CHEME 5440/7770: Structural Analysis of the Urea Cycle (PS2)

James Hayes
Smith School of Chemical and Biomolecular Engineering,
Cornell University, Ithaca NY 14850

```
html"""
<p style="font-size:20px;">James Hayes</br>
Smith School of Chemical and Biomolecular Engineering,
Cornell University, Ithaca NY 14850</p>
"""
```

```
read_reaction_file (generic function with 1 method)
```

```
include("C:/Users/labra/Documents/GitHub/CHEME-5440-7770-Cornell-Spring-2022/problem_sets/PS2/src/Include.jl")

```

Build the stoichiometric array

```
begin
    # Setup a collection of reaction strings -
    reaction_array = Array{String,1}()

    #Reactions -
    push!(reaction_array,"v₁,ATP+Citrulline+Aspartate,Argininosuccinate+AMP+Diphosphate,false")
    push!(reaction_array,"v₂,Argininosuccinate,Arginine+Fumarate,false")
    push!(reaction_array,"v₃,Arginine+H2O,Urea+Ornithine,false")
    push!(reaction_array,"v₄,CarbamoylPhosphate+Ornithine,Citrulline+Phosphate,false")
    push!(reaction_array,"v₅,3*NADPH+2*Arginine+4*O2+3*H,2*Citrulline+3*NADP+4*H2O+2*NO,true")

    #Exchanges
    push!(reaction_array,"b₁,∅,CarbamoylPhosphate,false")
    push!(reaction_array,"b₂,∅,Aspartate,false")
    push!(reaction_array,"b₃,Fumarate,∅,false")
    push!(reaction_array,"b₄,Urea,∅,false")
    push!(reaction_array,"b5,∅,ATP,false")
    push!(reaction_array,"b6,AMP,∅,false")
    push!(reaction_array,"b7,Diphosphate,∅,false")
    push!(reaction_array,"b8,Phosphate,∅,false")
    push!(reaction_array,"b9,∅,NADPH,true")
    push!(reaction_array,"b10,∅,O2,true")
    push!(reaction_array,"b11,∅,H,true")
    push!(reaction_array,"b12,NADP,∅,true")
    push!(reaction_array,"b13,H2O,∅,true")
    push!(reaction_array,"b14,NO,∅,true")



    # compute the stoichiometric matrix -
    (S, species_array, reaction_name_array) = lib.build_stoichiometric_matrix(reaction_array,expand=true);

    # show -
    nothing
end
```

```
(18, 26)
```

```
(ℳ,ℛ) = size(S)
```

```
["AMP", "ATP", "Arginine", "Argininosuccinate", "Aspartate", '
```

```
species_array
```

```
["v₁", "v₂", "v₃", "v₄", "Fv₅", "Rv₅", "b₁", "b₂", "b₃", "b₄",
```

```
reaction_name_array
```

```
18×26 Matrix{Float64}:
  1.0   0.0   0.0   0.0   0.0   0.0  0.0  …  0.0  0.0  0.0  0.
 -1.0   0.0   0.0   0.0   0.0   0.0  0.0     0.0  0.0  0.0  0.
  0.0   1.0  -1.0   0.0  -2.0   2.0  0.0     0.0  0.0  0.0  0.
  1.0  -1.0   0.0   0.0   0.0   0.0  0.0     0.0  0.0  0.0  0.
 -1.0   0.0   0.0   0.0   0.0   0.0  0.0     0.0  0.0  0.0  0.
  0.0   0.0   0.0  -1.0   0.0   0.0  1.0  …  0.0  0.0  0.0  0.
 -1.0   0.0   0.0   1.0   2.0  -2.0  0.0     0.0  0.0  0.0  0.
  ⋮                             ⋮        ⋱              ⋮
  0.0   0.0   0.0   0.0  -3.0   3.0  0.0     0.0  0.0  0.0  0.
  0.0   0.0   0.0   0.0   2.0  -2.0  0.0     0.0  0.0  0.0  0.
  0.0   0.0   0.0   0.0  -4.0   4.0  0.0     0.0  0.0  0.0  0.
  0.0   0.0   1.0  -1.0   0.0   0.0  0.0  …  0.0  0.0  0.0  0.
  0.0   0.0   0.0   1.0   0.0   0.0  0.0     0.0  0.0  0.0  0.
  0.0   0.0   1.0   0.0   0.0   0.0  0.0     0.0  0.0  0.0  0.
```

```
S
```

Convex analysis: compute the extreme pathways

```
begin
    # fill me in ...

    PM = lib.expa(S)

    # P contains the extreme pathways in its rows
    P = PM[:,1:ℛ]
    𝒩 = PM[:,(ℛ+1):end]

    # show -
    nothing
end
```

```
11×26 Matrix{Float64}:
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  …  0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  2.0  2.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  2.0  2.0  0.0  1.0  2.0  0.0     0.0  3.0  0.0  3.0
 6.0  6.0  4.0  4.0  1.0  0.0  4.0  6.0  …  3.0  0.0  3.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 2.0  2.0  0.0  0.0  1.0  0.0  0.0  2.0     3.0  0.0  3.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  1.0  1.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     1.0  1.0  0.0  0.0
 1.0  1.0  1.0  1.0  0.0  0.0  1.0  1.0  …  0.0  0.0  0.0  0.0
```

```
P
#5 extreme pathways (excluding ones which only involve
exchanges) (rows 3,5,6,8,11)
#Reactions v1 and v2 have 3/5 reaction frequency
#Reactions v3 and v4 have 3/5 reaction frequency
#Reaction v5 has reaction frequency of 4/5 (3/5 for
foward reaction and 2/5 for reverse)

#First path is only going back and forth in reversible v5
#Second path goes from v3>v4>v5
#Third path goes from v1>v2>v3>v4 and some v2>v5>v1 also
occurs
#Fourth path goes from v1>v2>v5
#Fifth path goes from v1>v2>v3>v4 (no v5 reaction occurs)
```

```
[0.0, 0.0, 0.0, 0.0, 2.0, 4.0, 0.0, 0.0, 0.0, 0.0, 1.0]
```

```
P[:,10]
#The tenth reaction is b5, the exchange of Urea out of
the system
#3 of the extreme pathways produce urea (the
second,third, and fifth)
```

```
11×18 Matrix{Float64}:
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  …  0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  …  0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0     0.0  0.0  0.0  0.0
 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  …  0.0  0.0  0.0  0.0
```

```
𝒩
#All zeros, as it should be
```

Metabolite connectivity array (MCA)

```
B =
18×26 Matrix{Int64}:
 1  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0
 1  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0  0
 0  1  1  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 1  0  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  1  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 1  0  0  1  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 ⋮              ⋮              ⋮              ⋮              ⋮
 0  0  0  0  1  1  0  0  0  0  0  0  0  0  1  1  0  0  0  0  0
 0  0  0  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  1  1  0  0  0  0  0  0  0  0  0  0  1  1  0  0  0
 0  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  1  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0
 0  0  1  0  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0  0  0
```

```
    B = binary_stoichiometric_matrix(S)
    #Binary stoichiometric matrix
```

```
18×18 Matrix{Int64}:
 2  1  0  1  1  0  1  1  0  0  0  0  0  0  0  0  0  0
 1  2  0  1  1  0  1  1  0  0  0  0  0  0  0  0  0  0
 0  0  4  1  0  0  2  0  1  2  3  2  2  2  2  1  0  1
 1  1  1  2  1  0  1  1  1  0  0  0  0  0  0  0  0  0
 1  1  0  1  2  0  1  1  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  2  1  0  0  0  0  0  0  0  0  1  1  0
 1  1  2  1  1  1  4  1  0  2  2  2  2  2  2  1  1  0
 ⋮              ⋮              ⋮              ⋮
 0  0  2  0  0  0  2  0  0  2  2  2  4  2  2  0  0  0
 0  0  2  0  0  0  2  0  0  2  2  2  2  4  2  0  0  0
 0  0  2  0  0  0  2  0  0  2  2  2  2  2  4  0  0  0
 0  0  1  0  0  1  1  0  0  0  1  0  0  0  0  2  1  1
 0  0  0  0  0  1  1  0  0  0  0  0  0  0  0  1  2  0
 0  0  1  0  0  0  0  0  0  0  1  0  0  0  0  1  0  2
```

```
begin
    # fill me in ...
    MCA=B*transpose(B)
end
```

```
ninosuccinate", "Aspartate", "CarbamoylPhosphate", "Citrulline",
```

```
species_array
```

```
#ATP and AMP are each connected to 4 metabolites (shared
reactions)
#Arginine shares reactions with 11 other metabolites
#Argininosuccinate shares reactions with 7 other
metabolites
#Aspartate shares reactions with 5 other metabolites
#CarbamoylPhosphate shares reactions with 4 other
metabolites
#Citrulline shares reactions with 15 other metabolites
#Diphosphate shares reactions with 5 other metabolites
#Fumarate shares reactions with 2 other metabolites
#H ions share reactions with 7 other metabolites
#H2O share reactions with 9 other metabolites
#NADP shares reactions with 7 other metabolites
#NADPH shares reactions with 7 other metabolites
#NO shares reactions with 7 other metabolites
#O2 shares reactions with 7 other metabolites
#Ornithine shares reactions with 6 other metabolites
#Phosphate shares reactions with 3 other metabolites
#Urea shares reactions with 3 other metabolites

#List of metabolites from most to least connected:
#Arginine, Citrulline, H2O,
Argininosuccinate/H+/NADP/NADPH/NO/O2, Ornithine,
Aspartate/Diphosphate, CarbamoylPhosphate/ATP/AMP,
Urea/Phosphate, Fumarate
```

```
[2, 2, 4, 2, 2, 2, 4, 2, 2, 4, 5, 4, 4, 4, 4, 2, 2, 2]
```

```
diag(MCA)
```

Reaction connectivity array (RCA)

```
26×26 Matrix{Int64}:
 6  1  0  1  1  1  0  1  0  0  1  1  1  0  0  0  0  0  0  0  0
 1  3  1  0  1  1  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0
 0  1  4  1  2  2  0  0  0  1  0  0  0  0  0  0  0  0  0  0  0
 1  0  1  4  1  1  1  0  0  0  0  0  0  1  0  0  0  0  0  0  0
 1  1  2  1  8  8  0  0  0  0  0  0  0  0  1  1  1  1  1  1  1
 1  1  2  1  8  8  0  0  0  0  0  0  0  0  1  1  1  1  1  1  1
 0  0  0  1  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 ⋮              ⋮              ⋮              ⋮              ⋮
 0  0  0  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1
 0  0  0  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1
 0  0  1  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 0  0  1  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  1  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
```

```
begin
    # fill me in ...
    RCA=transpose(B)*B

end
```

```
#All reactions are connected except v1/v3 and v2/v4 (ie
they don't share metabolites).

#The extreme pathway reaction frequency was only 3/5 for
v1,v2,v3, and v4. This means that metabolites produced in
reaction 1 and reaction 2, respectively, don't
necessarily have to continue on the path to reaction 3 or
4. However for v5 it was 4/5

#The reaction with the highest extreme pathway frequency
(v5) is also most connected to the other reactions (in
sharing metabolites). However, there isn't any clear
overall correlation.
```

```
[6, 3, 4, 4, 8, 8, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
```

```
diag(RCA)
```

```
begin
    # import some packages -
    using PlutoUI
    using PrettyTables
    using LinearAlgebra
    using Plots

    # setup paths -
    const _PATH_TO_NOTEBOOK = pwd()
    const _PATH_TO_SRC = joinpath(_PATH_TO_NOTEBOOK,"src")

    # load the PS2 code lib -
    lib = ingredients(joinpath(_PATH_TO_SRC,
"Include.jl"));

    # return -
    nothing
end
```

```
ingredients (generic function with 1 method)
```

```julia
function ingredients(path::String)

    # this is from the Julia source code (evalfile in base/loading.jl)
    # but with the modification that it returns the module instead of the last object
    name = Symbol("lib")
    m = Module(name)
    Core.eval(m,
        Expr(:toplevel,
             :(eval(x) = $(Expr(:core, :eval))($name, x)),
             :(include(x) = $(Expr(:top, :include))($name, x)),
             :(include(mapexpr::Function, x) = $(Expr(:top, :include))(mapexpr, $name, x)),
             :(include($path))))
    m
end
```

```
binary_stoichiometric_matrix (generic function with 1 method)
```

```julia
function binary_stoichiometric_matrix(matrix::Array{Float64,2})::Array{Int64,2}

    # initialize -
    (ℳ,ℛ) = size(matrix)
    B = Array{Int64,2}(undef,ℳ,ℛ)

    for row_index ∈ 1:ℳ
        for col_index ∈ 1:ℛ

            old_value = matrix[row_index,col_index]
            if (old_value == 0.0)
                B[row_index,col_index] = 0
            else
                B[row_index,col_index] = 1
            end
        end
    end

    # return -
    return B
end
```