Definitions of Box Model operations

Box Model

A **box model** is a directed graph whose vertices are **compartments** and edges are **transitions**. Each compartment has a **name**, which is a mathematical variable such as X. A transition is an ordered triple (s, t, r), where s and t are the **source** and **target** compartments of the transition, and r is its **rate**, which is a function of the compartments' names. Any variable involved in a transition rate that is not the name of a compartment is a **parameter** of the model. $V(M) = \{c_1, \ldots, c_n\}$, the vertex set of model M, is the set of its compartments, and $E(M)$ is the set of its edges. Let $X(M)$ be the set of the names of the compartments of model M, and let $P(M) = \{p_1, \ldots, p_m\}$ be the set of all its transitions' parameters; then each transition rate is a function $r(c_1, \ldots, c_n, p_1, \ldots, p_m)$ taking values in $\mathbb{R}$.

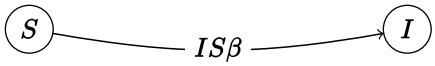
A box model is associated with an **ODE**, whose state variables are the compartments' names, and whose flow vector field is constructed from the transition rates:

$\frac{dX}{dt} = \sum_{\{e \in E(M) | \text{source}(e) = X\}} \text{rate}(e) - \sum_{\{e \in E(M) | \text{target}(e) = X\}} \text{rate}(e)$

for each compartment X.

There are also an SDE, a diffusion equation, individual-based models, and possibly other models that can be automatically generated to describe the behavior of this system.

Example: SI model.



This box model has two compartments, S and I. There is one transition from S to I, with rate βSI. Its ODE is

$\frac{dS}{dt} = -\beta SI$

$\frac{dI}{dt} = \beta SI.$

Products of box models

A **product** of box models, written $M_1 \times \cdots \times M_n$, where each M_i is a box model, is a box model whose set of compartments is the cross product, or Cartesian product, of the component box models' sets of compartments (see Appendix for definitions of graph product operations). The transitions include the edges of the Cartesian product of the components' graphs, but the product may include more transitions as well.

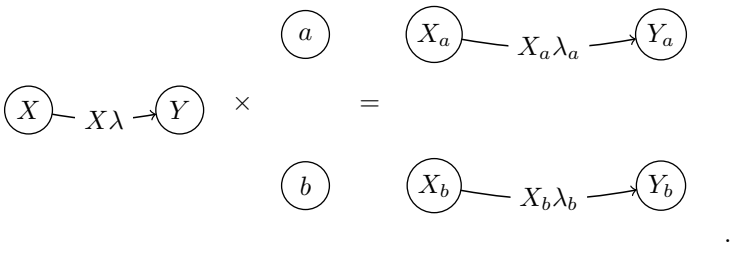
[TODO: it may include added compartments as well?]

- The compartments of the cross product are tuples $(c_1, \ldots, c_n)$ where each c_i is a compartment of M_i. The name of compartment $(c, d, e, \ldots, w)$ may be $c_{de\ldots w}$, or it may be named $X_{cde\ldots w}$, or some other naming may be used.
- For each transition T of component model M_i, with source s, target t, and rate r, there is a transition from each compartment $(c_1, \ldots, c_{i-1}, s, c_{i+1}, \ldots, c_n)$ of the product model to $(c_1, \ldots, c_{i-1}, t, c_{i+1}, \ldots, c_n)$. Its transition rate is derived from r.

I believe there are a few different products that are appropriate depending on the biology of the model, differing in what transitions are included and how their rates are constructed.

Simple Cross Product

Sometime it's sufficient to construct a product model with one transition for each edge of the graph Cartesian product. For example, imagine a simple infection model M_i, in which all compartments are stratified together by some demographic variable. Assume each component transition rate involves no compartments other than its source compartment. Then the stratified model is a simple product of the infection model with the stratification. The compartments of the product are labeled using subscripts, and the infection transitions are replicated in each stratum, with stratified compartment names substituted and subscript added to all parameters. For example, given a model with transition from X to Y at rate λX, stratified by crossing with a trivial model providing two demographic states a and b, we would construct a product model



.

This simple Cartesian product of models is defined by the following formula.

Definition. The *simple cross product* of models $M_1, \ldots, M_n$ is defined as follows:

- $V(M_1 \,\square\, \cdots \,\square\, M_n) = V(M_1) \times \cdots \times V(M_n)$ (see Appendix for set product operations).
- $E(M_1 \,\square\, \cdots \,\square\, M_n) = \cup_{\{\,(v_1,\ldots,v_n)\in V(M_1)\times\cdots\times V(M_n)\,\}} \cup_{i\in\{1,\ldots,n\}} \cup_{\{\,w,r|(v_i,w,r)\in E(M_i)\,\}}$ stratify$((v_1,\ldots,v_n), w, r, (M_1,\ldots,M_n), i)$

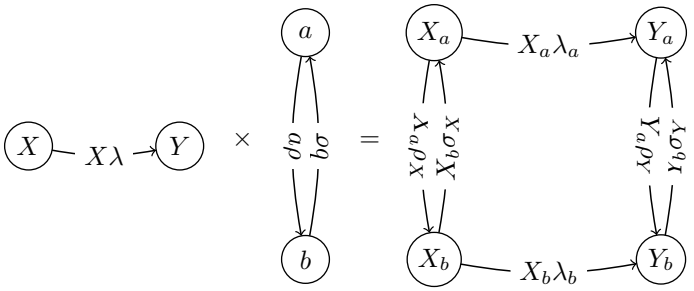
where

- stratify$((v_1,\ldots,v_n), w, r, (M_1,\ldots,M_n), i) = ((v_1,\ldots,v_n), (v_1,\ldots,v_{i-1}, w, v_{1+i},\ldots,v_n), r_s)$,

where r_s is the rate r with the name of v_i replaced by the name of the stratified vertex $(v_1,\ldots,v_n)$ and all parameters subscripted by $v_1,\ldots,v_{i-1},v_{i+1},\ldots,v_n$.

The name of a product compartment $(v_1,\ldots,v_n)$ can be constructed in various ways. It might be the name of v_1 subscripted by the names of the other compartments $v_2,\ldots,v_n$, or it might be X subscripted by all the compartments' names, or some other rule might be used.

Thus by crossing with the *a*-*b* model, the single row of the X-Y model gets replicated once for each compartment of the *a*-*b* model. In this case the stratifying model is trivial – there are no vertical arrows – but in general this product operation treats all the component models equally: if the "vertical" model has edges they are likewise replicated once for each compartment of the horizontal model:

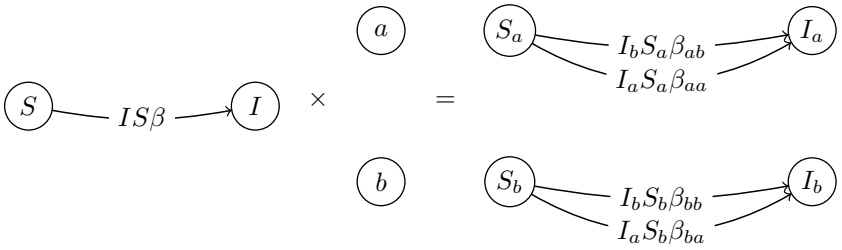


.

Cross product with interactions

Unfortunately, the above definition of the cross product is not suitable for even a simple SI example with transition rate βSI, because this model violates the assumption that the rate involves no compartments other than S. In this model it's reasonable to believe that either stratified S class can be infected by

individuals of both stratified I classes. This requires the S and I variables in that transition rate to be stratified separately, and a transition to be constructed for every combination of stratified compartments. The correct product model is this one, with the one edge replicated four times, not only twice:



.

If we used the simple cross product, we would not generate the correct sum over β and I variables.

To implement this, we need an extended product operation that provides stratification for the "catalyst" I in the βSI transition as well as the source S. The simple cross product operation can be extended in this way to become a full cross product operation.

Along with that extension, at this point I will introduce a complete definition of the cross product operation, including some details that will only become relevant in later examples.

Definition. A *cross product* $M_1 \,\square\, \cdots \,\square\, M_n$ of models $M_1, \ldots, M_n$ is a box model M such that

- $V(M) = V(M_1) \times \cdots \times V(M_n)$
- $E(M) = \cup_{i \in \{1,\ldots,n\}} \cup_{(s,t,r) \in E(M_i)} \Xi((s,t,r), i)$

where

- $\Xi((s,t,r),i) = \Xi_{\square}((s,t,r),i) = \begin{cases} \{\, (S, T, \Phi_r(r, S, i_s, T)) \mid (S, i_s) \in \Sigma(s, V(M)), \\ \quad T \in \Omega_1(S, i_s, s, t, r, i) \,\} \\ \qquad \text{if } r \text{ is linear in } s \\ \{\, (S, T, \Phi_{2r}(r, S, i_s, C, i_c, T)) \mid (S, i_s) \in \Sigma(s, V(M)), \\ \quad (C, i_c) \in \Sigma(c, V(M)), T \in \Omega_2(S, i_s, C, i_c, s, t, r, i) \,\} \\ \cup \{\, (S, T, \Phi_{1r}(S, i_s, i_c, s, t, r, i)) \mid (S, i_s) \in \Sigma(s, V(M)), \\ \quad (S, i_c) \in \Sigma(c, V(M)), T \in \Omega_{2i}(S, i_s, i_c, s, t, r, i) \,\} \\ \qquad \text{if } r \text{ is bilinear in } s \text{ and } c \end{cases}$

  generates the set of all product edges derived from the edge (s, t, r);

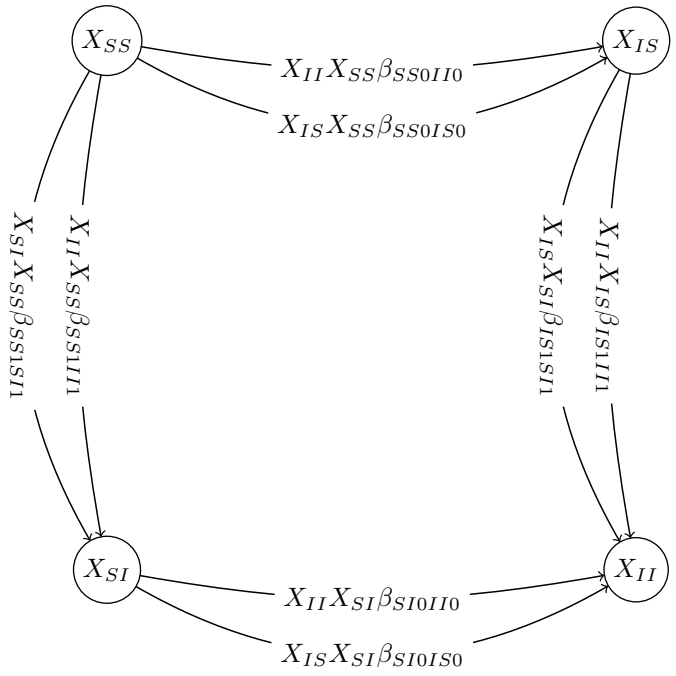
- $\Sigma(v, \mathcal{V})$ is a set of **inclusions** of compartment v in the compartments $\mathcal{V}$ of the product model. An inclusion is an ordered pair $(V, i) \in \mathcal{V} \times \mathbb{N}$, indicating that compartment v is in the ith position in product compartment V. In this stratification example, we only require simple inclusion in the ith model, $\Sigma_1(v, \mathcal{V}; i) = \{\, ((v_1, \ldots, v_n), i) \mid (v_1, \ldots, v_n) \in \mathcal{V}, v_i = v \,\}$, when stratifying a transition of that model. In some cases, as we will see, we will require a full inclusion function allowing "cross interactions", as we need to allow compartments from different component models to interact with each other: $\Sigma_c(v, \mathcal{V}) = \{\, ((v_1, \ldots, v_n), j) \mid (v_1, \ldots, v_n) \in \mathcal{V}, v_j = v \text{ for any } j \,\}$.
- $\Omega_1(S, i_s, s, t, r, i) = \{\, (S_1, \ldots, S_{i_s - 1}, t, S_{i_s + 1}, \ldots, S_n) \,\}$ is a **unary operation** defining the set of output (target) compartments produced by a given transition acting on source compartment S. Here we provide a simple definition where S just goes to the corresponding compartment T in the same stratification of the model, but we allow for variant products defined by a different unary operation.
- $\Omega_2(S, i_s, C, i_c, s, t, r, i) = \{\, (S_1, \ldots, S_{i_s - 1}, t, S_{i_s + 1}, \ldots, S_n) \,\}$ is a **binary operation** defining the set of output (target) compartments produced by a given transition acting on source compartment V when compartment C is the catalyst. Here as well, we simply transform S into its stratified counterpart T, but we allow for variant operations in which the same S transitions to different targets depending on C.
- $\Omega_{2i}(S, i_s, i_c, s, t, r, i)$ is a variant binary operation providing additional transitions in the case that s and c are both components of the same compartment S. This allows for a within-compartment interaction that is distinct from an interaction between two individuals of the same compartment.
- Φ_r, Φ_{2r}, and Φ_{1r} are **rewritings** of transition rates. We construct a rewritten rate for each edge of the product model based on its original rate r, by replacing the source s by V, the catalyst c by C (if present), and adding subscripts to all parameters. Parameters are subscripted by the compartment names indicating the stratification of the source compartment, in the simple case. When a single transition is replicated to multiple output compartments, it's necessary to add subscripts indicating the output compartments; when it's replicated using multiple catalyst compartments, we need subscripts naming the catalyst, and when it's replicated multiple times using different inclusions of compartments we need subscripts distinguishing the inclusions as well. The software tries to be smart about applying as few subscripts as are necessary to provide unique parameters for each transition.

This is a fairly general framework for defining products, at least in a world of linear and bilinear reactions among compartments. If transitions arise whose rates are arbitrary functions of compartment sizes, we'll need to allow users to supply additional Ξ and Ω functions to generate product transitions and rewrite their rates.

This product operation generates the correctly stratified SI model shown above. In this case it's not necessary to distinguish which inclusion function Σ it uses, because both produce the same result. In the next model it does make a difference.

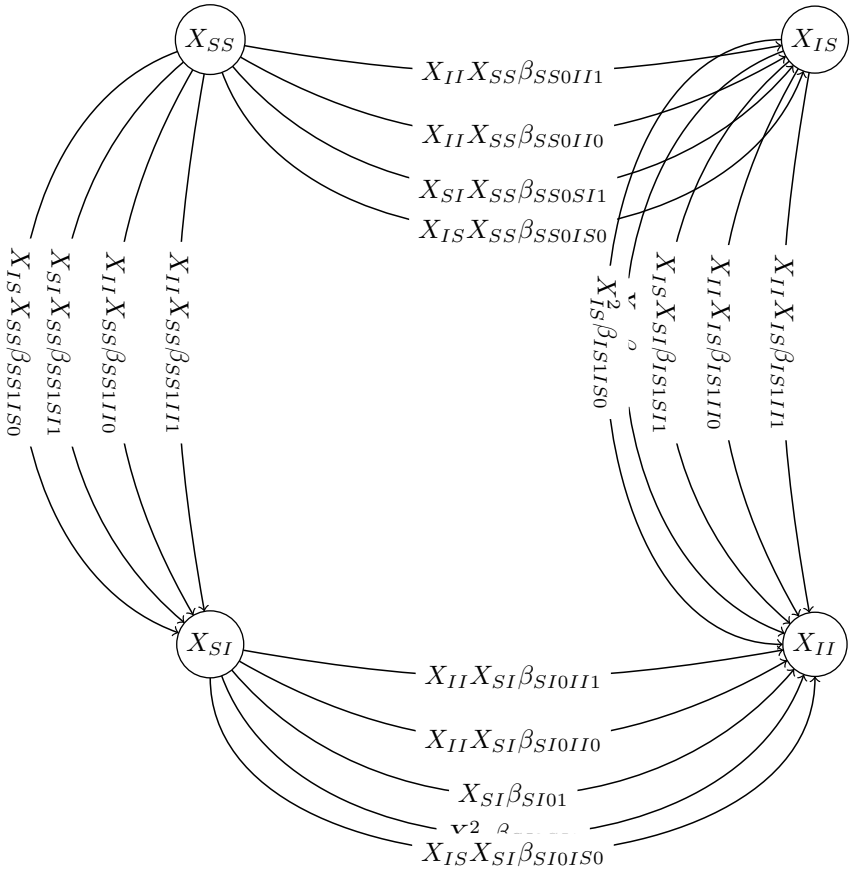
Example: Infectious dynamics of couples (ordered pairs)

Consider a system of pairs of individuals, where either can be susceptible or infected at any time. The above definition with the simple inclusion function Σ_1 produces this model:



This models pairs of individuals taken from two completely separate, non-interacting populations. The second individual of a pair can't infect the first individual of any pair. We want a model that allows interactions across that division, where each S individual in either position of any paired compartment can be infected by any I individual in either position of any compartment: there are four such S types and four such I types, so there will be 16 of these infection transitions.

In all we will have 18 transitions, including within-compartment infection events in compartments SI and IS. Using the full inclusion function Σ_c, which generates the full set of 18 interactions, we construct this model:



.

$$\begin{aligned}
\frac{dX_{SS}}{dt} &= -X_{II}X_{SS}\beta_{SS0II0} - X_{II}X_{SS}\beta_{SS0II1} - X_{IS}X_{SS}\beta_{SS0IS0} \\
&\quad - X_{SI}X_{SS}\beta_{SS0SI1} - X_{II}X_{SS}\beta_{SS1II0} - X_{II}X_{SS}\beta_{SS1III1} \\
&\quad - X_{IS}X_{SS}\beta_{SS1IS0} - X_{SI}X_{SS}\beta_{SS1SI1}
\end{aligned}$$

$$\begin{aligned}
\frac{dX_{SI}}{dt} &= -X_{II}X_{SI}\beta_{SI0II0} - X_{II}X_{SI}\beta_{SI0II1} - X_{IS}X_{SI}\beta_{SI0IS0} \\
&\quad - X_{SI}^2\beta_{SI0SI1} + X_{II}X_{SS}\beta_{SS1II0} + X_{II}X_{SS}\beta_{SS1III1} \\
&\quad + X_{IS}X_{SS}\beta_{SS1IS0} + X_{SI}X_{SS}\beta_{SS1SI1} - X_{SI}\beta_{SI01}
\end{aligned}$$

$$\begin{aligned}
\frac{dX_{IS}}{dt} &= -X_{II}X_{IS}\beta_{IS1II0} - X_{II}X_{IS}\beta_{IS1III1} - X_{IS}^2\beta_{IS1IS0} \\
&\quad - X_{IS}X_{SI}\beta_{IS1SI1} + X_{II}X_{SS}\beta_{SS0II0} + X_{II}X_{SS}\beta_{SS0II1} \\
&\quad + X_{IS}X_{SS}\beta_{SS0IS0} + X_{SI}X_{SS}\beta_{SS0SI1} - X_{IS}\beta_{IS10}
\end{aligned}$$

$$\begin{aligned}\frac{dX_{II}}{dt} = & X_{II}X_{IS}\beta_{IS1II0} + X_{II}X_{IS}\beta_{IS1II1} + X_{IS}^2\beta_{IS1IS0} \\ & + X_{IS}X_{SI}\beta_{IS1SI1} + X_{II}X_{SI}\beta_{SI0II0} + X_{II}X_{SI}\beta_{SI0II1} \\ & + X_{IS}X_{SI}\beta_{SI0IS0} + X_{SI}^2\beta_{SI0SI1} + X_{IS}\beta_{IS10} + X_{SI}\beta_{SI01}\end{aligned}$$

[Note: It's more common to consider pair models together with pair-formation and breakup events, with compartments for unpaired individuals. We can add this if needed.]

Example: Infectious dynamics of couples (unordered pairs)

[TODO]

If we want to consider pairs without distinguishing the two individuals by gender or otherwise differentiating the first from second individual, we reduce the model to fewer compartments – for instance, in our example, we will have one compartment that combines (S, I) and (I, S).

The unordered pair dynamics must be consistent with the ordered pair model because they are different descriptions of the same events. So we should be able to generate the unordered pair model from the ordered one, by combining compartments. We will define the unordered dynamics in that way, even though we're likely to end up generating them in a more direct way for efficiency.

The formal construction of the unordered model, then, is

- construct the ordered pair model
- collect the compartments of that model into equivalence classes: in the SI case, $\{\{(S,S)\},\{(S,I),(I,S)\},\{(I,I)\}\}$. Sum over those classes to generate the reduced compartments: in this case $Z_{SI} = X_{SI} + X_{IS}$, where we use the letter Z for reduced (unordered) compartment variables and X for the compartments of the ordered-pair model M^2.
- Construct reduced transition rates by transforming all X variables to Z variables, and assign their source and target vertices to the appropriate Z variables.

Note that this is a different way of combining compartments than the combining we did when constructing the ordered pair model. There we simply defined one compartment to be a synonym of the other and merged them by moving the arrows from one to the other; here we define a new compartment to be the sum of the old compartments and transform the transition rate expressions as well as relocating their arrows. Here is a definition for this operation:

An **aggregation of compartments** of a compartment model is a transformation $A(M)$ of that model by a pair of mappings $A : C_M \to C_A$ and $A : P_M \to P_A$, defined as for a relabeling, such that

- The compartments of $A(M)$ are the set $\{ A(C) | C \in C_M \}$
- There is an equation $C'_i = \sum C_j$ for each compartment C'_i of $A(M)$, where the sum is over all the compartments C_j such that $A(C_j) = C'_i$.
- The set of transitions of $A(M)$ is the set of triples $(A(s), A(t), (C'_1, \ldots, C'_{n'}, P'_1, \ldots, P'_{m'}) \mapsto \sum_{\{ (s',t',r') | A(s')=A(s), A(t')=A(t) \}} r'(C_1, \ldots, C_n, P_1, \ldots, P_m))$ generated by all transitions (s, t, r) of M. The sum is simplified to a function of the C'_i variables using the above sum equations.

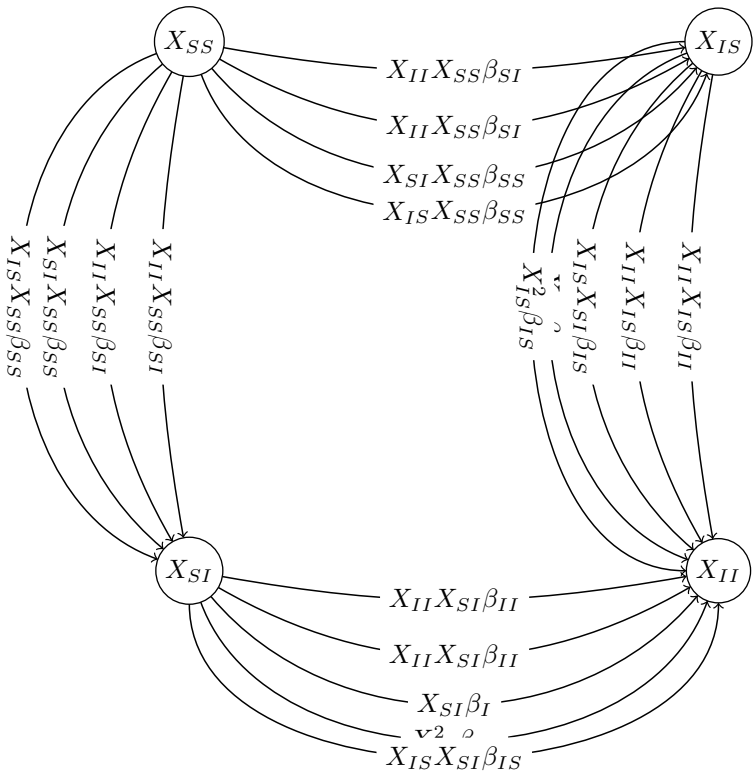
That is, the transitions of the model M are summed together where their sources and targets are mapped to the same places by the aggregation, and sums of compartments of M are simplified in terms of the compartments of $A(M)$. This reduction doesn't work for general box models, so this can only be done in particular cases. Also, it's probably necessary to relabel the parameters just so using $A(P_i)$, so that the sums will come out right.

Given that definition, we define a **sorting operation** Z such that $Z((S, I)) = Z((I, S)) = (S, I)$ (we sort the compartments in the order S, I, R). Using that as an aggregation function, the definition above gives us $Z_{SI} = X_{SI} + X_{IS}$, and we define the unordered pair model as

$M_U = Z(M^2)$

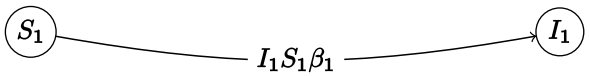
where M^2 is the ordered-pair model defined above. (Note: To make the transitions reduce to Z expressions we'll have to construct the right renaming for the indexed β parameters.)

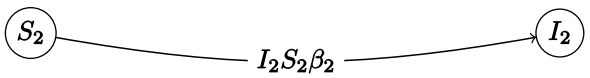
This will be a picture of the reduced pair model, with only 3 compartments, but it isn't yet:



Combining infection models into a multi-infection model

Suppose there are two infectious agents, numbered 1 and 2. Each is involved in an SI process (and maybe other compartments downstream from I). Let us call these models M_1 and M_2:

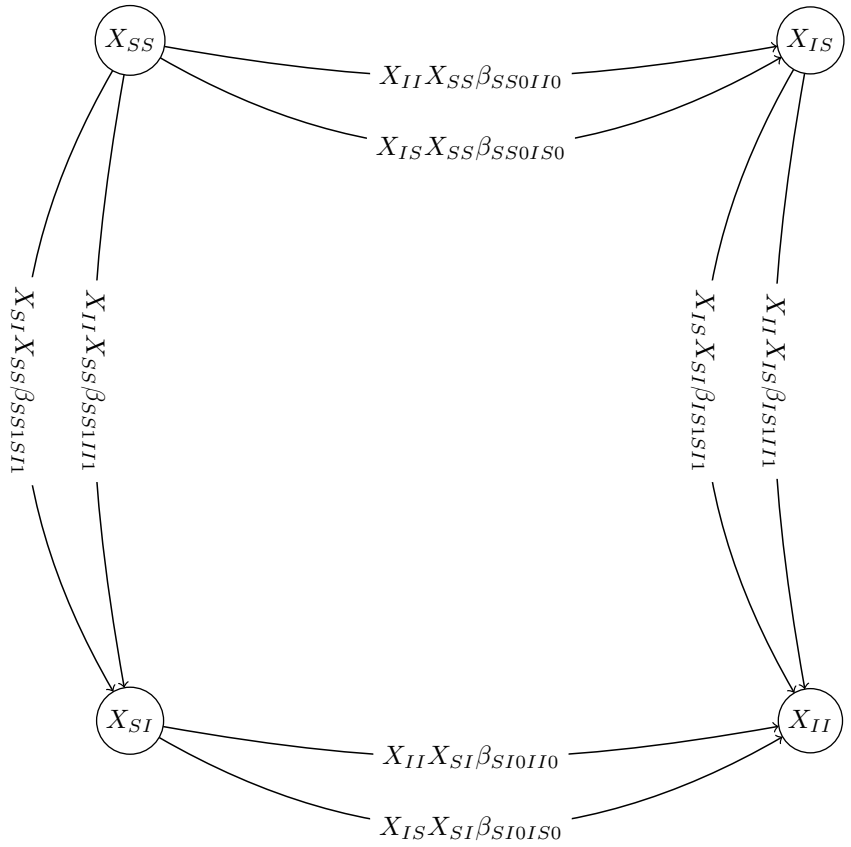




.

We wish to combine these infection processes into a single model M_{12}, in which a single susceptible population is infected by both agents, producing not only the I_1 and I_2 classes, but also one or more superinfected classes. Order of infection events might matter, in diseases in which an individual acquiring a second strain develops a weaker infection.

In this case, we need to use the simple inclusion function Σ_1, because there is no interaction between an individual infected with one strain and one susceptible to a different strain. With that, the cross product as defined gives us this product model:



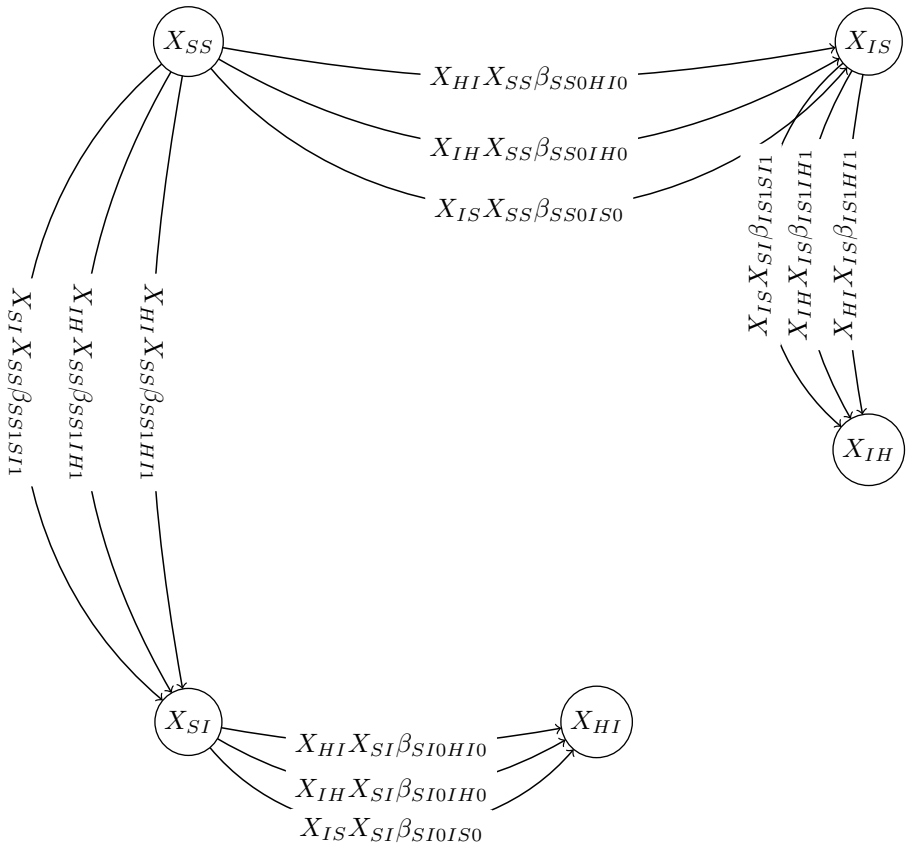
We need to modify this to account for order of infection, and to allow for the possibility that both strains can be transmitted in a single contact event.

[Note: we might want to explain why we didn't worry about double transmission events in the pair model]

To model partial superinfection, we need to introduce a "half-infected" class H, and provide a modified binary operation:

$$\Sigma_H(S, i_s, C, i_s, S, I, I, i) = \begin{cases} \{\,(S_1, \ldots, S_{i_s-1}, H, S_{i_s+1}, \ldots, S_n)\,\} & \text{if } S \text{ includes } I \\ \{\,(S_1, \ldots, S_{i_s-1}, I, S_{i_s+1}, \ldots, S_n)\,\} & \text{if not} \end{cases}$$

[TODO: the vertex set is generated by the Σ operation, not by the Cartesian product of vertex sets as claimed in the definition]



To account for simultaneous transmission of both strains, we need to construct a strong product rather than the cross product we have been studying. As described in the appendix, a strong product contains all the edges of the cross product, plus edges constructed from edges of multiple component graphs at once. In this model, the edges of the cross product are transitions where S changes to I in one or the other of the component models, i.e. where an individual acquires one of the disease strains. Transitions in which an individual acquires both strains in a single event are edges of the strong product.

The strong product of box models can be defined using most of the formal apparatus of the cross product, but with an edge generation function that generates derived edges from each combination of models' edges, rather than from each edge individually.

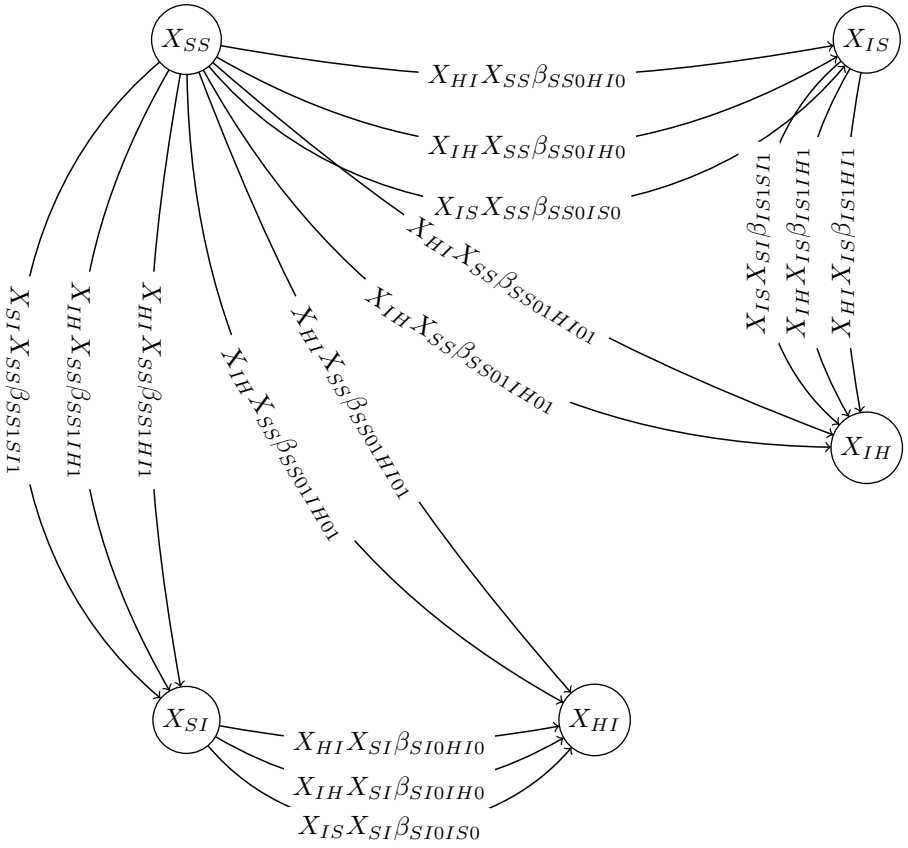
Definition. A *strong product* $M_1 \boxtimes \cdots \boxtimes M_n$ of box models $M_1, \ldots, M_n$ is a box model such that

- $V(M) = V(M_1) \times \cdots \times V(M_n)$

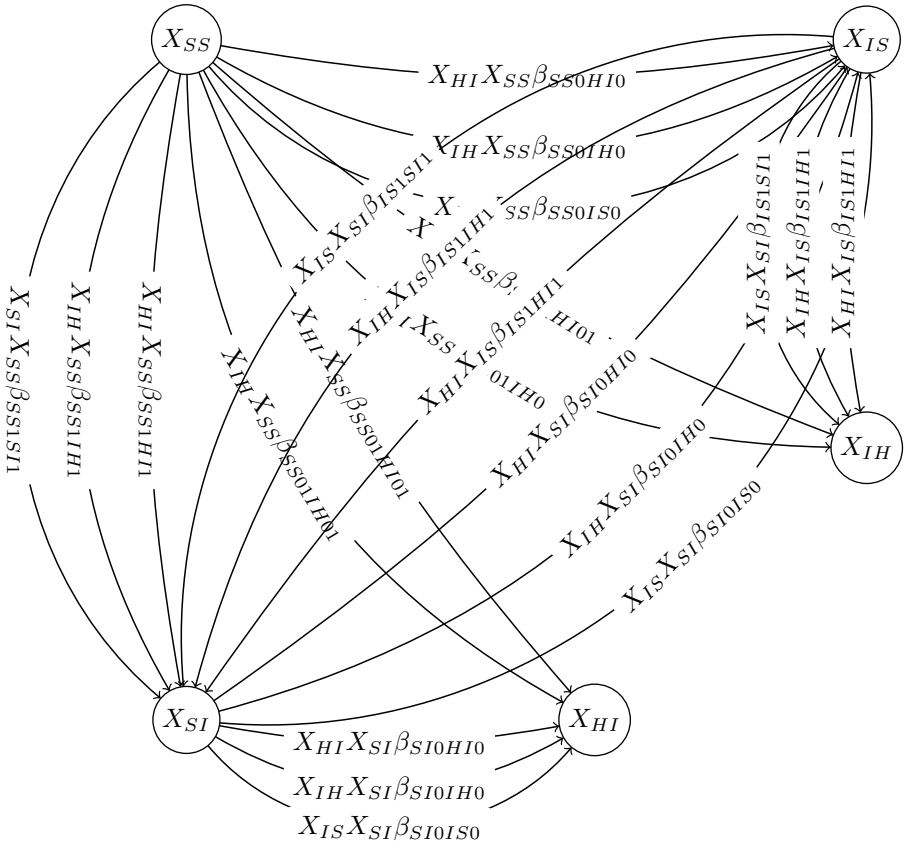
- $E(M) = \cup_{\sigma \subseteq \{1,\ldots,n\}} \cup_{\varepsilon \in \prod_{i \in \sigma} E(M_i)} \Xi_{\boxtimes}(\varepsilon)$
- $\Xi_{\boxtimes}(\varepsilon) = \begin{cases} \{\, (S, T, \tilde{\Phi}_r(r, S, \iota_s, T)) \mid S \in V(M), \{\, (S, i) \mid i \in \sigma \,\} \subseteq \Sigma(s, \{S\}), \\ \quad T \in \tilde{\Omega}_1(S, \iota_s, \varepsilon) \,\} \\ \quad \text{if } e = (s, t, r)\ \forall e \in \varepsilon \text{ and } r \text{ is linear in } s \\ \{\, (S, T, \tilde{\Phi}_{2r}(r, S, \iota_s, C, \iota_c, C)) \mid S \in V(M), \{S\} \times \sigma \subseteq \Sigma_c(s, \{S\}) \\ \quad C \in V(M), \{C\} \times \sigma \subseteq \Sigma_c(c, \{C\}), \\ \quad T \in \tilde{\Omega}_2(S, \{S\} \times \sigma, C, \{C\} \times \sigma, \varepsilon) \,\} \\ \quad \text{if } e = (s, t, r)\ \forall e \in \varepsilon \text{ and } r \text{ is bilinear in } s \text{ and } c \end{cases}$
- $\tilde{\Omega}_1$ and $\tilde{\Omega}_2$ generate the destination compartment for each transition given the transitions it's constructed from, and the source and catalyst compartments, and $\tilde{\Phi}_r$ and $\tilde{\Phi}_{2r}$ provide rewritten rate expressions.

This definition is surely opaque [and not actually correct: we distinguish the order of elements in ι_s and ι_c, we have strong products with and without cross interactions, and we may need within-compartment interactions for some applications-lw], but the upshot is that, in our example, for every way of locating one or more S labels in a compartment, and for every way of locating one or more I labels in a compartment, it uses the $\tilde{\Omega}_2$ function to generate a set of transitions resulting from the interactions of those pairs. In this model, we want the S and I labels to match up in the same positions (no cross interactions), so we provide an $\tilde{\Omega}_2$ function that yields outputs only in those cases.

Using a strong product allows us to produce this model:



We can construct products based on different assumptions by writing slightly different $\tilde{\Omega}_2$ functions. Here is a variant in which an encounter with an infected individual can knock out one or more strains, as well as infecting the contact with new ones:



Here's a variant in which it's also possible to get infected with both strains equally, in addition to having one infection weaker than the other:

Forward equations for a box model

Given a box model, rather than ODE as defined above, we can generate Kolmogorov forward and backward equations for the model. Both are ODEs, but the forward equations are the ODE of a box model, so we define that system as a box model.

Given an integer number N of total individuals, the compartments, that is the state variables, for the forward equations are $p_{(i_1/N,\ldots,i_n/N)}$ for each $(i_1,\ldots,i_n) \in \{0,1,\ldots,N\}^n$, where n is the number of compartments. If total mass is conserved, as it often is in box models, we reduce the dimension by eliminating the last of the model's compartments, and assume that the sum of all compartments is 1.

For each compartment X let $\mathbf{e}_X$ be the vector $(0, \ldots, 0, 1, 0, \ldots, 0)$ with 1 in the place corresponding to compartment X.

The transitions are

$(p_S, p_{S-\mathbf{e}_s/N+\mathbf{e}_t/N}, r(S)p_S)$

for each transition with source, target, and rate s, t, r of the original box model, and for every state S, where $r(S)$ is the rate r evaluated at the state S.

The forward equations themselves are the ODE of this box model, as defined above.

For example, for a simple SI model with transition βSI, the forward equations for $N = 3$ are

$$\begin{aligned}
\frac{dp_{01}}{dt} &= \frac{2}{9}\,\beta p_{\frac{1}{3}\frac{2}{3}} \\
\frac{dp_{\frac{1}{3}\frac{2}{3}}}{dt} &= -\frac{2}{9}\,\beta p_{\frac{1}{3}\frac{2}{3}} + \frac{2}{9}\,\beta p_{\frac{2}{3}\frac{1}{3}} \\
\frac{dp_{\frac{2}{3}\frac{1}{3}}}{dt} &= -\frac{2}{9}\,\beta p_{\frac{2}{3}\frac{1}{3}} \\
\frac{dp_{10}}{dt} &= 0
\end{aligned}$$

To do: forward equations of large power of infection model

In this section, we will document the Sage code (yet to be written) that implements large powers of a single model efficiently (as in the $SI \times SI$ pair model presented above), binds its transition rate parameters to something sane, and constructs its forward equations in a way that collapses to something usable.

Appendix: graph products

There are several standard product operations on directed graphs. All are ways of assigning arrows to the same set of vertices, which is the Cartesian product (cross product) of the component graphs' vertex sets. We use several of them in our definitions of products of box models.

The **Cartesian product of sets** S_1, S_2, ..., S_n is the set $S_1 \times S_2 \times \ldots \times S_n = \{\, (s_1, s_2, \ldots, s_n) \mid s_1 \in S_1, s_2 \in S_2, \ldots, s_n \in S_n \,\}$. The elements of

$S_1 \times S_2 \times \ldots \times S_n$ are referred to as **tuples** of elements of the component sets S_1, S_2, …, S_n.

We define a **directed graph** (and we are not concerned with undirected graphs) as a set $\{ (v, w, e) \} \subseteq V \times V \times E$, where V is the **vertex set** of the graph and E is its set of edge labels. Each of these tuples is visualized as an arrow from v to w with label e. [In our definition of box models, the elements of e are transition rates.] Note that this definition allows multiple edges between the same vertices v, w, so we are properly discussing directed multigraphs.

The cross product of graphs

The **cross product** $G_1 \,\square\, G_2 \,\square\, \cdots \,\square\, G_n$ **of directed graphs** G_1, G_2, …, G_n is a graph whose vertex set is the Cartesian product $V_1 \times V_2 \times \ldots \times V_n$ of the vertex sets V_i of each graph G_i, and whose edges are of the form $((v_1, v_2, \ldots, v_i, \ldots, v_n), (v_1, v_2, \ldots, w_i, \ldots, v_n), e)$, where the two tuples are identical in all but the i'th position, and where there is an edge connecting v_i to w_i in G_i.

Most definitions of this product do not provide labels for the edges of the product graph. In the body of this paper we construct these labels (transition rates) in a variety of ways.

The square symbol is chosen to suggest the shape of the cross product graph.

The tensor product

The **tensor product** of directed graphs G_1, …, G_n is a graph $G_1 \times \cdots \times G_n$ whose vertex set is the Cartesian product of the graphs' vertex sets, and which has an edge from $(v_1, v_2, \ldots, v_n)$ to $(w_1, w_1, \ldots, w_n)$ if and only if there is an edge from v_i to w_i in *every* graph G_i.

The tensor product is the natural product in the category of graphs (with graph homomorphisms as the category's morphisms), because the tensor product of graphs is the graph that has a homomorphism to each of the component graphs, and such that any other graph that has a homomorphism to each component also has a homomorphism to the tensor product graph that commutes.

In some sense, it might make sense to think of this product as embodying an "and" or "all" operation, where the Cartesian product embodies an "exactly one of" operation.

The strong product

The **strong product** $G_1 \boxtimes G_2$ of two graphs G_1, G_2 is the union of the Cartesian product and tensor product of the graphs. That is, it contains all the edges of those two product graphs.

If we extend that definition verbatim to n graphs, we get an "all or exactly one" product. In modeling infectious dynamics we are probably more interested in a "one or more" product. Therefore I prefer the following definition:

The **strong product** of graphs G_1, ..., G_n, written $G_1 \boxtimes G_2 \boxtimes \cdots \boxtimes G_n$, is the graph whose vertex set is the Cartesian product of the graphs' vertex sets, and which has an edge from $(v_1, \ldots, v_n)$ to $(w_1, \ldots, w_n)$ if and only if, for every i, either there is an edge from v_i to w_i, or $v_i = w_i$.

This graph is a subgraph of the transitive closure of the Cartesian product graph, but it is not the transitive closure, because it's limited to transitive combinations of at most one edge from each component graph.