Identifying Conjugative Plasmids and Integrative Conjugative Elements with CONJscan

Jean Cury, Sophie Abby, Olivia Doppelt-Azeroual, Bertrand Néron, Eduardo Rocha

Identifying conjugative plasmids and integrative conjugative elements with CONJscan

Jean Cury[1,2], Sophie Abby[3], Olivia Doppelt-Azeroual[4,5], Bertrand Néron[4,5], Eduardo P. C. Rocha[1,2]

[1] Microbial Evolutionary Genomics, Institut Pasteur, 28, rue Dr Roux, Paris, 75015, France

[2] CNRS, UMR3525, 28, rue Dr Roux, Paris, 75015, France

[3]Univ. Grenoble Alpes, Laboratoire Techniques de l'Ingénierie Médicale et de la Complexité - Informatique, Mathématiques et Applications, Grenoble (TIMC-IMAG), F-38000 Grenoble, France; Centre National de la Recherche Scientifique (CNRS), TIMC-IMAG, F-38000 Grenoble, France.

[4] Bioinformatics and Biostatistics Hub – C3BI, Institut Pasteur, 28, rue Dr Roux, Paris, 75015, France

[5]CNRS, USR 3756, 28, rue Dr Roux, Paris, 75015, France

Running head: Identification of conjugative elements.

Key Words: MacSyFinder, conjugation, integrative conjugative element, plasmid, protein profiles, comparative genomics, integrase, genomic islands.

1 Summary

We present a computational method to identify conjugative systems in plasmids and chromosomes using the CONJscan module of MacSyFinder. The method relies on the identification of the protein components of the system using hidden markov model profiles and then checking that the composition and genetic organization of the system is consistent with that expected from a conjugative system. The method can be assessed online using the Galaxy workflow or locally using a standalone software. The latter version allows to modify the models of the module, i.e., to change the expected components, their number, and their organization.

CONJscan identifies conjugative systems, but when the mobile genetic element is integrative (ICE), one often also wants to delimit it from the chromosome. We present a method, with a script, to use the results of CONJscan and comparative genomics to delimit ICE in chromosomes. The method provides a visual representation of the ICE location. Together, these methods facilitate the identification of conjugative elements in bacterial genomes.

2 Introduction

Conjugative elements transfer large amounts of genetic information between cells, having an important evolutionary role in bacterial evolution. There are several different types of conjugation. For example, some *Actinobacteria* are able to conjugate dsDNA, while others make distributive conjugation [1]. These alternative mechanisms will not be discussed here. Instead, we will focus on the most common mechanism of conjugation: the transfer of ssDNA through a type IV secretion system, or an analogous structure (reviewed in [2,3]). This mechanism involves three key molecular systems: the relaxosome, the type 4 coupling protein (T4CP), and the type IV secretion system (T4SS). The relaxase, often associated with other proteins in the relaxosome, interacts with the mobile genetic element (MGE) at the origin of transfer, produces a single-stranded cut, and becomes covalently linked with it. The T4CP is an ATPase that couples the nucleoprotein filament including the ssDNA and the relaxase with the T4SS. Finally, the T4SS is a large protein complex that spans both cell membranes (in diderms) and is able to deliver the nucleoprotein filament into another cell. The identification of a conjugative system requires the identification of these key

components, or at least the most conserved ones: the relaxase, the T4CP and VirB4, the only ubiquitous ATPase of the T4SS. Both relaxases and VirB4 can be divided in a number of families that have been used to type plasmids and to establish the resemblance between T4SS [4,5]. It is important to note that different combination of these three components can have very different functions. MGEs with a relaxase and a T4CP are mobilizable by conjugative systems but are not a conjugative system. Replicons with a T4SS and a T4CP, but devoid of relaxases, may be protein secretion systems or other co-options of the conjugation machinery that are not involved in conjugation.

The analysis of large-scale genome data requires reliable and flexible computational tools to identify and class conjugative elements. Ideally, these tools should allow the user to modify the number and type of components and their genetic organization. MacSyFinder was created with this goal in mind [6]. It can be used with predefined modules (set of protein profiles and definitions), but it can also be customized to meet the researcher needs. The program uses information on the presence and absence of a number of components and their genetic organization to identify systems matching these specifications in genome data. This makes it as simple to use as classical approaches based on blast searches, but allows more powerful queries. It is also more sensitive because it uses protein profiles for similarity search, as well as information on the presence and organization of the different components of the system.

The accuracy of CONJscan depends on its ability to identify the components of the system. When these are highly conserved, *e.g.*, the ATPase VirB4, they can be identified with high sensitivity. When they evolve fast (high sequence divergence) or are facultative components, like the lipoprotein VirB7, the task becomes more complicated. To allow for some flexibility, MacSyFinder searches for components that are expected to be almost always present ("mandatory" components) and those that may be either absent or non-identifiable (“accessory"). Some components can also be described as "forbidden" in which case they cannot occur in the conjugative system. This is useful to distinguish conjugative systems from other co-opted molecular systems (Note 5.1). For each type of components one can set up a minimal quorum. Decreasing the minimal quorum allows to search for more degenerate or distant systems, whereas the inverse only identifies systems closer to the prototypical system.

An important variable in MacSyFinder is the distance allowed between components. While the T4SS is usually encoded in one operon or a set of contiguous operons (contiguity formalized by an authorized inter-genic distance of 30 genes), the relaxase is often encoded apart (sometimes with the T4CP). Hence, the inter-genic distance between the relaxase and any other elements is increased to 60 genes. In total, three types of information facilitate the detection of conjugative systems: the identification of the components (and their type), the completeness of the set of components expected to form a full system, and their proximity in the genome (genetic organization). This information can be put together in a text file following a certain grammar that constitutes the *model* of the system that is given to MacSyFinder, which searches genomes for instances satisfying these descriptions [6].

The first part of this text shows how one can use MacSyFinder to identify conjugation systems with the pre-defined models of CONJscan. These models have been validated and shown to identify the vast majority of known systems [7,8]. Yet, they may be inadequate in certain specific situations. For example, the conjugative systems of a number of taxa (like Archaea, Actinobacteria, and Firmicutes) lack known relaxases [2,9]. In this case, the models can be adapted to identify novel components (or to accept the absence of the component), and then easily shared with others *via* text files. The second part of this text describes how the analysis of CONJscan can be complemented with a comparative genomics method to delimit ICEs in genomes. This uses the core-genome (*i.e.*, the list of gene families present in all the genomes available for the species), the genome annotations, the CONJscan results, and a script that we provide to plot all this together.

3 Materials

MacSyFinder reads a *model* and works in two steps: it uses Hidden Markov model (HMM) protein profiles to search for the system's components and then checks if their organization and quorum respects the specifications of the model (Figure 1).

3.1 Sequence data

Proteome data. MacSyFinder analyzes protein sequences stored in one single file in Fasta format. The search for conjugative elements usually requires a completely assembled replicon (or a known order of contigs, see Note 5.2). Hence, the file for the analysis should contain all the proteins encoded in the genome (or the replicon of interest) in the order of

their genomic position. The corresponding option for this file type is "--ordered_replicon".

The analysis of multiple genomes in one single batch is possible using the type "--gembase", which is similar to the "--ordered_replicon" but requires special sequence identifiers (see MacSyFinder's documentation).

Protein profiles. The protein profiles used by CONJscan are included in the package (https://github.com/gem-pasteur/Macsyfinder_models). They are described in [7]. The protein profiles for integrases can be retrieved from PFAM For the Tyrosine recombinase: PF00589 (one single profile). For the Serine recombinase: PF07508 and PF00239 (the protein should hit both profiles to be regarded as an integrase)).

3.2 Pre-defined models available in CONJscan

CONJscan is a MacSyFinder module that includes a set of pre-defined models and profiles to detect the eight types of ssDNA conjugation systems. These models are used as examples throughout the following sections. The standalone version of MacSyFinder expects to find the files of CONJScan (HMM profiles and model files) at a recognizable location (a folder for the HMM profiles and a folder for the models). Currently, only the standalone version allows the modification of the models and the introduction of novel protein profiles.

3.3 Software and availability

We listed resources of interest for this protocol in Table 2 (see Note 5.3 for issues related with installing the programs).

To run MacSyFinder one needs to install the NCBI/BLAST tools (in particular makeblastdb version ≥ 2.8, or formatdb), HMMER, and MacSyFinder [10,11,6]. The latter requires a Python interpreter (version 2.7) that must be installed beforehand. See MacSyFinder's online documentation for more details: http://macsyfinder.readthedocs.org/en/latest/installation.html.

To build HMM protein profiles, one also needs a program to make multiple sequence alignments (*e.g.,* MAFFT [12]), an alignment editor (*e.g.*, Seaview [13]), and a program to cluster proteins by sequence similarity (*e.g.*, Silix or usearch [14,15]). These programs are also required to build the pan-genomes.

CONJScan can be downloaded for local use with MacSyFinder (https://github.com/gem-pasteur/Macsyfinder_models) or it can be used online (http://galaxy.pasteur.fr/, search CONJScan or the direct link https://galaxy.pasteur.fr/tool_runner?tool_id=toolshed.pasteur.fr%2Frepos%2Fodoppelt%2Fconjscan%2FConjScan%2F1.0.2). Alternatively, one can query a database of conjugative systems already detected (http://conjdb.web.pasteur.fr).

The program MacSyView can be used to visualize the results of MacSyFinder [6]. It can be used locally (https://github.com/gem-pasteur/macsyview) or online (http://macsyview.web.pasteur.fr).

The script to plot the spots (region between two consecutive core genes) with ICEs can be downloaded for local use (https://gitlab.pasteur.fr/gem/spot_ICE).

4 Methods

The procedure to annotate conjugative elements contains two main steps. In the first step we show how to identify conjugative systems using CONJscan. In the second step we show how to delimit the conjugative element. For the use of the standalone version, we assume some familiarity with a Unix environment (Linux or Mac OS X).

4.1 Identifying conjugative systems

The identification of conjugative systems in a replicon relies on the CONJScan module for MacSyFinder.

4.1.1 Preparing the data

The protein file should be in multi fasta format (a succession of fasta entries in a text file) and must be in a directory where the user has permissions to write.

The models and the protein profiles must be in two different folders, typically called ”DEF” and “HMM”, respectively. The files with the protein profiles must have the same extension. The easiest is to download (or clone) the CONJScan module from the link provided above; where a folder called "Conjugation" has two other folders for the definitions of the models and for the profiles.

4.1.2 Running MacSyFinder

The standalone version of MacSyFinder requires a unix-like terminal. In the terminal, MacSyFinder can be started with a command line. For example, to detect a conjugative system of type F using the default model, one should type:

```
macsyfinder typeF \
            --db-type ordered_replicon \
            -d Conjugation/DEF \
            -p Conjugation/HMM \
            --profile-suffix .HMM \
            --sequence-db my_sequence.prt \
            -o my_sequence_typeF
```

Here, all the paths of the filenames are relative, meaning that MacSyFinder will look for the presence of the files starting from the folder where the command is executed. This can be changed by providing absolute paths. The meaning of the options is the following:

`--db-type` is a mandatory parameter that specifies whether the proteins in the multi-fasta file are sorted as they appear along the replicon (for drafts or metagenomes see Note 5.2).

`-d` the path to the definitions of the model.

`-p` the path to the set of protein profiles (`--profile-suffix` is the suffix of these files).

`--sequence-db` sets the fasta file with the protein sequences.

`-o` option specifies the name of the output folder. The default name contains the date and time of the command execution. It is advisable to provide meaningful names for these folders [16]. The standard output of the program is saved automatically in the file `macsyfinder.out`, in the output folder `my_sequence_typeF`.

In the previous example, we searched for only one of the eight types of MPF available. Figure 2 describes the different systems in terms of components and genetic organization. If the user wishes to run all the models, we advise to make a loop over each definition with the previous command line (see Note 5.4 for other possibilities). A `bash` command to do this follows:

```
for conj_type in typeF typeB typeC typeFATA typeFA typeG typeI
typeT; do
    macsyfinder "$conj_type" \
                --db-type ordered_replicon \
                -d Conjugation/DEF \
```

```
                        -p Conjugation/HMM \
                        --profile-suffix .HMM \
                        --sequence-db Data/plasmid_seq.prt \
                        -o plasmid\_seq\_"$conj_type"
    done
```

4.1.3 Analyzing the results

Table 1 presents the different output files with their description. The main output file is located in the folder `my_sequence_typeF/` in the previous example. It is named `macsyfinder.report`. It is a tabular file where each line corresponds to a protein identified as a component of a conjugative system, with its annotation and some results of the detection (including the hmmer i-evalue and the alignment coverage with the profile). It contains the predicted system (here `typeF`) (see Note 5.5 for how to class systems). This file is empty when there is no occurrence of a conjugation system that satisfies the model. If in spite of the negative result, one wishes to analyze the presence of proteins that might be components of a conjugation system, which might reveal an atypical or degraded system, this information can be found in the `macsyfinder.out` file. More specifically, this file contains the information on proteins that have matched certain protein profiles of the model and whether they formed a complete system (in which case this is reported in the `macsyfinder.report` file). If the analysis of these results suggests the existence of an atypical yet relevant system, the user can modify the model to account for such cases and re-run MacSyFinder with the novel model (see Note 5.6).

4.1.4 Running MacSyFinder with Galaxy

CONJScan was integrated on the public Galaxy@pasteur instance available at "https://galaxy.pasteur.fr". It is classified in the "genome annotation" category. Any user can connect to Galaxy (anonymously or with an account) and launch an execution of CONJScan. The functioning, input, and output of CONJScan in the Galaxy@pasteur instance is similar to the standalone version. The only differences between the two instances concern expert options and the ability to change the models, which are only available for the standalone version.

Before selecting a genome to scan, one must upload the data by opening the dialog box (highlighted in green on Figure 3.A). Then this dataset can be selected in the first parameter of the form. The option "Type of dataset to deal with" must be set as in the standalone, typically "ordered replicon" or "gembase" to analyze completely assembled replicons. The option

"Conjugative element to detect" allows to select one model from a precompiled set of models used by MacSyFinder. When one is not sure of which model to use, one can run the process consecutively, changing the model at each time.

Expert users can access and change the hmm search parameters by clicking on the select button under the Hmmer code option label. If so, the options, "*Maximal e-value*", "*Maximal independent e-value*" and "*Minimal profile coverage*" can be tuned before the execution (similar options are available in the standalone version). The two former options are specific to HMMER (see Table 2) and the latter represents the minimal accepted value for the fraction of the profile that is matched in the alignment with the protein sequence.

Once the options are set, the user needs to click on execute to launch the process. The user history (the panel on the right) will then be updated with information on the process and the associated files. New files will appear in grey when the job is waiting to be ran on the Institut Pasteur's cluster, in yellow when the process is running (Figure 3.B), and in green when it has correctly terminated (clicking the icon circled in red will update the job status).

Among the five outputs of CONJScan (Table 1), the "MacSyView output (CONJScan on data 1)" allows to visualize the results (Figure 3.C). Clicking on the eye on this file will display the link "Display in MacSyView" and clicking this link will open the MacSyView web application in a new tab of the web browser, automatically filled with the results of CONJScan. The user can then browse graphically these results.

The first page of MacSyView displays all occurrences of the systems found on the input data (Figure 4). The user can pick an instance to visualize it by clicking on it in the list. The page displaying the instance is divided in three parts. The first panel shows how the instance fits the model in terms of the components of the system. Boxes represent the number of each *mandatory*, *accessory*, and *forbidden* components. A tooltip gives the name of the component when the mouse hovers a box. The second panel shows the genetic context of the system (as transcribed from the input fasta file), with components drawn to scale. When the mouse hovers a box (circled in red), a tooltip displays information on the corresponding component, including the scores of the HMMER hit. This view can be exported as a SVG file (tools in blue at panel bottom). The third panel gives detailed information on the components of the system.

4.2 Delimiting an ICE

After the detection of a conjugative system, one is often interested in delimiting the associated mobile genetic element. If the element is a plasmid, then the delimitation is trivial (it's the replicon). However, if the element is integrated in a chromosome (ICE), one needs to delimit the ICE within the replicon. For this, we have developed a method using multiple genomes (usually more than four) of the same species. This is described in the following paragraphs.

4.2.1 Building core-genomes

The first step of the analysis consists in building the core genome of the species with the ICE. The core genome is the set of genes that are shared by all the genomes of the species, and can be built rapidly using the program Roary [17]. This requires the availability of `gff3` files for each genome in the same directory, a file format containing both annotations (list of genes) and the nucleotide sequences. These files can be downloaded from GenBank if the genome sequences are available there. They can also be generated by genome-annotation tools like Prokka [18]. The commands to obtain the core genome are as follow:

```
roary GFF/*.gff
```

Roary creates many output files whose description is not in the scope of this chapter. To obtain the information on the core genome one can type:

```
query_pan_genome -g 1493304436/clustered_proteins \
                 -a intersection \
                 /GFF/*.gff
```

Which will create a file called `pan_genome_result` containing the core genes identifiers. The script `query_pan_genome` is installed with Roary. The `-g` option takes the output of the previous command. The `-a intersection` option indicates the script to build the core-genome.

4.2.2 Defining the spot

Initially, one does not know the location of the ICE, except that its upper boundaries are the two flanking core genes in the genome (we assume here that the ICE is not part of the core genome). We define an interval as the genomic region between the two core genes. We define a spot as the set of intervals flanked by the same two families of core genes across genomes. Since there is only one member of a core gene per genome, the spot has one

interval in each genome at most. If there are rearrangements in this region there may not be an interval with the same two core gene families in certain genomes. The latter are not part of the spot and should be excluded from further analysis. The goal of the method it to focus on the interval with the ICE, while accounting for the gene repertoires of the spot, to delimit the ICE. We recommend to restrict such analyses to cases with a minimum of four genomes in the species (the fewer genomes, the weaker the statistical signal).

4.2.3 Delimiting the ICE within the spot

Once the spot with the conjugative system is defined, one needs to build the pan-genome of the genes in the spot. The pan-genome is the full set of protein families that are present in a given set of genomes (here in the set of proteins in the spot). To identify the pan-genome, one could re-use the pan-genome build earlier by Roary (built when constructing the core genome). However, if that step was skipped because the core-genome was already built independently, one can use usearch [15], a program that builds protein families rapidly using clustering by sequence similarity:

1- Gather all proteins from the spots in one file (`allproteins_spot.prt`).

2- Run usearch:

```
usearch -quiet -cluster_fast allproteins_spot.prt \
        -id 0.7\
        -uc allproteins_spot-70.uc
```

Where the options are:

`-quiet`, to remove the verbose standard output

`-cluster_fast` to use the algorithm uclust, a centroid-based clustering algorithm.

`-id` sets the percentage of identity for clustering.

`-uc` is a tabular file containing the results of the clustering

In the `*.uc` file, a family number is attributed to each protein, and one can easily build the protein families.

A visual representation of the spot focused on the genome with the ICE can be done using the script `plot_ICE_spot.py` (see https://gitlab.pasteur.fr/gem/spot_ICE). This website contains also a tutorial on how to delimit ICE. The genes of the ICE are expected to be at roughly the same frequency in the spot. Hence, their visual representation greatly facilitates

the delimitation of the ICE (for degenerate elements see Note 5.7). The Figure 5 shows an example of the visual representation of the data.

4.2.4 Disentangling tandem elements

When two ICEs are inserted into the same spot (in tandem or intermingled), they are both represented on the graph and this can be used to disentangle them. If the elements are intermingled, or if the two ICEs are present in the same set of strains, their discrimination can be difficult (see Note 5.8). When the ICEs are in tandem, the presence of an integrase between the elements can help in this process. Also, tandem ICEs have a similar succession of a tandem of integrases and conjugative systems.

When the ICE is in tandem with another mobile genetic element in the same set of strains (such as a prophage or an integrative mobilizable element), their delimitation may be facilitated by the presence of a separating integrase. However, it should be noted that ICEs can excise with neighboring elements, and thus mobilize them [19]. In this case, the difference (and thus the delimitation) between the elements may be questionable.

4.2.5 Annotating the elements

The delimited conjugative element can be functionally annotated using a range of computational tools (see Table 2). MGEs such as ICE tend to have many genes with no homologs in the sequence databases (see Note 5.9).

5 Notes

5.1 Co-optations of conjugative systems

Conjugative systems were often co-opted for other functions, notably protein secretion (pT4SS), but also for DNA secretion, and DNA uptake [20,5]. To distinguish between these and conjugative systems it is usually sufficient to identify relaxases in the system, since these are essential for the transfer of DNA, but not for protein secretion. Yet, some systems may perform both functions. In this case, CONJscan will correctly identify the conjugative system because there is a relaxase. Researchers interested in identifying pT4SS might use the TXSScan module of MacSyFinder [8].

5.2 Use of draft genomes and metagenomes

The analysis of draft genomes poses numerous challenges and typically precludes the identification of conjugative plasmids or ICEs. This is because these elements often carry repeats, such as transposable elements, that produce breaks in the assembly process when replicons are sequenced using short reads-technology. As a result, MGEs are split in several contigs and it is usually difficult to know which contigs belong to which element (except if a very similar element is available for comparison). CONJScan can only be used reliably in draft genomes if one knows the order of the contigs, in which case one can give to the program the ordered list of protein sequences, or if the entire conjugative system is a single contig (but that is rarely known *a priori*). Note that while T4SS tend to be encoded in one single locus, and thus are often in a single contig, the relaxase is often encoded apart and may be in another contig. Otherwise, CONJScan may be used to identify the components of the conjugative machinery in draft genomes.

Metagenomic datasets are even more difficult to analyze because the contigs belong to different and unassigned genomes (and are usually very small). In this case, CONJScan can be used to identify components of the conjugation system, but the complete systems are rarely identifiable.

The use of long reads sequencing technologies - that enable easier, longer assemblies, and powerful genome binning techniques that efficiently separate contigs between organisms using multi-variate analyses - are levering these challenges, and promise exciting developments concerning the study of conjugative systems in overlooked portions of the tree of life [21].

5.3 Installing issues

The installation of MacSyFinder requires previous installation of makeblastdb (or formatdb) and HMMER. It also needs Python 2.7. The detailed procedure for installation can be found online (http://macsyfinder.readthedocs.io/en/latest/installation.html).

In case of problems with installation or with the use of the programs, we encourage users to open an “issue” on the corresponding Github's web page (https://github.com/gem-pasteur/macsyfinder/issues). Before, users can check if this issue was not already solved and described in the sites' “closed issues”.

5.4 Running multiple jobs

One can run MacSyFinder with a number of models and on a number of replicons in several ways. As a rule, it is better to make a loop in a shell to run the program independently on each dataset and on each model (see the main text). The use of the "gembases" format as input allows MacSyFinder to analyze a large number of replicons at a time with the same model. There is also an option in MacSyFinder to run several models at a time (parameter "all", see documentation). Yet, if these models have homologous components or if the systems are scattered in the replicon, or in tandem, then the program may misidentify certain systems. Hence, we advise against the use of this option for non-expert users.

5.5 Classing the systems

Usually the output of CONJScan clearly indicates the MPF type of a system, because of the presence of components specific to this MPF. However, some closely related systems, notably FA and FATA, can sometimes jointly hit the same components. Usually the correct system is the one with more components assigned to, and for which the protein profiles have better (lower) i-evalues in the HMMER output. If the situation is unclear, it may be that the system is degenerated (few MPF-specific components), intermingled with another, or a set of tandem of systems (see Note 5.8).

5.6 Modifying the models

The models of MacSyFinder are written in a text file following a specific grammar (see MacSyFinder's documentation). It is very simple to change the models' specifications regarding the quorum, genetic organization, and role of each component. Most of them can be directly altered in the command line, or can be modified permanently in the model text file by following the predefined syntax. The models can also be improved by adding (or removing) novel components. In this case, one must provide the corresponding protein profile, which can be retrieved from public databases (like PFAM or TIGRFAM, see Table 2) or built specifically for the model (see [22] for a description of this process). They should be added to the profiles' directory.

5.7 Degenerate elements

Mobile genetic elements may endure inactivating mutations - including large deletions- that result in degenerate elements. These elements may lack a few components of an active conjugation system, but still be regarded as valid systems by CONJscan because they fit the minimal conditions of the model. Degenerate elements may also complicate the identification of the ICEs because the gene families of the element are present at different frequencies in the pan-genome. It is difficult to distinguish mildly degenerate elements from atypical functional ones without experimental data.

5.8 Intermingled ICE

Intermingled components of conjugative elements can occur in a number of cases. Some MPF types are close and their protein profiles cross-match, leading to a series of components from different MPFs and thus to an apparent intermingled set of conjugative elements. Some cases of true intermingled elements occur when an element integrates inside another element. These cases may be hard to disentangle without experimental evidence.

5.9 ORFs with unknown function

Many genes in mobile genetic elements have their function unknown. If the MGE of interest has many such elements after annotation with standard tools (generic tools like Prokka) then specific curation may be necessary, e.g., by using broader databases of protein profiles like EggNOG or using iterative searches with PSI-BLAST or jackhmmer (Table 2). It is not unusual that several genes remain of unknown function after all these analyses.

6 Acknowledgement

This work was supported by the ANR MAGISBAC project, the CNRS and the Institut Pasteur. We thank the collaborators who have worked with us on this topic in the last decade, notably Fernando de la Cruz, Chris Smillie, Marie Touchon, Maria Pilar Garcillan-Barcia, and Julian Guglielmini.

7 References

1. Grohmann E, Muth G, Espinosa M (2003) Conjugative plasmid transfer in gram-positive bacteria. Microbiol Mol Biol Rev 67:277-301.

2. Smillie C, Pilar Garcillan-Barcia M, Victoria Francia M, Rocha EPC, de la Cruz F (2010) Mobility of Plasmids. Microbiol Mol Biol Rev 74:434-452.

3. de la Cruz F, Frost LS, Meyer RJ, Zechner E (2010) Conjugative DNA Metabolism in Gram-negative Bacteria. FEMS Microbiol Rev 34:18-40.

4. Garcillan-Barcia MP, Francia MV, de la Cruz F (2009) The diversity of conjugative relaxases and its application in plasmid classification. FEMS Microbiol Rev 33:657-687.

5. Guglielmini J, de la Cruz F, Rocha EPC (2013) Evolution of Conjugation and Type IV Secretion Systems. Mol Biol Evol 30:315-331.

6. Abby SS, Neron B, Menager H, Touchon M, Rocha EP (2014) MacSyFinder: A Program to Mine Genomes for Molecular Systems with an Application to CRISPR-Cas Systems. PLoS One 9:e110726.

7. Guglielmini J, Neron B, Abby SS, Garcillan-Barcia MP, la Cruz FD, Rocha EP (2014) Key components of the eight classes of type IV secretion systems involved in bacterial conjugation or protein secretion. Nucleic Acids Res 42:5715-5727.

8. Abby SS, Cury J, Guglielmini J, Neron B, Touchon M, Rocha EP (2016) Identification of protein secretion systems in bacterial genomes. Sci Rep 6:23080.

9. Coluzzi C, Guedon G, Devignes MD, Ambroset C, Loux V, Lacroix T, Payot S, Leblond-Bourget N (2017) A Glimpse into the World of Integrative and Mobilizable Elements in Streptococci Reveals an Unexpected Diversity and Novel Families of Mobilization Proteins. Front Microbiol 8:443.

10. Camacho C, Coulouris G, Avagyan V, Ma N, Papadopoulos J, Bealer K, Madden TL (2009) BLAST+: architecture and applications. BMC Bioinformatics 10:421.

11. Eddy SR (2011) Accelerated Profile HMM Searches. PLoS Comput Biol 7:e1002195.

12. Katoh K, Toh H (2010) Parallelization of the MAFFT multiple sequence alignment program. Bioinformatics 26:1899-1900.

13. Gouy M, Guindon S, Gascuel O (2010) SeaView version 4: A multiplatform graphical user interface for sequence alignment and phylogenetic tree building. Mol Biol Evol 27:221-224.

14. Miele V, Penel S, Duret L (2011) Ultra-fast sequence clustering from similarity networks with SiLiX. BMC Bioinformatics 12:116.

15. Edgar RC (2010) Search and clustering orders of magnitude faster than BLAST. Bioinformatics 26:2460-2461.

16. White EP, Baldridge E, Brym ZT, Locey KJ, McGlinn DJ, Supp SR (2013) Nine simple ways to make it easier to (re) use your data. Ideas in Ecology and Evolution 6:1-10.

17. Page AJ, Cummins CA, Hunt M, Wong VK, Reuter S, Holden MT, Fookes M, Falush D, Keane JA, Parkhill J (2015) Roary: rapid large-scale prokaryote pan genome analysis. Bioinformatics 31:3691-3693.

18. Seemann T (2014) Prokka: rapid prokaryotic genome annotation. Bioinformatics 30:2068-2069.

19. Bellanger X, Morel C, Gonot F, Puymege A, Decaris B, Guedon G (2011) Site-specific accretion of an integrative conjugative element together with a related genomic island leads to cis mobilization and gene capture. Mol Microbiol 81:912-925.

20. Alvarez-Martinez CE, Christie PJ (2009) Biological diversity of prokaryotic type IV secretion systems. Microbiol Mol Biol Rev 73:775-808.

21. Hug LA, Baker BJ, Anantharaman K, Brown CT, Probst AJ, Castelle CJ, Butterfield CN, Hernsdorf AW, Amano Y, Ise K, Suzuki Y, Dudek N, Relman DA, Finstad KM, Amundson R, Thomas BC, Banfield JF (2016) A new view of the tree of life. Nat Microbiol 1:16048.

22. Abby SS, Rocha EPC (2017) Identification of Protein Secretion Systems in Bacterial Genomes Using MacSyFinder. In: Cascales E (ed) Bacterial protein secretion systems. Methods in Molecular Biology. Springer, p in press.

495

Table 1: List of MacSyFinder output files and folders.

Output	Description
`macsyfinder.conf`	Parameters used for the run
`macsyfinder.log`	Log information in case of problem
`macsyfinder.out`	Standard output
`macsyfinder.report`	Tabular report with the proteins of conjugation systems
`macsyfinder.summary`	Tabular report with the systems and their components
`macsyfinder.tab`	Tabular report with the number of systems detected per replicon
`results.macsyfinder.json`	JSON file summarizing the results for visualization with MacSyView
`hmmer_results`	Folder containing all the `hmmer` output files ("raw" and filtered)

500

Table 2: List of Resources or tools for useful for the detection and annotation of conjugative elements

Resource	Type	Description	Link
MacSyFinder	Generic	Program to identify molecular systems (of which CONJscan and TXSScan are modules)	https://github.com/gem-pasteur/macsyfinder
CONJScan	Conjugation systems	MacSyFinder module to identify conjugation systems	https://github.com/gem-pasteur/Macsyfinder_models https://galaxy.pasteur.fr/
MacSyView	Generic	To visualize MacSyFinder's results	https://github.com/gem-pasteur/macsyview
Blast tools	Sequence analysis	Rapid sequence similarity searches (incl. blast, psi-blast, makeblastdb)	https://blast.ncbi.nlm.nih.gov
HMMER	HMM analysis	Allows the use HMM profiles but also build one's own profiles (incl. hmmsearch and jackhmmer)	http://hmmer.org/
Usearch	Protein clustering	Very fast clustering method (uclust) for very similar proteins (>50% identity)	http://drive5.com/usearch/
MAFFT	Alignment	Multiple sequence alignment	http://mafft.cbrc.jp/alignment/software/
Seaview	Alignment Visualization	To visualize and edit multiple alignments	http://doua.prabi.fr/software/seaview
Roary	Pan-genomes	Construction of core and pan-genomes	https://sanger-pathogens.github.io/Roary/
PFAM	Proteins	Database of HMM profile	http://pfam.xfam.org/
TIGRFAM	Proteins	Database of HMM profile	http://www.jcvi.org/cgi-bin/tigrfams/index.cgi
EggNOG	Proteins	Database of HMM profile for functional annotation	http://eggnogdb.embl.de

CONJdb	Conjugation systems	Database of conjugative systems	http://conjdb.web.pasteur.fr/
TXSScan	Secretion systems	MacSyFinder module with model for all type of secretion system	https://github.com/gem-pasteur/Macsyfinder_models
IntegronFinder	Integrons	Detects integrons in genomes	https://github.com/gem-pasteur/Integron_Finder/
RFAM	RNA	Database of Covariance Models to find many types of RNA	http://rfam.xfam.org/
Infernal	RNA	Searches for RNAs using covariance models, notably for RFAM.	http://eddylab.org/infernal/
Silix	Protein clustering	Clustering method to build protein families from "blast all against all" results	http://lbbe.univ-lyon1.fr/-SiLiX-.html?lang=en
Prokka	Annotation	Rapid annotation of bacterial genomes	https://github.com/tseemann/prokka
ResFams	Antibiotic resistance	Database of HMM profiles specific of antibiotic resistance	http://www.dantaslab.org/resfams
CARD	Antibiotic resistance	Database of antibiotic resistance genes	https://card.mcmaster.ca/
Victors	Virulence factor	Database of Virulent factor	http://www.phidias.us/victors/

Figure 1. Screening genomes for conjugation systems using CONJscan with MacSyFinder. The components of a conjugation system are an ATPase, a coupling protein ("coupling p.", T4CP), and a relaxase, plus MPF-type specific genes which are found in a particular genetic organization, described in the models for T4SS (see Fig. 2). The CONJscan module turns MacSyFinder into a search engine for conjugation systems in genomes. First, the selected models of conjugation systems are read and the corresponding components are searched by sequence similarity in the genome (multi-fasta file) using HMMER with the HMM profiles of CONJscan. Then the genetic organization of the hits for the components is analyzed to identify sets of hits compatible with the models. Clusters of hits fulfilling the requirements are used to fill up occurrences of the systems. In the end, if the pre-defined number of components is found in the expected genetic organization, the presence of a conjugation system is predicted, for example here, a MPF of type T. Whether fruitful or not, the results of the search are stored in the output files (see Table 1) that can be used to guide the design of customized models.

Figure 2. The models of CONJscan for each MPF type. Each line is a graphical representation of a model, as defined in the main text. On the left-hand side, in grey, there are the three mandatory proteins, virb4, t4cp and MOB, common for all models. At the bottom, the box ”exchangeable” indicates that any of the relaxase profiles can be used for each MPF type. On the right-hand side, colored by type, there are the specific genes of each MPF type. They are coined "accessory" because they are not always identified in the locus for a number of reasons (missing, unidentifiable, etc). Hence, we set a quorum as the minimum number of components in a valid system (in parenthesis, in front of each line). In the model file, one can modify the quorum for the mandatory profiles and the quorum for the total number of profiles (mandatory + accessory).

Figure 3. Screenshots of the Galaxy interface for CONJscan. See main text for explanations.

Figure 4. Screenshot of the Galaxy interface for the visualization of the results of CONJscan using MacSyViewer. See main text for explanations.

Figure 5. Example of a visualization plot. The figure shows two ICEs that are in the same spot. Boxes represent genes (width proportional to its length), which are encoded on the direct strand (above the line), or the complementary one (below the line). The focal genome (containing the ICE) is in the middle of its subplot (focal ICEs are turquoise and orange, respectively). Each box above a gene in the focal genome belongs to the same pan-genome family. On the extremities, the core-genes are represented (these genes families have representatives in all genomes, thus the piling of boxes of all colors above them). The two ICEs have homologous conjugative systems (with hatches) but not the totality of the ICE. The grey line represents the GC% along the focal genome. The window on the right-hand side represents the number of genes in the interval divided by the number of genes in the interval with the largest number of genes of the spot. Here we see that there are few genes in the intervals lacking ICEs (this is not necessarily always the case). One can click on the genes to see their annotations. A right-click will select a gene as one of the limits (red-boxed genes near the core-genes). When the two limits are set, the intervening elements will be exported in a tabular file when closing this window.

Figure 1

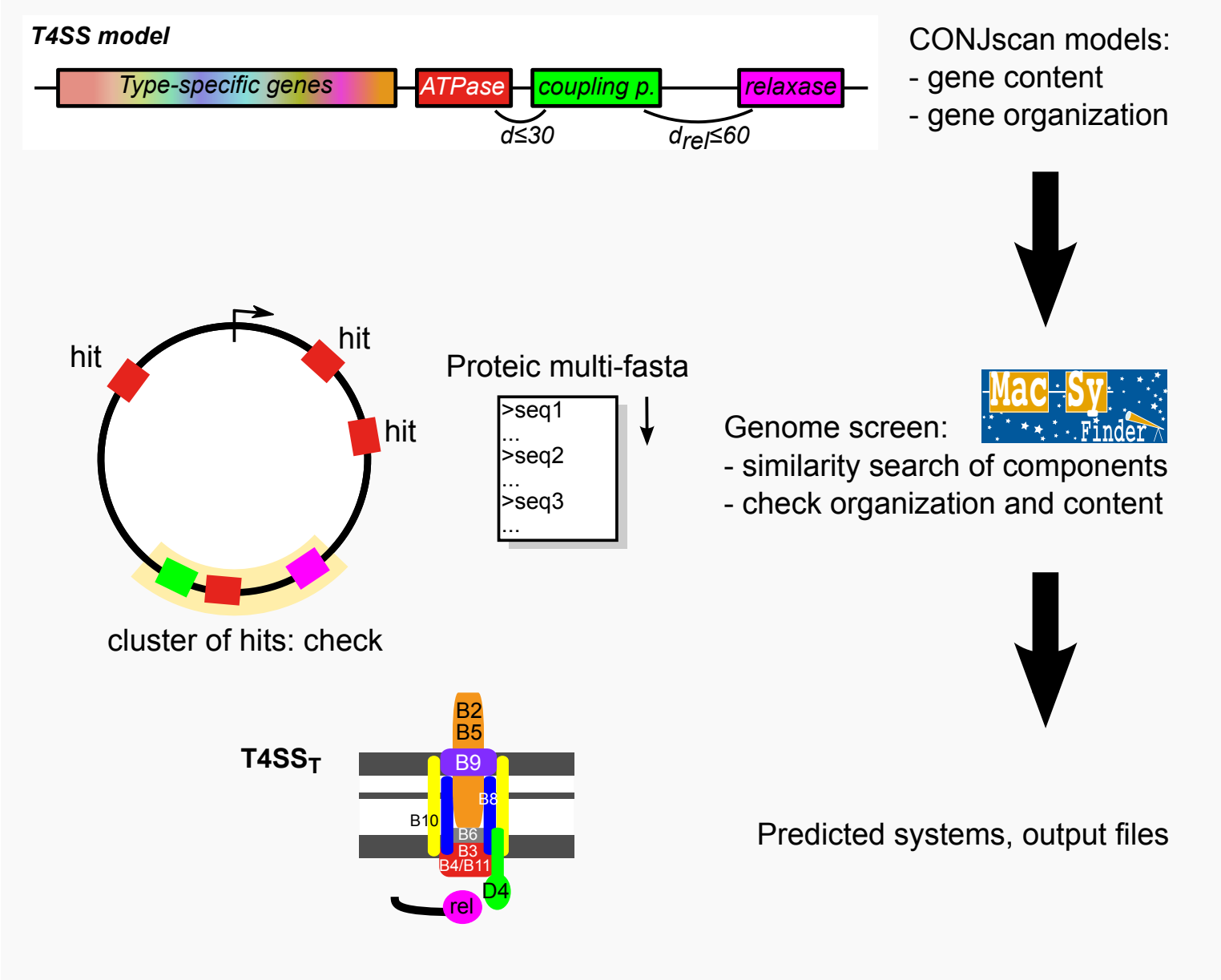

T4SS model
Type-specific genes
ATPase
coupling p.
relaxase
d≤30
drel≤60
CONJscan models:
- gene content
- gene organization
hit
hit
hit
cluster of hits: check
Proteic multi-fasta
>seq1
...
>seq2
...
>seq3
...
Genome screen:
- similarity search of components
- check organization and content
T4SST
B2
B5
B9
B8
B10
B6
B3
B4/B11
D4
rel
Predicted systems, output files

Figure 2

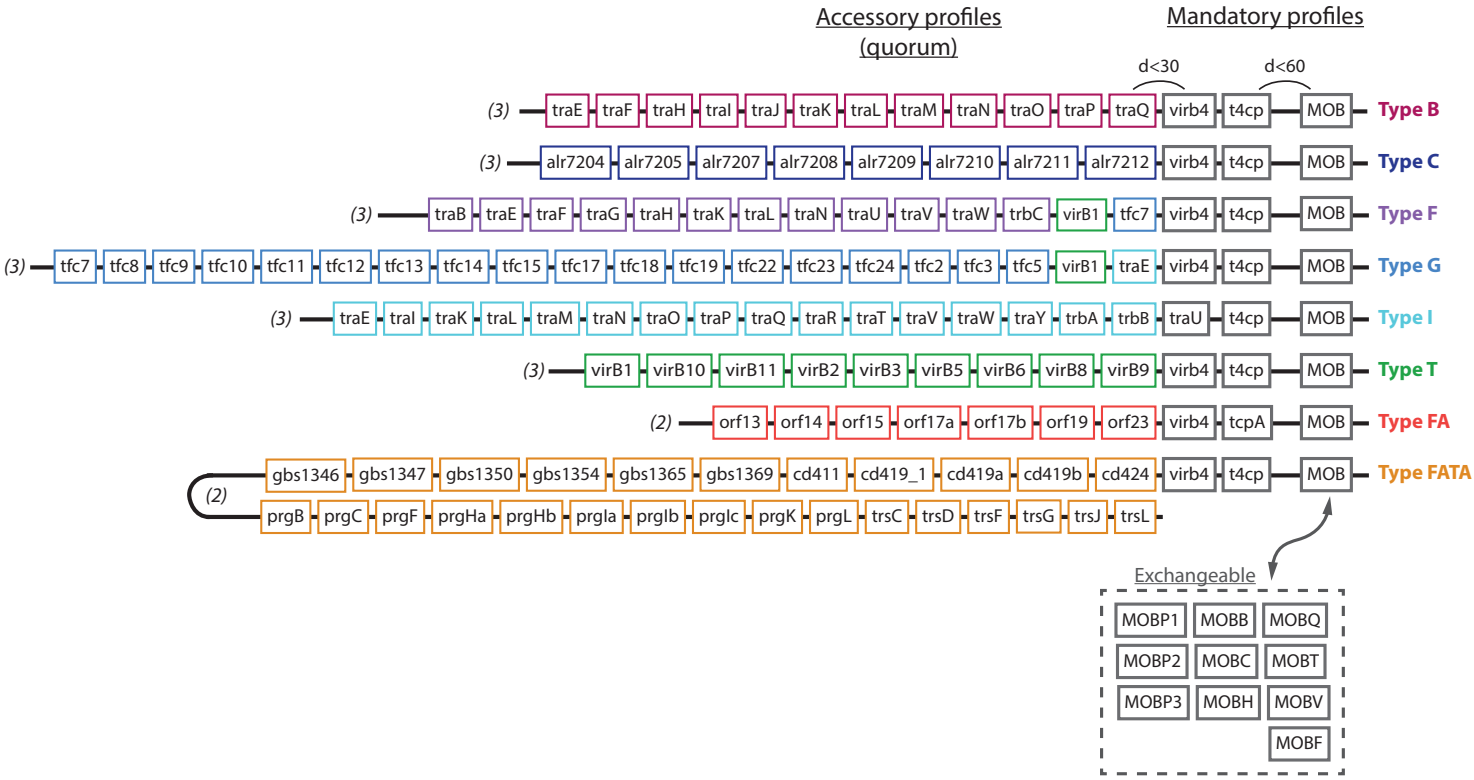

Accessory profiles
(quorum)
Mandatory profiles
d<30
d<60
(3) traE traF traH traI traJ traK traL traM traN traO traP traQ virb4 t4cp MOB Type B
(3) alr7204 alr7205 alr7207 alr7208 alr7209 alr7210 alr7211 alr7212 virb4 t4cp MOB Type C
(3) traB traE traF traG traH traK traL traN traU traV traW trbC virB1 tfc7 virb4 t4cp MOB Type F
(3) tfc7 tfc8 tfc9 tfc10 tfc11 tfc12 tfc13 tfc14 tfc15 tfc17 tfc18 tfc19 tfc22 tfc23 tfc24 tfc2 tfc3 tfc5 virB1 traE virb4 t4cp MOB Type G
(3) traE traI traK traL traM traN traO traP traQ traR traT traV traW traY trbA trbB traU t4cp MOB Type I
(3) virB1 virB10 virB11 virB2 virB3 virB5 virB6 virB8 virB9 virb4 t4cp MOB Type T
(2) orf13 orf14 orf15 orf17a orf17b orf19 orf23 virb4 tcpA MOB Type FA
(2) gbs1346 gbs1347 gbs1350 gbs1354 gbs1365 gbs1369 cd411 cd419_1 cd419a cd419b cd424 virb4 t4cp MOB Type FATA
prgB prgC prgF prgHa prgHb prgIa prgIb prgIc prgK prgL trsC trsD trsF trsG trsJ trsL
Exchangeable
MOBP1 MOBB MOBQ
MOBP2 MOBC MOBT
MOBP3 MOBH MOBV
MOBF

Figure 3

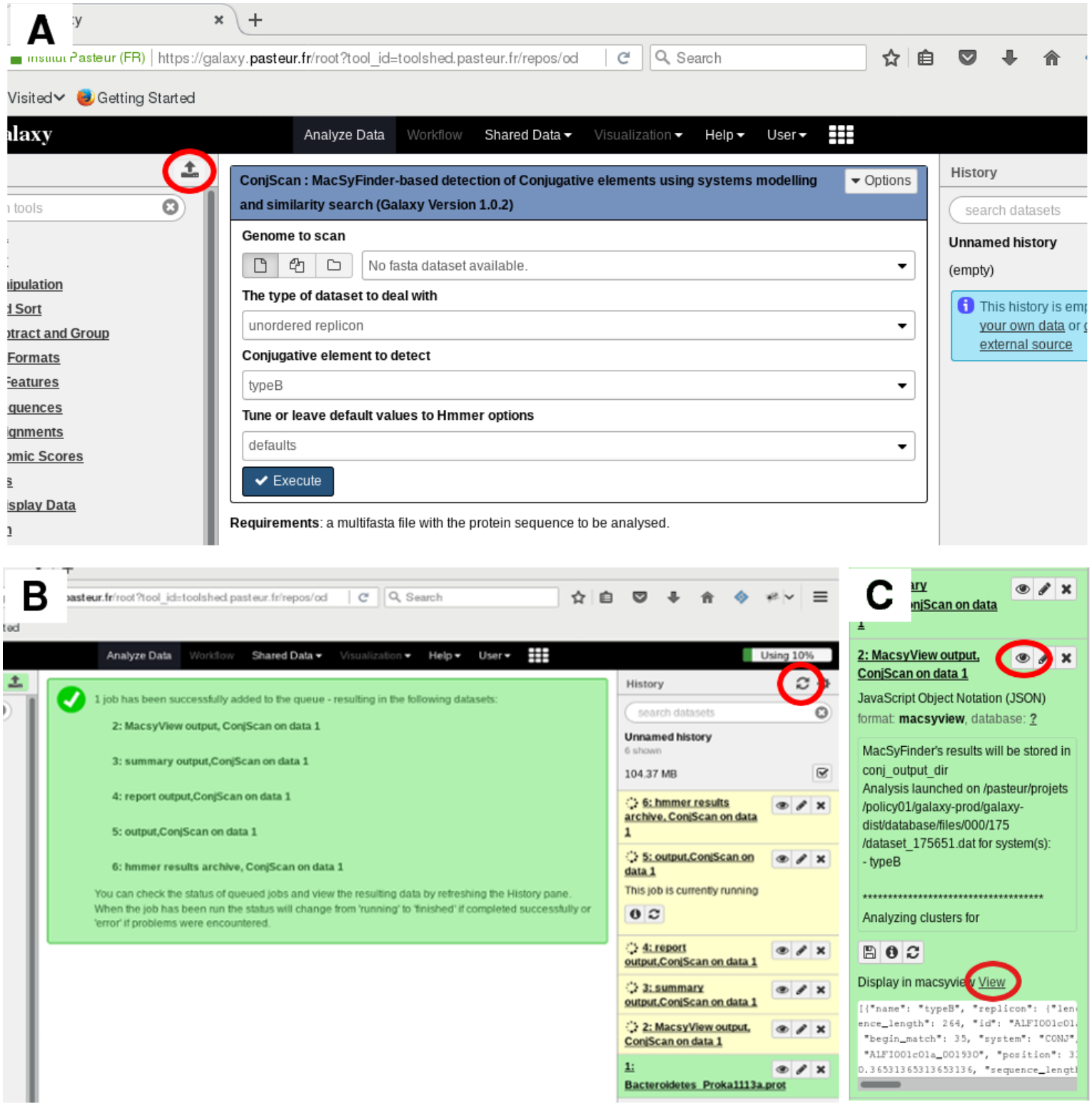
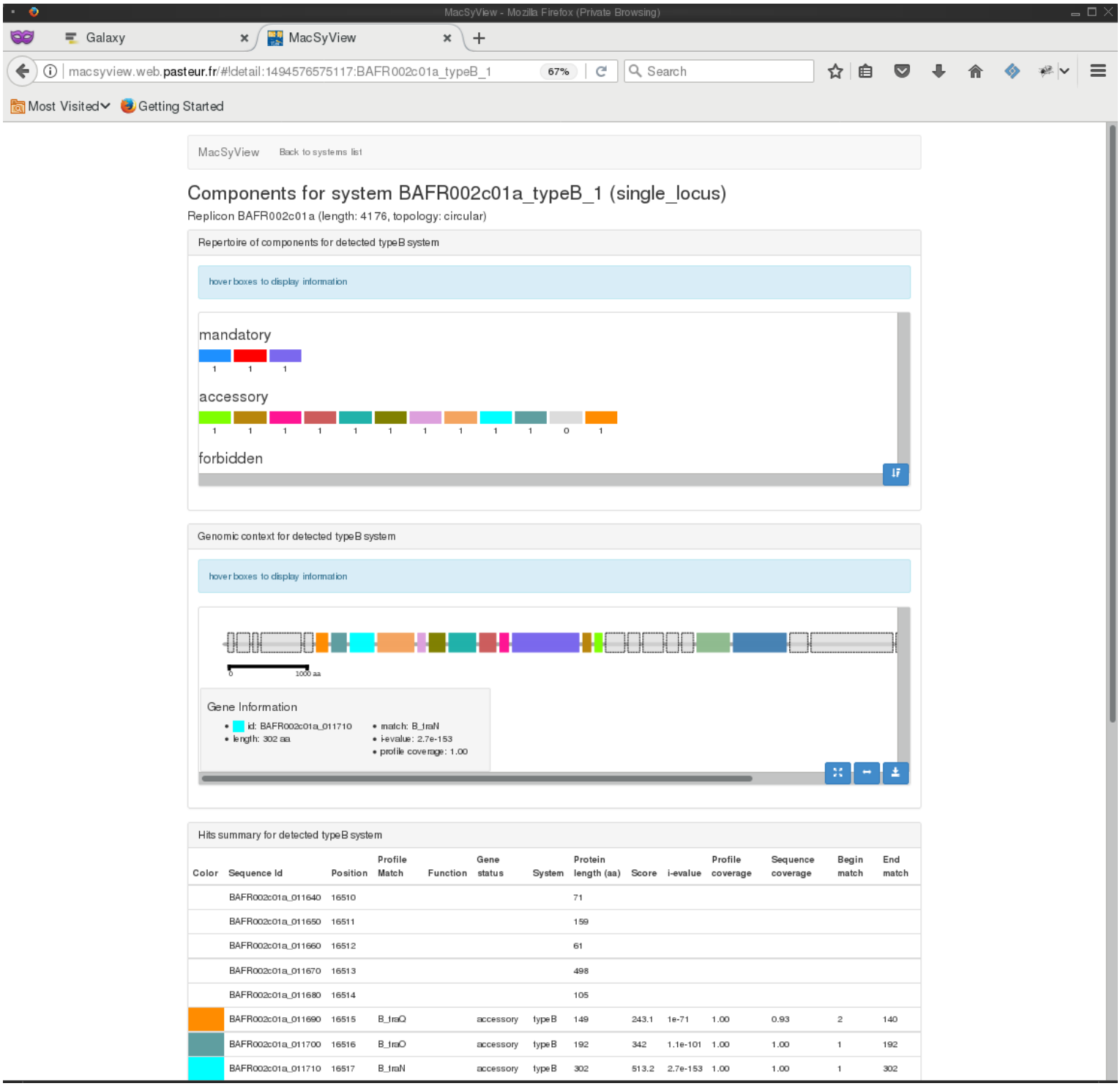
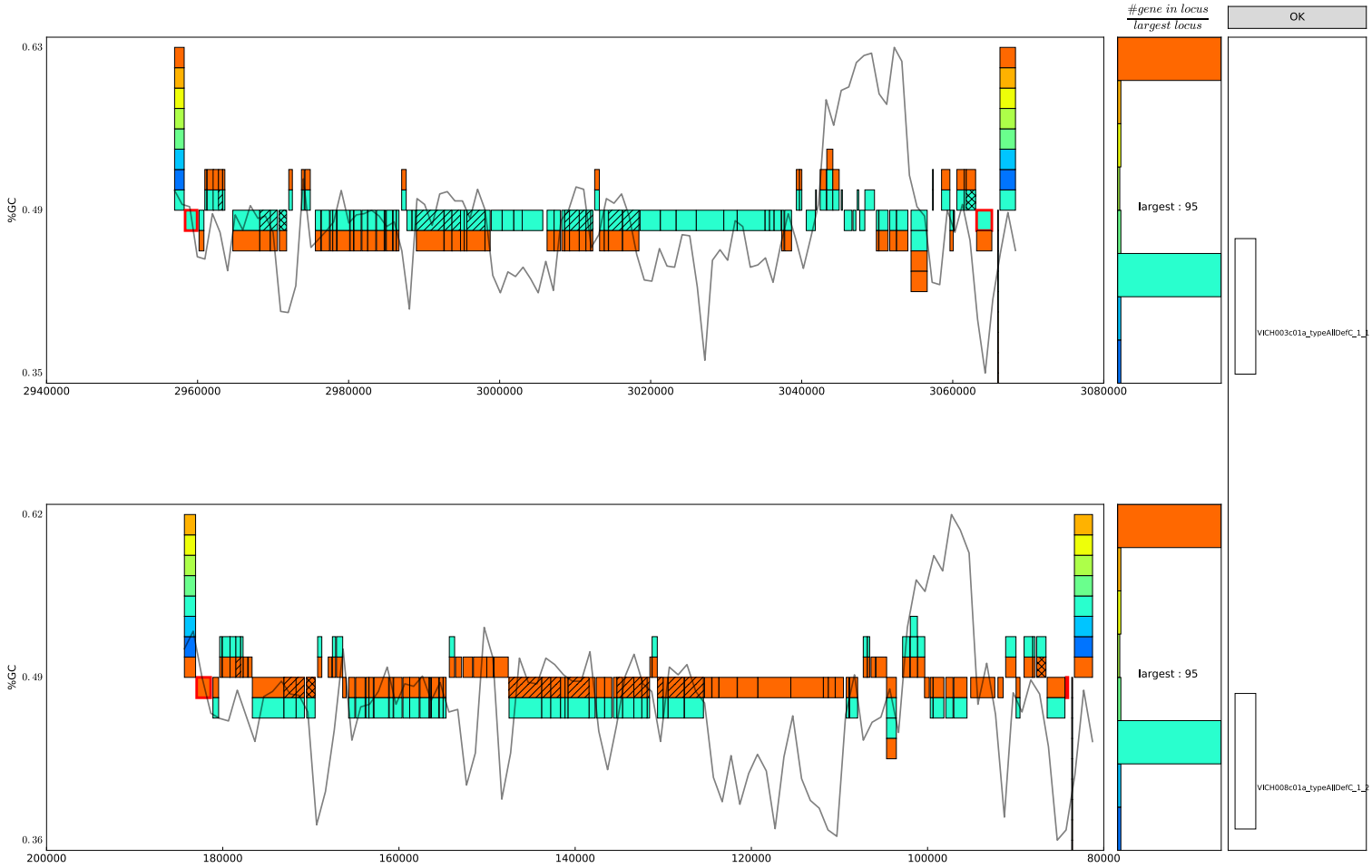

A
Institut Pasteur (FR) | https://galaxy.pasteur.fr/root?tool_id=toolshed.pasteur.fr/repos/od
Search
Visited
Getting Started
Analyze Data
Workflow
Shared Data
Visualization
Help
User
ConjScan : MacSyFinder-based detection of Conjugative elements using systems modelling and similarity search (Galaxy Version 1.0.2)
Options
Genome to scan
No fasta dataset available.
The type of dataset to deal with
unordered replicon
Conjugative element to detect
typeB
Tune or leave default values to Hmmer options
defaults
Execute
Requirements: a multifasta file with the protein sequence to be analysed.
History
search datasets
Unnamed history
(empty)
B
Using 10%
1 job has been successfully added to the queue - resulting in the following datasets:
2: MacsyView output, ConjScan on data 1
3: summary output,ConjScan on data 1
4: report output,ConjScan on data 1
5: output,ConjScan on data 1
6: hmmer results archive, ConjScan on data 1
You can check the status of queued jobs and view the resulting data by refreshing the History pane. When the job has been run the status will change from 'running' to 'finished' if completed successfully or 'error' if problems were encountered.
Unnamed history
6 shown
104.37 MB
6: hmmer results archive, ConjScan on data 1
5: output,ConjScan on data 1
This job is currently running
4: report output,ConjScan on data 1
3: summary output,ConjScan on data 1
2: MacsyView output, ConjScan on data 1
1: Bacteroidetes_Proka1113a.prot
C
2: MacsyView output, ConjScan on data 1
JavaScript Object Notation (JSON)
format: macsyview, database: ?
MacSyFinder's results will be stored in conj_output_dir
Analysis launched on /pasteur/projets/policy01/galaxy-prod/galaxy-dist/database/files/000/175/dataset_175651.dat for system(s):
- typeB
Analyzing clusters for
Display in macsyview View

Figure 4



Color	Sequence Id	Position	Profile Match	Function	Gene status	System	Protein length (aa)	Score	i-evalue	Profile coverage	Sequence coverage	Begin match	End match
	BAFR002c01a_011640	16510					71						
	BAFR002c01a_011650	16511					159						
	BAFR002c01a_011660	16512					61						
	BAFR002c01a_011670	16513					498						
	BAFR002c01a_011680	16514					105						
	BAFR002c01a_011690	16515	B_traQ		accessory	typeB	149	243.1	1e-71	1.00	0.93	2	140
	BAFR002c01a_011700	16516	B_traO		accessory	typeB	192	342	1.1e-101	1.00	1.00	1	192
	BAFR002c01a_011710	16517	B_traN		accessory	typeB	302	513.2	2.7e-153	1.00	1.00	1	302

Figure 5


$\frac{\#gene\ in\ locus}{largest\ locus}$
OK
%GC
0.63
0.49
0.35
2940000
2960000
2980000
3000000
3020000
3040000
3060000
3080000
largest : 95
VICH003c01a_typeAllDefC_1_1
%GC
0.62
0.49
0.36
200000
180000
160000
140000
120000
100000
80000
largest : 95
VICH008c01a_typeAllDefC_1_2