chauvot de Beauchene[1]
[1] LORIA, Campus Scientifique, BP 239, 54506, Vandœuvre-lès-Nancy Cedex, France
[2] Current: Duke Center for Genomic and Computational Biology, Duke University, NC 27708, Durham, USA

Corresponding author: antoine.moniot@loria.fr

Abstract *We introduce an extension of our fragment-based method for ssRNA-protein docking as it is still a challenging difficulty in docking. It is dedicated to hairpins and makes use of geometrical features of this secondary structure. An initial evaluation establishes that it is promising and could make it possible to overcome the limitations of the state-of-the-art fragment-based methods.*

Keywords RNA hairpin, protein, fragment-based docking

1 Introduction

Protein-RNA interactions are involved in many biological processes, including cell regulation [1] and diseases [2,3]. In that context, the structures of the complexes are major knowledge sources. However, their experimental inference is difficult, when possible [4]. As usual, the approach of choice to overcome this limitation is modeling. A difficulty arises when the interaction involves a single-stranded secondary structure of the RNA: single-stranded RNA is highly flexible and consequently difficult to model. This observation led to the introduction of two methods, one based on molecular dynamics [5] and ours, based on the assembling of structural fragments of RNA docked on the protein surface [6]. The current implementation of these methods, requiring the knowledge either of the exact coordinates of 2 nucleotides [5] or only of anchoring points on the protein surface [6], limits their applicability to few protein families. In this article, we introduce an extension of our method relaxing this requirement in the specific case when the single-stranded RNA is the loop of a hairpin. The additional pieces of information exploited to obtain this improvement are intervals on the distances between the nucleotides at the endpoints of the loop (intervals governed by the distances between the two nucleotides closing the loop). Thus, biological information is not needed except the identification of the nucleotides of the closure of the hairpin, which can be obtained by secondary structure prediction.

The organization of the paper is as follows. Section 2 introduces our method. Section 3 is devoted to its experimental evaluation. At last, we draw conclusions and outline our ongoing research in Section 4.

2 Methods

In this section, we first give a brief description of our fragment-based method, so as to highlight afterwards the specificities of the original contribution: the dedication to hairpins.

2.1 Fragment-Based Docking

Our fragment-based method consists of four main steps. To make the paper self-contained, they are now briefly summarized (details are available in [7]). First, for each of the 64 possible trinucleotides, hereafter referred to as *motifs*, a library containing all experimentally observed 3D structures is built, by browsing the Protein Data Bank (PDB [8]). This initial set is then refined by means of a clustering method, to retain a subset, ideally of minimal cardinality, "covering" it with a Root Mean Square Deviation (RMSD) below 1Å (1Å-net). The ensemble of 3D structures obtained for each of the 64 motifs is named *conformers*. With these 64 libraries at hand, the sequence of interest is first cut into trinucleotides with a step of one (so that two consecutive trinucleotides overlap by two nucleotides). For each of the corresponding motifs, the whole refined library of 3D structures is docked on the protein. This *rigid body docking*, using ATTRACT [9], generates for each trinucleotide a set of *poses*. Then, the assembly consists of searching possible paths in a directed graph. Its vertices are the poses

and two successive poses are connected by an edge provided that the RMSD between their two shared nucleotides is below a given threshold $T_{overlap}$. The output at this level is a list of *chains* of poses scored by ATTRACT, that cover the full RNA sequence. The last step consists in an ordering of the list. Two options are implemented. The first one is based on the geometric mean of the ranks of the poses [6]. The second one is a heuristic considering that the best chains are probably made of poses that participate in many chains, for probabilistic and/or entropic reasons [7]. It uses a forward-backward algorithm to count the number of chains in with each pose participates, then selects for every fragment the most connected poses and assemble only those.

In this previous framework, the anchorage takes the form of the knowledge of an interaction between two given nucleotides and two given residues of the protein.

2.2 Dedication to Hairpins

A feature of hairpins useful for modeling is the knowledge of the distance between the nucleotides at the endpoints of the loop [10]. Our dedicated method is based on the conjecture that an appropriate exploitation of this feature can prove enough to constrain the assembly so as to relax the initial need for anchoring points. The implementation is based on an enrichment of the graph described in Section 2.1. A new type of edge is introduced, to connect poses of the last and first fragment. This connection is added when the Euclidean distance $D_{closure}$ between the phosphate of first nucleotide of the first fragment and the phosphate of the last nucleotide of the last fragment belongs to the interval 11.8-24.7Å (interval observed in the benchmark). The new graph is depicted in Figure 1.

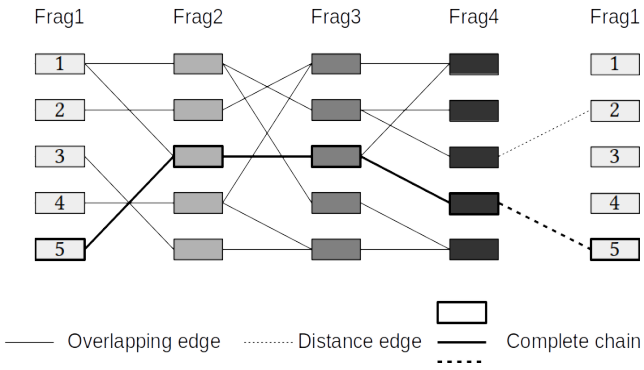


Fig. 1. Graph dedicated to hairpins. This graph represents the connection between the poses of a chain of 6 nucleotides. There are 5 poses for each fragment. If two consecutive poses are overlapping there is an edge between them. A complete chain has 4 poses, one for each fragment.

The set of chains considered here is smaller than in the general case, since it retains only those included in a cycle of the new graph.

3 Assessment of the New Method

To assess the method, a data set of hairpin-protein complexes was produced.

3.1 Selection of a Benchmark

The algorithm for this derivation is made up of two main steps. It takes in input the set of all available non-redundant experimental structures of hairpin-protein complexes. For every hairpin, the docking of all the conformers of all the motifs present in its loop is performed on the corresponding protein using ATTRACT. This leads to retaining only the complexes for which all fragments have at least one *near native* pose. A near native pose is a pose whose RMSD with the native (experimental) position is below 3Å. The second step is a refinement that consists in doing the docking again with a subset of conformers structurally close enough to the native position. This corresponds to eliminating conformers that are structurally too different from the native position to be a near native pose. At this level, the criterion to retain a complex is the following one: for every fragment, the rank of

the ATTRACT score of the first-ranked near native pose must be below a number of pose N_{pose}. Obviously, this step, involving pieces of information in principle unknown, turns our experimental approach into a proof of concept.

The initial set of hairpin-protein complexes is obtained by application of NAfragDB [11]. It contains 19 complexes. At the first step of the algorithm, one obtains roughly $10 \cdot 10^3$ poses per conformer, i.e., $30 \cdot 10^6$ poses per fragment, given the fact that the libraries for the motifs contain on average 3000 conformers. At this level, only 2 complexes are selected: 5UDZ [12] and 1RKJ [13]. For both of them, at least one fragment has the top-ranked near native pose at a rank higher than 10^6. This explains, at least for our data set, the need for the second step of the algorithm, to keep a chance to obtain a relevant assembly. This second step is parameterized as follows. For every motif, a subset of conformers is created that contains only the ten closest (according to the RMSD) to the experimental structure after optimal fitting. On the contrary, the number of poses per conformer is increased to $50 \cdot 10^3$, so that the new number of poses per fragment is $500 \cdot 10^3$. Table 1 provides the set of hairpins selected at the first step, with the corresponding docking results.

		frag1	frag2	frag3	frag4	frag5
5UDZ	first docking	5743309	423163	15403	27578	2138595
	second docking	**90424**	2114	285	1334	34291
1RKJ	first docking	1382237	1110963	13187	599859	
	second docking	7056	**14949**	191	1275	

Tab. 1. First rank for a near native pose for the fragments of the two hairpins.

The figures in Table 1 establish that performing the assembly for 5UDZ and 1RKJ requires to consider a maximum of $6 \cdot 10^{24}$ and $5 \cdot 10^{16}$ possible chains respectively. Those numbers are based on the following computation. For each fragment of each sequence, the number of poses considered is the rounded largest value among the ranks of the first-ranked near native poses (here $90 \cdot 10^3$ for 5UDZ and $15 \cdot 10^3$ for 1RKJ). This arbitrary choice corresponds to a reasonable assumption on what information could be inferred from data. With the processing for 5UDZ being the most time consuming, and currently underway, in the sequel, the results are provided for 1RKJ only.

3.2 Assessment for the Hairpin of 1RKJ

The sequence of the hairpin is UCCCGA (thus four fragments). The three parameters to be set to derive the chains are $D_{closure}$, N_{pose}, and $T_{overlap}$. The two first values have been given above. As for the third one, the value of 2.6Å was retained since it is the smallest value ensuring to generate a chain connecting near native poses only. For this parameterization, the number of chains is 47617288. In that set we obtain 732 acceptable solutions (with an RMSD below 5Å) and 122 good solutions (RMSD below 3Å), with a best model at 1.9Å. The best one and the experimental one are represented in Figure 2.

Thus, the method appears sensitive, but not specific enough, which calls for an investigation of the set of chains. Section 2.1 has introduced the two methods implemented to sort it. We now discuss their effectiveness. When using as criterion the geometric mean of the ranks of the poses, the smallest rank of a solution among the 732 satisfactory ones is 4783648. This is close to the top 10%, but leaves too many false positives above.

	frag1	frag2	frag3	frag4
All poses	206742	502600	654520	1105597
Poses in a solution	5701	16849	366	1672
Average for all poses	11985	15830	24865	20151

Tab. 2. Highest number of chains in which a pose of the 47617288 chains (all poses) and a pose of the 732 solutions (poses in a solution) are involved. The last line is the average of chains for all poses.

The second criterion is evaluated by looking at the number of chains in which poses are involved (see Table 2). We found that the poses which are involved in good chains are not involved in more chains than the rest of the poses.

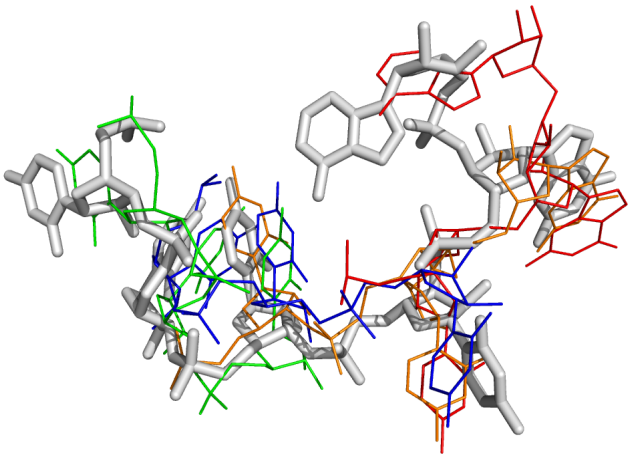


Fig. 2. Poses (in colors) composing the best chain with respect to RMSD with the native chain (in white). Frag1 in green, frag2 in blue, frag3 in orange, frag4 in red.

In order to decrease the number of retained chains, we tried to narrow to 10-15Å the interval for the distance $D_{closure}$ (which was initially set to 11.8-24.7Å), since this distance is 13.2Å in 1RJK. This resulted in a decrease of the cardinality of more than 30%, obtaining 32765494 chains, but also decreasing the number of good solutions from 732 to 517. It did not increase the percentage of correct solutions in the chains.

4 Conclusions and Ongoing Work

Given the difficulty of the task tackled, our initial results appear promising. The knowledge of the secondary structure seems to be relevant enough to replace the knowledge of the anchoring points. The results were better with anchoring points, with a higher percentage of correct chains and a more accurate best chain in most cases [6]. This was to be expected since distance and position represent a stronger constraint than distance only. Indeed, this knowledge allowed to assemble and evaluate all the possible chains, without our previous heuristic pre-filtering of the most-connected poses [7], and to obtain a more precise model (1.9Å, instead of 3.6-5.7Å). On the other hand, the hypothesis of a loop closure being weaker than that of the exact position of the chain extremities on the protein, the current approach retains more false positives than the anchored docking did (more than 1% correct models in the assembled chains).

An improvement of our method should result from a change of target for the distance. The distance between the phosphate of the first nucleotide and the sugar of the last nucleotide seems to be a stronger constraint. The interval of distance is 13.7-21.6Å, which is tighter than the phosphate-phosphate interval (see Section 2.2). The variance for the phosphate-sugar distance, 2.4, is smaller than for the phosphate-phosphate distance (5.9). These values are observed on our bigger benchmark of 191 structures. Currently, the limiting factor of the new method is still the docking of the trinucleotides by ATTRACT, as in most complexes, not all fragments have at least one near-native pose. A major reason for this problem is inherent to the fragment-based approach: minimizing the interaction energy for such small fragments is not equivalent to minimizing this energy for the whole sequence. Another one, but directly related, is the inadequation of ATTRACT scoring function for ssRNA, developed on protein- double-stranded RNA complexes. Thus, our current research consists in developing a new scoring function specific for ssRNA fragments. In parallel, we consider an addition to the distance constraint, to check if the loop-closing nucleotides have specific geometries with the base-pairing of the neighbor nucleotides.

Finally, a natural extension of this work consists in applying its principle to different RNA secondary structures (known or predicted), with the aim of the global docking of the complex. Eventually, the constraint should not necessarily involve the knowledge of the secondary structure, but could benefit from any knowledge of distance between two nucleotides.

References

[1] Y. Huang, JL. Zhang, XL. Yu, TS. Xu, ZB. Wang, and XC. Cheng. Molecular functions of small regulatory non-coding RNA. *Biochemistry Moscow*, 78(3):221–230, 2013.

[2] T. Vanderweyde, K. Youmans, L. Liu-Yesucevitz, and B. Wolozin. Role of stress granules and RNA-bindingproteins in neurodegeneration: a mini-review. *Gerontology*, 59(6):524–533, 2013.

[3] M. Derrigo, A. Cestelli, G. Savettieri, and I. Di Liegro. RNA-protein interactions in the control of stability andlocalization of messenger RNA (review). *International Journal of Molecular Medicine*, 5(2):111–134, 2000.

[4] S. Jones. Protein–RNA interactions: structural biology and computational modeling techniques. *Biophysical Reviews*, 8(4):359–367, 2016.

[5] K. Kappel and R. Das. Sampling native-like structures of RNA-protein complexes through rosetta folding and docking. *Structure*, 27(1):140–151, 2019.

[6] Chauvot de Beauchene I, S. de Vries, and M. Zacharias. Fragment-based modelling of single stranded RNA bound to RNA recognition motif containing proteins. *Nucleic Acids Research*, 44(10):4565–4580, 2016.

[7] Chauvot de Beauchene I, S. de Vries, and M. Zacharias. Binding site identification and flexible docking of single stranded RNA to proteins using a fragment-based approach. *PLoS Computational Biology*, 12(1), 2016.

[8] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic Acids Research*, 18(1):235–242, 2000.

[9] M. Zacharias. Protein-protein docking with a reduced protein model accounting for side-chain flexibility. *Protein Science*, 12(6):1271–1282, 2003.

[10] P. Clote, Y. Ponty, and JM. Steyaert. Expected distance between terminal nucleotides of RNA secondary structures. *Journal of Mathematical Biology*, 65(3):581–599, 2011.

[11] A. Moniot, S. de Vries, D. Ritchie, and I. Chauvot de Beauchene. NAfragDB: a multi-purpose structural database of nucleic-acid–protein complexes for advanced users. In *GGMM*, page 21, 2019.

[12] L. Wang, Y. Nam, A.K. Lee, C. Yu, K. Roth, C. Chen, E.M. Ransey, and P. Sliz. LIN28 zinc knuckle domain is required and sufficient to induce let-7 oligouridylation. *Cell Reports*, 18(11):2664–2675, 2017.

[13] C. Johansson, L.D. Finger, L. Trantirek, T.D. Mueller, S. Kim, I.A. Laird-Offringa, and J. Feigon. Solution structure of the complex formed by the two N-terminal RNA-binding domains of nucleolin and a pre-rRNA target. *Journal of Molecular Biology*, 337(4):799–816, 2004.

Paper 194

RCPred: RNA complex prediction as a constrained maximum weight clique problem

Audrey Legendre, Mandy Ibene, Eric Angel, Fariza Tahi

IBISC, Univ Evry, Université Paris-saclay, Evry, 91000 France
2

Corresponding Author: fariza.tahi@univ-evry.fr

Paper Reference: **Legendre et al. BMC Bioinformatics 2019, 20(Suppl 3):128. https://doi.org/10.1186/s12859-019-2648-1**

RNAs can interact and form complexes with catalytic functions. This is for example the case of the ribosome, composed of the 5S, 5.8S, 18S and 28S RNAs in eukaryotes, that is responsible of the translation of messenger RNAs into proteins. The ribosome is also composed of proteins, but it is the RNAs that are responsible in the catalytic activity of the complex. The prediction of RNA complexes, and more precisely the prediction of their structure, is therefore an import task, and very few tools have been proposed for this purpose.

Recently, we have proposed a new method, called RCPred, that allows to predict RNA complexes secondary structures with pseudoknots, including the so-called "external" pseudoknots, which are pseudoknots occurring in RNA-RNA interactions. RCPred is also able to return several solutions, optimal and sub-optimal ones. This is an important point since, in the one hand, RNAs can have several structures, and on the other hand, the real structure does not correspond always to the solution of minimum free energy.

Our method is based on an original approach that takes advantage of the high number of RNA secondary structures and RNA-RNA interaction prediction tools: the problem of RNA complex prediction as the determination of the best combination (according to the free energy) of RNA secondary structures and RNA-RNA interactions predicted upstream using existing tools. We model those predicted structures and interactions as a graph in order to have a combinatorial optimization problem that is a constrained maximum weight clique problem (MWCP). We propose a heuristic based on Breakout Local Search to solve this problem, and which is able to return several solutions.

Users can have some information on the structure of interest that can help its prediction. These can be patterns like helices, pseudoknots, terminal loops, internal loops, multiple loops. They can also be in possession of experimental probing data such as SHAPE data.

Several tools have been developed for integrating user information or probing data to improve RNA secondary structure. However, to our knowledge, no method predicting RNA complexes allows to consider user constraints or experimental probing data. We developed a new version of our method based on the MWCP for RNA complexes secondary structure prediction. In order to integrate probing data and user constraints along with the free energy criteria, our new method, called C-RCPred, has extended the previously used heuristic to approximatively solve a three-criteria variant of the maximum clique problem.

We have evaluated our methods on a large number of complexes, which shows competitive results compared to the methods of the state of the art. Note that RCPred as well as C-RCPred are, to our knowledge, the only one tools that allow to predict (internal and external) pseudoknots and to return sub-optimal solutions.

RCPred and C-RCPred have been implemented as interactive tools. They are available as web servers and on EvryRNA platform (http://EvryRNA.ibisc.univ-evry.fr).

Paper 208

Bio2M platform: find everything in your RNA-Seq data

Benoit Guibert[1], Florence Ruffle[1], Anthony Boureux[1] and Thérèse Commes[1]
IRMB U1183, 80 rue Augustin Fliche, 34295, Montpellier, France

Corresponding author: Anthony.Boureux@inserm.fr

URL: https://bio2m.montp.inserm.fr/platform/

Abstract *The Bioinformatics and BioMarquers (Bio2M) platform is dedicated mainly to the analysis of RNA-Seq data obtained from next-generation sequencer (NGS). As others platforms, we can do gene expression analysis, but also, transcript assembly for coding-protein gene and long non-coding gene already annotated or new. We are able also to look for rare expression variant splice transcripts and chimera transcripts. We are also able to search in large data bank all of these kind of transcripts by using kmer methods. It will take few minutes for about 1000 samples.*

The platform is also involved in bioinformatics training for biologist, from R beginners, RNA-Seq data analysis to mutation research.

Keywords NGS, RNA-Seq, Chimera, lncRNA, kmers.

1 Introduction

The aim of this platform is to provide bioinformatics analysis to researcher in the NGS field. We have an extensive range of expertise in the RNA-Seq analysis data. The platform is located at the Institute of Regenerative Medicine and Biotherapies (IRMB) in Montpellier. It has exchange with different laboratories around Montpellier in the academic or private research.

2 Actitivy

Computational techniques are used to analyze high-throughput sequences data. In the last 10 years, the sequencing technology has grown faster than the computational biology, which makes more difficult and complex to analyze of all generated data. Therefore, the software to process sequencing data evolved rapidly and require a lot of expertise before using them. The team members are associated with a research team which analyze, develop pipeline and software dedicated to the RNA-Seq data.

Major services provided by the platform:

1. Help to design experiment for NGS
2. RNA-Seq analysis
   - Quality check of sequencing data: FastQC, kmerTool
   - Alignment on reference sequence: CRAC, STAR, Hisat2
   - Differential gene expression based on k-mers: DE-kupl[1] , CountTags
   - Differential gene expression based on transcripts or genes sequence: Kallisto, SLEUTH, DESeq2, EdgeR
   - Search for chimeric genes and/or RNA: ChimCT[2,3], STARFusion
   - Transcripts assembly for new genes, unannotated genes, coding genes or no-coding genes (lncRNA): Stringtie
3. DNA-Seq analysis
   - Variation and mutation search: CracTools, Hisat2/Freebayes
4. Specific analysis on demand

The research team associated with the platform also develop new software based on kmer search, which permits to look in large sample data libraries (more than 1000 samples) for gene or transcript expression, chimera transcript or other transcriptional events.

3 IT infrastructure

Computational biology, in the NGS field, require huge computer resources for a wide range of computationally intensive tasks. The platform shares IT infrastructure with other team in our building. Different servers are available for services like web server as well as the IT management. The low and small intensive analysis are done on a local cluster (about 20 nodes and some with large memory). This cluster, as IT managed entirely by the team, give the freedom to the team to do what they want.

For large computer intensive tasks, the team use the high-throughput computing (HPC) cluster provided by the Meso@LR (https://meso-lr.umontpellier.fr/) located in Montpellier.

All the pipeline analysis are under control of a bioinformatics wokflow management system (snakemake) and use singularity container that can be run directly in any HPC cluster available. All results and files are given to the scientist via a web site, with restricted access for each projects. All generated files are under versioning control to track the provenance of the workflow execution results and to certify for quality control.

4 Training

The platform manage few training in the bioinformatics fields. Theses training are designed principally towards biologist that are interested to understand what is a behind a bioinformatics workflow.

The next one will be about how to find a mutation in a gene panel or exome experiment. The purpose of this training is to learn:

— how to do control quality check on fastq sequences
— how to do the mapping on reference sequences
— how to do variant calling
— how to do variant filtering and why

Theses training are organized once or two per year, generally between March and June of each year, in collaboration with the Mobidic bioinformatics team of Montpellier CHU and Seq.one corporate. The expertise require for these courses are provided by associate professor and professor of Montpellier University, and researchers from INSERM.

Acknowledgements

This work is supported by CHRU of Montpellier, Montpellier University and INSERM.

References

[1] Jérôme Audoux, Nicolas Philippe, Rayan Chikhi, Mikaël Salson, Mélina Gallopin, Marc Gabriel, Jérémy Le Coz, Emilie Drouineau, Thérèse Commes, and Daniel Gautheret. DE-kupl: exhaustive capture of biological variation in RNA-seq data through k-mer decomposition. **2017** - *Genome Biology* - 18(1):243.

[2] Florence Rufflé, Jerome Audoux, Anthony Boureux, Sacha Beaumeunier, Jean-Baptiste Gaillard, Elias Bou Samra, Andre Megarbane, Bruno Cassinat, Christine Chomienne, Ronnie Alves, Sebastien Riquier, Nicolas Gilbert, Jean-Marc Lemaitre, Delphine Bacq-Daian, Anne Laure Bougé, Nicolas Philippe, and Therese Commes. New chimeric RNAs in acute myeloid leukemia. **2017** *F1000Research* [version 2; peer review: 2 approved]. 6(ISCB Comm J):1302

[3] Bougé, Anne-Laure and Rufflé, Florence and Riquier, Sébastien and Guibert, Benoit and Audoux, Jérôme and Commes, Thérèse RNA-Seq Analysis to Detect Abnormal Fusion Transcripts Linked to Chromothripsis **2018** - *Methods in Molecular Biology* - 1769: 133-156

Paper 216

Towards CNNs Representations for Small Data Classification

Khawla Seddiki[1,2], Philippe Saudemont[2], Frédéric Precioso[3], Nina Ogrinc[2], Maxence Wisztorski[2], Michel Salzet[2], Isabelle Fournier[2] and Arnaud Droit[1]

[1] Centre de Recherche du CHU de Québec - Université Laval, Québec City, Canada

[2] Université Cote d'Azur, CNRS, I3S, Sophia Antipolis, France

[3] Univ. Lille, Inserm, CHU Lille, U1192 - Protéomique Réponse Inflammatoire Spectrométrie de Masse - PRISM, F-59000 Lille, France

Corresponding author: khawla.seddiki.1@ulaval.ca

Abstract *Rapid and accurate clinical diagnosis from Mass Spectrometry (MS) remains highly challenging. Some machine learning (ML) approaches, including Support Vector Machine or Random Forest for instance, have been investigated for this purpose. An important component of this development is the building of effective classification models with MS data. These ML algorithms require time-consuming preprocessing steps such as baseline correction, denoising, and spectra alignment to remove non-sample-related data artifacts. They also depend on the laborious extraction of features, making them unsuitable for rapid analysis. Convolutional Neural Networks (CNNs) have been found to perform well under such circumstances since they can learn efficient representations from data without the need for preprocessing. However, their effectiveness drastically decreases when the number of MS spectra available is small, which is a common situation in medical applications. Transfer learning strategies can extend an accurate representation model learnt from a large dataset to a smaller one. We first investigated transfer learning by a 1D-CNN model designed to classify MS data and then we developed a new cumulative learning method when transfer learning was not powerful enough as in cases of low-resolution or data heterogeneity. What we proposed is to train the same model through several classification tasks over various small datasets in order to accumulate MS knowledge in the resulting representation. Using a cumulative learning approach resulted in a classification accuracy exceeding 98% for 1D clinical canine sarcoma cancer cells, human ovarian cancer serums, and pathogenic microorganisms. We showed for the first time the use of cumulative representation learning using datasets generated in different biological contexts, on different organisms, and acquired by different instruments. Our approach thus illustrates a promising strategy for improving classification accuracy when only small numbers of samples are available as prospective cohorts.*

Keywords Transfer learning, Cumulative learning, CNNs

1 Introduction

Accurate and rapid identification of cancer tissues has a crucial impact on medical decisions. Conventional histopathological examinations are resource intensive and time-consuming, requiring 30–45 minutes per sample processed and the presence of a skilled pathologist [1]. A similar need exists in the treatment of infections, where accurate identification of microorganisms responsible for human infections is important to ensure the most appropriate and effective treatment for a patient, in the shortest possible time [2]. In this context, it is essential to use tools which provide accurate identification and correct interpretation of the analyzed samples. Mass spectrometry (MS) is particularly useful for such purposes since it provides non-targeted molecular information on the millisecond time scales. Its sensitivity, reproducibility, and suitability for analyzing complex mixtures are well established. New methods of analysis of crude samples are making diagnostics even faster and easier. Simultaneously, the development of MS-based bacterial biotyping clearly illustrates the value of MS in clinical applications [3].

For cancer-related diagnostics and microbial pathogen identifications, many popular classification Machine Learning models, such as Support Vector Machine (SVM) [4], Random Forest (RF) [5], and Linear Discriminant Analysis (LDA) [6] have been already used and compared [7] [8]. However, these

methods are applied generally to preprocessed MS data, and differences in preprocessing pose a major challenge to any comparison of MS data analysis. Classification model design for rapid applications thus becomes a highly complex task, since it must follow a workflow involving several interdependent preprocessing steps. Data preprocessing is used to improve the robustness of subsequent multivariate analysis and to increase data interpretability by correcting issues associated with MS acquisition [9]. Preprocessing quality is important, and if inadequate, can lead to biased or biologically irrelevant conclusions [10]. Several factors, often related to the experimental conditions including sample heterogeneity, sample processing and MS analysis (e.g. electronic noise, instrument calibration stability, temperature stability,...) can contribute to spectral variations. In addition, the *curse of dimensionality*, must be avoided. This is a well-known problem that arises when analyzing MS data having a large number of dimensions, and is lessened using data dimensionality reduction techniques [11]. Various MS classification workflows have been developed so far, but there is no golden standards for the optimal choice of parameters at each individual step, for their quality evaluation or for their best combination [12]. It has been shown that the choice of preprocessing parameters for a specific dataset can decrease the performance of the classification model and that preprocessing may be effective only for that dataset and not any others generated from different instruments or with different settings [13]. A standard pipeline for MS classification using SVM, RF or LDA must include these preprocessing steps and must consider aforementioned constraints, which makes such algorithms unsuitable for rapid analysis. Convolutional Neural Networks (CNNs) are one of the most successful deep learning architectures designed to learn representation from an input signal with different levels of abstraction [14]. To address rapid clinical MS data classification tasks, CNNs represent an attractive approach offering various advantages over conventional Machine Mearning algorithms. These include significantly higher accuracy, effectiveness on raw spectrum classification even in presence of signal artifacts (noise, baseline distortion, etc.) and hence discards the need for data preprocessing before classification [15], integration of features extraction with classification and without a feature-engineering step since all layers are trained together, and finally exploitation of spatially stable local correlations by enforcing the local connectivity patterns [16]. However, CNNs classification efficiency trained using a small number of spectra drops rapidly [15]. Unfortunately, many real-world applications do not have access to big training sets because of data scarcity, or because of the difficulty and expense in labeling data [17]. In medicine, it is often the case that some samples are only accessible in limited amounts, especially for rarer diseases and pathologies (e.g. patient biopsies, at advanced stage of infection). Therefore the size of clinical datasets is constrained by data availability and by the experiments complexity and high cost [18]. For such applications, transfer learning has emerged as an interesting approach [19]. This technique is applicable to small datasets and therefore requires fewer computational resources while increasing the classification accuracy as compared to CNNs models built from scratch. Transfer learning is a two-step process. An accurate data representation is first learned, by training a model on a dataset containing a large amount of annotated data covering many categories. This representation (i.e. its model weights) are then reused to build a new model based on a smaller annotated dataset containing fewer categories, by training only the final decision layer(s) or by also fine-tuning the whole model with the reduced set of categories. Transfer learning has proven useful in many engineering areas including computer vision, robotics, image classification and natural language processing (NLP) applications [20]. With MS data, it would use basic similarities in spectral shape gathered from different datasets and adapted to address new classification problems. This has yet to be explored for 1D spectral data, since no 1D spectral dataset as large as the ImageNet database in the 2D image analysis domain is available [21]. Most MS classification by CNN is therefore focused on MS 2D imaging analysis [22] [23] [24]. We have found no description of their use or of transfer learning or representation learning in conjunction with 1D MS data. The aim of this study was to build CNNs-based classification models for 1D mass spectra by transfer learning or representation learning. Pattern recognition models were built using small clinical datasets generated for the diagnosis of cancers or microbial infections.

2 Methods

2.1 Datasets

We evaluated our proposed approach on independent MS datasets (Table 1):

	MS instrument	Dataset	Classes	# spectra	Description
Target domain data	Synapt G2-S Q-TOF (Waters, SpiderMass)	Canine sarcoma	Normal	482	Contained 1 normal and 11 heterogeneous sarcoma types as described previously [25]
			Myxosarcoma	60	
			Fibrosarcoma	404	
			Hemangiopericytoma	134	
			Malignant peripheral nerve tumor	60	
			Osteosarcoma	339	
			Undifferentiated pleomorphicsarcoma	376	
			Rhabdomyosarcoma	66	
			Splenic fibrohistiocytic nodules	63	
			Histiocytic sarcoma	105	
			Soft tissue sarcoma	69	
			Gastrointestinal stromal sarcoma	70	
			Total	**2228**	
	Hybrid quadrupole (QSTAR pulsar I)	Human ovary cancer 1	Normal	95	Contained two classes of low-resolution spectra, normal and cancerous publicly available at home.ccr.cancer.gov/ncifdaproteomics/ppatterns.asp
			Cancer	121	
			Total	**216**	
	Synapt G2-S Q-TOF (Waters, SpiderMass)	Microorganisms	Staphylococcus aureus	26	Contained a five human pathogen as described previously [26]
			E.coli D31	26	
			Pseudomonas aeruginosa	24	
			Enterococcus faecalis	19	
			Candida albicans	23	
			Total	**119**	
Source domain data	PBSII SELDI-TOF	Human ovary cancer 2	Normal	91	Contained two classes of high-resolution spectra , normal and cancerous publicly available at home.ccr.cancer.gov/ncifdaproteomics/ppatterns.asp
			Cancer	162	
			Total	**253**	
	Rapiflex MALDI-TOF (Bruker)	Rat brain	Gray matter	4635	Contained spectra of rat gray and white brain matter
			White matter	5465	
			Total	**10100**	
	Synapt G2-S Q-TOF (Waters, SpiderMass)	Beef liver	Positive mode	1372	Contained two types of spectra of healthy beef liver samples, one acquired in positive and the other in negative ion mode
			Negative mode	1265	
			Total	**2637**	

Tab. 1. Description of datasets

We focused on lipids and metabolites as the main species observed in the 100-1.600 mass/charge (m/z) range with our MS instrument, namely the SpiderMass. Multiple studies have shed light on the role of lipid metabolism deregulation in cancer development [27]. Recent microbial taxonomy studies have also demonstrated the possibility of biotyping pathogens using their lipid composition [28]. The classification models obtained using public ovarian datasets were based on lipids and proteins patterns since the m/z range is 700-12.000 [29].

2.2 Evaluation protocol

All datasets were imported without undergoing any preprocessing step. Each dataset were binned at 0.1 Da, linearly scaled between 0 and 1, and divided randomly into training, validation, and test with ratios of 60%, 20%, and 20%, respectively. Performance of trained classifiers was measured by global accuracy on test subsets averaged over 10 independent iterations. For each iteration a stratified 5-fold cross validation was used to maintain the original proportion of minority classes. A weighted loss function was used during the training for samples from under-represented classes.

Hyper-parameter search We evaluated the effects of hyper-parameter value alterations on the classification accuracy of the clinical datasets by CNNs. MS data hyper-parameters selection have a huge impact on the performance, as strong as images. We began with an investigation of the optimal convolutional filter size for the extraction of spectral features, followed by a search of various learning rate, including 0.1, 0.01, and 0.001 with reducing learning rate when validation set accuracy stopped improving during 10 epochs. We investigated the use of two optimizer algorithms, including Adam and Stochastic gradient descent (SGD). We also searched the use of various batch sizes, including 64, 128, and 256. This evaluation was also done in terms of regularizer technique by adding either batch normalization, dropout of 0.5 or L1/L2 regularization after each convolutional layer. Using this approach, we expected to determine what model depth and hyper-parameters are optimal for classification of MS spectra, especially in the case of highly heterogeneous biological classes such as canine cancer types.

Protocol for evaluating 2D-CNN adapted to 1D We evaluated and compared the application of three prominent CNN architectures for classifying spectra in clinical datasets. The first of these was variant_Lecun contained two convolutional layers and two fully connected layers (model 1), adapted from [30], the second was variant_LeNet included three convolutional layers and two fully connected layers (model 2) [15], and the third was variant_VGG9 with six convolutional layers and three fully connected layers (model 3), adapted from [20].

Protocol for evaluating transfer learning The three CNN architectures were trained on the large rat brain dataset with all weights initialized according to He normal distribution. Rat brain dataset was chosen as the source domain data as it was the largest one. The decision layers of the

network were not useful, since the rat brain and clinical datasets were from different contexts. The convolutional weights were then frozen so that they would not be updated during back-propagation, the decision layers were removed, and the new specific decision layers dedicated to smaller clinical datasets were trained (target domain datasets). Transfer learning from the rat brain dataset allowed the model to learn and detect generic representations of MS peaks. By freezing the lower CNN levels, we are assuming that the model has extracted the right patterns, and that only the high level is needed to take into account specific peak's features.

Protocol for evaluating cumulative learning Transfer learning in some cases may not be enough as an aid to classifying biologically similar materials using CNN models. This proximity is reflected in a high degree of confusion between classes. This is typically the case when the biggest dataset which is supposed to be used to learn the pivotal data representation is not big enough. In addition, low-resolution or data heterogeneity can further complicate the classification task. We therefore proposed two approaches to developing 1D CNN cumulative learning:

Scenario A The first step was to train CNN architectures on the rat brain dataset as described before for transfer learning. The model weights are then fine-tuned, the decision layers are removed, and new decision layers are trained with the beef liver dataset, then its weights were frozen and new specific decision layers were added and trained using the canine cancer dataset. For the human ovary 2 dataset, rat brain weights were frozen and new specific decision layers were added and trained using the human ovary 1 dataset.

Scenario B CNN architectures were trained on the rat brain and fine-tuned with the beef liver dataset as described in Scenario A, but instead of testing this model on the canine cancer dataset, an additional learning was added. Beef liver CNN weights were fine-tuned, decision layers were removed and new specific decision layers were added and trained using the microorganisms dataset, before freezing convolutional layer weighting and training new specific decision layers on the canine cancer dataset. The resulting CNN model from Scenario B was tested with changes to the dimensionality of the output space (number of classes) and the activation function of the last fully connected layer on rat brain, beef liver and microorganisms datasets separately. The objective was to assess how much learning skill the final CNN gained or lost of MS knowledge through successive training.

Protocol for comparing our approach with conventional Machine Learning algorithms To make such a comparison valid, all spectra were binned similarly, and the same ratio of training, validation and test subsets was conserved. These conventional algorithms are not designed to classify MS spectra that have not been preprocessed. In order to compare their performance to that of CNNs on raw data, spectra were corrected using sequential preprocessing of five steps: (1) Savitzky-Golay-Filter denoising, (2) baseline subtraction using the statistics-sensitive non-linear iterative peak-clipping, (3) normalization on the total ion count, (4) alignment using a cubic warping function, (5) and peaks detection using the median absolute deviation. Chi-square (χ^2) statistic was used to reduce data dimensionality before feeding to the classification algorithms.

3 Results

Hyper-parameter search The regularizer technique, the optimizer algorithm and the learning rate revealed significant effects on classification accuracy. Batch normalization, used after each convolutional layer to avoid over-fitting, was found superior to the dropout technique and L1/L2 regularization. The Adam optimizer with default hyper-parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$ and a fixed learning rate of $\eta = 0.001$ was found superior to the SGD algorithm. Adam was carried out using a cross-entropy loss function. We also found that Max-Pooling was very important in order to account for peak shift invariance along the m/z dimension. Since batch size did not affect the results, it was set at 256. ReLu (models 1 and 3) and Leaky Relu (model 2) were chosen as the activation function for each convolutional layer. We found that large filter size was more effective than image-optimized filtering (pixel features). This indicates that features extracted from spectral data differ from those seen in images. The three CNNs architectures and their best hyper-parameters are shown in Figure 1.

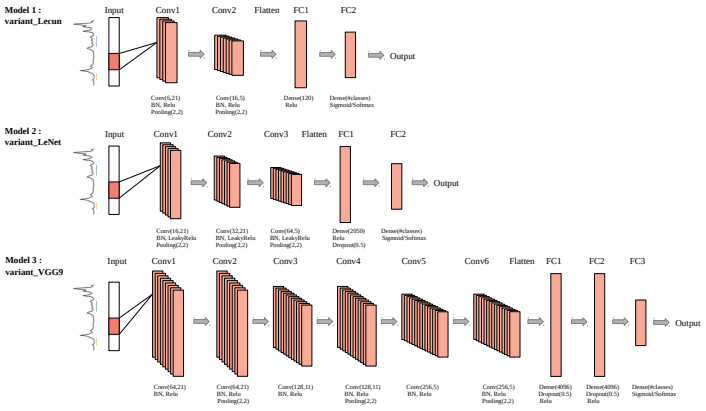


Fig. 1. Architectures of the three CNN models. Convolutional layers are labeled as Conv, flatten layer as Flatten, and fully connected layers as FC

CNNs classification performance

For canine sarcoma classification, binary (2 classes) classification of tissues as healthy or cancerous was sought first, followed by differentiation of sarcoma type (12 classes).

Datasets	# classes	variant_Lecun	variant_LeNet	variant_VGG9
Canine sarcoma	2	**0.98 ± 0.00**	0.96 ± 0.01	0.96 ± 0.01
	12	0.88 ± 0.03	0.88 ± 0.02	**0.90 ± 0.01**
Microorganisms	5	**0.89 ± 0.02**	0.68 ± 0.03	0.61 ± 0.13

Tab. 2. Overall accuracy of classification of SpiderMass spectra using three CNN architectures. The best result for each task over 10 independent iterations is indicated in boldface.

As shown in Table 2, variant_Lecun was the best at binary classification of canine sarcoma, but when the number of classes was expanded to 12, variant_VGG9 was slightly better. This suggests that deep CNNs might be better at sorting out heterogeneous samples. Variant_Lecun was the best at classifying microorganisms. Accuracy suffers quickly from over-fitting when a deep architecture such as variant_LeNet and variant_VGG9 are used on data of this size. The only classification that could be described as accurate was for canine sarcoma versus healthy tissue (binary classification) by variant_Lecun with an average accuracy of 0.98. Based on this result, we focused our subsequent efforts on the canine sarcoma and microorganism multi-class classifications.

Transfer learning

Datasets	# classes	variant_Lecun	variant_LeNet	variant_VGG9
Canine sarcoma	12	0.90 ± 0.01 (2%)	**0.92 ± 0.01** (3%)	**0.93 ± 0.02** (3%)
Microorganisms	5	**0.99 ± 0.00** (10%)	**0.99 ± 0.00** (31%)	0.96 ± 0.02 (36%)

Tab. 3. Overall accuracy of classification of SpiderMass spectra using three CNN architectures after transfer learning. The improvement in performance from scratch is expressed as a percentage

As shown in Table 3, transfer learning clearly improved the accuracy of classification of both small SpiderMass datasets compared to the models trained from scratch (without transfer learning). Gains in the accuracy of canine sarcoma differentiation were obtained for all three architectures, although much room for improvement remained. variant_LeNet and variant_VGG9 predicted the correct classes with almost equal success. Improvements was considerable also for the 5-class microorganism task, and huge in the case of variant_VGG9. These results suggest that training a CNN model with extracted spectral features transferred even from an unrelated field is better than training it with spectral features learned from scratch with a small dataset. The aim of the following experiments was to improve the canine sarcoma multi-class classification performance.

Cumulative learning

Two scenarios were tested: (A) training on intermediate beef liver and then on canine cancer dataset; (B) training on beef liver, then on microorganisms and lastly on canine cancer dataset.

Protocol	**variant_Lecun**	**variant_LeNet**	**variant_VGG9**
Scenario A	0.92 ± 0.01 (4%* 2%**)	**0.95 ± 0.01** (7%* 4%**)	0.94 ± 0.01 (4%* 1%**)
Scenario B	0.95 ± 0.02 (7%* 5%** 3%***)	**0.99 ± 0.00** (10%* 7%** 3%***)	0.96 ± 0.00 (6%* 3%** 2%***)

Tab. 4. Overall accuracy of canine sarcoma classification by the three CNN architectures. The improvement in performance is expressed as a percentage relative to learning from scratch*, to transfer learning**, and to Scenario A***

As shown in Table 4, Scenario A improved the classification accuracy considerably relative to learning from scratch and slightly relative to transfer learning, the best improvements being obtained for variant_LeNet. Scenario B provided a slight additional improvement over Scenario A, and the greatest accuracy was achieved also with variant_LeNet architecture. The effectiveness of the cumulative knowledge method is thus apparent, enabling the CNNs to distinguish not only cancerous versus healthy tissues (binary classification), but also the different cancer types despite the small size and the heterogeneity of the dataset.

Classification accuracy obtained by CNN from scratch on data used for the training (rat brain and beef liver) and after transfer learning for microorganism (Table 3) was equal to 0.99. Testing the final cumulative representation of variant_LeNet (from Scenario B) on rat brain, beef liver and microorganism datasets separately did not show improvement of the classification accuracy from 0.99. This indicates that the CNN model accumulates MS knowledge through the successive training phases without any losses.

Public MS datasets classification

We assessed CNNs performance following the same training and evaluation approach, but with variant_LeNet architecture only, because of its superior performance with SpiderMass datasets and its low computational resources needed. Variant_LeNet was thus trained on the rat brain dataset as the source domain, followed by the transfer learning protocol using the high-resolution dataset and representation-learning Scenario A using the low-resolution dataset.

Dataset	**# classes**	**variant_LeNet**	**Transfer learning**
Human ovary 1	2	0.78 ± 0.02	**0.98 ± 0.00** (24%*)

Dataset	**# classes**	**variant_LeNet**	**Transfer learning**	**Cumulative learning**
Human ovary 2	2	0.80 ± 0.00	0.83 ± 0.02 (3%*)	**0.99 ± 0.00** (24%* 19%**)

Tab. 5. Overall accuracy of a variant_LeNet architecture at classifying ovarian cancer serums; percent improvement relative to learning from scratch* and to transfer learning**

Transfer learning improved classification accuracy from 0.78 for training from scratch to 0.98 for the high-resolution dataset (Table 5). With the low-resolution dataset, accuracy was improved from 0.80 to 0.83 by transfer learning and to 0.99 by cumulative learning. These results show that in contrast with the previously reported lack of sensitivity and specificity of low-resolution MS datasets for cancer diagnosis [29], our CNN representation model was up to the task and without any need for spectral preprocessing steps.

Performance of conventional algorithms applied to preprocessed datasets

As shown in Table 6, RF outperformed the other methods, while LDA was best only for human ovary 2 classification. Performance of RF and LDA was not comparable to that of CNNs. In addition, RF and LDA require more time to carry out the necessary preprocessing steps and to determine the optimal hyper-parameters since datasets had different artifacts and therefore required different preprocessing strategies.

Datasets	# classes	SVM	RF	LDA
Canine sarcoma	2	0.76 ± 0.16 (22%*)	**0.96 ± 0.01** (2%*)	0.88 ± 0.17 (10%*)
	12	0.52 ± 0.19 (47%***)	**0.65± 0.01** (34%***)	0.61 ± 0.02 (38%***)
Microorganisms	5	0.54 ± 0.35 (45%**)	**0.86 ± 0.01** (13%**)	0.51 ± 0.26 (48%**)
Human ovary 1	2	0.66 ± 0.24 (49%***)	**0.91 ± 0.02** (8%***)	0.85 ± 0.06 (15%***)
Human ovary 2	2	0.60 ± 0.05 (65%**)	0.88 ± 0.03 (12%**)	**0.97 ± 0.00** (2%**)

Tab. 6. Overall accuracies of clinical spectra classifications by SVM, RF, and LDA; percent of difference to 1D CNN trained from scratch*, from transfer learning**, and from representation learning***

4 Discussion and Conclusions

CNNs have become common tools in several research areas. They are designed to extract spatial features from input signals with different levels of abstraction. We have investigated here the performance of CNNs in the classification of 1D mass spectra generated for clinical purposes. This study shows for the first time the use of cumulative learning for 1D spectrum classification of datasets generated in vastly different biological contexts, on different organisms, acquired by a variety of instruments and technologies at different resolutions. Our CNN model was designed by accumulating mass spectral knowledge through multiple training steps on small datasets. It provided a viable alternative when transfer learning was inadequate, as was the case for low-resolution, heterogeneous MS data, or when the source domain dataset was not large enough. The novelty is that the model can be pre-trained on a dataset containing only two output categories and yet predict 2, 5 and even 12 outputs, that are unlikely to share common features. Our CNN model was able to classify raw MS data without preprocessing steps, thus bypassing the expert parameter setting step. This performance capability is due to convolutional filters that allow CNN architecture to learn peak patterns rather than only considering each m/z intensity value separately as do conventional algorithms. More importantly, significant variations of the overall signal intensity due to biological heterogeneity and non-reproducible technical factors (not all peaks showing up in each sample) are taken into account in the pattern recognition of CNNs. In the present study, we investigated the performance of our learning approach for MS data classification. It would be interesting to extend the investigation to analyze which data characteristics are similar between the different datasets. The focus of future research will be the interpretation of classification results in order to identify regions of interest in spectra, which may correspond to new biomarkers. It would be also interesting to understand how the relative abundance and position of such biomarkers might be used to discover new therapeutic or diagnostic targets.

References

[1] Jialing Zhang, John Rector, John Q Lin, Jonathan H Young, Marta Sans, Nitesh Katta, Noah Giese, Wendong Yu, Chandandeep Nagi, James Suliburk, et al. Nondestructive tissue analysis for ex vivo and in vivo cancer diagnosis using a handheld mass spectrometry system. *Science translational medicine*, 9(406):eaan3968, 2017.

[2] Anand Kumar, Daniel Roberts, Kenneth E Wood, Bruce Light, Joseph E Parrillo, Satendra Sharma, Robert Suppes, Daniel Feinstein, Sergio Zanotti, Leo Taiberg, et al. Duration of hypotension before initiation of effective antimicrobial therapy is the critical determinant of survival in human septic shock. *Critical care medicine*, 34(6):1589–1596, 2006.

[3] Markus Kostrzewa. Application of the maldi biotyper to clinical microbiology: progress and potential. *Expert review of proteomics*, 15(3):193–202, 2018.

[4] Corinna Cortes and Vladimir Vapnik. *Support-vector networks*, volume 20. Springer, 1995.

[5] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[6] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

[7] Baolin Wu, Tom Abbott, David Fishman, Walter McMurray, Gil Mor, Kathryn Stone, David Ward, Kenneth Williams, and Hongyu Zhao. Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. *Bioinformatics*, 19(13):1636–1643, 2003.

[8] Devin A Gredell, Amelia R Schroeder, Keith E Belk, Corey D Broeckling, Adam L Heuberger, Soo-Young Kim, D Andy King, Steven D Shackelford, Julia L Sharp, Tommy L Wheeler, et al. Comparison of

machine learning algorithms for predictive modeling of beef attributes using rapid evaporative ionization mass spectrometry (reims) data. *Scientific reports*, 9(1):5721, 2019.

[9] Melanie Hilario, Alexandros Kalousis, Christian Pellegrini, and Markus Mueller. Processing and classification of protein mass spectra. *Mass spectrometry reviews*, 25(3):409–449, 2006.

[10] Akin Ozcift and Arif Gulten. Assessing effects of pre-processing mass spectrometry data on classification performance. *European Journal of Mass Spectrometry*, 14(5):267–273, 2008.

[11] Melanie Hilario and Alexandros Kalousis. Approaches to dimensionality reduction in proteomic biomarker studies. *Briefings in bioinformatics*, 9(2):102–118, 2008.

[12] Alejandro Cruz-Marcelo, Rudy Guerra, Marina Vannucci, Yiting Li, Ching C Lau, and Tsz-Kwong Man. Comparison of algorithms for pre-processing of seldi-tof mass spectrometry data. *Bioinformatics*, 24(19):2129–2136, 2008.

[13] Jasper Engel, Jan Gerretzen, Ewa Szymańska, Jeroen J Jansen, Gerard Downey, Lionel Blanchet, and Lutgarde MC Buydens. Breaking with trends in pre-processing? *TrAC Trends in Analytical Chemistry*, 50:96–106, 2013.

[14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[15] Jinchao Liu, Margarita Osadchy, Lorna Ashton, Michael Foster, Christopher J Solomon, and Stuart J Gibson. Deep convolutional neural networks for raman spectrum recognition: a unified solution. *Analyst*, 142(21):4067–4074, 2017.

[16] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[17] George Forman and Ira Cohen. Learning from little: Comparison of classifiers given little training. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 161–172. Springer, 2004.

[18] Torgyn Shaikhina and Natalia A Khovanova. Handling limited datasets with neural networks in medical applications: A small-data approach. *Artificial intelligence in medicine*, 75:51–63, 2017.

[19] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[20] Jie Lu, Vahid Behbood, Peng Hao, Hua Zuo, Shan Xue, and Guangquan Zhang. Transfer learning using computational intelligence: a survey. *Knowledge-Based Systems*, 80:14–23, 2015.

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[22] Jens Behrmann, Christian Etmann, Tobias Boskamp, Rita Casadonte, Jörg Kriegsmann, and Peter Maaβ. Deep learning for tumor classification in imaging mass spectrometry. *Bioinformatics*, 34(7):1215–1223, 2017.

[23] Jannis van Kersbergen, Farhad Ghazvinian Zanjani, Svitlana Zinger, Fons van der Sommen, Benjamin Balluff, Naomi Vos, Shane Ellis, Ron M.A. Heeran, Marit Lucas, Henk A. Marquering, Ilaria Jansen, C. Dilara Savci-Heijink, Daniel M. de Bruin, and Peter H. N. de With. Cancer detection in mass spectrometry imaging data by dilated convolutional neural networks. In *Medical Imaging 2019: Digital Pathology*, 2019.

[24] Lei Huang and Tong Wu. Novel neural network application for bacterial colony classification. *Theoretical Biology and Medical Modelling*, 15(1):22, 2018.

[25] Philippe Saudemont, Jusal Quanico, Yves-Marie Robin, Anna Baud, Julia Balog, Benoit Fatou, Dominique Tierny, Quentin Pascal, Kevin Minier, Mélissa Pottier, et al. Real-time molecular diagnosis of tumors using water-assisted laser desorption/ionization mass spectrometry technology. *Cancer cell*, 34(5):840–851, 2018.

[26] Benoit Fatou, Michel Salzet, and Isabelle Fournier. Real time human micro-organisms biotyping based on Water-Assisted Laser Desorption/Ionization. *The EuroBiotech Journal*, 3(2), 2019.

[27] Claudio R Santos and Almut Schulze. Lipid metabolism in cancer. *The FEBS journal*, 279(15):2610–2623, 2012.

[28] Simon JS Cameron, Zsolt Bodai, Burak Temelkuran, Alvaro Perdones-Montero, Frances Bolt, Adam Burke, Kate Alexander-Hardiman, Michel Salzet, Isabelle Fournier, Monica Rebec, et al. Utilisation of ambient laser desorption ionisation mass spectrometry (aldi-ms) improves lipid-based microbial species level identification. *Scientific reports*, 9(1):3006, 2019.

[29] Thomas P Conrads, Vincent A Fusaro, Sally Ross, Don Johann, Vinodh Rajapakse, Ben A Hitt, Seth M Steinberg, Elise C Kohn, David A Fishman, Gordon Whitely, et al. High-resolution serum proteomic features for ovarian cancer detection. *Endocrine-related cancer*, 11(2):163–178, 2004.

[30] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Paper 226

Evolution of replication origins in vertebrate genomes: rapid turnover despite selective constraints

Florian Massip[1,2], Marc Laurent[3], Caroline Brossas[3], José Miguel Fernández-Justel[4], María Gómez[4], Marie-Noelle Prioleau[3], Laurent Duret[1,5] and Franck Picard[1,5]

[1] Université de Lyon, Université Lyon 1, CNRS, Laboratoire de Biométrie et Biologie Evolutive UMR 5558, 43 bd du 11 novembre 1918 69622 Villleurbanne, France

[2] Berlin Institute for Medical Systems Biology, Max Delbrueck Center for Molecular Medicine, Hannoversche Str. 28, 10115 Berlin , Germany

[3] Institut Jacques Monod, CNRS UMR7592, Université Paris Diderot, 15 rue helene Brion, 75013, Paris, France

[4] Centro de Biología Molecular Severo Ochoa CBMSO (CSIC/UAM). Nicolás Cabrera 1. 28049, Madrid, Spain

[5] These authors contributed equally

Corresponding author: florian.massip@gmail.com

Reference paper: **Massip *et al.* (2019) Evolution of replication origins in vertebrate genomes: rapid turnover despite selective constraints *Nucleic Acids Research.*** https://academic.oup.com/nar/article/47/10/5114/5420529

Introduction: DNA replication follows a spatiotemporal program that ensures the faithful replication of genomes at each cell cycle. In vertebrate genomes, DNA replication initiates at precise genomic regions, called replication origins. While DNA replication is one of the most important process of cells' life, the molecular mechanisms inducing the firing of replication are still poorly understood [1]. Notably, the question of whether origins positioning in vertebrates is determined by sequence structures or by epigenetic marks remains unresolved. To study the genetic determinants of replication origins, we conducted an evolutionary analysis of replication origins in vertebrates.

Methods: We generated a genome-wide map of chicken origins (the first of a bird genome), and reanalyzed published SNS sequencing data from human [2] and mouse genomes [3] using the same peak-calling methodology to ensure that the sensitivity of origin detection was similar in all species.

Results: Comparing maps of replication origins in vertebrates, we find origins to be associated to the same genomic elements (namely G-quadruplexes, CpG islands and transcription start sites) in all species, confirming the importance of these elements in replication firing. Next, we analysed the intra-species polymorphism at origins loci. Our study revealed a strong depletion of genetic diversity at the core of replication initiation loci, showing that origins are associated to strong sequence constraints and that mutations in these regions have a deleterious effect. In contrast, inter-species comparisons revealed very limited conservation of replication origins on larger evolutionary scale. Indeed, we found that replication landscapes have been largely remodeled during the evolution of vertebrates. While the replication initiation activity in human and chicken genomes is concentrated in clusters of very active loci, the mouse genome presents a more uniform distribution. In addition, we showed that origins experienced a rapid turnover during vertebrate evolution, since pairwise comparisons of origin maps revealed that $< 24\%$ of them are conserved among vertebrates.

Conclusion: Our study highlights the flexibility of the spatial program of replication in vertebrate genomes, unraveling the existence of a novel genetic determinant of replication origins in vertebrates, the precise nature of which remains to be determined.

References

[1] Marie-Noëlle Prioleau and David M MacAlpine. Dna replication origins—where do we begin? *Genes & development*, 30(15):1683–1697, 2016.

[2] Emilie Besnard, Amélie Babled, Laure Lapasset, Ollivier Milhavet, Hugues Parrinello, Christelle Dantec, Jean-Michel Marin, and Jean-Marc Lemaitre. Unraveling cell type–specific and reprogrammable human replication origin signatures associated with G-quadruplex consensus motifs. *Nature Structural & Molecular Biology*, 19(8):837–844, July 2012.

[3] Christelle Cayrou, Benoit Ballester, Isabelle Peiffer, Romain Fenouil, Philippe Coulombe, Jean-Christophe Andrau, Jacques van Helden, and Marcel Méchali. The chromatin environment shapes dna replication origin organization and defines origin classes. *Genome research*, 25(12):1873–1885, 2015.

Paper 227

InterPro: the protein classification database

Typhaine PAYSAN-LAFOSSE[1], Matthias Blum[1], Hsin-Yu Chang[1], Swaathi Kandasaamy[1], Gift Nuka[1], Matloob Qureshi[1], Lorna Richardson[1], Gustavo A. Salazar[1] and Robert D. FINN[1]

1 European Molecular Biology Laboratory, European Bioinformatics Institute (EMBL-EBI), Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SD, UK

Corresponding Author: typhaine@ebi.ac.uk

Paper Reference: **Mitchell AL, Attwood TK, Babbitt PC, et al. InterPro in 2019: improving coverage, classification and access to protein sequence annotations. Nucleic Acids Research. 2019 Jan;47(D1):D351-D360.** http://doi.org/10.1093/nar/gky1100

Over the past few years the volume of nucleic acid sequencing has continued to grow dramatically. This sequence data encodes millions of proteins, the vast majority of which have never been experimentally characterized. To overcome this paucity of information, the function of these protein sequences is inferred through the automatic transfer of information from a few experimentally characterized sequences. InterPro (http://www.ebi.ac.uk/interpro/) is the largest source for automatic annotation of sequences in the UniProt Knowledgebase (UniProtKB) [1]. InterPro consists of a consortium of 13 member databases: CATH-Gene3D [2], the Conserved Domains Database (CDD) [3], HAMAP [4], PANTHER [5], Pfam [6], PIRSF [7], PRINTS [8], PROSITE Patterns [9], PROSITE Profiles [9], SMART [10], the Structure–Function Linkage Database (SFLD) [11], SUPERFAMILY [12] and TIGRFAMs [13]. These member databases use different methodologies, such as profile hidden Markov models (HMMs) or regular expressions, in order to predict protein signatures. These different signatures are integrated to InterPro entries, containing one or more equivalent signatures. As, each InterPro member database has a different area of expertise, collectively they offer complementary levels of protein classification, reflected in the InterPro classification. A few member databases also offer amino acid residue-level annotation, including catalytic residues and those that are involved in ligand binding, to date two of them are available in InterPro: CDD and SFLD. InterPro also provides additional information about sequence features, such as consensus annotation of long-range intrinsic disorder (provided by MobiDB-lite [14]), and prediction of signal peptides, transmembrane regions and coiled-coils, via the SignalP, Phobius, TMHMM and Coils software packages [15-18]. InterPro is widely disseminated and utilised by the scientific community, and the database is recognised by ELIXIR as a core data resource [19].

InterPro releases are made available every two months through the InterPro website and download. To deal with the growing volume of sequence data and an increasing demand to retrieve subsets of the data, often via programmatic access, we have developed an entirely new website, released in 2019. It provides additional features and more flexibility in querying, presenting and retrieving data. InterPro can be searched through different ways including a protein sequence search, relying on the InterProScan software [20], a text search and a search by domain architecture. The website is based on an Application Programming Interface (API) which can also be utilised by users for direct access to the data. The API is designed around a Representational State Transfer (REST) framework, it offers six main endpoints, each corresponding to a key data type in InterPro: Entries, Proteins, Structures, Sets, Proteomes and Taxonomies. One of the new features available on the website is the Browse page: users can explore, search and filter the Entries, Proteins, Structures, Taxonomies, Proteomes and Sets data types. Another additional feature is the Download page, which allows users to select data types, apply filters and format as required. The website utilizes a series of web components to display different data types. The representations of protein sequences in the Protein pages, Structure pages and in the domain architectures section of the Entry pages use an extended version of ProtVista [21] to display sequence match positions. An adapted version of the LiteMol viewer [22] enables 3-dimensional (3D) visualization of entries and structures. A link between those two components has been made on the Structure pages to enable users to highlight regions on 3D representations of protein structures corresponding to the ProtVista linear representation of families and domains. These developments extend and enrich the information

provided by InterPro and provide unparalleled flexibility in terms of data access.

References

1. The UniProt Consortium UniProt: the universal protein knowledgebase. Nucleic Acids Res. 2017; 45:D158–D169.
2. Lewis T.E., Sillitoe I., Dawson N., Lam S.D., Clarke T., Lee D., Orengo C., Lees J. Gene3D: extensive prediction of globular domains in proteins. Nucleic Acids Res. 2018; 46:D435–D439.
3. Marchler-Bauer A., Bo Y., Han L., He J., Lanczycki C.J., Lu S., Chitsaz F., Derbyshire M.K., Geer R.C., Gonzales N.R. et al. CDD/SPARCLE: functional classification of proteins via subfamily domain architectures. Nucleic Acids Res. 2017; 45:D200–D203.
4. Pedruzzi I., Rivoire C., Auchincloss A.H., Coudert E., Keller G., de Castro E., Baratin D., Cuche B.A., Bougueleret L., Poux S. et al. HAMAP in 2015: updates to the protein family classification and annotation system. Nucleic Acids Res. 2015; 43:D1064–D1070.
5. Mi H., Huang X., Muruganujan A., Tang H., Mills C., Kang D., Thomas P.D. PANTHER version 11: expanded annotation data from Gene Ontology and Reactome pathways, and data analysis tool enhancements. Nucleic Acids Res. 2017; 45:D183–D189.
6. Finn R.D., Coggill P., Eberhardt R.Y., Eddy S.R., Mistry J., Mitchell A.L., Potter S.C., Punta M., Qureshi M., Sangrador-Vegas A. et al. The Pfam protein families database: towards a more sustainable future. Nucleic Acids Res. 2016; 44:D279–D285.
7. Nikolskaya A.N., Arighi C.N., Huang H., Barker W.C., Wu C.H. PIRSF family classification system for protein functional and evolutionary analysis. Evol. Bioinform. Online. 2007; 2:197–209.
8. Attwood T.K., Coletta A., Muirhead G., Pavlopoulou A., Philippou P.B., Popov I., Romá-Mateo C., Theodosiou A., Mitchell A.L. The PRINTS database: a fine-grained protein sequence annotation and analysis resource–its status in 2012. Database (Oxford). 2012; 2012:bas019.
9. Sigrist C.J.A., de Castro E., Cerutti L., Cuche B.A., Hulo N., Bridge A., Bougueleret L., Xenarios I. New and continuing developments at PROSITE. Nucleic Acids Res. 2013; 41:D344–D347.
10. Letunic I., Bork P. 20 years of the SMART protein domain annotation resource. Nucleic Acids Res. 2018; 46:D493–D496.
11. Akiva E., Brown S., Almonacid D.E., Barber A.E., Custer A.F., Hicks M.A., Huang C.C., Lauck F., Mashiyama S.T., Meng E.C. et al. The Structure-Function Linkage Database. Nucleic Acids Res. 2014; 42:D521–D530.
12. Oates M.E., Stahlhacke J., Vavoulis D.V., Smithers B., Rackham O.J.L., Sardar A.J., Zaucha J., Thurlby N., Fang H., Gough J. The SUPERFAMILY 1.75 database in 2014: a doubling of data. Nucleic Acids Res. 2015; 43:D227–D233.
13. Haft D.H., Selengut J.D., Richter R.A., Harkins D., Basu M.K., Beck E. TIGRFAMs and genome properties in 2013. Nucleic Acids Res. 2013; 41:D387–D395.
14. Piovesan D., Tabaro F., Paladin L., Necci M., Micetic I., Camilloni C., Davey N., Dosztányi Z., Mészáros B., Monzon A.M. et al. MobiDB 3.0: more annotations for intrinsic disorder, conformational diversity and interactions in proteins. Nucleic Acids Res. 2018; 46:D471–D476.
15. Nielsen H. Predicting secretory proteins with SignalP. Methods Mol. Biol. 2017; 1611:59–73.
16. Käll L., Krogh A., Sonnhammer E.L.L. Advantages of combined transmembrane topology and signal peptide prediction–the Phobius web server. Nucleic Acids Res. 2007; 35:W429–W432.
17. Krogh A., Larsson B., Heijne von, Sonnhammer E.L. Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. J. Mol. Biol. 2001; 305:567–580.
18. Lupas A., Van Dyke M., Stock J. Predicting coiled coils from protein sequences. Science. 1991; 252:1162–1164.
19. Durinx C., McEntyre J., Appel R., Apweiler R., Barlow M., Blomberg N., Cook C., Gasteiger E., Kim J.-H., Lopez R. et al. Identifying ELIXIR Core Data Resources. [version 2; referees: 2 approved]. F1000Res. 2017; 5:2422.
20. Jones P., Binns D., Chang H.-Y., Fraser M., Li W., McAnulla C., McWilliam H., Maslen J., Mitchell A., Nuka G. et al. InterProScan 5: genome-scale protein function classification. Bioinformatics. 2014; 30:1236–1240.
21. Watkins X., Garcia L.J., Pundir S., Martin M.J. The UniProt Consortium . ProtVista: visualization of protein sequence annotations. Bioinformatics. 2017; 33:2040–2041.
22. Pravda L., Sehnal D., Toušek D., Navrátilová V., Bazgier V., Berka K., Svobodová Vareková R., Koca J., Otyepka M. MOLEonline: a web-based tool for analyzing channels, tunnels and pores (2018 update). Nucleic Acids Res. 2018; 46:W368–W373.

Paper 228

Using the Human Cell Atlas for tracking SARS-CoV-2 entry factors

Christophe BÉCAVIN[1], Marie DEPREZ[1], Laure-Emmanuelle ZARAGOSI[1], Pascal BARBRY[1], AND HCA LUNG BIOLOGICAL NETWORK

[1] Université Côte d'Azur, CNRS, IPMC, Sophia-Antipolis 06560, France

Corresponding Author: becavin@ipmc.cnrs.fr

Paper Reference:

Sungnak et al. (2020), SARS-CoV-2 Entry Genes Are Most Highly Expressed in Nasal Goblet and Ciliated Cells within Human Airways, Nature Medecine, In Press

Deprez et al. (2020), A single-cell atlas of human healthy airways, Submitted (bioRxiv - https://doi.org/10.1101/2019.12.21.884759)

The SARS-CoV-2 coronavirus, the etiologic agent responsible for COVID-19 coronavirus disease, is a global threat. This pathogen propagates in the respiratory tract and can lead to acute respiratory distress. The airways are the first line of defense to this virus. Prior to COVID-19 pandemic, we characterized the respiratory tract unique cellular ecosystem by single-cell profiling methods, investigating the cell population distributions and transcriptional changes along the airways.

Analysis of the human airway epithelium in 10 healthy living volunteers by single-cell RNA profiling (https://www.genomique.info/cellbrowser/HCA/) was performed on 77,969 cells were collected at 35 distinct locations, from the nose to the 12^{th} division of the airway tree. We performed a deep quality control analysis, removing doublet cells and ambient RNA background. We integrated all samples together in one single dataset after appropriate batch removal. The resulting atlas is composed of a high percentage of epithelial cells (89.1%), but also immune (6.2%) and stromal (4.7%) cells with distinct cellular proportions in different regions of the airways. It reveals differential gene expression between identical cell types (suprabasal, secretory, and multiciliated cells) from the nose and tracheobronchial airways. By contrast, cell-type specific gene expression is stable across all tracheobronchial samples. Our atlas improves the description of rare cells like ionocytes [1], pulmonary neuro-endocrine (PNEC) and brush cells.

Our laboratory belongs to the Lung Biological Network of the Human Cell Atlas, the international consortium involved in the construction of a full atlas of the human cells. In a community effort, we used our datasets [2], [3] and a previously publish lung atlas [4] to better understand viral tropism of the SARS-CoV-2 (https://www.covid19cellatlas.org/). We assessed the RNA expression of the coronavirus receptor, *ACE2*, as well as the viral S protein priming protease *TMPRSS2* that governs viral entry [5], [6]. In-depth bioinformatic analysis of epithelial cells in the respiratory tree reveals that nasal goblet/secretory cells and multiciliated cells display the highest *ACE2* expression of all analyzed airway epithelial cells. We demonstrated that many of the top genes associated with *ACE2* airway epithelial expression are innate immune-associated, antiviral genes, highly enriched in the nasal epithelial cells. This suggests a particular relevant role for nasal goblet and ciliated cells as early viral targets and potential reservoirs of SARS-CoV-2 infection. We expect that our study will serve as a biological framework for dissecting viral transmission and developing clinical strategies for prevention and therapy.

[1] D. T. Montoro *et al.*, “A revised airway epithelial hierarchy includes CFTR-expressing ionocytes,” *Nature*, vol. 560, no. 7718, pp. 319–324, Aug. 2018.

[2] S. Ruiz García *et al.*, “Novel dynamics of human mucociliary differentiation revealed by single-cell RNA sequencing of nasal epithelial cultures,” *Development*, vol. 146, no. 20, p. dev.177428, Sep. 2019.

[3] M. Deprez *et al.*, “A single-cell atlas of the human healthy airways,” *bioRxiv*, 2019.

[4] F. A. Vieira Braga *et al.*, “A cellular census of human lungs identifies novel cell states in health and in asthma,” *Nat. Med.*

[5] P. Zhou *et al.*, “A pneumonia outbreak associated with a new coronavirus of probable bat origin,” *Nature*, vol. 579, no. 7798, pp. 270–273, Mar. 2020.

[6] M. Hoffmann, H. Kleine-Weber, S. Schroeder, M. A. Mü, C. Drosten, and S. Pö, “SARS-CoV-2 Cell Entry Depends on ACE2 and TMPRSS2 and Is Blocked by a Clinically Proven Protease Inhibitor,” *Cell*, vol. 181, pp. 271-280.e8, 2020.