Chapter 15

Tn-Core: Functionally Interpreting Transposon-Sequencing Data with Metabolic Network Analysis

George C. diCenzo, Marco Galardini, and Marco Fondi

Abstract

Transposon-sequencing (Tn-seq) is a powerful tool facilitating the genome-scale identification of genes required for bacterial growth or survival in an environment of interest. However, Tn-seq suffers from two primary drawbacks: (1) genetic interactions masking phenotypes thereby resulting in important cellular functions remaining undiscovered and (2) a difficulty in easily going from a list of essential genes to a functional understanding of cell physiology. Tn-Core is a computational toolbox to help overcome these limitations through combining the output of Tn-seq studies with in silico genome-scale metabolic networks. In this chapter, we outline how to use Tn-Core to contextualize Tn-seq data (and optionally RNA-seq data) with metabolic models to: (1) generate a complete view of essential metabolism, (2) prepare context-specific metabolic models for further computational analyses, and (3) refine genome-scale metabolic models. All functions of Tn-Core are provided for download from a freely available repository (github.com/diCenzo-GC/Tn-Core), and a web-app requiring limited computational experience is also available (combo.dbe.unifi.it/tncore).

Key words Transposon-sequencing, Insertion-sequencing, Genome-scale metabolic network analysis, GENRE, Functional genomics, RNA-sequencing, Bacterial metabolism, Essential genes

1 Introduction

Transposon-sequencing (Tn-seq, also known as insertion-sequencing or INseq) approaches, and derivatives of this method involving barcoded transposons (BarSeq), have rapidly gained in popularity due to their unparalleled ability to functionally interrogate prokaryotic genomes [1–4]. However, it can be difficult to progress from a list of essential genes to a functional understanding of the data. Tn-seq screens also may fail to identify some essential cellular functions due in part to functional redundancy or other types of genetic interactions. Genetic interactions are surprisingly common in bacterial genomes [5, 6], and they can result in one-third of essential cellular functions remaining undiscovered when using Tn-seq [6].

Mario Andrea Marchisio (ed.), *Computational Methods in Synthetic Biology*, Methods in Molecular Biology, vol. 2189, https://doi.org/10.1007/978-1-0716-0822-7_15,

One approach to overcome these limitations is to combine Tn-seq with genome-scale metabolic modeling [6–8]. Metabolic modeling can be divided into two stages: (1) preparation of an in silico reconstruction of the entire metabolic network of an organism [9], and (2) application of mathematical methods to simulate flux distribution throughout the network [10]. As every network reaction is linked to the corresponding gene(s) encoding the protein(s) catalyzing the reaction, metabolic modeling allows for a genome-scale prediction of essential metabolic genes [11]. Researchers have recently begun to recognize the potential of integrating these two highly complementary approaches [6–8, 12–17]. Often, this involved using Tn-seq data to identify and correct errors (i.e., incorrect reactions or gene-reaction pairings) in genome-scale metabolic networks [6–8, 12–14], while others have demonstrated that metabolic networks serve as powerful tools to contextualize Tn-seq data [6, 7, 13]. In doing so, metabolic networks provide a platform to aid functional interpretation of Tn-seq data, and to help identify errors or missing data (Fig. 1).

We recently reported the development of Tn-Core [15], a toolbox for integration of Tn-seq (or BarSeq) data with metabolic modeling that is available as a web application and for download to run locally. Here, we describe how to use Tn-Core to: (1) generate a complete view of essential metabolism, (2) prepare context-specific metabolic models for further computational analyses, and (3) refine genome-scale metabolic models. Instructions for both the local version and the web application are provided.

2 Software and Inputs

2.1 The Stand-Alone Version of Tn-Core

Tn-Core is dependent on various software, all of which are freely available for academic users with the exception of MATLAB. Below, we provide a list of all the required software and the necessary input data. Software version numbers are provided when we expect compatibility with Tn-Core to be version dependent; if no version number is provided, we expect that any version will be compatible with Tn-Core (*see* **Note 1**).

2.1.1 Software

1. Tn-Core. Available at: github.com/diCenzo-GC/Tn-Core [15]. Tn-Core comes with a detailed manual describing the functions and the parameters of each function that can be modified.
2. MATLAB. Available at: mathworks.com.
3. COBRA Toolbox. Available at: opencobra.github.io/cobra-toolbox/stable [10].
4. TIGER Toolbox version 1.2-beta. Available at: csbl.bitbucket.io/tiger/download.html [16].

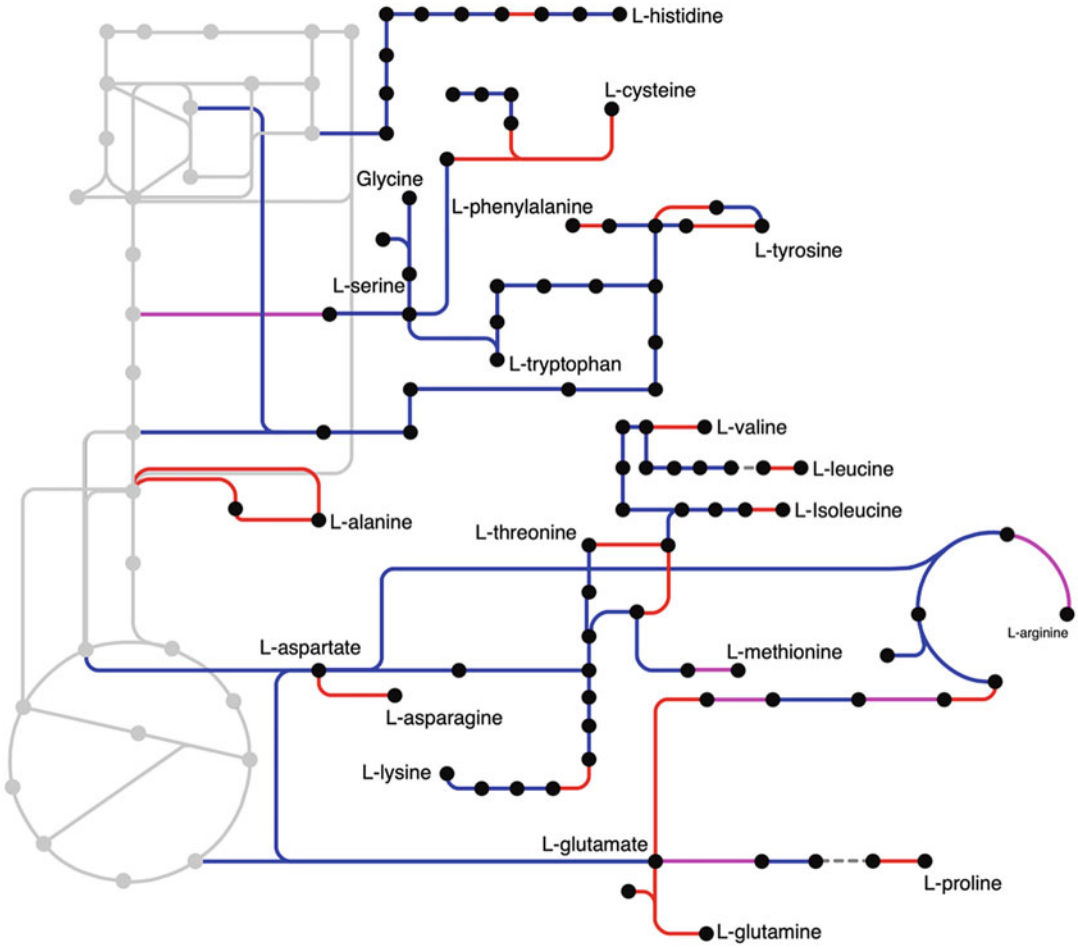

Fig. 1 Effects of genetic interactions on the interpretation of Tn-seq data. A summary schematic of the amino acid biosynthetic pathways of *Sinorhizobium meliloti* is shown. Central carbon metabolism is displayed in light gray. Lines indicate metabolic reactions while dots represent metabolites. Blue lines indicate reactions whose associated genes were essential for the growth of wild-type *S. meliloti* in minimal medium, based on a published Tn-seq screen [6]. The genes associated with reactions in magenta appeared to be nonessential, but important for growth, in the published Tn-seq data, whereas reactions in red are associated with genes that appeared nonessential in the wild-type. This figure highlights that ~25% of core amino acid biosynthetic reactions in a wild-type *S. meliloti* strain remained undetected based solely on Tn-seq data. (Figure adapted from [6])

5. FASTCORE version 1.0. Available at: uni.lu/forschung/fstc/life_sciences_research_unit/research_areas/systems_biology/software/fastcore [17].
6. libSBML. Available at: sourceforge.net/projects/sbml/files/libsbml [18].
7. SBML Toolbox. Available at: sourceforge.net/projects/sbml/files/SBMLToolbox [19].

8. iLOG CPLEX Studio (available at: ibm.com/products/ilog-cplex-optimization-studio) or the Gurobi solver (available at: gurobi.com).

2.1.2 Input Data

1. Tn-seq or BarSeq data as a file entitled 'TnSeqData.txt' (*see* **Note 2**). The data must be provided as a tab-delineated file, with the data in the first column and the gene names in the second column (*see* **Note 3**). The data must compile to the following specifications: (1) it should not be log-transformed, (2) all values should be ≥ 0, (3) the data must be normalized (e.g., by gene length), and (4) they should be provided in a format in which a smaller number is indicative of a greater fitness effect when the gene is mutated (*see* **Note 4**).
2. (Optional) RNA-seq data as a file entitled 'RnaSeqData.txt' (*see* **Note 2**). If provided, the data must be present as a tab-delineated file, with the data in the first column and the gene names in the second column (*see* **Note 3**). The data should be provided as RPKM (Reads per kilo base per million mapped reads) or TPM (Transcript per million) and should not be log-transformed.
3. A genome-scale metabolic network reconstruction (GENRE for short) as a file entitled 'inputModel.xml' (*see* **Note 2**). The metabolic network can be prepared by the user or downloaded from online databases (*see* **Notes 5**–**7**). If using the automated Tn-Core pipeline, the reconstruction should be provided in SBML format as a .XML file. If not using the fully automated pipeline, the reconstruction can be provided as a COBRA-formatted model (*see* **Note 8**).

2.1.3 Additional Requirements Specific to the Automated Pipeline

1. Exchange reactions as a file entitled 'ExRxns.txt' (*see* **Note 2**). This file should be a tab-delineated file containing information on the exchange reactions that should have a nonzero lower bound (*see* **Notes 9** and **10**). The first column should contain the identifiers of the exchange reactions of interest. The second column should contain the lower bound (a negative number; we recommend a value of −10 for all rows if unsure of which values to use). Although this can be an empty file, we generally recommend against this (*see* **Note 10**).
2. The objective reaction as a file entitled 'ObjectiveRxn.txt' (*see* **Note 2**). A tab-delineated file with a single row containing information on the objective function (*see* **Note 11**), which will generally be the biomass reaction. The first column should contain the identifier of the objective function. The second column should contain the objective coefficient (generally, this value is set to 1).

2.2 Requirements for the Tn-Core Web Application

Aside from a web browser and an internet connection, no local software is required to run the Tn-Core web application. The input data required for running the Tn-Core web application are the same as for running the local version of Tn-Core using the automated pipeline, as described in Subheadings 2.1.2 and 2.1.3 (*see* **Notes 2** and **12**).

3 Methods and Outputs

In this section, we provide descriptions of how to use the local and web versions of Tn-Core to: (1) contextualize Tn-seq data, and (2) refine GENREs. The local version of Tn-Core is distributed with a fully automated pipeline, whose use is optional. We therefore describe how to use Tn-Core with or without the automated pipeline. In general, we recommend that the automated pipeline is used by those without prior experience in metabolic modeling, whereas we recommend users experienced in the COBRA Toolbox to try Tn-Core without the automated pipeline.

3.1 Contextualizing Tn-Seq Data (Local Version)

3.1.1 Prepare the Workspace

1. Ensure all of the required software has been downloaded and ideally placed within the same folder on your machine.
2. Place all input files into a new folder (different from where the software files are located).
3. Open MATLAB and navigate to the directory containing the input data.
4. Ensure all the software are in your MATLAB path. This can be done with the following command, where '`/PATH/TO/SOFTWARE/`' is replaced with the path of the software folder:

```
addpath(genpath(/PATH/TO/SOFTWARE/));
```

5. Initialize the COBRA Toolbox with the following command:

```
initCobraToolbox;
```

6. Set the COBRA solver with the following command, where '`SOLVER`' is replaced with either '`gurobi`' or '`ibm_cplex`':

```
changeCobraSolver('SOLVER', 'all');
```

7. Initialize the TIGER Toolbox with the following command, where '`SOLVER`' is replaced with either '`gurobi`' or '`ibm_cplex`':

```
start_tiger('SOLVER');
```

3.1.2 Prepare the Input Data

1. Load the Tn-seq data with the following command:

```
tnseq = table2cell(readtable('TnSeqData.txt', ...
'ReadRowNames',
false, 'ReadVariableNames', false, ...
'Delimiter', '\t'));
```

2. (Optional) Load the RNA-seq data with the following command:

```
rnaseq = table2cell(readtable('RnaSeqData.txt', ...
'ReadRowNames', false, 'ReadVariableNames', false, ...
'Delimiter', '\t'));
```

3. Load the GENRE with the following command (*see* **Note 13**):

```
model = readCbModel('inputModel.xml');
```

4. Ensure that the imported model contains the grRules and rxnGeneMat fields by running the following command that will add these fields if they are missing (*see* **Note 14**):

```
model = tncore_fix(model);
```

5. Set the objective function of the model and set the bounds of the exchange reactions using standard procedures and functions of the COBRA Toolbox (*see* **Note 13**).

3.1.3 Run Tn-Core

1. Integrate the Tn-seq data with the input GENRE using Tn-Core. If no RNA-seq data are provided, initiate Tn-Core with the following command:

```
[coreModel, reducedModel] = tncore_core(model, tnseq);
```

If RNA-seq data are provided, initiate Tn-Core with the following command:

```
[coreModel, reducedModel] = tncore_core(model, tnseq, ...
[], [],[], rnaseq);
```

The above commands will run Tn-Core with default parameters. Numerous parameters can be modified. Detailed information on available parameters can be obtained in MATLAB with the following command:

```
help tncore_core
```

2. The MATLAB workspace can be saved with the following command (*see* **Note 15**):

```
save('tncore_output.mat');
```

3. Export the core model as an easy-to-read Excel file. If no RNA-seq data are provided, export the model with the following command:

```
tncore_export(coreModel, reducedModel, tnseq);
```

If RNA-seq data are provided, export the model with the following command:

```
tncore_export(coreModel, reducedModel, tnseq, rnaseq);
```

3.2 Contextualizing Tn-Seq Data (Local Version: Automated)

3.2.1 Prepare the Workspace

1. Ensure all of the required software has been downloaded and ideally placed within the same folder on your machine.
2. Place all input files into a new folder (different from where the software files are located).
3. Open MATLAB and navigate to the directory containing the input data.
4. Ensure all the software are in your MATLAB path. This can be done with the following command, where '`/PATH/TO/SOFTWARE/`' is replaced with the path to the software folder:

```
addpath(genpath(/PATH/TO/SOFTWARE/));
```

3.2.2 Run Tn-Core

Integrate the Tn-seq data with the input GENRE using Tn-Core. If no RNA-seq data are provided, initiate Tn-Core with the following command:

```
tncore_overall_workflow();
```

If RNA-seq data are provided, initiate Tn-Core with the following command:

```
tncore_overall_workflow('rnaseqData', true);
```

The above commands will run Tn-Core with default parameters. Numerous parameters can be modified. Detailed information on available parameters can be obtained in MATLAB with the following command (*see* **Notes 16** and **17**):

```
help tncore_overall_workflow
```

The core model will be automatically exported as an Excel file. At the time of writing, a COBRA-formatted model will not be returned.

3.3 Contextualizing Tn-Seq Data (Web Application)

3.3.1 Run Tn-Core

1. Upload the GENRE in SBML format in the 'Metabolic reconstruction' field.
2. Upload the exchange reaction file in the 'Exchange reactions' field.
3. Upload the objective function file in the 'Objective reaction' field.
4. Upload the Tn-seq data in the 'TnSeq data' field.
5. (Optional) Upload RNA-seq data in the 'RNAseq data' field.
6. (Optional) Insert a value in the 'Minimum growth fraction' field. This value determines the minimum growth rate of the output core model as a fraction of the maximal growth rate of the input core model. The value must be >0 and ≤ 1. The default is 0.5.
7. (Optional) Insert a value in the 'Expression threshold' field. This value determines the minimum RPKM or TPM value for a gene to be considered highly expressed. The value must be >0. A common value would be the average of the RPKM or TPM values for all genes in the dataset. The default is set to 0.02% the sum of all RPKM or TPM values.
8. Enter your email address for anonymous statistics on the server usage.

9. (Optional) Enter a passphrase to protect your results to ensure that only you can access the results.
10. Click 'submit job'.

3.3.2 Obtain the Output Files

1. Navigate to the results page. If the web browser was not closed, then it will be automatically redirected to the results page when Tn-Core finishes.

   Otherwise, ensure to record the URL prior to closing the web browser. Then, paste the URL in the address bar to navigate to the results page. If a passphrase was entered when starting Tn-Core, enter the passphrase when prompted. Results older than 1 week will be cleared from the server.
2. Download the MATLAB file containing the core metabolic model in COBRA format.
3. Download the Excel file containing the core metabolic model.

3.4 Contextualizing Tn-Seq Data: Output Files

When contextualizing the Tn-seq data using either of the three methods described in Subheadings 3.1, 3.2, and 3.3, the primary output will be a context-specific core metabolic model derived from the input model on the basis of the Tn-seq data (*see* **Note 18**). This model will contain the core (essential) metabolism required for the cell to produce biomass, embedding reactions associated with genes identified as essential in the Tn-seq screen (*see* **Note 19**). The core model will also contain reactions whose associated genes were not essential in the Tn-seq dataset if the reaction is nevertheless required for biomass production. The model will be returned in one or two formats:

1. A COBRA-formatted model will be returned when running the Tn-Core web application (*see* **Note 20**) or the stand-alone version of Tn-Core without the automated pipeline. This output is currently not returned when running the local version of Tn-Core with the automated pipeline. The COBRA-formatted model can be directly used in downstream analysis pipelines.
2. An Excel-formatted model will be returned when running the Tn-Core web application or the local version of Tn-Core with the automated pipeline. This output can also be obtained with the local version of Tn-Core without the automated pipeline by following **step 3** of Subheading 3.1.3. By providing a comprehensive picture of the core metabolism of the organism, the Excel file provides a functional interpretation of the Tn-seq data.

The Excel file will consist of three worksheets: (1) the reactions, (2) the genes, and (3) the metabolites (*see* **Note 21**). Each row in the reactions worksheet represents one core metabolic reaction and will include: (1) the name of the reaction, (2) the reaction formula

using human-understandable metabolite names, and (3) the genes associated with the reaction using Boolean operators (*see* **Note 22**). Each row in the genes worksheet represents one core metabolic gene and will include: (1) the gene name, (2) the Tn-seq data for the gene, and (3) the RNA-seq data for the gene (when provided).

3.5 Refining a GENRE (Local Version)

At the time or writing, refining a GENRE cannot be performed with the automated pipeline or on the Tn-Core web application, although this functionality may be added in the future.

3.5.1 Prepare the Workspace

1. Ensure all of the required software has been downloaded and ideally placed within the same folder on your machine.
2. Place all input files into a new folder (different from where the software files are located).
3. Open MATLAB and navigate to the directory containing the input data.
4. Ensure all the software are in your MATLAB path. This can be done with the following command, where '`/PATH/TO/SOFTWARE/`' is replaced with the path to the software folder:

```
addpath(genpath(/PATH/TO/SOFTWARE/));
```

5. Initialize the COBRA Toolbox with the following command:

```
initCobraToolbox;
```

6. Set the COBRA solver with the following command, where '`SOLVER`' is replaced with either '`gurobi`' or '`ibm_cplex`':

```
changeCobraSolver('SOLVER', 'all');
```

7. Initialize the TIGER Toolbox with the following command, where '`SOLVER `' is replaced with either '`gurobi`' or '`ibm_cplex`':

```
start_tiger('SOLVER');
```

3.5.2 Prepare the Input Data

1. Load the Tn-seq data with the following command:

```
tnseq = table2cell(readtable('TnSeqData.txt', ...
'ReadRowNames', false, 'ReadVariableNames', false, ...
'Delimiter', '\t'));
```

2. Load the GENRE with the following command (*see* **Note 13**):

```
model = readCbModel('inputModel.xml');
```

3. Ensure that the imported model contains the grRules and rxnGeneMat fields by running the following command that will add these fields if they are missing (*see* **Note 14**):

```
model = tncore_fix(model);
```

4. Set the objective function of the model and set the bounds of the exchange reactions using standard procedures and functions of the COBRA Toolbox (*see* **Note 13**).

3.5.3 Run Tn-Core

1. Run Tn-Core to identify potential errors in the gene-reaction associations using the following command:

```
[refinedModel, report] = tncore_refine(model, tnseq);
```

The above command will run Tn-Core with default parameters. Several parameters can be modified. Detailed information on available parameters can be obtained in MATLAB with the following command:

```
help tncore_refine
```

2. The MATLAB workspace can be saved with the following command (*see* **Note 15**):

```
save('tncore_output.mat');
```

3. Two output files are provided:

   A refined COBRA-formatted metabolic model. Two types of changes may be present compared to the input model. First, the gene-reaction associations will be updated by removing genes that can compensate for an essential gene; however, reactions will not be removed (*see* **Note 23**). Second, reactions producing dead-end metabolites will be removed from the model; however, this is an optional parameter that can be set to false.

A summary table of the changes made to the model and additional suggestions. This table will contain five columns: reaction identifier, type of change, the original and the new gene associations of the reaction, and an indication of whether the change was made in the model. Three types of changes are listed: (1) changed gene-reaction association, (2) dead-end removal, and (3) OR to AND transition in the gene-reaction associations (*see* **Note 24**). The latter class of changes is not present in the refined model, but they can be manually added by the user.

4 Notes

1. At the time of writing, the following software was used to run Tn-Core on the web-app, and we are therefore certain that this set-up is compatible with Tn-Core: Tn-Core v.2.2, MATLAB R2016b, libSBML v.5.13.0, SBML Toolbox v.4.1.0, iLOG CPLEX Studio v.12.7.1, COBRA Toolbox commit 9b10fa1, TIGER Toolbox v.1.2-beta, and FASTCORE v1.0.
2. Sample input files are provided at: github.com/diCenzo-GC/Tn-Core-webserver.
3. The Tn-seq and RNA-seq data files can contain data for genes not present in the GENRE, and these files do not have to contain data for all genes in the GENRE. Additionally, it is not necessary for all genes in the Tn-seq data file to be present in the RNA-seq data file, and vice versa. However, it is important to ensure that the gene naming convention in the Tn-seq, RNA-seq, and GENRE are the same (i.e., ensure the same version of the genome annotation has been used).
4. Tn-Core was initially written to handle Tn-seq data provided as the number of reads (or number of insertions) mapping to a gene normalized by the length of the gene. In this case, the data values are ≥ 0, and nonessential genes follow a normal distribution when log-transformed. Although we have not tested other types of data (i.e., change in number of insertions from the input to output populations), we expect that Tn-Core will work with any data set where (1) the minimum value is zero, (2) essential genes have a low value and nonessential genes have a high value, (3) nonessential genes follow a normal distribution, and (4) the gene lengths have been accounted for.
5. Freely available GENREs are now available for many prokaryotic organisms. They can be found through online repositories or as supplementary files to published manuscripts. Databases storing GENREs include: BiGG (bigg.ucsd.edu) [20], VMH (vmh.life) [21], and BioModels (ebi.ac.uk/biomodels) [22].

6. If a GENRE is not available for your organism of interest and you wish to try Tn-Core using a draft GENRE, various online applications exist for the rapid preparation of draft GENREs on the basis of a genome annotation. One platform with which we are familiar is the KBase web application (kbase.us) [23]. KBase provides an easy-to-use interface and good tutorials on the preparation of a GENRE, which can be prepared within minutes even with no knowledge of metabolic reconstruction methods.
7. Although GENREs of any quality can be used as input with Tn-Core, the quality of the output data is dependent on the quality of the starting GENRE. Thus, we recommend using a manually curated GENRE where possible. It is of particular importance to ensure that the biomass reaction is comprehensive. A pathway will only be identified as essential by Tn-Core if it is required for the production of one or more biomass components. For example, the biosynthesis of NAD^+ may be absent from the Tn-Core output, regardless of its essentiality in the Tn-seq data, if NAD^+ is not present in the biomass reaction.
8. If not using the automated pipeline, the GENRE can be directly provided to Tn-Core as a COBRA-formatted model. In this case, it is necessary to set the objective and to add all desired constraints (including exchange reactions) to the model prior to running Tn-Core.
9. Exchange reactions are used to define the composition of the growth medium. They can be thought of as setting the maximal rate (not concentration) that a nutrient becomes available to the cells. In order for a nutrient to be available to the cell (i.e., present in the growth medium), the lower bound must be a nonzero negative value. The exchange reactions listed in this file will be set to the indicated value. All other exchange reactions will have a lower bound set to zero (i.e., the corresponding nutrient will not be present in the medium), unless doing so results in the model being unable to produce biomass.
10. The closer the in silico medium (as determined by the active exchange reactions) resembles the growth medium used for the Tn-seq experiment, the better the output is likely to be. However, Tn-Core will function even if the in silico medium is either not fully defined or not defined at all. If only a few exchange reactions are indicated (i.e., setting only the carbon source), then a minimal set of additional exchange reactions required to support growth will be identified to turn on. Similarly, if no exchange reactions are indicated, Tn-Core will automatically select a minimal set of exchange reactions to turn on. Thus, if unsure of how to identify the correct

exchange reactions, an empty ExRxns.txt file can be provided to Tn-Core. However, we caution that in these cases, it is likely that the nutrients present in the in silico medium will differ from those present in the actual growth medium, which is likely to reduce the quality of the output data. Thus, we suggest that, at minimum, the correct carbon, nitrogen, phosphorus, and sulphur sources are indicated in the ExRxns.txt file.

11. The objective function is the reaction that should be optimized (i.e., flux through this reaction should be maximized) during simulations. In essence, the objective function can be considered as the goal of the model. When using Tn-Core, we expect that biomass production (i.e., growth of the cell) will always be the objective function. Thus, this file should contain the reaction identifier corresponding to the biomass reaction. Often, the biomass reaction contains the word 'Biomass' in the reaction identifier, but this is not always the case.
12. At the time of writing, the functionality of the Tn-Core web application is limited compared to the local version of Tn-Core. Over time, we expect to increase the capabilities of the web application. In the meantime, we suggest users to use the local version if they are looking for functionality not currently available on the web application, such as the refinement of GENREs.
13. If unsure how to set up the COBRA model, there are two alternatives. The first option is to import the model with the `tncore_import` function using the following command:

```
model = tncore_import();
```

In this case, it is necessary to include the input files of Subheading 2.1.3 in the same folder as the model. The second option is to instead run Tn-Core using the automated pipeline.

14. Currently, the Tn-Core Toolbox requires that models contain the grRules and rxnGeneMat fields to function. Depending on the version of the COBRA Toolbox being used, the imported model may or may not already contain these fields. If missing, the `tncore_fix` command will add these fields (based on the rules field); the `tncore_fix` command will do nothing if they already exist.
15. The saved workspace can be imported into MATLAB with the following command:

```
load('tncore_output.mat');
```

16. The following parameters of the overall pipeline are relevant to the contextualization of Tn-seq data: `solver`, `prepCore`, `tnseqData`, `rnaseqData`, `epsilon`, `binThresh`, `essThresh`, `growthFrac`, `expressThresh`, `keepHigh`, `deadends`.
17. For each additional parameter to be set, the user must indicate the parameter to be set followed by the desired value. For example, to run an analysis without RNA-seq data with an essentiality threshold (`essThresh`) of 4.5 instead of the default of 3.5, the following command can be used:

```
tncore_overall_workflow('essThresh', 4.5);
```

Similarly, to run an analysis with RNA-seq data, with an essentiality threshold (`essThresh`) of 4.5 instead of the default of 3.5, and with a growth fraction (`growthFrac`) of 0.7 instead of the default of 0.5, the following command can be used:

```
tncore_overall_workflow('rnaseqData', true, ...
'essThresh', 4.5,'growthFrac', 0.7);
```

When setting multiple parameters, their order in the command is not important.

18. When provided, RNA-seq data are used to guide the selection of which nonessential genes (as determined by Tn-seq) to include in the model. For example, if two genes encode functionally redundant proteins, and the reaction catalyzed by these proteins is essential for biomass production, only the gene with the higher expression is included in the output model. If no RNA-seq data are provided, one gene is chosen at random.
19. The output model will not necessarily contain all genes of the input model that were essential in the Tn-seq screen. By default, Tn-Core removes reactions producing dead-end metabolites; i.e., reactions are removed if they involve a metabolite that is not involved in any other reaction in the model. If an essential gene is associated with a reaction producing a dead-end metabolite (this could be caused, for example, by the reaction being involved in producing an essential biomass component that has not been included in the biomass reaction), then by default the gene will be removed by Tn-Core. When running the local version of Tn-Core, either with or without the automated pipeline, it is possible to change the settings to keep reactions producing dead-end metabolites.

However, at the time of writing, this is not possible on the Tn-Core webserver.

20. When running the Tn-Core web application, the COBRA-formatted model will be returned within a .MAT file that has to be imported to MATLAB for further use.
21. For further information on the data included in the output Excel file, please refer to: github.com/diCenzo-GC/Tn-Core-webserver.
22. A reaction can be associated with more than one gene. In these cases, the genes will be related using the Boolean operators 'OR' and/or 'AND'. An 'OR' statement means that only one of the genes need to be present for the reaction to function. For example, (GeneA or GeneB) means that deleting either GeneA or GeneB will have no effect, whereas deleting both will cause the reaction to no longer be present. An 'AND' statement means that both of the genes need to be present for the reaction to function (i.e., they may be subunits of a protein complex). For example, (GeneA and GeneB) means that deleting either GeneA or GeneB will cause the reaction to no longer be present. More complex associations may also be present, such as ((GeneA and GeneB) or GeneC).
23. Briefly, a core metabolic model is first produced on the basis of the Tn-seq data. Then, for reactions in the core model that contains a gene identified as essential in the Tn-seq data, the genes associated with the corresponding model in the input model are modified to match those in the core model. In essence, this process takes reactions with an essential gene (based on the Tn-seq data) and removes other genes from the reaction if they allow the gene to be nonessential (e.g., if the reaction has the gene [GeneA or GeneB] and only GeneA was essential in the Tn-seq data, GeneB will be removed from the reaction).
24. If two or more genes that are essential in the Tn-seq data are always associated with the same reactions in the model, and there are only 'OR' Boolean operators in the gene association (i.e., no 'AND' Boolean operators in the gene association), then a suggestion is made to replace the 'OR' operators with 'AND' operators.

References

1. van Opijnen T, Camilli A (2013) Transposon insertion sequencing: a new tool for systems-level analysis of microorganisms. Nat Rev Microbiol 11:435–442. https://doi.org/10.1038/nrmicro3033
2. Barquist L, Boinett CJ, Cain AK (2013) Approaches to querying bacterial genomes with transposon-insertion sequencing. RNA Biol 10:1161–1169. https://doi.org/10.4161/rna.24765

3. Kwon YM, Ricke SC, Mandal RK (2016) Transposon sequencing: methods and expanding applications. Appl Microbiol Biotechnol 100:31–43. https://doi.org/10.1007/s00253-015-7037-8
4. Wetmore KM, Price MN, Waters RJ et al (2015) Rapid quantification of mutant fitness in diverse bacteria by sequencing randomly bar-coded transposons. MBio 6: e00306–e00315. https://doi.org/10.1128/mBio.00306-15
5. Babu M, Arnold R, Bundalovic-Torma C et al (2014) Quantitative genome-wide genetic interaction screens reveal global epistatic relationships of protein complexes in *Escherichia coli*. PLoS Genet 10:e1004120. https://doi.org/10.1371/journal.pgen.1004120
6. diCenzo GC, Benedict AB, Fondi M et al (2018) Robustness encoded across essential and accessory replicons of the ecologically versatile bacterium *Sinorhizobium meliloti*. PLoS Genet 14:e1007357. https://doi.org/10.1101/209916
7. Blazier A, Papin JA (2018) Reconciling high-throughput gene essentiality data with metabolic network reconstructions. bioRxiv. https://doi.org/10.1101/415448
8. Burger BT, Imam S, Scarborough MJ et al (2017) Combining genome-scale experimental and computational methods to identify essential genes in *Rhodobacter sphaeroides*. mSystems 2:e00015–17. https://doi.org/10.1128/mSystems.00015-17
9. Thiele I, Palsson BØ (2010) A protocol for generating a high-quality genome-scale metabolic reconstruction. Nat Protoc 5:93–121. https://doi.org/10.1038/nprot.2009.203
10. Schellenberger J, Que R, Fleming RMT et al (2011) Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2.0. Nat Protoc 6:1290–1307. https://doi.org/10.1038/nprot.2011.308
11. Lewis NE, Nagarajan H, Palsson BØ (2012) Constraining the metabolic genotype-phenotype relationship using a phylogeny of in silico methods. Nat Rev Microbiol 10:291–305. https://doi.org/10.1038/nrmicro2737
12. Broddrick JT, Rubin BE, Welkie DG et al (2016) Unique attributes of cyanobacterial metabolism revealed by improved genome-scale metabolic modeling and essential gene analysis. Proc Natl Acad Sci U S A 113: E8344–E8353. https://doi.org/10.1073/pnas.1613446113
13. Bartell JA, Blazier AS, Yen P et al (2017) Reconstruction of the metabolic network of *Pseudomonas aeruginosa* to interrogate virulence factor synthesis. Nat Commun 8:14631. https://doi.org/10.1038/ncomms14631
14. Minato Y, Gohl DM, Thiede JM, et al (2019) Genome-wide assessment of Mycobacterium tuberculosis conditionally essential metabolic pathways. bioRxiv 534289. https://doi.org/10.1101/534289
15. diCenzo GC, Mengoni A, Fondi M (2019) Tn-Core: a toolbox for integrating Tn-seq gene essentiality data and constraint-based metabolic modelling. ACS Synth Biol 8:158–169. https://doi.org/10.1021/acssynbio.8b00432
16. Jensen PA, Lutz KA, Papin JA (2011) TIGER: toolbox for integrating genome-scale metabolic models, expression data, and transcriptional regulatory networks. BMC Syst Biol 5:147. https://doi.org/10.1186/1752-0509-5-147
17. Vlassis N, Pacheco MP, Sauter T (2014) Fast reconstruction of compact context-specific metabolic network models. PLoS Comput Biol 10:e1003424. https://doi.org/10.1371/journal.pcbi.1003424
18. Bornstein BJ, Keating SM, Jouraku A, Hucka M (2008) LibSBML: an API library for SBML. Bioinformatics 24:880–881. https://doi.org/10.1093/bioinformatics/btn051
19. Keating SM, Bornstein BJ, Finney A, Hucka M (2006) SBMLToolbox: an SBML toolbox for MATLAB users. Bioinformatics 22:1275–1277. https://doi.org/10.1093/bioinformatics/btl111
20. Schellenberger J, Park JO, Conrad TM, Palsson BØ (2010) BiGG: a biochemical genetic and genomic knowledgebase of large scale metabolic reconstructions. BMC Bioinformatics 11:213. https://doi.org/10.1186/1471-2105-11-213
21. Noronha A, Modamio J, Jarosz Y et al (2019) The virtual metabolic human database: integrating human and gut microbiome metabolism with nutrition and disease. Nucleic Acids Res 47:D614–D624. https://doi.org/10.1093/nar/gky992
22. Glont M, Nguyen TVN, Graesslin M et al (2018) BioModels: expanding horizons to include more modelling approaches and formats. Nucleic Acids Res 46:D1248–D1253. https://doi.org/10.1093/nar/gkx1023
23. Arkin AP, Cottingham RW, Henry CS et al (2018) KBase: The United States Department of Energy Systems Biology Knowledgebase. Nat Biotechnol 36:566–569. https://doi.org/10.1038/nbt.4163