Determining MinSpan vectors of COBRA model

Author: Aarash Bordbar.

Affiliation: Sinopia Biosciences, San Diego, CA USA

Reviewer: James T. Yurkovich. Catherine Clancy, LCSB, University of Luxembourg.

INTRODUCTION

In this tutorial, we show how to calculate MinSpan vectors [1] for a COBRA model. COBRA models are predominantly studied under steady-state conditions, thus the null space of the **S** matrix is of high interest. MinSpan vectors represent the sparsest linear basis of the null space **S** while still maintaining the biological and thermodynamic constraints of the COBRA model (Figure 1). The **S** matrix has dimensions (**m** x **n**) where **m** is the number of metabolites and **n** is the number of reactions. The linear basis of the null space (**N**) has dimensions (**n** x **n-r**) where **r** is the rank of **S**. Thus this algorithm calculates **n-r** vectors that are linearly independent of each other and also are minimal. For further info on MinSpan, it's derivation, implementation, and uses, see Bordbar et al. 2014 [1].

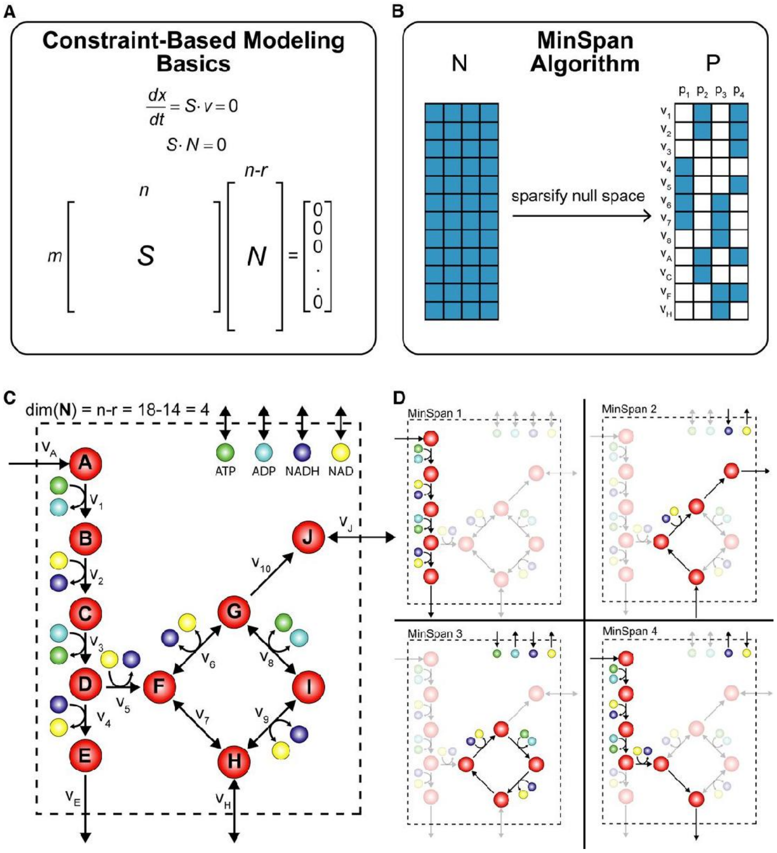


Figure 1 | Overview of the MinSpan algorithm. (A) A metabolic network is mathematically represented as a stoichiometric matrix (**S**). Reaction fluxes (**v**) are determined assuming steady state. All potential flux states lie in the null space (**N**). (B) The MinSpan algorithm determines the shortest, independent pathways of the metabolic network by decomposing the null space of the stoichiometric matrix to form the sparsest basis. (C) A simplified model for glycolysis and the TCA cycle is presented with 14 metabolites, 18 reactions, and a 4-dimensional null space. Reversible reactions are shown. (D) The four pathways calcuIed by MinSpan for the simplified model are presented, two of which recapitulate glycolysis and the TCA cycle, while the other two represent other possible metabolic pathways. The flux directions of a pathway through reversible reactions are shown as irreversible reactions.

MATERIALS

Initialize The COBRA Toolbox.

Initialize The Cobra Toolbox using the `initCobraToolbox` function.

```
initCobraToolbox
```

```
     _____   _____   _____   _____     _____     |
    /  ___| /  _  \ |  _  \ |  _  \   / ___ \    |   COnstraint-Based Reconstruction and Analysis
    | |     | | | | | |_| | | |_| |  | |___| |   |   The COBRA Toolbox - 2017
    | |     | | | | |  _  { |  _  /  |  ___  |   |
    | |___  | |_| | | |_| | | | \ \  | |   | |   |   Documentation:
    \_____| \_____/ |_____/ |_|  \_\ |_|   |_|   |   http://opencobra.github.io/cobratoolbox
                                                 |

> Checking if git is installed ...  Done.
> Checking if the repository is tracked using git ...  Done.
> Checking if curl is installed ...  Done.
> Checking if remote can be reached ...  Done.
> Initializing and updating submodules ... Done.
> Adding all the files of The COBRA Toolbox ...  Done.
> Define CB map output... set to svg.
> Retrieving models ...   Done.
> TranslateSBML is installed and working properly.
> Configuring solver environment variables ...
  - [----] ILOG_CPLEX_PATH :  --> set this path manually after installing the solver ( see instructions
  - [----] GUROBI_PATH :  --> set this path manually after installing the solver ( see instructions )
  - [---*] TOMLAB_PATH: C:\Program Files\tomlab\
  - [----] MOSEK_PATH :  --> set this path manually after installing the solver ( see instructions )
  Done.
> Checking available solvers and solver interfaces ... Done.
> Setting default solvers ... Done.
> Saving the MATLAB path ... Done.
  - The MATLAB path was saved in the default location.

> Summary of available solvers and solver interfaces

    Support           LP   MILP     QP   MIQP    NLP
----------------------------------------------------------------------
cplex_direct  active             0      0      0      0      -
dqqMinos      active             0      -      -      -      -
glpk          active             1      1      -      -      -
gurobi        active             1      1      1      1      -
ibm_cplex     active             0      0      0      -      -
matlab        active             1      -      -      -      1
mosek         active             0      0      0      -      -
pdco          active             1      -      1      -      -
quadMinos     active             0      -      -      -      0
tomlab_cplex  active             1      1      1      1      -
qpng          passive            -      -      1      -      -
tomlab_snopt  passive            -      -      -      -      1
gurobi_mex    legacy             0      0      0      0      -
lindo_old     legacy             0      -      -      -      -
lindo_legacy  legacy             0      -      -      -      -
lp_solve      legacy             1      -      -      -      -
opti          legacy             0      0      0      0      0
----------------------------------------------------------------------
Total         -                  6      3      4      2      2
```

```
+ Legend: - = not applicable, 0 = solver not compatible or not installed, 1 = solver installed.


> You can solve LP problems using: 'glpk' - 'gurobi' - 'matlab' - 'pdco' - 'tomlab_cplex' - 'lp_solve'
> You can solve MILP problems using: 'glpk' - 'gurobi' - 'tomlab_cplex'
> You can solve QP problems using: 'gurobi' - 'pdco' - 'tomlab_cplex' - 'qpng'
> You can solve MIQP problems using: 'gurobi' - 'tomlab_cplex'
> You can solve NLP problems using: 'matlab' - 'tomlab_snopt'

> Checking for available updates ...
--> You cannot update your fork using updateCobraToolbox(). [9b5f12 @ tut_minSpan].
    Please use the MATLAB.devTools (https://github.com/opencobra/MATLAB.devTools).
```

Equipment Setup

Running the MinSpan algorithm requires the installation of a mixed-integer linear programming (MILP) solver. We have used Gurobi v5+ (http://www.gurobi.com/downloads/download-center) which is freely available for academic use. This tutorial and the algorithm has been rigorously tested and support Gurobi v5+. `detMinSpan` will not work with GLPK; other solvers are untested. For detailed information on installation of Gurobi , refer to The Cobra Toolbox solver instalation guide.

Implementation

For MinSpan vectors to be calculated, the model must (1) consist of only reactions that are able to carry flux under that particular condition, (2) allow for the trivial solution (**v** = 0) to be feasible, and (3) have the biomass function removed. `detMinSpan` will automatically check and complete the first two modifications, but the biomass must be removed manually.

The algorithm is an iterative pruning of null space basis vectors to the sparsest possible matrix. The problem is NP-hard, meaning that an optimal solution is not guaranteed for large COBRA models; an approximate solution is found by setting a time limit on the MILP calculation.

Procedure

In this example, we will calculate the MinSpan vectors for the *E. coli* core network.

Ensure that the Gurobi MILP and LP solvers are working:

```
test1 = changeCobraSolver('gurobi', 'MILP');
```

```
> Gurobi interface added to MATLAB path.
```

```
test2 = changeCobraSolver('gurobi', 'LP');
```

```
> Gurobi interface added to MATLAB path.
```

```
if test1 == 0 | test2 == 0
    error('Gurobi v5+ not detected');
end
```

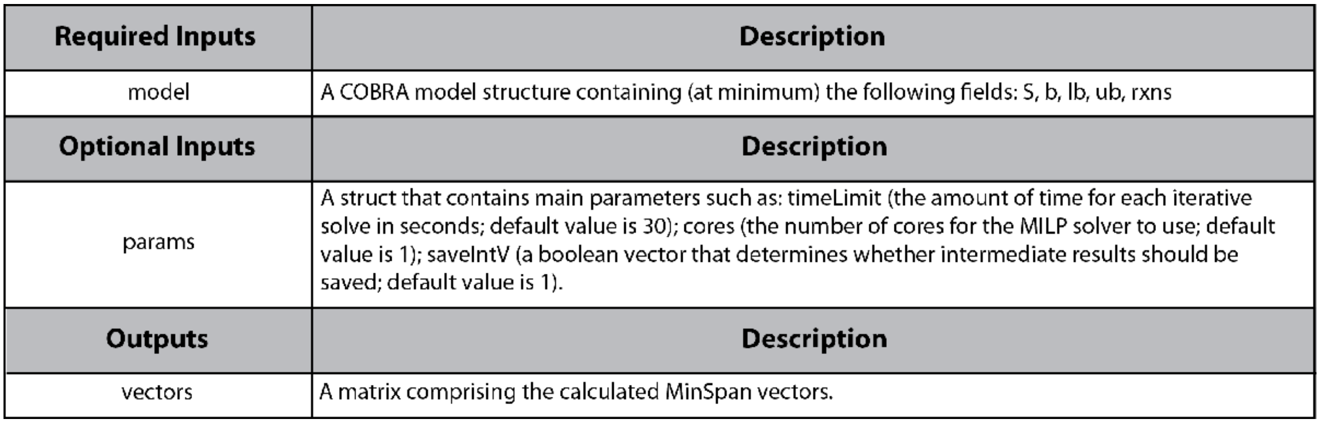
Load the core model:

```
load('ecoli_core_model.mat', 'model');
```

The biomass function is then removed from the model using the COBRA function `removeRxns`.

```
bmName = {'Biomass_Ecoli_core_w_GAM'};
model = removeRxns(model, bmName);
```

The MinSpan algorithm takes the model as input and returns a matrix containing the calculated MinSpan vectors (Table 1).



Required Inputs	Description
model	A COBRA model structure containing (at minimum) the following fields: S, b, lb, ub, rxns
Optional Inputs	**Description**
params	A struct that contains main parameters such as: timeLimit (the amount of time for each iterative solve in seconds; default value is 30); cores (the number of cores for the MILP solver to use; default value is 1); saveIntV (a boolean vector that determines whether intermediate results should be saved; default value is 1).
Outputs	**Description**
vectors	A matrix comprising the calculated MinSpan vectors.

Table 1 | Inputs and outputs of the `detMinSpan` function.

Running the algorithm on the modified *E. coli* core model returns the calculated MinSpans for the network:

```
tic
minSpanVectors = detMinSpan(model);
```

```
Starting parallel pool (parpool) using the 'local' profile ...
connected to 4 workers.
```

```
toc
```

```
Elapsed time is 0.138651 seconds.
```

`minSpanVectors` is a matrix that consists of 23 linearly independent vectors. A further description of these vectors is provided and their comparison to Extreme Pathways [2] is provided in the Supplementary Material of Bordbar et al 2014 [1] (Section 1 and Figure S2).

Numerical properties of the stoichiometric matrix:

```
[nMet,nRxn]=size(model.S)
```

```
nMet = 72
nRxn = 94
```

Independent dimensions of the right nullspace of the stoichiometric matrix

```
fprintf('%s%g\n','Number of right nullspace basis vectors expected: ',nRxn-rank(full(model.S))
```

```
Number of right nullspace basis vectors expected: 27
```

Numerical properties of the minSpan basis:

```
[nRxn2,nMinSpanVectors]=size(minSpanVectors)
```

```
nRxn2 = 94
nMinSpanVectors = 23
```

Rank of the minSpanVectors

```
fprintf('%s%g\n','Rank of minSpanVectors matrix:',rank(full(minSpanVectors)))
```

```
Rank of minSpanVectors matrix:23
```

Check the minSpan is really a basis for the nullspace:

```
fprintf('%s%g\n','Should be zero: ',norm(model.S*minSpanVectors))
```

```
Should be zero: 7.10218e-10
```

```
fprintf('%s%g\n','Should be zero (?): ',norm(nRxn-rank(full(model.S))-rank(full(minSpanVectors
```

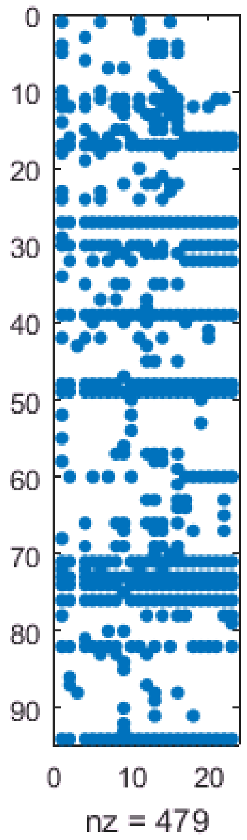
```
Should be zero (?): 4
```

Investigate the sparsity pattern of the minSpan basis:

```
fprintf('%s%g\n','Sparsity ratio of minSpanVectors: ',nnz(minSpanVectors)/(nRxn2*nMinSpanVecto
```

```
Sparsity ratio of minSpanVectors: 0.221554
```

```
spy(minSpanVectors)
```



References

[1] Bordbar A, Nagarajan H, Lewis NE, Latif H, Ebrahim A, Federowicz S, Schellenberger J, Palsson BO. "Minimal metabolic pathway structure is consistent with associated biomolecular interactions" *Mol Syst Biol* **10:**737 (2014).

[2] Schilling CH, Letscher D, Palsson BO. "Theory for the systemic definition of metabolic pathways and their use in interpreting metabolic function from a pathway-oriented perspective. *J Theor Biol* **203:**229-248 (2000).