Integrative COVID-19 Biological Network Inference with Probabilistic Core Decomposition: Supplementary Methods

YANG GUO, FATEMEH ESFAHANI, XIAOJIAN SHAO, VENKATESH SRINIVASAN, ALEX THOMO, LI XING, XUEKUI ZHANG

1. SARS-COV-2 AND HOST PROTEIN-PROTEIN INTERACTION NETWORK EXTENSION

In this section we describe the procedure of extending the SARS-CoV-2-host protein-protein interaction (PPI) network identified by Gordon et al. [1] with the *human* Biomine database. And we also introduce the pre-processing steps for duplicated entries and loops in the extended dataset.

Each entry in the *human* Biomine database possesses the form:

$$\mathcal{E} = (from, to, relationship, link\_goodness)$$

A small sample of the *human* Biomine database is presented in Table S1. For edges in the PPI network, we set *relationship* to be *COVID_before_stage1*, meaning they are predefined SARS-CoV-2 protein interactions; *link_goodness* is set to be 1.0 representing the edge as known with full confidence. For the *human* organism set, if two entries have the same $from$ and *to* nodes but with different *relationship* and *link_goodness*, we regard them as (semi-)duplicates and merge them into a single entry with the second *relationship* appended after the first and the new *link_goodness* being the average of the two previous *link_goodness* value.

We now append the formatted PPI network after the *human* Biomine data. This operation will surely introduce new duplicated entries in the merged dataset. We will re-do the duplicate removal operation on the new dataset. However, when merging the newly found duplicates, the new *link_goodness* will be the maximum *link_goodness* between the two to account for the 100% confidence edges in the PPI network. After duplicate removal, an additional loop removal procedure will be applied. Currently, our peeling algorithm (PA) can only operate on undirected graphs, hence any two edges with opposite direction (forming a loop) will be regarded as the same and processed. The maximum *link_goodness* between the two edges will be assigned to the processed edge. The idea of duplicate removal and loop removal is illustrated in Figure S1.

Table S1. *human* Biomine database sample

from	*to*	*relationship*	*link_goodness*
Gene_EntrezGene:10144	BiologicalProcess_GO:GO:0051056	participates_in	0.220
Gene_EntrezGene:10144	Gene_EntrezGene:23071	interacts_with	0.581
Gene_EntrezGene:10144	Protein_UniProt:O94988	codes_for	1.000
Protein_STRING:ENSP00000473166	Organism_TAXON:9606	belongs_to	0.066
Protein_STRING:ENSP00000473166	Protein_UniProt:Q01892	functionally_associated_to	0.061

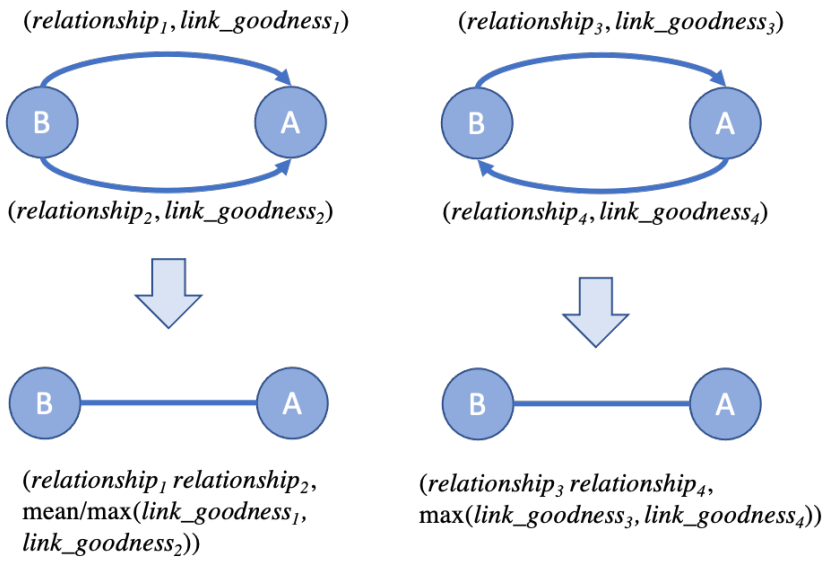

Fig. S1. Remove duplicated and looped edges

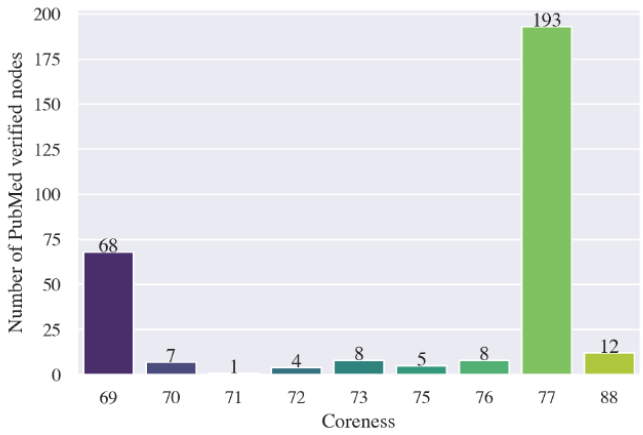

Fig. S2. Distribution of coreness in the list of PubMed verified nodes

Algorithm S1. Data analysis pipeline

Notation:

v, $v\_current$ denote vertex in the network

Input:

User-defined threshold $threshold_1$ based on degree expectation;
User-defined threshold $threshold_2$ based on η-degree lower-bounds;
User-defined parameter η for peeling algorithm;
The extended dataset (denote as *network*): made from Biomine probabilistic database and the original PPI network;
List of experimentally validated nodes, *ground_truth*, in the *network*.

Algorithm:

$nodelist \leftarrow$ list of nodes in *network*
for all $v \in ground\_truth$ **do**
  $level\_1\_connections \leftarrow$ find nodes directly connected to the experimentally validated nodes
$retaining\_nodelist \leftarrow ground\_truth.append(level\_1\_connections)$ ▷ list of all nodes to be retained: ground truth + level 1 connections
$to\_filter\_nodelist \leftarrow set(nodelist) - set(retaining\_nodelist)$ ▷ list of nodes to be passed into data screening steps
$degree\_expectation \leftarrow \{\}$ ▷ empty hash table
for all $v \in to\_filter\_nodelist$ **do**
  $current\_vertex\_edge\_prob \leftarrow [\,]$ ▷ empty array
  $current\_vertex\_edge\_prob \leftarrow$ list of edge probabilities for all edges incident to v
  $degree\_expectation[v] \leftarrow sum(current\_vertex\_edge\_prob)$ ▷ compute degree expectation of v
for all $v \in degree\_expectation.keys()$ **do**
  if $degree\_expectation[v] < threshold_1$ **then**
    delete $degree\_expectation[v]$ ▷ delete v from hash table keys
$\eta\_degree\_LB \leftarrow \{\}$ ▷ η-degree lower-bounds
for all $v \in degree\_expectation.keys()$ **do**
  $\eta\_degree\_LB[v] \leftarrow CLT(v)$ ▷ compute initial η-degree lower-bound of v based on the algorithm proposed in [2]
for all $v \in \eta\_degree\_LB.keys()$ **do**
  if $\eta\_degree\_LB[v] < threshold_2$ **then**
    delete $\eta\_degree\_LB[v]$
$alive\_nodelist \leftarrow \eta\_degree\_LB.keys()$
$final\_nodelist \leftarrow alive\_nodelist.append(retaining\_nodelist)$
$final\_nodelist\_coreness \leftarrow \{\}$
while $len(final\_nodelist) \neq 0$ **do** ▷ peeling algorithm
  $v\_current \leftarrow$ find the vertex $v\_current$ with the smallest η-degree computed by DP algorithm [2]
  $final\_nodelist\_coreness[v\_current] \leftarrow \eta$-degree of $v\_current$ ▷ assign the core number of $v\_current$ to be equal to its η-degree
  $final\_nodelist.remove(v\_current)$ ▷ removing vertex $v\_current$
  recompute the η-degree of $v\_current$'s neighbours
$dense\_subgraph \leftarrow$ identify high-activity subgraph based on $final\_nodelist\_coreness$ and connections to *ground_truth*
functional enrichment analysis on selected nodes ← clusterProfiler, Metascape, DAVID

Output:

Enriched GO terms (clusterProfiler);
Enriched pathway and process terms (Metascape);
Enriched clusters from functional annotation clustering (DAVID);
Generated hypotheses (proteins, genes, tissues, diseases, etc.).

Table S2. Node types in core 69 (sorted by node count)

Node type	Count
UniProt indexing nodes (prefix: 'Protein_UniProt')	3485
STRING indexing nodes (prefix: 'Protein_STRING')	113
PubMed indexing nodes (prefix: 'Article_PubMed')	8
UniProt_Tissue indexing nodes (prefix: 'Tissue_UniProt/tissue')	7
Gene Ontology (GO) Molecular Function indexing nodes (prefix: 'MolecularFunction_GO')	4
GO Cellular Component indexing nodes (prefix: 'CellularComponent_GO')	4
GO Biological Process indexing nodes (prefix: 'BiologicalProcess_GO')	1
InterPro Domain indexing nodes (prefix: 'Domain_InterPro')	1
TAXON Organism indexing nodes (prefix: 'Organism_TAXON')	1
InterPro Homologous Superfamily indexing nodes (prefix: 'HomologousSuperfamily_InterPro')	1

Each type of nodes has specific prefix to its name (listed in parentheses).

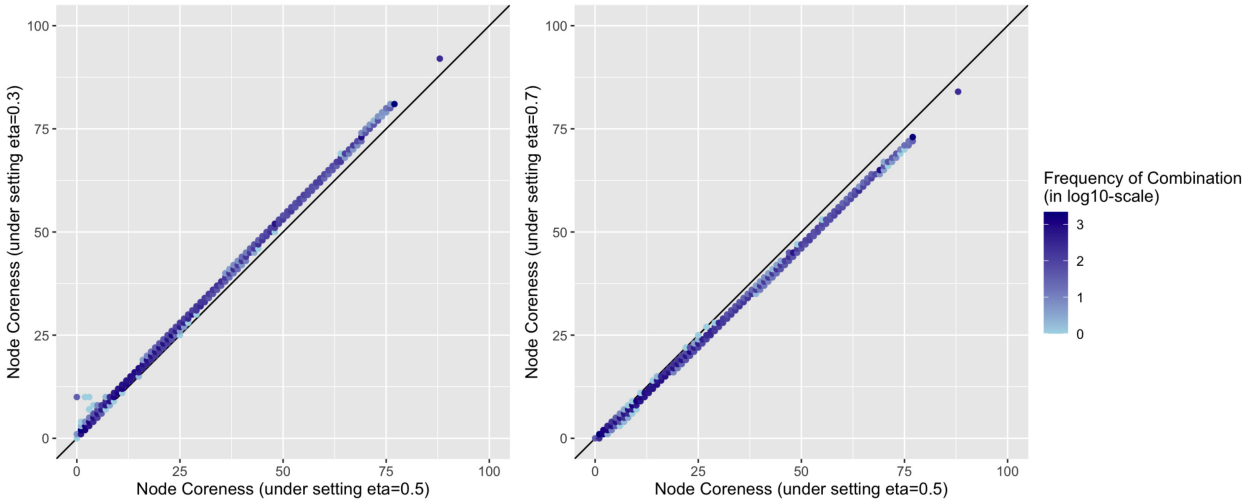

Fig. S3. Comparison for coreness results in different settings of η

REFERENCES

1. D. E. Gordon, G. M. Jang, M. Bouhaddou, J. Xu, K. Obernier, K. M. White, M. J. O'Meara, V. V. Rezelj, J. Z. Guo, D. L. Swaney *et al.*, "A SARS-CoV-2 protein interaction map reveals targets for drug repurposing," Nature **583**, 459–468 (2020).
2. F. Esfahani, V. Srinivasan, A. Thomo, and K. Wu, "Efficient computation of probabilistic core decomposition at web-scale," in *Proceedings of the 22nd International Conference on Extending Database Technology (EDBT)*, (2019), pp. 325–336.