Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Genome analysis

HyAsP, a hybrid assembler for plasmids

Robert Müller[1] **and Cedric Chauve**[2]

[1]Computational Methods for the Analysis of the Diversity and Dynamics of Genomes, Bielefeld University, 33615 Bielefeld, Germany;
[2]Department of Mathematics, Simon Fraser University, Vancouver BC V5A 1S6, Canada

Abstract

Motivation: Plasmids are ubiquituous in bacterial genomes, and have been shown to be involved in important evolutionary processes, in particular the acquisition of antimicrobial resistance. However separating chromosomal contigs from plasmid contigs and assembling the later is a challenging problem.
Results: We introduce HyAsP, a tool that identifies and assembles plasmid contigs following a hybrid approach based on a database of known plasmids and a greedy assembly algorithm. We test HyAsP on a large sample of bacterial data sets and observe that it generally outperforms other tools.
Availability: https://github.com/cchauve/HyAsP
Contact: cedric.chauve@sfu.ca

1 Introduction

Plasmids are extra-chromosomal DNA molecules common in Bacteria. Plasmids differ from chromosomes in various features, such as their length – they tend to be much shorter than chromosomes –, copy number – plasmids can be present in multiple copies in a cell – and GC content. They play an important part in horizontal gene transfer and, thus, in the transmission of virulence factors and antibiotic resistance (Dolejska and Papagiannitsis, 2018; Carattoli, 2013). Therefore, the effective identification of plasmids from bacterial samples is important toward mitigation strategies against the proliferation of drug-resistant bacteria.

Various approaches have been explored for the detection of plasmids, with a recent focus on methods using whole-genome sequencing (WGS) data, as WGS is now a standard approach in microbial genomics, including in a clinical context. PLACNET (Lanza *et al.*, 2014) uses information from the assembly (e.g. scaffold links and read depth), reference sequences and plasmid-diagnostic sequence features, but also needs a subsequent expert analysis. Recycler (Rozov *et al.*, 2017) and plasmidSPAdes (Antipov *et al.*, 2016) do not depend on reference sequences and are fully unsupervised. Recycler predicts plasmids by repeatedly peeling off cycles of the assembly graph based on read-depth and length features. plasmidSPAdes assumes that the read depth of plasmids differs from the chromosome, estimates the chromosomal read depth, removes those contigs that are presumably of chromosomal origin and predicts plasmids from the connected components of this reduced assembly graph. The recent reference-based tool MOB-recon (Robertson and Nash, 2018) uses a database of reference plasmids and collections of known replicons and relaxases. Contigs of an assembly are mapped against the reference database and grouped into putative plasmid units, that are further refined by discarding those units without a replicon or relaxase. We refer to (Orlek *et al.*, 2017; Arredondo-Alonso *et al.*, 2017; Laczny *et al.*, 2017) for recent reviews on plasmids detection and assembly tools.

We present HyAsP, a novel tool for extracting plasmids from WGS assemblies in a fully automatic way. It combines ideas from both reference-based and de-novo methods to identify plasmids using information on the occurrences of known plasmid genes and considering characteristics such as read depth and GC content. We compare the prediction quality of HyAsP with plasmidSPAdes and MOB-recon on a data set comprising 66 genomes and show that HyAsP generally outperformes competing tools.

2 Methods

HyAsP is a greedy algorithm for the reconstruction of plasmids from an assembly graph using information from known plasmid genes, read depth and GC content. It combines idea of reference-based methods such as MOB-recon and assembly-based methods such as plasmidSPAdes. We provide below a high level description of the algorithm and experiments, while all technical details are provided in Supplementary Material.

Identifying seeds. HyAsP starts from an assembly graph – obtained by Unicycler (Wick *et al.*, 2017) by default – and first identifies *seeds*, defined as contigs which contain at least one plasmid gene (according to a database of known plasmid genes). Seeds are used to identify parts of the assembly graph that serve as starting points to construct chains of contigs potentially defining plasmid fragments. Dubious seeds, of potential chromosomal origin, are filtered out if they do not satisfy some *eligibility* criteria defined in terms of plasmid gene density, read depth and length.

Greedy extension. Next, seeds are enumerated based on their plasmid gene density and GC content, preferring those with a high gene density and a GC content differing from the average GC content of the assembly graph, and each seed is extended into a contig chain following a greedy approach. To extend the current contig chain, its two endpoints are searched for *eligible extensions*, using eligibility criteria aimed at averting likely errors by, e.g., limiting the length of the plasmid and avoiding overly long gene-free stretches as well as fluctuations in GC content that are too high. The eligible extensions are scored based on these features and the extension with the best score is chosen.

Updating read depth. Once all eligible extensions have been performed, the resulting contig chain is considered as a potential plasmid fragment; it is then checked whether it can be circularised, if its first and last contigs overlap sufficiently. The construction of a plasmid concludes by determining a final read-depth value for each contig it contains. To this end, the average read depth (by default the mean) is computed and a contig is assigned the minimum of this average and its current read depth in the assembly graph. If a contig has a larger reads depth than the one it is assigned in the current plasmid, it is kept in the assembly graph for being potentially used by other plasmids, after having decreased its depth by the value assigned to it, thus allowing contigs to occur in several plasmids.

Postprocessing. Once all plasmids are constructed, those which do not meet some quality requirements, determined from a large collection of known plasmids, are marked as *questionable* and separated from the other ones, called *putative* plasmids. Finally, the (putative) non-circular plasmids are grouped into bins, where plasmid fragments are binned together based on read depth and GC content.

Integrated pipeline. We incorporated the HyAsP algorithm into a pipeline accepting FASTQ files and a collection of known plasmid genes as input. The reads are assembled using Unicycler (Wick *et al.*, 2017) and, subsequently, the genes are mapped to the assembly contigs using BLAST. Finally, the plasmids are predicted in the assembly graph using HyAsP as described above. HyAsP is a single-threaded program written completely in Python (requiring Biopython, NumPy and pandas) with system calls to BLAST (blastn, makeblastdb). After downloading HyAsP, it can be used directly or installed as a package through pip.

Experimental evaluation. We evaluated HyAsP on a collection of real plasmids compiled in (Robertson and Nash, 2018). We compared HyAsP with plasmidSPAdes and MOB-recon, that represent the two approaches (de-novo and reference-based) that are combined in our tool, and performed best in previous benchmarks.

The data set consists of 133 bacterial samples comprising 377 plasmids. To simulate the use of plasmids identification tools on a newly WGS dataset, using only knowledge from previously existing plasmid data, the data set was split in half based on the release date of the read data. The 66 samples released on 19 December 2015 or later formed the *test samples*, spanning 147 plasmids from 8 different species (the *ground truth plasmids*). The other 67 samples were used as the *database samples* and we refer to the corresponding 230 plasmids as the *MOB-database*. Similarly, we created a second set of references by collecting all plasmids from NCBI released before 19 December 2015 (the *NCBI-database*).

We assessed the predictions of all tools in terms of their *precision* (proportion of predicted plasmids corresponding to expected references) and *recall* (proportion of expected reference plasmids corresponding to predictions) by mapping the putative plasmids (HyAsP) resp. plasmid contigs (MOB-recon, plasmidSPAdes) against the ground truth plasmids using BLAST. Precision and recall were then summarized with the *F1 score* per test sample. In addition, the total precision, recall and F1 score over all test samples (per species) were determined by summing over the predicted plasmids of all test samples (per species).

3 Results

As shown in Table 1, HyAsP outperformed plasmidSPAdes and MOB-recon in both total precision and recall. Consequently, the F1 score of HyAsP was notably higher as well. In addition, plasmidSPAdes and MOB-recon showed a notable trade-off between precision and recall.

	Tool	Precision	Recall	F1 score
NCBI	HyAsP	0.871445	0.898822	0.884922
	plasmidSPAdes	0.659211	0.741983	0.698152
	MOB-recon	0.760241	0.583909	0.660509
MOB	HyAsP	0.934515	0.775169	0.847416
	plasmidSPAdes	0.659211	0.741983	0.698152
	MOB-recon	0.760241	0.583909	0.660509

Table 1. Precision, recall and F1 score of HyAsP, plasmidSPAdes and MOB-recon, over the 66 test samples using the NCBI-database and MOB-database.

We also observed a trade-off in precision and recall between the MOB-database and the NCBI-database: HyAsP attained a higher precision (0.934515) and lower recall (0.775169) using the less extensive MOB-database, which led to a similar but slightly lower F1 score (0.847416).

However, note that we excluded from the reference sequences three *Salmonella enterica* plasmids from the NCBI-database, which exhibited unusually chromosome-like characteristics. These plasmids (resp. their genes) caused a severe drop in precision on test samples from *Salmonella enterica* but had hardly any effect on other test samples. We refer to Section 2.1 of Supplementary Material for more information on this issue.

Subsequently, we analyzed the predictions after grouping the samples based on the associated organism at the species level. HyAsP performed similarly well or better than plasmidSPAdes and MOB-recon for the majority of species. Only for *Klebsiella aerogenes*, did it fall notably behind plasmidSPAdes but still outperformed MOB-recon. While there are species, for which the other tools achieved a higher recall or precision, the trade-off between both is usually smaller for HyAsP leading to a better overall prediction quality (in terms of the F1 score).

A more extensive set of results is available in Supplementary Material.

References

Antipov, D. *et al.* (2016). plasmidSPAdes: assembling plasmids from whole genome sequencing data. *Bioinformatics*, **32**(22), 3380–3387.

Arredondo-Alonso, S. *et al.* (2017). On the (im)possibility of reconstructing plasmids from whole-genome short-read sequencing data. *Microb Genom*, **3**(10). doi: 10.1099/mgen.0.000128.

Carattoli, A. (2013). Plasmids and the spread of resistance. *Int J Med Microbiol*, **303**(6-7), 298–304.

Dolejska, M. and Papagiannitsis, C. C. (2018). Plasmid-mediated resistance is going wild. *Plasmids*. doi: j.plasmid.2018.09.010.

Laczny, C. C. *et al.* (2017). Assessing the heterogeneity of in silico plasmid predictions based on whole-genome-sequenced clinical isolates. *Brief Bioinform*. doi: 10.1093/bib/bbx162.

Lanza, V. F. *et al.* (2014). Plasmid Flux in Escherichia coli ST131 Sublineages, Analyzed by Plasmid Constellation Network (PLACNET), a New Method for Plasmid Reconstruction from Whole Genome Sequences. *PLOS Genet*, **10**(12), 1–21.

Orlek, A. *et al.* (2017). Plasmid Classification in an Era of Whole-Genome Sequencing: Application in Studies of Antibiotic Resistance Epidemiology. *Front Microbiol*, **8**, 182.

Robertson, J. and Nash, J. H. E. (2018). MOB-suite: software tools for clustering, reconstruction and typing of plasmids from draft assemblies. *Microb Genom*, **4**(8). doi: 10.1099/mgen.0.000206.

Rozov, R. *et al.* (2017). Recycler: an algorithm for detecting plasmids from de novo assembly graphs. *Bioinformatics*, **33**(4), 475–482.

Wick, R. R. *et al.* (2017). Unicycler: Resolving bacterial genome assemblies from short and long sequencing reads. *PLOS Comput Biol*, **13**(6), 1–22.