

rksim Networks

July 20, 2020

1 Construction

Networks in *rksim* are subclasses of *NetworkX DiGraph* and are thus directed graphs, but with additional methods. From a set of reactions

$$\begin{aligned} s = \{ & R1_a + R2_a + \dots \rightarrow P1_a + P2_a + \dots, \\ & R1_b + R2_b + \dots \rightarrow P1_b + P2_b + \dots, \\ & \vdots \quad \quad \quad \rightarrow \quad \quad \quad \vdots \quad \quad \quad \vdots \} \end{aligned} \quad (1)$$

a **Reaction** is constructed for each by removing identical components and setting the stoichiometries of unique **Species**. For a reaction $A + A \rightarrow P$: **A.stoichiometry** = 2 and **P.stoichiometry** = 1. From the set of reactions a network is initialised by first adding nodes as simply all unique species. For example, for the set of three reactions $s_1 = \{A + B \rightarrow I, I \rightarrow P, C \rightarrow P\}$ the nodes are shown in Figure 1a, where each node is indexed, contains name, initial concentration (c_0) and **Species**.

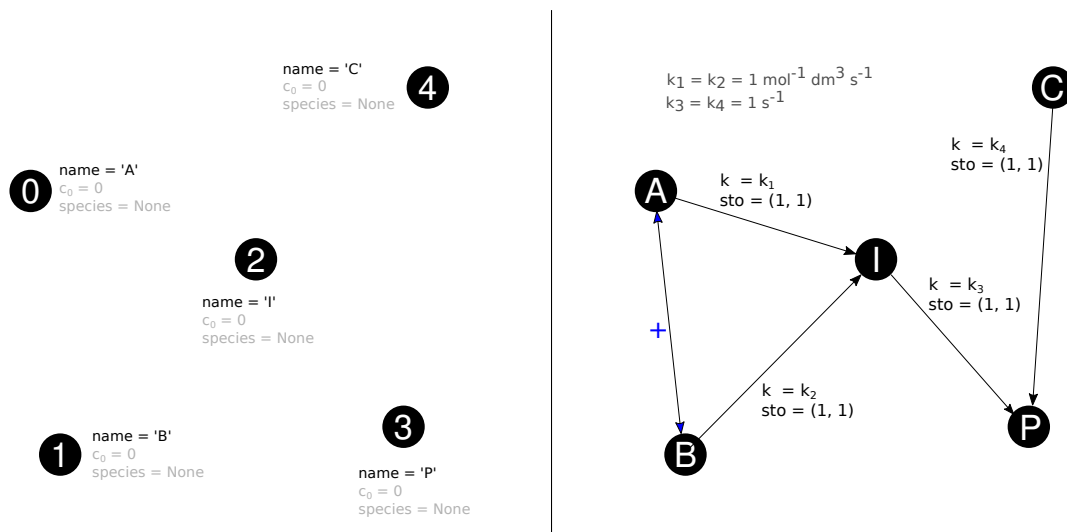


Figure 1: Reaction network node (a) and edge (b) initialisation for s_1 .

Edges are then added for the addition of components ('+' edges) and reactions ('reaction' edges). Addition edges denote more than one component is required for a reaction, in s_1 there is one addition edge ($A-B$, represented with two directed edges $A \rightleftharpoons B$). Reaction edges are added for species that convert from one into another and contain a rate constant k and a stoichiometry (sto) change (n, m) . The full initialised **Network** for s_1 is shown in Figure 1b, where for simplicity nodes are labelled with their name rather than index.

The overabundant rate constants are initialised with $k = 1$ and a mapping generated to enable setting the unique rate constants i.e here $k_1 \equiv k_2$. A mapping is generated by, for each reaction edge, populating a list of edges that must have the same rate constant. These are neighbours over addition edges that also have reaction edges to the product and likewise over the addition edges of products.

In s_1 stoichiometry changes are all $1 \rightarrow 1$ along the reaction edges. For a set $s_2 = \{2A \rightarrow B, A \rightarrow C\}$ stoichiometries are (2, 1) and (1, 1) for edges $A-B$ and $A-C$ respectively. When a system is assigned to a set of data initial concentrations are populated as the first in the **TimeSeries** of concentrations.

2 Derivative

Simulated kinetics ($c_i(t)$ for a component i at a time t) are generated by numerical propagation of a set of ordinary differential equations (ODE) from the initial value of the concentrations using *scipy*. The ODEs ($d\mathbf{c}/dt$, where \mathbf{c} is the concentration vector) are constructed from the **Network** as follows.

For each node i the change in concentration is the inflow minus the outflow

$$\frac{dc_i}{dt} = \text{inflows}(i) - \text{outflows}(i) \quad (2)$$

$$\text{outflows}(i) = \sum_{j \in \{O_i\}} \frac{k_{i \rightarrow j}}{N_{n_j} + 1} a_i c_i^{a_i} \prod_{k \in \{n_i \rightarrow j\}} c_k^{a_k} \quad (3)$$

where $\{O_i\}$ is the set of direct successor nodes to i , $k_{i \rightarrow j}$ is the rate constant from i to j , N_{n_j} is the number of neighbours along addition edges to node j , a_i is the stoichiometry of i along the $i-j$ reaction edge, c_i is the concentration of component i , $\{n_i \rightarrow j\}$ is the set of nodes which are neighbours to i and also have a reaction edge to j . An example is shown in Figure 2.

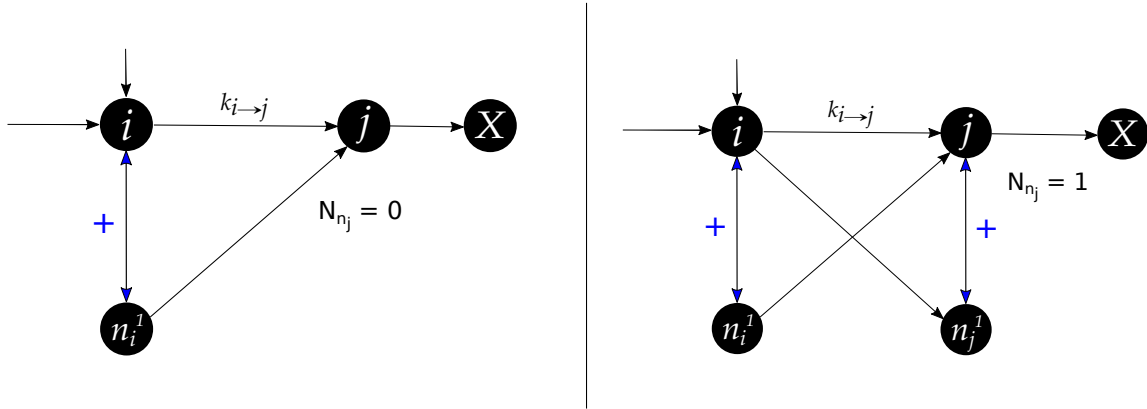


Figure 2: Visualisation of an outflow node j originating from i .

Likewise for the inflow nodes

$$\text{inflows}(i) = \sum_{j \in \{I_i\}} \frac{k_{j \rightarrow i}}{N_{n_j} + 1} a_j c_j^{a_j} \prod_{k \in \{n_j \rightarrow i\}} c_k^{a_k} \quad (4)$$

where j now enumerates over direct predecessor nodes of i , the set $\{I_i\}$.