Optimal Scaffolding for Chloroplasts' Inverted Repeats

Victor Epain, Rumen Andonov, Dominique Lavenier

Optimal Scaffolding for Chloroplasts' Inverted Repeats

Victor Epain, Rumen Andonov and Dominique Lavenier
Univ. Rennes, Inria, IRISA, CNRS, F-35000 Rennes, France

Corresponding author: victor.epain@irisa.fr

Abstract *Scaffolding step in the genome assembly aims to determine the order and the orientation of a huge number of previously assembled genomic fractions (contigs/scaffolds). Here we introduce a particular case of this problem and denote it by Nested Inverted Fragments Scaffolding (NIFS). We formulate it as an optimisation problem in a particular kind of directed graph that we call Multiplied Doubled Contigs Graph (MDCG). Furthermore, we prove that the NIFS problem is NP-Hard. We also discuss how the chloroplast data have been generated by filtering the reads sequenced both from plants and chloroplasts. Moreover, we propose a graph structure to visualise the solution and to highlight the particularity of chloroplast's regions structure.*

Keywords Genome assembly, Inverted repeats, De Bruijn graph, Assembly graph, NP-Hard

1 Introduction

Sequencing an organism is extracting DNA molecules contained in its cells and getting it as sequences of ATGC letters corresponding to nucleotides. Until now, sequencing technologies cannot return one complete sequence for each DNA molecule in cells but fragmented copies of them, called reads. Furthermore, sequencing DNA molecules from plants cells does sequence not only the DNA of the plant organism, but also the organelles living in them. This paper focuses on how to assemble chloroplast genome, a plant organelle responsible for photosynthesis metabolism, which confers the ability to transform sunlight energy to chemical one. The knowledge of chloroplast genomes allows evolutionary analyses [1], (meta)-barcoding [2], and is useful for biophotovoltaic process development [3].

Given a set of reads, assembling them needs comparisons between their sequences in order to detect if they overlap or not. The first difficulty for assembly methods is provoked by genomes' repeats: reads from one region can overlap reads from a repeat of it, while they do not correspond to same genomic location. Another difficulty is yielded by the fact that reads are sequenced from the two complemented DNA strands with no distinction. *Inverted repeats* (one region is the reverse-complement of the other one) can produce repeat-induced overlaps and so can lead to assemble a mix of two strands regions. One more specific issue here is how to separate reads from plant and the ones from the chloroplast genome. Indeed, partitioning them is not so trivial as some plant's reads

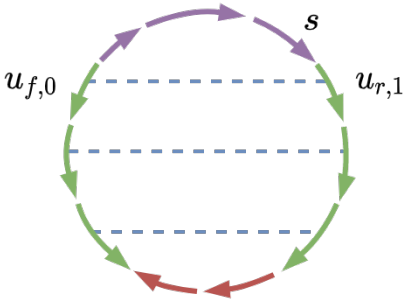

Fig. 1. Chloroplast genome's structure. Each arrow represents an oriented DNA sequence (contigs). Chloroplast's genome is circular thus it is a circular sequence of contigs, that begins at s and ends in s. Purple and red regions are unique, while green ones are *inverted repeats* (one is obtained by reversing and complementing the sequence of the other). For a given contig $u_{f,0}$ on the left-side green region, the contig in front of it (on the right-side green region) has the reverse orientation (so r). Because orientations are mutually exclusive, it is necessary to chose an occurrence that differs 0 (so 1). The couple $u_{f,0}, u_{r,1}$ is defined as inverted fragments. Each inverted fragments couple is linked by blue dashed line. Inverted repeats can be modelled as a sequence of nested inverted fragments (as illustrated with parallel blue dashed lines).

can overlap some chloroplast's reads. As a consequence, choosing repeat-induced or plant-chloroplast overlaps can lead to misassembled genome.

Dedicated chloroplast genomes assemblers have been already developed [4, 5] and have been compared in [6]. Exclusively based on the knowledge that chloroplast genome is circular and has two inverted repeats [7] (as illustrated in Fig. 1), we propose a dedicated scaffolding strategy that focuses on inverted repeats reconstruction [8]. This scaffolding approach is a linear programming formulation, which requires as input an assembly graph obtained from a pre-assembly step. Our method does not need any distance information between the reads and only uses genomic regions decomposition. Although our formulation is similar to the one proposed in [9] for RNA folding, here we deal with contigs instead of nucleotides and *a priori* without any knowledge concerning their order — which obviously increases the underlying challenges.

2 Method

Input data are reads that come from short reads sequencing technology, and can be part of plant's or/and chloroplasts' genomes. We also use chloroplast proteome data from a near plant species, and a well known gene sequence that is contained with a very high probability in one of the chloroplast's unique genomic regions, defined as the *seed gene*. An overview of the method is illustrated in Fig. 2. In the next section we briefly describe how the reads from the chloroplast's and the plant's genomes are separated, and how the remaining reads are assembled into contigs.

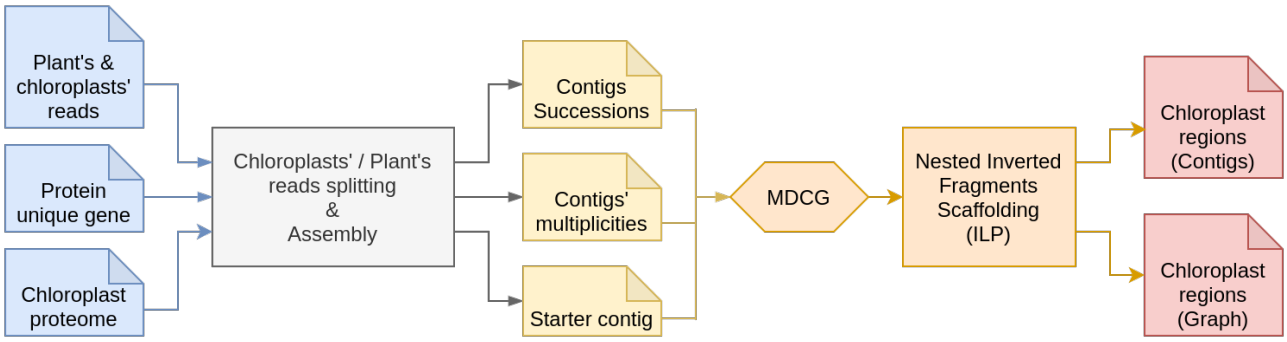

Fig. 2. Method overview. The method can be cut in two main parts. First step is a driven chloroplast assembly with reads that come from both plant's genome and chloroplasts' genome. This assembly is a De Bruijn graph assembly approach. It outputs contigs' successions and an estimated multiplicity for each contig. These results are structured in a Multiplied Doubled Contigs Graph (MDCG). We finish the assembly thanks to an Integer Linear Programming (ILP) approach. Finally, we output a fewer number of larger contigs that correspond to chloroplast's genomic regions, and a scaffolding enriched assembly graph, as illustrated Fig. 4.

2.1 Data Filtering & Contigs Generation

Under the hypothesis that chloroplasts' genome are over-represented comparing to the plant's genome in the plant's cell, only a subset of order of million randomly chosen reads is considered. Thus, reads are hashed into k-mers, and we keep reads with a high k-mer coverage. This implies that the assembly is done using exclusively k-mers with a high coverage presumably sequenced from chloroplasts' genome (and repeats in the plant).

Reads from this subset are assembled into contigs with a De Bruijn graph (DBG) approach using the tool Minia [10]. For each contig its *mean coverage* is computed according to the mean k-mer distribution over it. Then, the seed gene is mapped against contigs. The contig with the best alignment is defined as the *seed contig*, and its mean coverage becomes the *reference coverage*. Other contigs are mapped against translated chloroplast's proteome from a near plant's species thanks to tblastn tool [11]. Those which obtain a high alignment score are tagged as *contigs near to the proteome*.

Then we use MindTheGap [12] to find k-mers paths in the DBG from tagged contigs to other tagged ones. These paths are defined as *links*. These links can pass through a limited number of non-tagged contigs: in that case, these contigs are selected to be in the final contigs set.

We define the contigs set $\mathcal{C}$ as the set containing near-to-the-proteome tagged contigs and those which are intermediate in paths between two tagged ones. For each contig we compute a *multiplicity value* as an approximated ratio between contig's estimated coverage and seed contig's coverage (the reference coverage). The seed contigs is expected to participate only once into the final assembly. Thus, a contig's multiplicity may be interpreted as an upper-bound of the number of times the associated contig can be met in the final assembly. Contigs and their attributes are illustrated in Tab. 1a.

c_{id}	c_{len}	c_{mult}
2	18914	2
3	19212	1
4	88398	1
5	7596	2

(a) Contigs data

c_{id}	c_{or}	d_{id}	d_{or}
2	r	3	r
2	r	3	f
2	f	5	r
4	f	5	f

(b) Links data

Tab. 1. Example of scaffolding input data. It contains two types of data: **(a)** Set $\mathcal{C}$ of contigs with their length (c_{len}) and multiplicity (c_{mult}); **(b)** Set of links $\mathcal{L}$. An identifier (c_{id}) and an orientation (c_{or}) is provided for each contig c. For the sake of saving memory, for each link $(orc, ord) \in \mathcal{L}$, only one of the existing two links is reported in the table, the one with $c_{id} < d_{id}$.

If one contig is participating in the solution, its sequence is oriented: it can be either in its *forward* orientation (so the sequence does not change), or in its *reverse* orientation (the sequence is read in inverse reading and each nucleotide is complemented). For this reason links in the set $\mathcal{L}$ are succession relations between two oriented contigs. Tab. 1b shows links examples, where orientation attribute is indicated as 'f' (resp. 'r') for forward (resp. reverse) orientation.

2.2 Multiplied Doubled Contigs Graph

Building chloroplast genomic regions implies finding a sequence of linked oriented contigs, in the limit of multiplicity for the two orientations for each contig. Links between two multiplied (by their multiplicity) and oriented (forward or reverse) contigs can be structured in an oriented graph. Thus, we define $MDCG = (V, E)$ the *Multiplied Doubled Contigs Graph*, where V is the set of vertices and E is the set of edges. Even if in practice we do not multiply and double contigs and links data in memory, each vertex $v \in V$ is one occurrence of an oriented contig and each edge $(u, v) \in E$ corresponds to a link between occurrences of two oriented contigs. Let us show some MDCG properties:

- For each vertex $v \in V$, its reverse $\overline{v}$ is in the graph too. By definition, vertices' identifier remain the same $v_{id} = \overline{v}_{id}$, and same for occurrences $v_{occ} = \overline{v}_{occ}$.
- For each edge $e = (u, v) \in E$, its reverse $\overline{e} = (\overline{v}, \overline{u})$ is in the graph too.

Furthermore, we add what we define as *inverted fragments* data into MDCG. Each inverted fragments is an unoriented couple of vertices (i, j), such that they are two different occurrences of the same contig, but one is in forward orientation, while the other is in reverse orientation. These couples are candidates to be part of inverted repeats pairs. Fig. 1 shows an example where $i = u_{f,0}$ and $j = u_{r,1}$.

2.3 Nested Inverted Fragments Scaffolding

Thanks to the MDCG structure, building a sequence of linked oriented contigs is equivalent to finding an elementary path in MDCG. This is a necessary, but not a sufficient condition to correctly retrieve a circular genome with pairs of inverted repeats. Especially, no more that one from the vertices $v \in V$ and its reverse $\overline{v} \in V$ can be in the path. The seed contig is illustrated as a big black dot in Fig. 1 & 3 (seed vertex $s \in V$, in arbitrarily forward orientation). The circularity of the genome corresponds to a path that begins at s and ends in s.

Finding a path requires giving a position for each vertex participating in it. As illustrated in Fig. 1, two inverted fragments (i, j) and (k, l) in inverted repeats regions are nested *i.e.* if you draw a line from i to j, and another from k to l, then the two lines do not intersect. In order to know if two inverted fragments intersect, we must compare the positions of the associated vertices.

Finally, the goal of the Nested Inverted Fragments Scaffolding (NIFS) is to find a path from s to s, that passes through at most one of the orientations of multiplied contigs, and that maximises the number of nested inverted fragments. To solve this problem we give in [8] a linear programming formulation, such that vertices' position, their relative location, chosen edges and chosen inverted

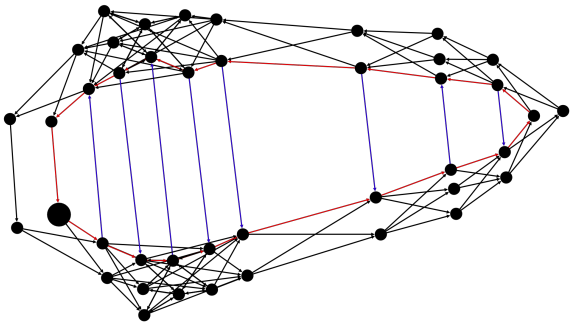

Fig. 3. An instance of *Multiplied Doubled Contigs Graph*. The input contigs were multiplied by their multiplicity number, then doubled according to two DNA strands. The obtained graph possesses 42 nodes and 130 edges. Nodes candidate to participate in inverted repeats have one of their reverse oriented versions linked by a blue edge. The solution (the assembled genome) is represented as a path in red. It passes through 8 adjacent inverted fragments that represent inverted repeats. It begins with the biggest node (a given starter, as illustrated in Fig. 1) and finishes in the same node since the chloroplast genomes are circular.

fragments are all integer or binary variables. We also demonstrate that the NIFS problem is NP-Hard. Moreover, the number of variables and constraints is polynomial according the vertices and edges number ($O(|V|^2 + |E|)$ for both).

3 Results

The mathematical formulation was implemented in Python3 using the PuLP package where we run Gurobi solver with academic licence. All the instances have been executed on a Linux laptop computer (32GB RAM, Intel® Core™ i7-10610U CPU @ 1.80GHz ×8).

Input data are contigs and links obtained as described in Section 2.1. When all links between contigs are provided, the computed solution enables to assemble the contigs and to output final sequences (fewer and longer contigs). As the genomes for these instances are known, it is possible to asses our solution. We used for this purpose Quast — a well known assembly evaluation tool [13]. All genome references are from NCBI database. Each instance has been run 10 times. Tab. 2 reports some solver run statistics and major Quast measures.

For each feasible instance, our scaffolding program outputs 3 contigs: two of them correspond to two unique regions known as long and short single copy (LSC & SSC), and the third one corresponds to one of the inverted repeats region (IRa or IRb). In addition, our tool outputs an assembly graph that visualises the connections between the three genomic regions. This assembly graph is enriched by reverse properties as illustrated in Fig. 4. Indeed, the graph in Fig. 4 shows that there are two possible circular sequences of genomic regions: $0 - 1_{f,0} - 2_f - 1_{r,1}$ and $0 - 1_{f,0} - 2_r - 1_{r,1}$. These results are not contradictory because chloroplast genomes are multimeric: two conforma-

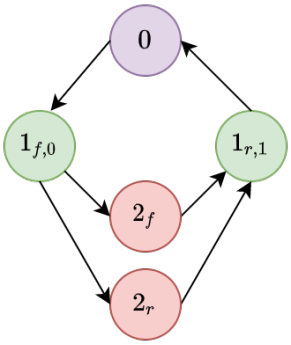

Fig. 4. Scaffolding final enriched assembly graph. Nodes correspond to genomic regions while edges indicate their neighbourhoods. Purple, green and red nodes correlate respectively with purple, green and red regions in Fig. 1. Two green nodes' label differ by their orientation 'f' and 'r' (one is the reverse complement sequence of the other). They also differ by their occurrence number (0 and 1). It simply means that their associated genomic regions are simultaneously present in the genome sequence (*i.e.* the green nodes both participate in the solution). On the opposite, red nodes just differ by their orientation: it implies that only one genomic region from one of these two nodes appears in the genome (one genomic region is the reverse complement of the other).

tions can simultaneously exist in the same plant's cell. In fact, one of the unique genomic region between the two inverted repeats (*e.g.* the red region in Fig. 1 & 4) exists in forward orientation in some genomes, and in reverse orientation in others. This multimeric property for chloroplast genomes has been studied in [14].

instance	**size**	$\|V\|$	$\|E\|$	**time**	**%gnm**	**NGA50**	**#mis**	**|mis|**	**mism**	**indels**
Aloysia citriodora	154699	42	74	.04	99.85	85509.0	0	0	3.88	10.36
Altheaea officinalis	159987	10	24	.01	99.91	87847.0	0	0	0.0	0.0
Amborella trichopoda	162686	106	168	.19	98.82	88782.0	0	0	162.35	36.08
Citrus limon	160101	174	358	.44	80.32	26819.0	2	102284	35.0	39.66
Coffea arabica	155188	134	222	.01	-	-	-	-	-	-
Dendrobium nobile	152018	36	54	.03	97.77	33841.0	2	105360	837.25	187.93
Digitalis lanata	153108	48	86	.08	83.15	83859.0	0	0	11.0	9.43

Tab. 2. Scaffolding results on chloroplast benchmark with reference genomes. Instance: species name; **size**: genome length in number of nucleotides (in base-pairs); $|V|$ and $|E|$ are respectively the number of vertices and edges; **time**: CPU solver time (in seconds); **%gnm**: percentage of genome length covered by the solution computed by our approach; **NGA50**: NG50 corrected of assembly errors; **#mis**: number of misassemblies in the proposed solution; **|mis|**: misassemblies total length (in bp); **mism**: number of alignment mismatches per 100 kbp; **indels**: number of alignment indels per 100 kbp.

Excluding three of the instances (Citrus limon, Dendrobium nobile and Coffea arabica) all other instances have been successfully solved. Citrus limon and Dendrobium nobile are misassembled, while Coffea arabica is not a feasible instance. Concerning the first two instances, further post-optimal analysis revealed two causes: unique regions are misassembled — this fact by itself is not a surprise since NIFS problem exclusively focuses on inverted repeat building — and/or some reads from the plant have not been filtered and were wrongly assembled into remaining contigs. On the other hand the Coffea arabica instance is unfeasible because some links are missing and/or multiplicities are under-estimated.

Quast's assessment indicated that Digitalis lanata solution is covering only 83.15% of the reference genome. However, further analysis highlighted that Quast was not able to map the contig corresponding to inverted repeat sequence to only one of the two reference genome's inverted repeats.

4 Conclusion & Discussion

In this study we show that the knowledge of the genomic regions structure in the case of chloroplasts is sufficient for the scaffolding assembly. We achieve this task by designing and implementing a linear programming tool that focuses on inverted repeats building in a circular genome context.

Although the problem is NP-Hard, our numerical experiments show that Gurobi solves real datasets extremely fast. Our first results evaluated by Quast are very encouraging even if they reveal some issues: the first assembly step is a highly heuristic-based approach and can suffer from unremoved plant's reads that remain in contigs set. Also, estimating the multiplicities remains a hard task.

Since chloroplast genomes are multimeric (regions between inverted repeats can be considered in their both orientations) this raises the question of their suitable representation. To answer this question

our strategy here is to explicitly reveal this behaviour using an enriched assembly graph where each region is one node and its associated sequence can participate in the final genome independently of the other nodes.

Acknowledgements

The authors are thankful to Abdelkader Ainouche for numerous discussions and for providing real plants and chloroplasts data.

References

[1] Zheng Xiao-Ming, Wang Junrui, Feng Li, Liu Sha, Pang Hongbo, Qi Lan, Li Jing, Sun Yan, Qiao Weihua, Zhang Lifang, Cheng Yunlian, and Yang Qingwen. Inferring the evolutionary mechanism of the chloroplast genome size by comparing whole-chloroplast genome sequences in seed plants. *Scientific Reports*, 7(1):1555, May 2017. Number: 1 Publisher: Nature Publishing Group.

[2] Natasha de Vere, Tim C. G. Rich, Sarah A. Trinder, and Charlotte Long. DNA Barcoding for Plants. In Jacqueline Batley, editor, *Plant Genotyping: Methods and Protocols*, Methods in Molecular Biology, pages 101–118. Springer, New York, NY, 2015.

[3] Jenny Tschörtner, Bin Lai, and Jens O. Krömer. Biophotovoltaics: Green Power Generation From Sunlight and Water. *Frontiers in Microbiology*, 10, 2019.

[4] Nicolas Dierckxsens, Patrick Mardulyn, and Guillaume Smits. NOVOPlasty: de novo assembly of organelle genomes from whole genome data. *Nucleic Acids Research*, 45(4):e18, February 2017.

[5] Jian-Jun Jin, Wen-Bin Yu, Jun-Bo Yang, Yu Song, Claude W. dePamphilis, Ting-Shuang Yi, and De-Zhu Li. GetOrganelle: a fast and versatile toolkit for accurate de novo assembly of organelle genomes. *Genome Biology*, 21(1):241, September 2020.

[6] Jan A. Freudenthal, Simon Pfaff, Niklas Terhoeven, Arthur Korte, Markus J. Ankenbrand, and Frank Förster. A systematic comparison of chloroplast genome assembly tools. *Genome Biology*, 21:254, September 2020.

[7] Ralph Bock and Volker Knoop, editors. *Genomics of Chloroplasts and Mitochondria*, volume 35 of *Advances in Photosynthesis and Respiration*. Springer Netherlands, Dordrecht, 2012.

[8] Victor Epain and Rumen Andonov. Integer Programming Approach for Nested Pairs Genome Scaffolding, March 2022.

[9] Dan Gusfield. The RNA-Folding Problem. In Dan Gusfield, editor, *Integer Linear Programming in Computational and Systems Biology: An Entry-Level Text and Course*, pages 105–121. Cambridge University Press, Cambridge, 2019.

[10] Rayan Chikhi and Guillaume Rizk. Space-Efficient and Exact de Bruijn Graph Representation Based on a Bloom Filter. In Ben Raphael and Jijun Tang, editors, *Algorithms in Bioinformatics*, Lecture Notes in Computer Science, pages 236–248, Berlin, Heidelberg, 2012. Springer.

[11] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, October 1990.

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

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

[14] Xing-Wang Deng, Rod A. Wing, and Wilhelm Gruissem. The chloroplast genome exists in multimeric forms. *Proceedings of the National Academy of Sciences*, 86(11):4156–4160, June 1989. Publisher: National Academy of Sciences Section: Biological Sciences: Genetics.