Vesicle traffic system and constraints.

## 1 Formula variables

Basic notation for the formula we are trying to generate.

1. edge = e(i,j,q) There is an qth edge between node i and j.
2. dump = d(i,j,q) drop edge(i,j,q) from the graph.
3. presence node = n(i,k) kth molecule is present on the node i.
4. active_node = a(i,k) kth molecule is active on the node i.
5. presence_edge = e(i,j,q,k) There is an qth edge between node i and j with kth molecule present.
6. active_edge = b(i,j,q,k) There is an qth edge between node i and j with kth molecule active.
7. reachability = r(i,j,k,z) Node i and j are reachable with kth molecule present in z steps.
8. pairing matrix = p(k,k') Molecule k and k' are pairing molecules in the matrix.
9. Type of the Boolean function = sorts (takes k-1 arguments Bool and return Bool)
10. Boolean function on nodes Function (fn_m,*sorts)
11. Boolean function on edge Function(fe_m,*sorts)

## 2 Regulation on nodes and edges

Regulation of a model is based on the kind of the model under consideration. for example model 1 (-V 1) has two different arbitrary Boolean function dictating the activity of the molecules on edge and nodes and for variation 3 (-V 3) activity on edge is driven by SNARE-SNARE inhibition.

We have build constraints A0, A1 for these two regulations.

A0: No regulation on nodes.

$$\bigwedge_{i,k} n_{i,k} == a_{i,k}$$

A1: No regulation on edges.

$$\bigwedge_{i,j,q,k} e_{i,j,q,k} == b_{i,j,q,k}$$

A0: Boolean regulation on nodes. Activity of a molecule k on a node is defined as a Boolean function of presence of other molecule present on that node.

$$\bigwedge_{i,k} n_{i,k} \supset a_{i,k} == f_n[k](\bigvee_{k'!=k} n_{i,k'})$$

A1: Boolean regulation on edges. Activity of a molecule k on a node is defined as a Boolean function of presence of other molecule present on that node.

$$\bigwedge_{i,j,q,k} e_{i,j,q,k} \supset b_{i,j,q,k} == f_e[k](\bigvee_{k'!=k} e(i,j,q,k'))$$

A1: SNARE-SNARE inhibition. Inhibition of the edges are driven by the pairing matrix.

$$\bigwedge_{i,j,i!=j,q,k} (\bigwedge_{k'!=k} p_{k,k'} \supset e_{i,j,q,k'}) \supset \neg b(i,j,q,k))$$

$$\bigwedge_{i,j,i!=j,q,k} \neg[(\bigwedge_{k'!=k} p_{k,k'} \supset e_{i,j,q,k'})] \supset b(i,j,q,k))$$

## 3 Basic constraints on the edges and nodes

The edge labels are subset of the node label of source and target compartment.

C0: The edge labels are subset of the node label of source compartment.

$$\bigwedge_{i,j,k} e_{i,j,k} \supset a_{i,k}$$

C1: The edge labels are subset of the node label of target compartment.

$$\bigwedge_{i,q} e_{i,j,k} \supset a_{j,k}$$

C2: Self edges are not allowed.

$$\bigwedge_{i,q} \neg e_{i,i,q}$$

C4: Condition on p_kk'. Diagonal blocks should be all 0's.

$$\bigwedge_{(x<M/2 \wedge y<M/2) \vee (x>=M/2 \wedge y>=M/2)} \neg p(x,y)$$

C5: Activity on the node. A molecule should be present to be active on a node.

$$\bigwedge_{i,k} a_{i,k} \supset n_{j,k}$$

## 4 Main constraints

Main constraints to be followed.

F0: An edge has to have one present molecule. F0 looks redundant.

$$\bigwedge_{i,j,q} \bigvee_{k} e_{i,j,q,k} \supset e_{i,j,q}$$

F1: If molecule is active on an edge then it should be present on the edge.

$$\bigwedge_{i,j,q,k} a_{i,j,q,k} \supset e_{i,j,q,k}$$

## 5 Steady state specification

Each molecule leaving the node on a vesicle should come back to its source node in a cycle, i.e., **for every molecule leaving the node there exists a cycle with that molecule present on *each of the edges and nodes* of the path taken**.

### 5.1 Reachability definition and stability condition

If molecule is active on an edge then it should be present on the edge.

F3: stability condition. Source node is reachable by target node with that molecule present in p steps.

$$\bigwedge_{i,j,k} (\bigvee_{q} e_{i,j,q,k}) \supset (r_{j,i,k,0} \vee r_{j,i,k,1} \vee ... \vee r_{j,i,k,p})$$

F2: Reachability definition.

$$\bigwedge_{i,j,k,p} r_{i,j,k,p} \supset (\bigvee_{q} e_{i,j,q,k} \vee \bigvee_{i \neq i'} (\bigvee_{q} e_{i,i',q,k}) \wedge r_{i',j,k,p-1})$$

## 6 Fusion rule Formula

Fusion rules consist of two different mechanisms.

1. A **SNARE pairing mechanism** which determines compatible Q-R pairs on vesicles and compartments that can cause fusion.
2. **Regulatory mechanisms** on edges and on nodes (possibly distinct) which regulate SNARE activity based on the presence/absence of other molecules on the corresponding node or edge.

F4: For an edge to be valid, at least one SNARE pair on the vesicle and target compartment must be active, and have a non-zero entry in the pairing matrix.

$$\bigwedge_{i,j,q} e_{i,j,q} \supset \bigvee_{k,k'} (b_{i,j,q,k} \wedge a_{j,k'} \wedge p_{k,k'})$$

F5: To ensure that fusion respects the graph structure by the edge under consideration, it should not be possible to fuse with any other node.

$$\bigwedge_{i,j,k} b_{i,j,k} \supset \neg \bigvee_{j \neq j', k''} (a_{j',k''} \wedge p_{k,k''})$$

## 7 Connectivity

To check the whether that n connected is necessary condition, we remove (drop) n edges from the graph and if it disconnects the graph and we get an assignment we have a n connected satisfying graph. We can go down or up using -C _ option.

D0: Only present edges can be dropped.

$$\bigwedge_{i,j,q} d_{i,j,q} \supset e_{i,j,q}$$

D1,D2: We are dropping c edges from the graph. exactly c are dropped.

$$\sum_{i,j,q} d_{i,j,q} == c$$

D3: Graph becomes disconnected. Ensure that there is no path between some nodes i,j in the underlying undirected graph. Different reachability.

$$\bigwedge_{i,j} \neg(r'_{i,j} \vee r'_{j,i})]$$

D4: New reachability definition for grph connectedness: dReachable. Node i,j are reachable either if there is a direct edge and its not dropped. Or there is an node i' such that, there is a direct edge between i,i' which is not dropped and i' and j is dReachable.

$$\bigwedge_{i,j}[\bigvee_{q}(e_{i,j,q} \wedge \neg d_{i,j,q}) \vee (\bigvee_{i'!=i} r'(i',j) \wedge \bigvee_{q}(e_{i,i',q} \wedge \neg d_{i,i',q})] \supset r'(i,j)$$