A. iNAP 2.0: Metabolic Modeling Analysis

Here, we introduce the integrated Network Analysis Pipeline 2.0 (iNAP 2.0), which features an innovative metabolic complementarity network for microbial studies from metagenomics sequencing data. iNAP 2.0 sets up a 4-module process for metabolic interaction analysis, namely: (I) Prepare genome-scale metabolic models; (II) Infer pairwise interactions of genome-scale metabolic models; (III) Construct metabolic interaction networks; (IV) Analyze metabolic interaction networks. Starting from metagenome-assembled or complete genomes, iNAP 2.0 offers a variety of methods to quantify the potential and trends of metabolic complementarity between models, including the PhyloMint pipeline based on phylogenetic distance-adjusted metabolic complementarity, the SMETANA approach based on cross-feeding substrate exchange prediction, and metabolic distance calculation based on parsimonious flux balance analysis. Notably, iNAP 2.0 integrates the random matrix theory (RMT) approach to find the suitable threshold for metabolic interaction network construction. Finally, the metabolic interaction networks can be sent for analysis using topological feature analysis such as hub node determination.

If you find this pipeline useful for your research and you want to cite it, refer to this:
Peng, Xi, Kai Feng, Xingsheng Yang, Qing He, Bo Zhao, Tong Li, Shang Wang, and Ye Deng. 2024. " iNAP 2.0: Harnessing metabolic complementarity in microbial network analysis." *iMeta* e235. https://doi.org/10.1002/imt2.235

Encounter any problems? Contact its maker: Xi Peng (emmettpengxi@hotmail.com)
Any bug report is welcomed!

0. Input File Requirements

Zipped genome sets (.zip). The zipped genome set contains all genome sequence files (.fasta/.fa) to be analyzed. Ensure all sequence files are directly compressed rather than stored in folders and then compressed. Each sequence file name should be unique, not starting with numbers, and not contain spaces, hyphens, or other special characters (underscore is recommended). If genome sets are planned for SMETANA analysis, the number of genome files should not exceed 300 due to SMETANA's high consumption of computational resources.

Prokka predicted protein sequences (.zip). The predicted protein set contains all protein sequence files (.faa) corresponding to the genomes. This file can be obtained using the Prokka tool or protein sequence files already

obtained or downloaded from reference databases. Compression and naming requirements are the same as above.

Growth medium for gap-filling (.txt/.tabular). When using CarveMe (gap filling) in Step 2-B, users can upload customized media in addition to the default five media. The media description file should contain four columns: medium, description, compound, and name. Note that compound names and IDs must be consistent with the BiGG database (http://bigg.ucsd.edu/, [13]).

1. Prepare genome-scale metabolic models

1.1 **[Step 1] Prokka: Rapid prokaryotic genome annotation.** iNAP 2.0 utilizes Prokka with default settings for genome annotation. Alternatively, users can employ tools like Prodigal or EGGNOG-mapper for this step.
Input file: Zipped genome sets (.zip)
Parameter(s): 1. Genome file extension: fasta (default), fa, fna
Output file: 1. The output protein sequence file (Prokka_faa.zip); 2. The log file (Prokka_log.txt)

1.2 **[Step 2-A] CarveMe.** iNAP 2.0 offers CarveMe, a fast and automated tool for building GSMMs. The output format (sbml-fbc2) ensures compatibility with most constraint-based modeling tools. In the GSMM reconstructed by CarveMe, the upper and lower bounds of the reaction flux (ready for flux balance analysis) directly call the default values in Cobra (*cobra_default_ub, cobra_default_lb*), which are 1000 and -1000 mmol/gDW/h.
Input file: Prokka predicted protein sequences (.zip)
Parameter(s): N/A
Output file: 1. CarveMe models (CarveMe_models.zip); 2. The log file (carveme_log.txt)

1.3 **[Step 2-B] CarveMe (gap filling).** We recommend using the gap-filling function to correct the model derived from metagenome-assembled genomes (MAGs) of environmental metagenomes. CarveMe provides five pre-defined media compositions for gap filling (default: Lysogeny broth, LB), allowing growth simulation of the model on corresponding media. Users can also create and utilize custom media compositions, such as dietary components for gut microbiome models. **It is important to note that defining media for environmental microbiomes can be challenging due to the difficulty of culturing most microorganisms and the limitations of rich media. Using rich media for gap-filling can ease over-gap-filling.**
Input file: 1. Prokka predicted protein sequences (.zip); 2. Your own media library (.tsv) [If you choose to provide your own media library for gap filling]
Parameter(s): 1. Source of media composition library (selection); 2. Growth media for gap filling (No matter whether you use pre-built libraries or your own uploaded library, do fill in the media name.

Output file: 1. CarveMe models (CarveMe_models.zip); 2. The log file (carveme_log.txt)

2. Infer pairwise interactions of genome-scale metabolic models

2.1 PhyloMint

2.1.1 [Step 3] PhyloMint. The program calculates $MI_{complementarity}$/$MI_{competition}$ between n input GSMMs (output from Step 2-A/B), resulting in n^2 sets of indices calculated. The parameter *MaxCC* indicates the maximum number of members in a strongly connected component (recommended set to default: 5). *MaxCC* only influences the results of competition index (See the calculation formula).

Input file: Zipped CarveMe models (.zip)

Parameter(s): MaxCC (default: 5)

Output file: PhyloMint general result (PhyloMint_result.txt)

2.1.2 [Step 4] PhyloMint PTM. $MI_{complementarity}$ index, as noted earlier, represents A's potential to utilize metabolic substances from B. Potentially transferable metabolites (PTMs) are defined as the intersection of A's seed set and B's non-seed set, as per in the calculation formula of $MI_{complementarity}$ index. This step takes in the zipped model set (output of Step 2-A/B) and PhyloMint result (output of Step 3) as inputs and outputs these substances in a tabular file, displaying the donor and receptor of each compound along with their nomenclature and full index in the BiGG database. Note that if your GSMMs are obtained elsewhere instead of generated by CarveMe, the file extension might be. sbml (Systems Biology Markup Language). Remember to change the extension setting in this step.

Input file: 1. Zipped CarveMe Models (.zip); 2. PhyloMint general result (.txt)

Parameter(s): 1. Model file extention: xml (extensible markup language), sbml (systems biology markup language); 2. MaxCC (Must be equal to this time's PhyloMint MaxCC)

Output file: 1. PTM list (PhyloMint_PTMs.txt); 2. The log file (getPTM_log.txt)

2.1.3 [Step 5] Create PhyloMint Matrix. This step converts the result generated by Step 3 from a tabular form to a matrix form for network threshold determination. In particular, we provide two processing modes for the asymmetric *MI* index: 1) Keep the original value, that is, directly convert the results of Step 3 into an asymmetric matrix, and then the metabolic interaction network constructed using this matrix will be directed; 2) Select the larger value in the pairwise index of each pair of models to represent the interaction strength between the models, so that the generated matrix is symmetric, and the metabolic interaction network constructed using it will be undirected, which we set to default and recommend for better using the RMT-based method for network threshold determination.

Input file: PhyloMint general result (.txt)

Parameter(s): For asymmetric index: Keep the original value or take the maximum value?

Output file: 1. Adjacent matrix of competition index (PhyloMint_Competition_Matrix.txt); 2. Adjacent matrix of complementarity index (PhyloMint_Complementarity_Matrix.txt)

2.2 SMETANA

2.2.1 [Step 6] Create Community List. To calculate the interaction between pairwise GSMMs, iNAP 2.0 provides the required table of all pairwise GSMMs in a given model set (output of Step 2-A/B) by default,. Actually, SMETANA can calculate scores for communities of more than 2 species (models). You can follow the instructions on SMETANA page and calculate multi-member community's SMETANA score using Step 7 and 8 as well.

Input file: Zipped CarveMe Models (.zip)

Parameter(s): N/A

Output file: Community list (community_list.txt)

2.2.2 [Step 7-A] SMETANA Global. For a given community, SMETANA defines two indices, MRO (metabolic resource overlap) and MIP (metabolic interaction potential), to represent the competition and complementarity levels of the community, respectively.

Input file: 1. Zipped CarveMe Models (.zip); 2. Community list

Parameter(s): N/A

Output file: SMETANA result of global mode (smetana_result.txt)

2.2.3 [Step 7-B] Iterative SMETANA Global. SMETANA has been confirmed by its developers to have a drawback: the results of each run may vary slightly. This inconsistency is due to the solution pool feature of the CPLEX solver. The developers recommend running multiple runs and calculating the average index value to represent the final value. To achieve this, the program is designed to run 2-10 runs of the same input as Step 7-A and output the average results.

Input file: 1. Zipped CarveMe Models (.zip); 2. Community list

Parameter(s): Number of repeats (2-10)

Output file: 1. Several SMETANA results of global mode in a zipped file (smetana_results.zip); 2.Average SMETANA results of global mode (smetana_results_average.txt)

2.2.4 [Step 8-A] SMETANA Detailed. In addition to using MRO/MIP to quantify the metabolic interactions of the community, SMETANA also provides a detailed mode to calculate a series of indices to quantify further the interspecies interactions: SCS (species coupling score), metabolite uptake score (MUS), and metabolite production score (MPS). These three indices are combined and recorded as the SMETANA score to represent the sum of interspecies dependencies in the community.

Input file: 1. Zipped CarveMe Models (.zip); 2. Community list

Parameter(s): N/A

Output file: SMETANA result of detailed mode (smetana_result.txt)

2.2.5 [Step 8-B] Iterative SMETANA Detailed.

Input file: 1. Zipped CarveMe Models (.zip); 2. Community list

Parameter(s): Number of repeats (2-10)

Output file: 1. Several SMETANA results of detailed mode in a zipped file (smetana_results.zip); 2.Average SMETANA results of detailed mode (smetana_results_detailed_average.txt)

2.2.6 [Step 9] Create SMETANA Matrix. This step converts the SMETANA MIP/MRO results from a tabular to two matrices.

Input file: 1. Community list (.txt); 2. SMETANA result of global mode

Parameter(s): value to replace n/a with (there might still be n/a in the result even though using repeat runs, default: 0.1).

Output file: 1. Adjacent matrix of MRO (smetana_mro_adjacent_matrix); 2. Adjacent matrix of MIP (smetana_mip_adjacent_matrix)

2.3 Metabolic distance

2.3.1 [Step 10] Metabolic distance. This program helps to calculate the pairwise metabolic distances of GSMMs. Specifically, the program first conducts flux balance analysis (FBA) on each model with the biomass reaction as the objective function to optimize (maximize) the biomass reaction flux (e.g., growth rate or ATP yield). By default, iNAP 2.0 fills in "Growth", representing the growth rate reaction in GSMMs generated by CarveMe. Then, the optimized biomass reaction flux is fixed, and a second parsimonious FBA (pFBA) is conducted to minimize the sum of absolute flux in each model. Subsequently, the reactions whose flux is not zero in at least one model are selected as representatives, the flux vectors of the models are generated, and the Euclidean distances between them are calculated as proxies of metabolic distances.

Input file: Zipped CarveMe Models (.zip)

Parameter(s): Biomass reaction ID (default for CarveMe models: Growth)

Output file: 1. Table of metabolic distance (metabolic_distance_tabular.txt); 2. Adjacent matrix of metabolic distance (metabolic_distance_matrix.txt); 3. Adjacent matrix of Standardized Euclidean version of metabolic distance (metabolic_distance_matrix_standardized.txt)

3. Construct metabolic interaction networks

3.1 [Step 11] Random Matrix Theory (cutoff, Chi-square test). **See A.4.3.**

3.2 [Step 12] Random Matrix Theory (cutoff, Kolmogorov-Smirnov test). This step and Step 11 use the RMT-based method to determine the network threshold. The chi-square test is utilized in Step 11, and the Kolmogorov-Smirnov test is used in this step. Compared with the chi-square test, the Kolmogorov-Smirnov test

is expected to give a more relaxed threshold, which may be more practical when dealing with values like the $MI_{competition}$ and $MI_{complementarity}$ indices.

Input file: Adjacent Matrix

Parameter(s): N/A

Output file: RMT result (RMThreshold result)

3.3 [Step 13] Z-score outlier detection. iNAP 2.0 provides a Z-score outlier detection method for the adjacent matrix to filter interactions for constructing the networks.

2.4 [Step 14] Construct Network Adjacent Matrix. See A.4.4.

4. Analyze metabolic interaction networks

4.1 [Step 15] Global network properties and individual nodes' centrality.

4.2 [Step 16] Module separation and module hubs.

4.3 [Step 17] Integrate node attributes. The node centrality (output of Step 15), the node between- and among-module connectivity (z_i-P_i value and roles in the network, output of Step 16), and the taxonomic annotation of the node (optional and uploaded by users, tabular-separated .txt file with first column as node IDs, referred to example file in iNAP 2.0) are essential components of the microbial network information. This step can merge the above three files for subsequent analysis or as an annotation file for visualization.

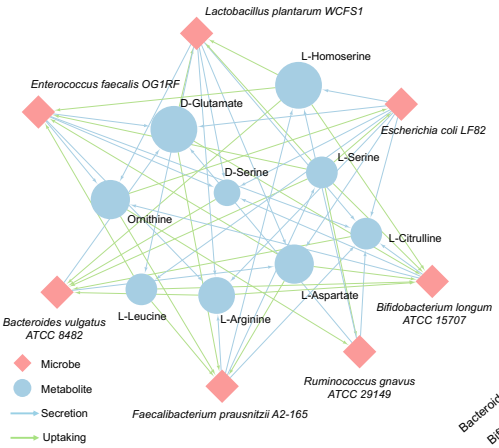
4.4 [Step 18] Network intersection. This step compares the adjacent matrices of two networks and finds the subnetworks composed of their common edges. It then outputs the intersected network with an adjacent matrix and edge list of the subnetwork.

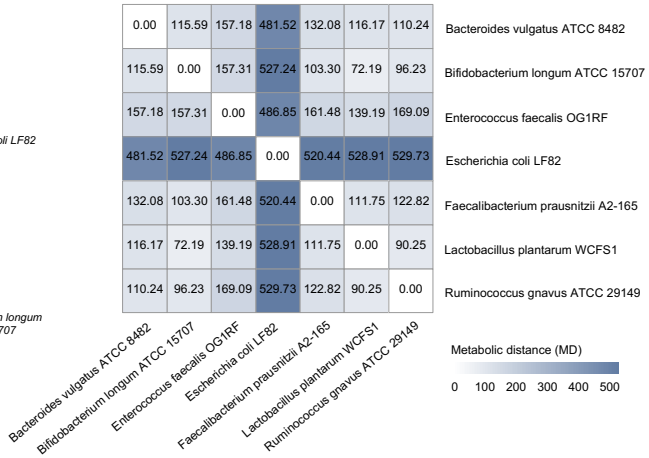
4.5 [Step 19] Visualize the PTM network. This step uses the output of Step 5 as input to transform the potentially transferable metabolites in the specific network into a directed bipartite microbe-metabolite network. All metabolite annotations stored in the input are integrated and output in a node attribute table.

Example Figure: (A)

4.6 [Step 20] Metabolic distance heatmap. This step generates a heatmap using the metabolic distance matrix (output of Step 10), which allows for the significant identification of distinctive metabolic profiles in microbial consortiums.

Example Figure: (B)


Lactobacillus plantarum WCFS1
L-Homoserine
Enterococcus faecalis OG1RF
D-Glutamate
Escherichia coli LF82
L-Serine
D-Serine
Ornithine
L-Citrulline
L-Aspartate
Bacteroides vulgatus ATCC 8482
L-Leucine
L-Arginine
Bifidobacterium longum ATCC 15707
Ruminococcus gnavus ATCC 29149
Faecalibacterium prausnitzii A2-165
Microbe
Metabolite
Secretion
Uptaking


0.00 115.59 157.18 481.52 132.08 116.17 110.24 Bacteroides vulgatus ATCC 8482
115.59 0.00 157.31 527.24 103.30 72.19 96.23 Bifidobacterium longum ATCC 15707
157.18 157.31 0.00 486.85 161.48 139.19 169.09 Enterococcus faecalis OG1RF
481.52 527.24 486.85 0.00 520.44 528.91 529.73 Escherichia coli LF82
132.08 103.30 161.48 520.44 0.00 111.75 122.82 Faecalibacterium prausnitzii A2-165
116.17 72.19 139.19 528.91 111.75 0.00 90.25 Lactobacillus plantarum WCFS1
110.24 96.23 169.09 529.73 122.82 90.25 0.00 Ruminococcus gnavus ATCC 29149
Bacteroides vulgatus ATCC 8482
Bifidobacterium longum ATCC 15707
Enterococcus faecalis OG1RF
Escherichia coli LF82
Faecalibacterium prausnitzii A2-165
Lactobacillus plantarum WCFS1
Ruminococcus gnavus ATCC 29149
Metabolic distance (MD)
0 100 200 300 400 500