The stoich module

Peter Gawthrop. *peter.gawthrop@unimelb.edu.au*

May 21, 2024

Contents

Note: this is the stoich.ipynb notebook. The PDF version "The stoich module" is available here.

1 Introduction

BondGraphTools is a python based toolkit for the creation and analysis of bond graph models of physical systems. Such physical systems include biomolecular systems; **stoich** is a toolkit for the stoichiometric analysis of such systems.

This document provides a simple introduction to **stoich** by means of a built-in bond graph model of a simple enzyme-catalysed reaction.

2 Example.

stoich.model() implements the enzyme-catalysed reaction: A+E = C = B+E where A is the substrate, B the product, E the enzyme and C an intermediate compound. This can be analysed using the following code.

First import some code:

[1]:
```
import BondGraphTools as bgt
import numpy as np
import sympy as sp
import IPython.display as disp
import stoich as st
```

2.1 Basic analysis

Now perform stoichiometric analysis on the model:

[2]:
```
s = st.stoich(st.model())
```

```
Computing N ...
Swapping Re:r1 for two Sf in ABCE
Swapping Re:r2 for two Sf in ABCE
Done.
Computing K ...
Done.
Computing G ...
Done.
```

s is a Python dict containing the stoichiometric information. For example, the stoichometric matrix N where $\dot{X} = NV$ is revealed as:

[3]:
```
print(s['N'])
```

```
[[-1  0]
 [ 0  1]
 [ 1 -1]
 [-1  1]]
```

This can be displayed in a more readable form as:

[4]:
```
disp.Latex(st.sprintl(s,'N'))
```

[4]:

$$N = \begin{pmatrix} -1 & 0 \\ 0 & 1 \\ 1 & -1 \\ -1 & 1 \end{pmatrix} \tag{1}$$

The state (vector of concentrations) X and the vector V of reaction flows are:

[5]:
```
disp.Latex(st.sprintl(s,'species'))
```

[5]:

$$X = \begin{pmatrix} X_A \\ X_B \\ X_C \\ X_E \end{pmatrix} \tag{2}$$

[6]:
```
disp.Latex(st.sprintl(s,'reaction'))
```

[6]:

$$V = \begin{pmatrix} V_{r1} \\ V_{r2} \end{pmatrix} \tag{3}$$

The corresponding reactions can be displayed:

[7]:
```
disp.Latex(st.sprintrl(s))
```

[7]:

$$A + E \Leftrightarrow C \tag{4}$$

$$C \Leftrightarrow B + E \tag{5}$$

The stoichometric matrix N gives the species state X in terms of reaction flow V from $\dot{X} = NV$. N is also used together with thermodynamic constants K and rate constants κ to give an explicit expresion for reaction flow V in terms of species state X. **stoich** computes the symbolic expression as:

[8]:
```
disp.Latex(st.sprintl(s,'N'))
```

[8]:

$$N = \begin{pmatrix} -1 & 0 \\ 0 & 1 \\ 1 & -1 \\ -1 & 1 \end{pmatrix} \tag{6}$$

[9]:
```
disp.Latex(st.sprintvl(s))
```

[9]:

$$v_{r1} = \kappa_{r1} \left(K_A K_E x_A x_E - K_C x_C \right) \tag{7}$$

$$v_{r2} = \kappa_{r2} \left(-K_B K_E x_B x_E + K_C x_C \right) \tag{8}$$

2.2 Conserved moieties and Pathways

Conserved moieties are revealed by the matrix G where $G^T N = 0$. In this case:

[10]:
```
disp.Latex(st.sprintl(s,'G'))
```

[10]:

$$G = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & -1 & 0 & 1 \end{pmatrix} \tag{9}$$

The first row corresponds to $\dot{x}_A + \dot{x}_B + \dot{x}_C = 0$, the sum of the rows $(0\ 0\ 1\ 1)$ corresponds to $\dot{x}_C + \dot{x}_E = 0$

Pathways are revealed by the matrix K where $NK = 0$. In this case:

[11]:
```
disp.Latex(st.sprintl(s,'K'))
```

[11]:

$$K =) (\tag{10}$$

There are no pathways: there is zero flow ($V = 0$) in the steady state.

2.3 Chemostats.

Consider the case where both substrate A and product B are chemostats:

[12]:
```
chemostats = ['A','B']
```

The same system, but with the chemostats, can be analysed using:

[13]:
```
sc = st.statify(s,chemostats=chemostats)
```

The stoichometric matrix N is now:

[14]:
```
disp.Latex(st.sprintl(sc,'N'))
```

[14]:

$$N = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & -1 \\ -1 & 1 \end{pmatrix} \tag{11}$$

The first two rows are zero, corresponding to $\dot{x}_A = \dot{x}_B = 0$: this is because both substrate A and product B are chemostats.

The pathway matrix K is now:

[15]:
```
disp.Latex(st.sprintl(sc,'K'))
```

[15]:

$$K = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \tag{12}$$

This means that the flow though reactions r1 and r2 are the same and can be non-zero at steady-state. The conserved moieties of this chemostated system are revealed by the matrix G

[16]:

```
disp.Latex(st.sprintl(sc,'G'))
```

[16]:

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \tag{13}$$

The three rows correspond to: - $\dot{x}_A = 0$ (x_A is constant) - $\dot{x}_B = 0$ (x_B is constant) - $\dot{x}_C + \dot{x}_E = 0$ ($x_C + x_E$ is constant)

2.4 Pathway analysis

[17]:

```
## Find the pathway stoichiometric matrix
sp = st.path(s,sc)
## And show the corepondīng reaction
disp.Latex(st.sprintrl(sp))
```

[17]:

$$A \Leftrightarrow B \tag{14}$$

2.5 Simulation

Although **BondGraphTools** has its own simulation tool, the particular form of stoichiometric equations allows for a special purpose simulation tool taking advantage of explicit equations and reducing the state dimension in the presence of conserved moieties.

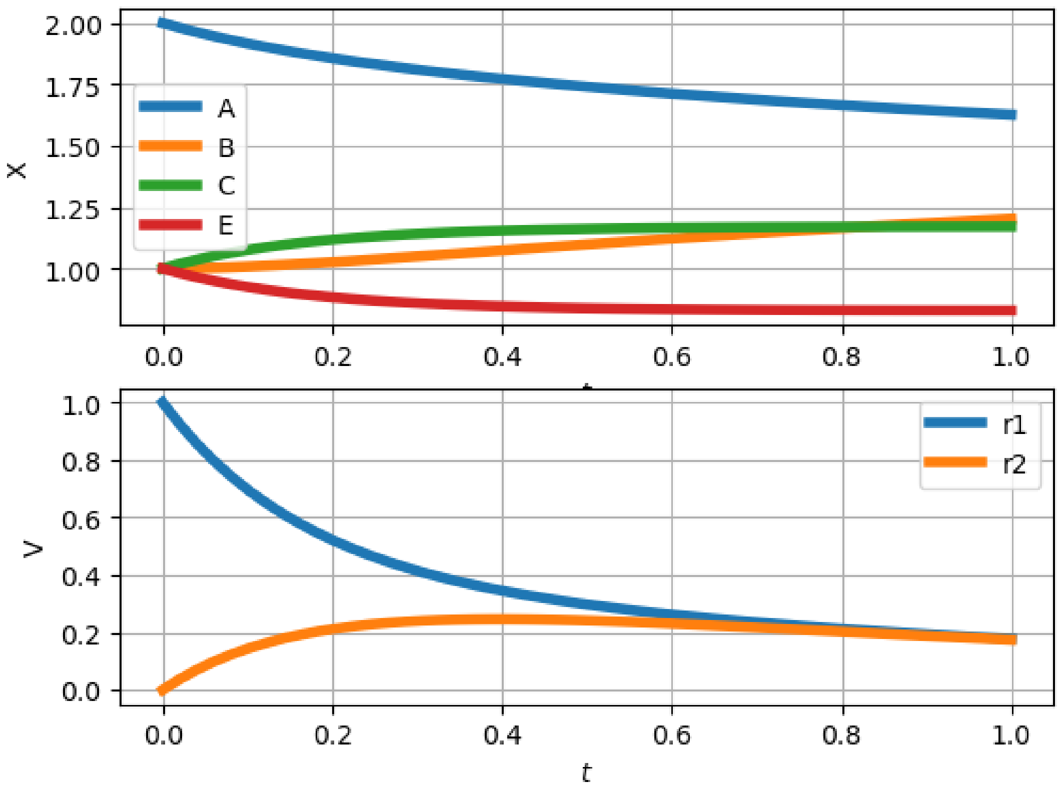
The system (without chemostats) can be simulated as:

[18]:

```
X0 = np.array([2,1,1,1]) # Set initial states
result = st.sim(s,X0=X0) # Simulate
```

[19]:

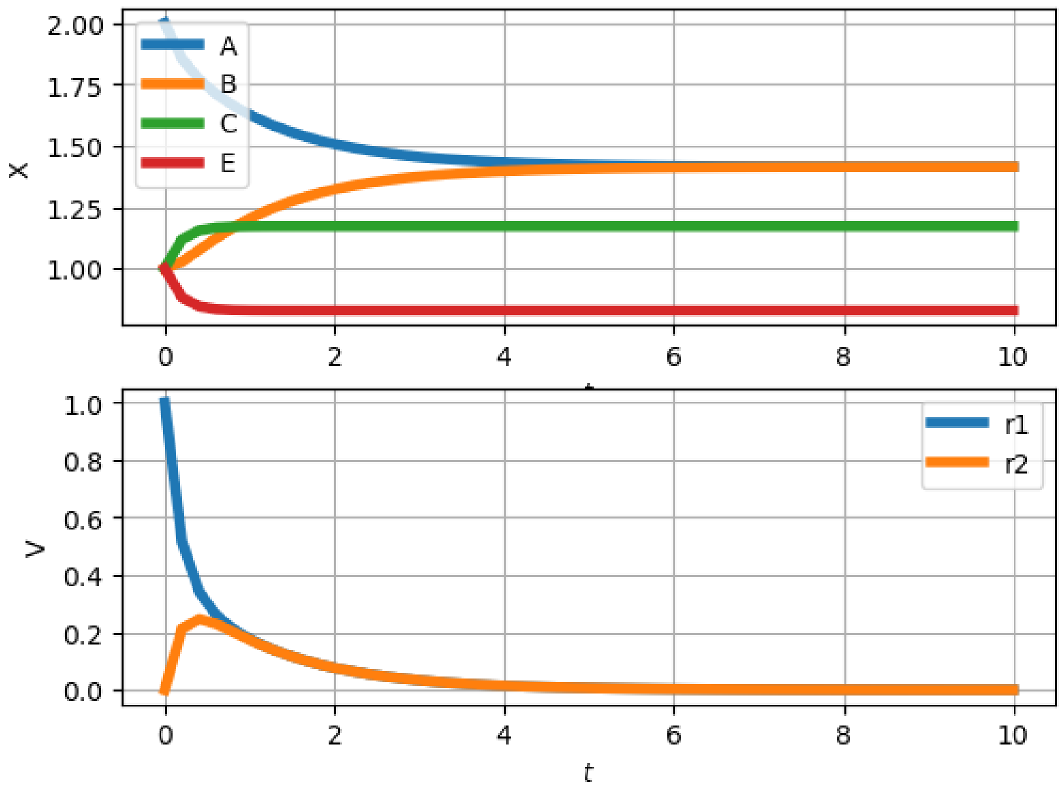
```
st.plot(s,result)
```



The flows V though r1 and r2 seem to be heading towards zero as predicted by pathway analysis. This can ve verified by simulating over a longer time:

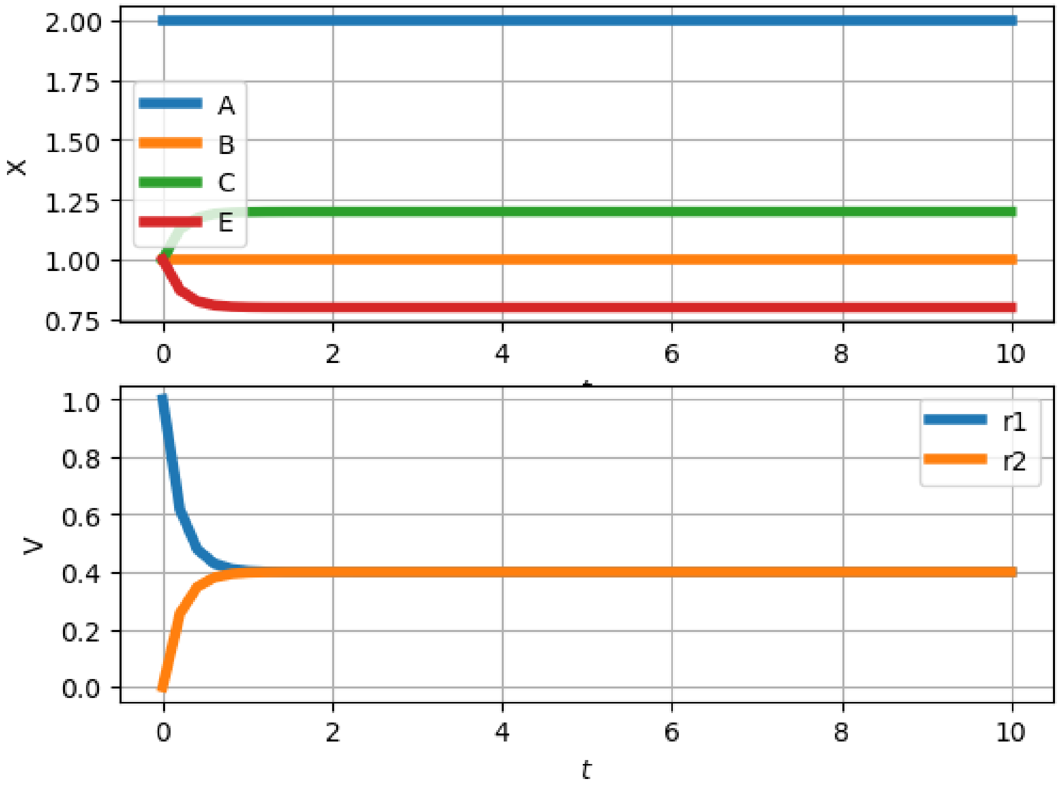
[20]:

```
t = np.linspace(0,10)
result = st.sim(s,X0=X0,t=t)
st.plot(s,result)
```



The system with chemostats can be simulated as:

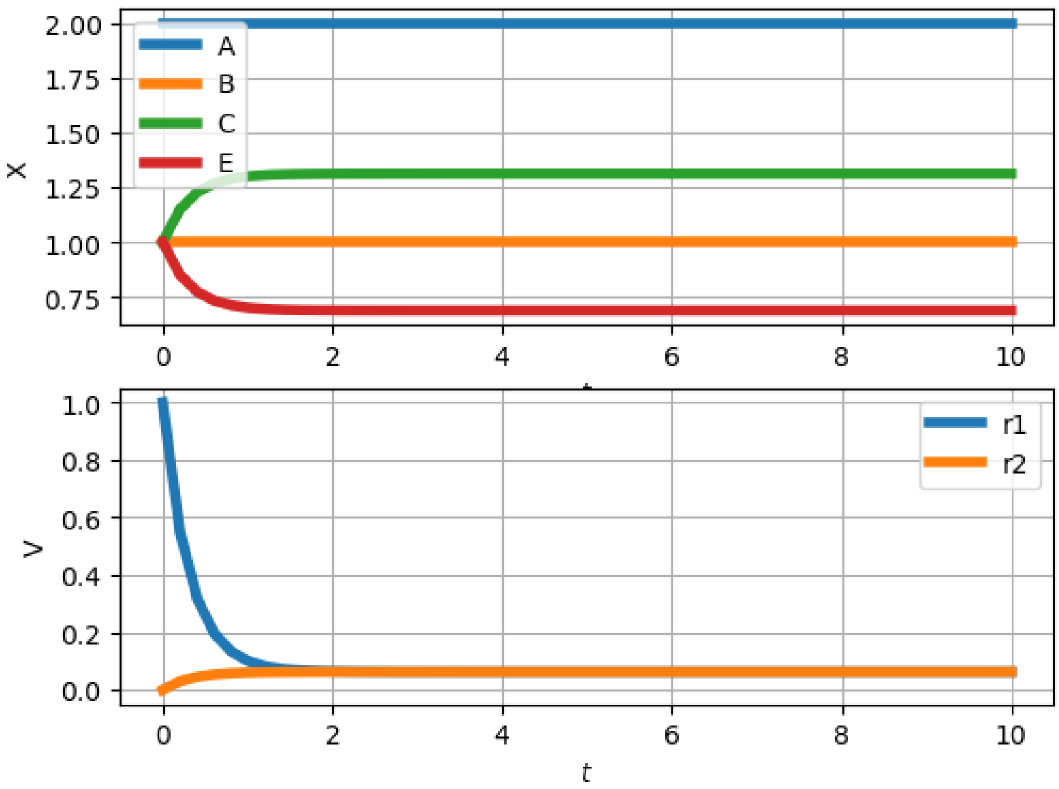
[21]:
```
t = np.linspace(0,10)
result = st.sim(s,X0=X0,t=t,sc=sc)
st.plot(s,result)
```



As predicted by pathway analysis, the two flows converge on a non-zero value.

Of course, these simulations have been using default (unity) values for parameters. These defaults can be changed by explicitly supplying parameters:

[22]:
```
parameter={'kappa_r2':0.1}
result = st.sim(s,X0=X0,t=t,sc=sc,parameter=parameter)
st.plot(s,result)
```



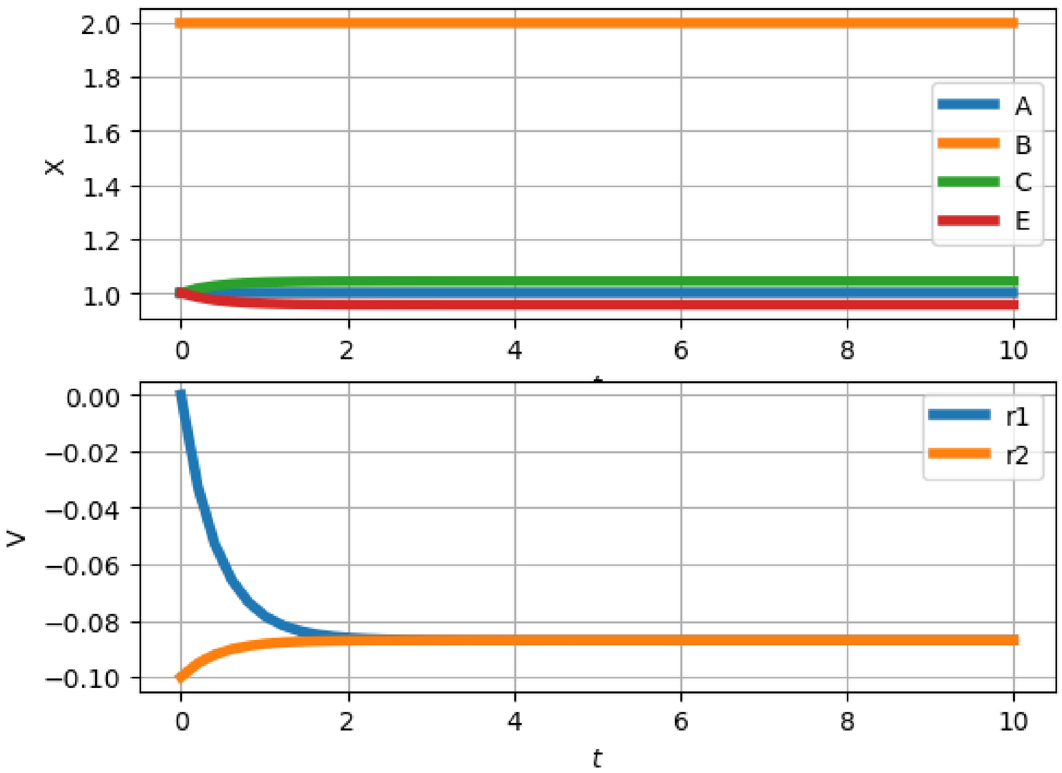
Moreover, the four initial conditions can be explicity chosen, for example: -

$$X_0 = \begin{pmatrix}1 & 2 & 1 & 1\end{pmatrix}^T$$

Note that the default value was -

$$X_0 = \begin{pmatrix}2 & 1 & 1 & 1\end{pmatrix}^T$$

[23]:
```
X0 = np.array([1,2,1,1])
result = st.sim(s,t=t,sc=sc,parameter=parameter,X0=X0)
st.plot(s,result)
```



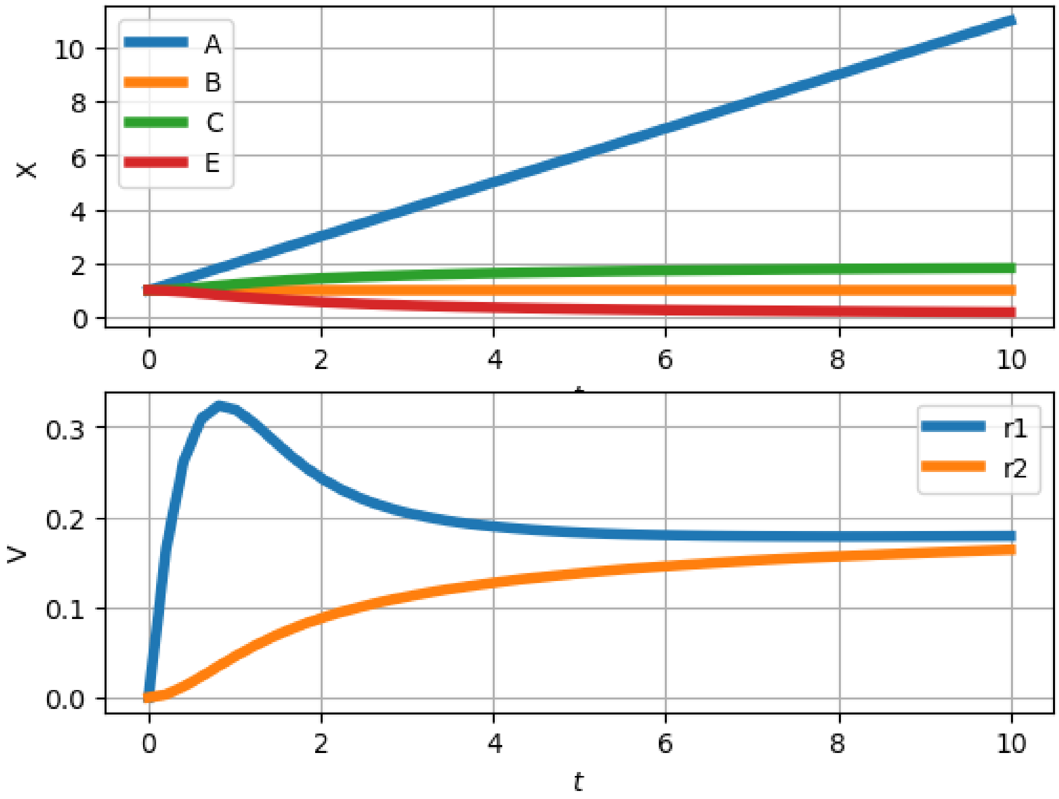
As the product B amount is greater than that of the substrate A, the flow proceeds in reverse.

2.5.1 Time-varying chemostats

By default, chemostats remain at the corresponding initial state. This can be changed by declaring a time-varying expression for the chemostat state. For example, set the chemostat for substrate A to have a value of $1 + t$:

[24]:

```
X_chemo = {'A':'1+t'}
result = st.sim(s,t=t,sc=sc,parameter=parameter,X_chemo=X_chemo)
st.plot(s,result)
```



Note that the flow rates reach a maximum value as the amount of enzyme x_E reduces to zero. This behaviour is typical of systems with conserved moities in general and enzyme catalysed reactions in particular.

References