

Toward a comprehensive system for constructing compartmental epidemic models

Darren Flynn-Primrose^{1*}, Steve Walker¹, Michael Li^{1,2},
Benjamin M. Bolker^{1,3}, David J. D. Earn¹, Jonathan Dushoff³

^{1*}Department of Mathematics & Statistics, McMaster University, 1280
Main Street West, Hamilton, Ontario, L8S 4K1, Canada.

²Public Health Risk Science Division, National Microbiology
Laboratory, Public Health Agency of Canada, 370 Speedvale Avenue
West, Guelph, Ontario, N1H 7M7, Canada.

³Department of Biology, McMaster University, 1280 Main Street West,
Hamilton, Ontario, L8S 4L8, Canada.

*Corresponding author(s). E-mail(s): flynnprd@mcmaster.ca;
Contributing authors: swalk@mcmaster.ca; bolker@mcmaster.ca;
earn@math.mcmaster.ca; dushoff@mcmaster.ca;

Abstract

Compartmental models are valuable tools for investigating infectious diseases. Researchers building such models typically begin with a simple structure where compartments correspond to individuals with different epidemiological statuses, e.g., the classic SIR model which splits the population into susceptible, infected, and recovered compartments. However, as more information about a specific pathogen is discovered, or as a means to investigate the effects of heterogeneities, it becomes useful to stratify models further — for example by age, geographic location, or pathogen strain. The operation of constructing stratified compartmental models from a pair of simpler models resembles the Cartesian product used in graph theory, but several key differences complicate matters. In this article we give explicit mathematical definitions for several so-called “model products” and provide examples where each is suitable. We also provide examples of model stratification where no existing model product will generate the desired result.

Keywords: epidemiology, transmission dynamics, compartmental model, graph theory, Cartesian product

MSC Classification: 92D30 , 92-10 , 05C76

1 Introduction

The COVID-19 pandemic has reemphasized the importance of compartmental models (Abou-Ismaïl, 2020; Massonis et al, 2021; Adam, 2020; Currie et al, 2020; Lofgren et al, 2014; McBryde et al, 2020; Enserink and Kupferschmidt, 2020) and has resulted in a flood of new compartmental models (Friston et al, 2020; Fields et al, 2021; Chang and Liu, 2022; Lavielle et al, 2020; Balabdaoui and Mohr, 2022; (e.g., Friston et al (2020); Fields et al (2021); Chang and Liu (2022); Lavielle et al (2020); Balabdaoui and Mohr (2022))). This abundance of new model variants is to be expected given the number of public health modelers seeking to integrate the current scientific understanding of emerging infectious diseases in a way that will have policy impact. Modelers must be able to build models rapidly to explore scenarios and generate high quality forecasts; public health recommendations have the biggest impact if they can be acted on promptly. However, the speed at which modelers can develop new models typically trades off with model quality. It is therefore useful to develop tools that make it easier for modelers to build high-quality models more quickly.

One way to address this speed-quality trade-off is to build infectious disease models incrementally. Information is scarce early in an epidemic, and so early models should be simple to reflect ignorance. As epidemics progress, more information is gathered and policy choices require fast input from scientists. Public health modelers will then need to quickly add complexity to their models if they are to be relevant to policy.

Proceeding in this way can eventually result in extremely complex models, much of whose complexity is no longer relevant. Complexity makes it more difficult to add new features to the model when they become necessary. Therefore, modelers need tools that make it easier to flexibly add and remove model features.

~~Savageau and Voit~~ (Savageau, 1988; Voit, 1988, 1990) Savageau (1988) and Voit (1988, 1990) made an early attempt to create such a toolbox by recasting the underlying differential equations of a model into a canonical form they call an “S-model”. Unfortunately this effort focused on the model’s differential equations rather than its graphical structure, thus making it unsuitable for less mathematically inclined modelers. It does not seem to have been widely adopted.

Friston et al (2020) describe how the state space of a complex epidemiological model can be constructed from the product of different latent state dimensions (their Figure 1: infection status, clinical status, testing status, and location), but the definition of which compartments are connected, and the rates of flow between them, is left up to the modeller.

A recent and promising effort to formalize the construction of compartmental models employs the language of category theory (Fong and Spivak, 2018; Libkind et al, 2022, 2021; Baez et al, 2022; Baez and Pollard, 2017). This powerful approach addresses many of the concepts we discuss here; however, at its current stage of development it requires considerable knowledge of advanced mathematics to use effectively. An ongoing project to implement the category theoretic approach in the Julia language can be found at <https://github.com/AlgebraicJulia/AlgebraicPetri.jl> (Halter et al, 2022).

Worden and Porco (2017) use the relatively simple language of graph theory to describe common methods of combining compartmental models. The current paper is

a result of the our efforts to implement the products described by Worden and Porco in software. At a high level, we view model “multiplication” as a three-step operation:

DJDE: I don’t think it is natural to annoint this as a formal mathematical Remark... We can hyperlink to it without this, as I’ve now done.

Procedure for Model Multiplication

1. Generate the vertices of the product model by combining the vertices of the factor models. This typically means taking the Cartesian product of the vertices of factor models. In many cases, we will want only a subset of the Cartesian product in the final model (e.g., some combinations are physically or biologically impossible). ~~(Subsetting is equivalent to setting the population of superfluous compartments to zero and disallowing any flows in or out of them.)~~
2. Generate the edges of the product model. Again, we will typically take the Cartesian product of edges in each factor model with the vertices in the other. Some transitions may be disallowed, in which case we would drop those edges from the product or set the flows across them to zero. In other cases, we may want to add edges to the product to allow state changes in multiple strata to occur simultaneously.
3. Resolve ~~the~~ ambiguities in how ~~the magnitudes of flow rates~~ flow functions are generalized to accommodate the presence of additional strata.
4. Set parameters for the product model

JD: What do we think of the new item? It would have been clarifying for me. The old way implies that we generally expect to just get new flow rates from old, which seemed hard; later it seemed like we knew even *less* about the new flow-rate parameters than I thought.

The combination of the first two steps is just a graph-theoretic Cartesian product. Recognizing the need to adjust flow rates (the third step above) is Worden and Porco’s contribution. This requirement can arise in a number of ways; for example, when stratifying a standard SIR model, we need to decide whether the susceptible population of one stratum can be infected by the infectious population of another stratum. If the stratification is based on age groups then it is reasonable to let different strata infect one another, i.e. old people infecting young people and *vice versa*. On the other hand, if the populations in different strata are isolated from each other it may be preferable to prohibit inter-stratum infection. Our approach follows Worden and Porco’s; when computing the magnitude of a flow between compartments we separately compute the contribution from each individual stratum and then sum the resulting quantities to find the total magnitude of the flow. There are other possible approaches which we will revisit in Section 5.1, but this additive approach appears to be the most common. When stated in these simple terms it is easy to imagine that we have solved the problem of combining models, but the devil is in the details. It is one thing to develop a framework for combining models and their various components but it is a much greater challenge to determine precisely which models can (and cannot) be expressed within that framework. JD: Ben and I discussed adding flows recently. Adding flows is generally what you want, but also a tiny bit weird. It works because FoI contributions are generally linear. In other words if $f(\sum w_x s_x) = \beta(\sum w_x s_x)$, then it also equals $\sum f(w_x s_x)$. In an ideal world (not sure if anyone is willing to take the time), we could

use the weighted states as a default, and point out the in a many models it’s equivalent to the WP special case. JD: It’s probably necessary to either drop weighted states, or adopt it, since we don’t really have “a” weighted-states product, right? Can’t any product have weighted-states or a summed-flows *version*?

Given the diversity of the audience potentially interested in this material we have attempted to strike a ~~healthy~~ balance between mathematical rigour and intuitive explanation. In Section 2 we introduce some mathematical terminology that will be used in the remaining sections to rigorously define a number of distinct model products. Sections 3 and 4 share similar structure, in the first subsection we focus on mathematical details and in the second subsection we provide examples which we hope will communicate our main points in a more intuitive way. Section 3 restates two products introduced by Worden and Porco (2017), Section 4 describes a generalization of the products from the previous section. Section 5 has three subsections each dedicated to different complexity that can make models challenging to construct using products alone. We conclude with Section 6 where we summarize our primary results and discuss the potential for further investigation.

2 Defining Compartmental Models

Compartmental models closely resemble directed graphs, which are graphs where the connections between nodes have a specific direction (Roberts and Tesman, 2009); in this analogy the directional connections correspond to flows between compartments. The relationship between directed graphs and compartmental models has been studied before, for example by Walter and Contreras (1999) — however, such investigations have been largely limited to the case where the magnitudes of flows between compartments are governed by linear equations. While certainly an important special case, this framework is insufficient for a disease-transmission model. In fact, a central point we hope to communicate in this paper is that, while constructing the nodes and edges of compartmental models is straightforward, making choices about how to calculate flows between compartments is a challenge with considerable nuance.

We consider a compartmental model with n compartments. Flows can go from one compartment to another, from outside the system into a compartment (e.g., births), or from a compartment to the outside (e.g., deaths). There are thus up to $n(n + 1)$ possible flows, although most of these will be missing in any particular model. Each flow that is present will be described by a function that may depend on the state of any of the n compartments and on any of m (possibly time-varying) parameters. Flows out of a compartment can be either *per capita* flows, in which case their value is multiplied by the current state value of that compartment, or they may be *absolute* flows where no further computation is required.

Historically much of the work done investigating compartmental models through a mathematical lens has been focused on the case where flow rates between compartments are restricted to linear equations (i.e., flows strictly proportional to the current state of the originating compartment). Formally, we let $\mathcal{F}(n, m)$ denote the set of allowable flow-rate functions $f : \mathbb{R}_{\geq 0}^n \times \mathbb{R}^m \rightarrow \mathbb{R}_{\geq 0}$; we will often abbreviate this as \mathcal{F} . More generally, we write the dimension of a model *state space* as $n = \mathfrak{s}(f)$ and of

the *parameter space* as $m = \mathbf{p}(f)$. It is also frequently convenient to write functions in the form $f(\vec{\beta}, \vec{x})$ where $\vec{\beta}$ is a vector in f 's parameter space and \vec{x} is a vector in f 's state space.

To move from flow-rate functions to actual flows, it is convenient to define a family of *filtering functions* which select specific subsets of the state space. We have two slightly different uses for these filtering functions: (1) we use them to select the “from” compartment of per-capita flows and (2) they will be used later to select compartments of a model product that belong to particular strata, for example selecting all compartments that correspond to a specific age group or location. We denote these families with the symbol Υ ; for example $\Upsilon(3, 2)$ is the set of all simple projections from \mathbb{R}^3 to \mathbb{R}^2 (in this case, the three ways of choosing two variables from a list of three). We also include in all such families the projection onto unity to allow for the possibility of flows specified in absolute rather than per-capita terms. A filtering function being used to select a single “from” compartment for a per-capita flow will be denoted with τ^{from} ; one being used to select all compartments from a single stratum of the model, which will typically then be used as arguments to a flow function, will be denoted with τ^{arg} .

We now have two families of functions. The first family is a set of flow rate functions, \mathcal{F} , which we will use to describe either the per-capita or absolute rate of flow between compartments. The second family is a set of filtering functions, Υ , which will be used to select the population of the “from” compartment for a given per-capita flow from all compartments in a model (in the case of absolute flow rates this function will map all inputs to 1). With these definitions we can now define our third and final family of functions which will be denoted by \mathfrak{F} and is used to describe the absolute rate of flow between compartments. Every function in \mathfrak{F} can be expressed as a product of one function in Υ and another in \mathcal{F} . That is, if f is a member of \mathfrak{F} then there must be a τ^{from} in Υ and an f_r in \mathcal{F} such that $f = \tau^{\text{from}} \cdot f_r$. Since Υ contains functions that select the population of a single compartment as well as the projection of state space onto the number 1, \mathfrak{F} will contain functions that describe the magnitude of a flow both in per-capita terms and in absolute terms. In the former case we will have $f = \tau^{\text{from}} \cdot f_r$ where τ^{from} corresponds to the population of the from compartment; in the latter case we will have the same equation but τ^{from} will reduce to the number 1, leaving f_r as the absolute flow rate between compartments. Below we give a more technical definition.

Definition 1 (Flow Functions). Given $n, m \in \mathbb{N}$ we define $\mathfrak{F}(n, m)$ to be the set of all functions $f : \mathbb{R}_{\geq 0}^n \times \mathbb{R}^m \rightarrow \mathbb{R}_{\geq 0}$ such that for some $f_r \in \mathcal{F}(n, m)$ and $\tau^{\text{from}} \in \Upsilon(n, 1)$, for every $\vec{\beta} \in \mathbb{R}^m$ and $\vec{x} \in \mathbb{R}_{\geq 0}^n$

$$f(\vec{\beta}, \vec{x}) = \tau^{\text{from}}(\vec{x}) \cdot f_r(\vec{\beta}, \vec{x}) \quad (1)$$

Remark 1 (Notation, Ordered sets and Labels). Where sets are labeled we assume their labels correspond to the order of the set (i.e. the first element in the set is labeled “1” and the i^{th} element is labeled “i”). Where sets are associated with each other we assume their labels (and/or order) correspond to the association map (i.e. if A and B are associated then $a_i \in A$ maps to $b_i \in B$ and vice versa).

A closely related concept to compartmental models is that of directed graphs (aka **digraphs**). Intuitively, digraphs are just graphs where the connections between vertices have a specific direction associated with them. A more technical definition is that a digraph consists of two sets: a set of vertices V and a set of ordered pairs of vertices E that defines directed connections between vertices. As we will see below, compartmental models are digraphs with an additional set of functions that regulate the magnitude of the flow across edges. Later when we discuss model products it will be helpful to recognize that the digraph underlying a product model is the Cartesian product of the digraphs underlying the factor models. First, though, we define compartmental models in the language developed thus far.

Definition 2 (Compartmental Models). Let $D_0 = (V, E)$ be a labeled, finite digraph and let $n = |V|$. Suppose there is a set $F \subset \mathfrak{F}(n)$ associated with E . Then we call $D = (D_0, F)$ a **compartmental model**. We call $\mathbb{R}_{\geq 0}^n$ the **state space of D** and if $m = \sum_{f \in F} \mathfrak{p}(f)$, then we call \mathbb{R}^m the **parameter space of D** .

2.1 Parameter Space of factor and product models

In Definition 2 and in much of the rest of this paper we assume the product model parameters are entirely independent of each other. So if one factor model has k_1 compartments and l_1 parameters and the other factor model has k_2 compartments and l_2 parameters then every the product model will have a minimum of $k_1 l_2 + k_2 l_1$ parameters (when no interaction between strata is allowed) and a maximum of $k_1^2 l_2 + k_2^2 l_1$ parameters (when all strata are allowed to interact with each other). This approach has the advantage of preserving generality.

In practice, parameters in the product model are often related to parameters in the original model factors in simple mechanistic ways. However, there is an enormous range of possible relationships between the parameters of the factor models and the parameters of their product. Some parameters, such as those describing intrinsic properties of a pathogen, may be constant across all strata of a product model. Others such as recovery time may be constant with respect to some dimensions of stratification (e.g. location) but variable with respect to others (e.g. age). In other cases the relationships may depend on the degree of available data — we may know that recovery time varies with age in reality, but choose to treat it as constant for modeling purposes. It does not appear possible, with just the information present in the factor models, to deduce the desired relationship between factor model and product model parameters. Thus we choose for now to default to the most general possible case and trust that where more convenient relationships between parameters exist modelers will construct appropriate mappings for themselves. That said, we will discuss a few common scenarios here for the purpose of illustration.

A parameter may be constant across multiple strata of a product model. This is common for parameters that describe intrinsic properties of the pathogen being modeled, where the strata represent variation among hosts. A related case occurs when the value of a parameter at each stratum in the product model is related to the factor model version of the parameter by a simple scalar. For example a pathogen may have an average recovery time across the entire population and the specific recovery

time for different age groups could be specified as a percentage of the overall average. In both these cases if we let $\alpha \in \mathbb{R}$ denote the parameter value in the factor model and $\vec{\beta} \in \mathbb{R}^k$ denote the values of the derived parameters at k different stratum in the product model then we can write $\vec{\beta} = \alpha \vec{w}$ where $\vec{w} \in \mathbb{R}^k$ is a vector of weights.

Multiple parameters in a factor model may be related to each other, as when a single compartment has multiple flows emanating from it. Consider the case of a person who has been exposed to a pathogen. Some models will allow them several possible fates: for example they could go on to be asymptomatic or have mild or severe symptoms. In that case the factor model in question will have three parameters ($\alpha_1, \alpha_2, \alpha_3 \in (0, 1)$) related by the fact that their sum must be one ($\alpha_1 + \alpha_2 + \alpha_3 = 1$). In other words, if α_1 and α_2 are given then $\alpha_3 = 1 - \alpha_1 - \alpha_2$. When included in a product model every stratum of the new model will have three parameters derived from the original α values; however each stratum may have different values for those parameters. For example people in different age groups may be more or less likely to experience severe, mild, or no symptoms. Mathematically, if we say $\vec{\beta}_1, \vec{\beta}_2, \vec{\beta}_3$ denote the parameters at every stratum of the product model derived from $\alpha_1, \alpha_2, \alpha_3$ respectively, then we can write $\vec{\beta}_1 = \alpha_1 \vec{w}_1$, $\vec{\beta}_2 = \alpha_2 \vec{w}_2$, and $\vec{\beta}_3 = \vec{1} - \vec{\beta}_1 - \vec{\beta}_2$.

A more complex case occurs when different strata of a product model interact. For example consider a simple SI model, that is, a model with only susceptible and infected compartments. In the standard formulation the force of infection of such a model is given by $\Lambda = \frac{\beta I}{N}$, so the total number of newly infected people is $S \cdot \Lambda = \frac{\beta SI}{N}$. Suppose we now stratify this model to represent a scenario where each person lives in one of three different locations but may come in contact with people living in the other locations. Our model would then have three infected compartments (i.e. $\vec{I} = (I_1, I_2, I_3)$) and three susceptible compartments (i.e. $\vec{S} = (S_1, S_2, S_3)$); the force of infection would be generalized to a vector with three entries, one for each location (i.e. $\vec{\Lambda} = (\lambda_1, \lambda_2, \lambda_3)$). In the most general case, where the force of infection does not take the standard form given above, each λ_i would be expressed as *some* function of the infected populations as well as a vector of parameters $\vec{\beta}_i$ which gives some information about how people at different locations interact with each other. Thus we would be left with $\vec{\Lambda} = (f(\vec{\beta}_1, \vec{I}), f(\vec{\beta}_2, \vec{I}), f(\vec{\beta}_3, \vec{I}))$. In the standard formulation the force of infection is a linear equation with respect to the population of infected compartments, so we can be more specific. The factor model parameter β is generalized to nine new, presumed unrelated, parameters which can be written as a 3×3 matrix

$$B = \begin{pmatrix} \beta_{11} & \beta_{12} & \beta_{13} \\ \beta_{21} & \beta_{22} & \beta_{23} \\ \beta_{31} & \beta_{32} & \beta_{33} \end{pmatrix}$$

with the end result that we can write the expression

$$\vec{\Lambda} = \frac{1}{N} B \vec{I}$$

While it still preserves a degree of generality, this approach unfortunately expands the model's parameter space significantly. In practice the likelihood of a person residing in

one location coming into contact with a person in a different location is not arbitrary but can reasonably be expected to vary according to the distance between the two locations. So if D is a three-by-three matrix where the $d_{i,j}$ entry denotes the distance between location i and location j then it is possible to construct a contact matrix C that assigns numerical values to the likelihood of contact between people at two locations based on the known distance between them. One way to do this would be to let every entry $c_{i,j}$ in C be given by

$$c_{i,j} = e^{-\gamma d_{i,j}}$$

where $\gamma \in \mathbb{R}_{\geq 0}$ is some fixed parameter. In this way we can write

$$\vec{\Lambda} = \frac{\beta}{N} C \vec{I}$$

which preserves the original meaning of the parameter β and only introduces one new parameter (γ) instead of nine. Notice that it is possible to translate between the two approaches using the map

$$B = \beta C = \beta \begin{pmatrix} e^{-\gamma d_{1,1}} & e^{-\gamma d_{1,2}} & e^{-\gamma d_{1,3}} \\ e^{-\gamma d_{2,1}} & e^{-\gamma d_{2,2}} & e^{-\gamma d_{2,3}} \\ e^{-\gamma d_{3,1}} & e^{-\gamma d_{3,2}} & e^{-\gamma d_{3,3}} \end{pmatrix}$$

(BB: there are other ways to handle this kind of parameter simplification (construction of WAIFW matrices); see e.g. <https://github.com/bbolker/bbmisc/blob/master/solve-beta.Rnw> (section 3), and possibly use references therein?) There are of course other ways to handle this kind of parameter simplification, see for example [Anderson and May \(1985, 1992\)](#); [Grenfell and Anderson \(1985\)](#). Most situations will allow for a parameter space mapping of this kind that relates the default parameter space generated by model products to a smaller parameter space dictated by the specific data available to the modeler. However, as we have tried to make clear in this subsection, the sheer variety of parameter space mappings that could be useful in some situation is so great that the only way we can confidently maintain generality is by treating all parameters in the product model as independent.

In our general parameterization the l_i parameters in the product model that come from the i 'th factor model (of two) can be organized in a $l_i \times k_{2-i} \times k_{2-i}$ order three tensor, $B^{(i)}$. The modeler will have some set of parameters known to them which we call $\vec{\theta}$ and will be able to compose, from a library of standard relations, a mapping g so that

$$B_{hij}^{(i)} = g_{hij}^{(i)}(\vec{\theta})$$

3 Naive and Modified Products

[Worden and Porco \(2017\)](#) describe several distinct model products, two of which are particularly relevant to us. Here, we would like to restate these two definitions in a mathematically rigorous way. These formal definitions provide a foundation for us to

define a new, generalized version of a model product in Section 4 and have been valuable in our efforts to automate the process of model stratification. However, we will begin with a description of the Cartesian product of digraphs as they are a significant component of model products; treating them separately will simplify our later definitions.

Definition 3 (Cartesian Product of Digraphs). Let $D^a = (V^a, E^a)$ and $D^b = (V^b, E^b)$ be digraphs with vertices V^\bullet and edges E^\bullet . Then the **Cartesian Product** of D^a and D^b (i.e. $D^a \times D^b$) can be written as $D = (V, E)$ where $V = V^a \times V^b$ and E is the union of two sets E^c and E^d which are given by

$$E^c = \{((w, x), (y, x)) \in V \times V \mid (w, y) \in E^a\} \quad (2)$$

and

$$E^d = \{((w, x), (w, z)) \in V \times V \mid (x, z) \in E^b\} \quad (3)$$

respectively.

It is not always the case that the digraph underlying a product model is the Cartesian product of the digraphs underlying the factor models, one example is the so-called “strong product” defined in Worden and Porco (2017) which includes additional edges that are not present in the Cartesian product. Alternatively, as we shall see in Section 5.3, there are some cases where the digraph underlying the product model is a proper subset of the Cartesian product of the factor models. This is particularly the case when combining models with different pathogen strains while disallowing the possibility of being infected by multiple strains simultaneously. However we believe the practical applications of these non-Cartesian model products are relatively limited; in this article we focus on model products where the related digraph product is Cartesian. Worden and Porco (2017) discuss two Cartesian-like products; since the underlying digraphs of these products are identical, the only difference between them is the set of flow functions in the product model (with respect to the procedure for model multiplication they agree on the first two steps and only differ on the third). Notice that in Definition 3 the edge set of the product digraph is the union of two other sets E^c and E^d . The set of flow functions of a model product F will be formed by a similar union of sets F^c and F^d , where F^c contains the flow functions related to edges in E^c and F^d contains those related to E^d .

Definition 4 (Naive and Modified Products). Suppose we have two compartmental models $D^a = (V^a, E^a, F^a)$ and $D^b = (V^b, E^b, F^b)$. Let V , E^c , E^d , and E be as in Definition 3. Since E is the union of E^c and E^d every edge $e \in E$ is associated with an edge in one factor model and a vertex in the other; we use $p(e)$ to denote the associated factor model edge and $s(e)$ to denote the associated factor model vertex. For example if $e \in E^c$, then $p(e) = (w, y) \in E^a$ and $s(e) = x \in V^b$ (Diagram 1). By Definition 2 for every $e \in E^a$ there are functions $f_e^a \in F^a$ and $f_{r,e} \in \mathcal{F}(|V^a|)$, as well as a filtering function $\tau_e^{\text{from}} \in \mathcal{T}(|V^a|, 1)$ such that

$$f_e^a(\vec{\beta}, \vec{x}) = \tau_e^{\text{from}}(\vec{x}) \cdot f_{r,e}(\vec{\beta}, \vec{x}). \quad (4)$$

Let τ_e^{arg} be the filtering function from V to $V^a \times s(e)$. For every $e \in E^c$ and $\vec{x} \in \mathbb{R}_{\geq 0}^{|V|}$ we define $f_e^1 : \mathbb{R}^{\mathfrak{p}(f_{p(e)}^a)} \times \mathbb{R}_{\geq 0}^{|V|} \rightarrow \mathbb{R}_{\geq 0}$ by

$$f_e^1(\vec{\beta}, \vec{x}) = \tau_{p(e)}^{\text{from}}(\tau_e^{\text{arg}}(\vec{x})) \cdot f_{r,p(e)}(\vec{\beta}, \tau_e^{\text{arg}}(\vec{x})) \quad (5)$$

and we define $F^{1,c} = \bigcup_{e \in E^c} f_e^1$. A symmetric definition can be given for $e \in E^d$ yielding $F^{1,d} = \bigcup_{e \in E^d} f_e^1$. We define $F^1 = F^{1,c} \cup F^{1,d}$ and we say $D^1 = (V, E, F^1)$. We define

$$D^a \boxminus D^b = D^1 \quad (6)$$

We call \boxminus **the naive product**.

Alternatively, we can for every $v^b \in V^b$ let $\tau_{v^b}^{\text{arg}}$ be the filtering function from V to $V^a \times v^b$. For every $e \in E^c$ and $\vec{x} \in \mathbb{R}_{\geq 0}^{|V|}$ we define $f_e^2 : \mathbb{R}^{\mathfrak{p}(f_{p(e)}^a) \times |V^b|} \times \mathbb{R}_{\geq 0}^{|V|} \rightarrow \mathbb{R}_{\geq 0}$ by

$$f_e(\vec{\beta}, \vec{x}) = \tau_{p(e)}^{\text{from}}(\tau_{s(e)}^{\text{arg}}(\vec{x})) \sum_{v^b \in V^b} f_{r,p(e)}(\vec{\beta}_{v^b}, \tau_{v^b}^{\text{arg}}(\vec{x})) \quad (7)$$

where we have used the convention that $\vec{\beta} \in \mathbb{R}^{\mathfrak{p}(f_{p(e)}^a) \times |V^b|}$ can be expressed as $\vec{\beta} = (\vec{\beta}_{v_1^b}, \dots, \vec{\beta}_{v_{|V^b|}^b})$ and for every $i \in \mathbb{N}_{|V^b|}$, $\vec{\beta}_{v_i^b} \in \mathbb{R}^{\mathfrak{p}(f_{p(e)}^a)}$. We define $F^{2,c} = \bigcup_{e \in E^c} f_e^2$.

A symmetric definition can be given for $e \in E^d$ yielding $F^{2,d} = \bigcup_{e \in E^d} f_e^2$. We define $F^2 = F^{2,c} \cup F^{2,d}$ and we say $D^2 = (V, E, F^2)$. We define

$$D^a \square D^b = D^2. \quad (8)$$

We call \square **the modified product**.

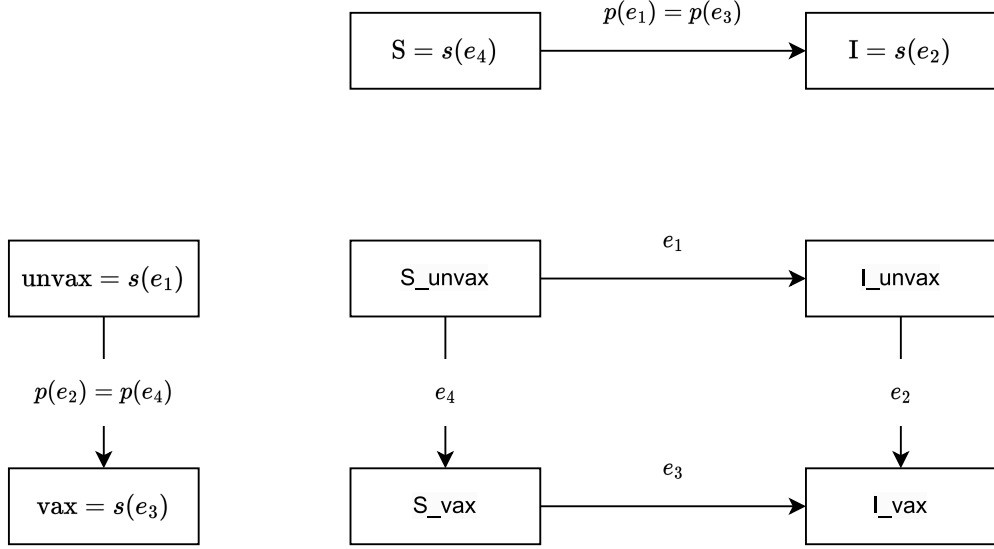


Fig. 1 A Susceptible-Infected model combined with a Unvaccinated-Vaccinated model. Every edge in the product model maps to an edge in one of the factor models (via the mapping p) and a vertex in the other factor model (via the mapping “ s ”).

Given two models, their naive and modified products will have identical vertex and edge sets. In fact, the only difference between the two is their flow functions.

3.1 Example Naive and Modified Products

We now describe specific examples of naive and modified products. We will note here that in our example models we have erred on the side of simplicity. We occasionally omit minor details which, while important for the actual job of modeling, are unnecessary and potentially distracting for the purpose of clearly communicating our point.

To illustrate the difference between naive and modified products, consider the two compartmental models in Figure 2. The top model tracks the epidemiological status of a population of susceptible, infected, and recovered individuals, and can be written in our notation as $G = (V^g, E^g, F^g)$ where $V^g = \{S, I, R\}$, $E^g = \{(S, I), (I, R)\}$, and $F^g = \{S \frac{\beta I}{S+I+R}, I\gamma\}$. The bottom model divides the population into three age groups, young, medium, and old, and can be written as $H = (V^h, E^h, F^h)$ where $V^h = \{Y, M, O\}$, $E^h = \{(Y, M), (M, O)\}$, and $F^h = \{\frac{Y}{\mu}, \frac{M}{\mu}\}$, where μ is the expected residency time for people in the young and medium age groups. We can then compute both $G \boxtimes H = M$ (Figure 4) and $G \boxdot H = N$ (Figure 3). Note that $V^m = V^n$ and $E^m = E^n$; the only difference between M and N is the difference between F^m and F^n . So

$$V^m = V^n = \{SY, SM, SO, IY, IM, IO, RY, RM, RO\} \quad (9)$$

$$E^m = E^n = \{(SY, IY), (IY, RY), (SM, IM), (IM, RM), (SO, IO), (IO, RO), (SY, SM), (SM, SO), (IY, IM), (IM, IO), (RY, RM), (RM, RO)\} \quad (10)$$

$$F^m = \left\{ SY \left(\frac{\beta_1 IY}{SY + IY + RY} + \frac{\beta_2 IM}{SM + IM + RM} + \frac{\beta_3 IO}{SO + IO + RO} \right), IY\gamma_1, \right. \\ SM \left(\frac{\beta_1 IY}{SY + IY + RY} + \frac{\beta_2 IM}{SM + IM + RM} + \frac{\beta_3 IO}{SO + IO + RO} \right), IM\gamma_2, \\ SO \left(\frac{\beta_1 IY}{SY + IY + RY} + \frac{\beta_2 IM}{SM + IM + RM} + \frac{\beta_3 IO}{SO + IO + RO} \right), IO\gamma_3, \\ \left. \frac{SY}{\mu}, \frac{SM}{\mu}, \frac{IY}{\mu}, \frac{IM}{\mu}, \frac{RY}{\mu}, \frac{RM}{\mu} \right\} \quad (11)$$

$$F^n = \left\{ SY \frac{\beta_1 IY}{SY + IY + RY}, IY\gamma_1, SM \frac{\beta_2 IM}{SM + IM + RM}, IM\gamma_2, SO \frac{\beta_3 IO}{SO + IO + RO}, IO\gamma_3, \right. \\ \left. \frac{SY}{\mu}, \frac{SM}{\mu}, \frac{IY}{\mu}, \frac{IM}{\mu}, \frac{RY}{\mu}, \frac{RM}{\mu} \right\} \quad (12)$$

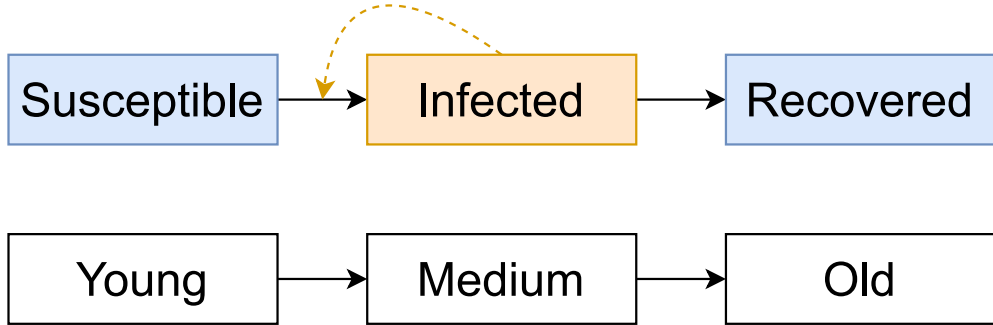
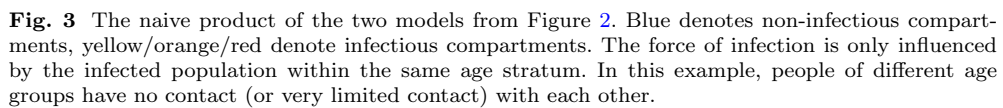


Fig. 2 A standard SIR model and a simple age stratification model. Orange shading denotes the infectious compartment, while blue shading denotes non-infectious compartments.



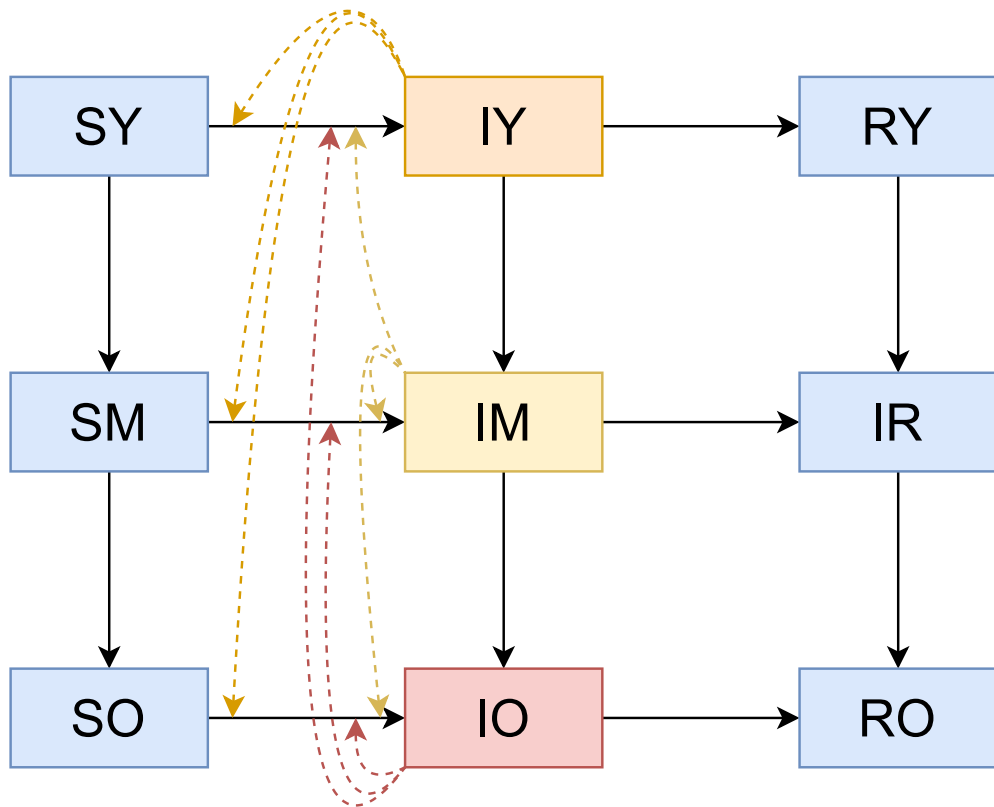


Fig. 4 The modified product of the two models from Figure 2. Unlike in Figure 3, individuals make epidemiological contacts across age strata, so the force of infection for each age stratum is influenced by the infected population in all age strata.

The term “naive product” is not pejorative; in this specific example the modified product is likely to be preferred because people of all ages commonly interact with each other. In some scenarios, however, the naive product would be preferred. In the case of spatial stratification, for example, one might want to use the naive, the modified, or another alternative depending on the specifics of the epidemiological system. At first glance the naive product seems promising because it incorporates the idea that people at different physical locations cannot interact and so do not infect one another. This, however, assumes that the model in question simulates movement explicitly which is not always the case. In product models that do simulate movement explicitly (e.g. [Mohammadi et al, 2023](#)), the flows between different locations are included in the original factor model describing spatial structure. Other models, such as [Dietz and Sattenspiel \(1995\)](#), model movement implicitly. In this sort of model a person’s location stratum might determine where they live but the possibility of their movement to another location is represented as a contact rate between people in their home stratum and the stratum they might visit. In this case the modified product would seem the most appropriate. However, the situation can be even more complex; for example, depending on the scales involved, it may be beneficial to allow contact between people who are either in the same location *or* in neighbouring locations, for some appropriate definition of a neighbourhood. In this case we want to go beyond the naive or modified products to some kind of *generalized product*.

4 Generalized Product

The naive product restricts people in each stratum to interacting only with other people in the same stratum; the modified product allows people in any strata to interact. We propose a new product that allows for people in each stratum to interact with people in an arbitrary subset of the other strata. This allows the creation of a model where people at a given location can interact at the same location or neighboring locations. The following definition is very similar to Definition 4; however unlike Equation 7, where the sum is over all vertices in a factor model, the sum in this definition will only include a subset of a factor model’s vertices.

Definition 5 (Generalized Product). Let $D^a = (V^a, E^a, F^a)$ and $D^b = (V^b, E^b, F^b)$ be two compartmental models and let V , E , E^c , and E^d be as in Definition 3. As in Definition 2 we know that every edge $e \in E$ is associated with an edge in one factor model and a vertex in the other; we use $p(e)$ to denote the associated factor model edge and $s(e)$ to denote the associated factor model vertex. For every $e \in E^a$ there are functions $f_e^a \in F^a$ and $f_{r,e} \in \mathcal{F}(|V^a|)$, as well as a filtering function $\tau_e^{\text{from}} \in \Upsilon(|V^a|, 1)$ such that

$$f_e^a(\vec{\beta}, \vec{x}) = \tau_e^{\text{from}}(\vec{x}) \cdot f_{r,e}(\vec{\beta}, \vec{x}). \quad (13)$$

For every $v^b \in V^b$ let $\tau_{v^b}^{\text{arg}}$ be the filtering function from V to $V^a \times v^b$ and let $V_{v^b} \subset V^b$ denote those vertices in V^b that belong to strata that interact with the $V^a \times v^b$ stratum. For every $e \in E^c$ and $\vec{x} \in \mathbb{R}_{\geq 0}^{|V|}$ we define $f_e : \mathbb{R}^{\mathbf{p}(f_{p(e)}) \times |V_{v^b}|} \times \mathbb{R}_{\geq 0}^{|V|} \rightarrow \mathbb{R}_{\geq 0}$ by

$$f_e(\vec{\beta}, \vec{x}) = \tau_{p(e)}^{\text{from}}(\tau_{s(e)}^{\text{arg}}(\vec{x})) \sum_{v^b \in V_{v^b}} f_{r,p(e)}(\vec{\beta}_{v^b}, \tau_{v^b}^{\text{arg}}(\vec{x})) \quad (14)$$

We define $F^c = \bigcup_{e \in E^c} f_e$. A symmetric definition can be given for $e \in E^d$ yielding $F^d = \bigcup_{e \in E^d} f_e$. We define $F = F^c \cup F^d$ and we say $D = (V, E, F)$. We define

$$D^a \boxtimes D^b = D. \quad (15)$$

We call \boxtimes **the generalized product**.

4.1 Example Generalized Products

Below we show three different ways an SI model could be stratified with location. Figures 5 and 6 show the naive and modified products respectively. Figure 7 shows one example of a generalized product where interactions can only occur within a single geographic region or between neighboring regions. So for example an infected person in the Toronto region could infect a susceptible person in Toronto or Ottawa but not one in Montreal. However, an infected person in Ottawa could infect a susceptible person anywhere.

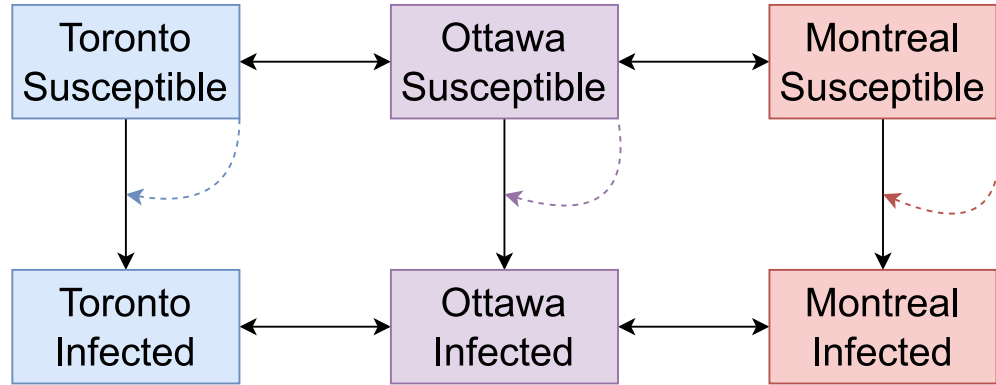


Fig. 5 The naive product of an SI model with location model including Toronto, Ottawa, and Montreal. Notice the force of infection at any given location is influenced by the size of the infectious population at the same location. This approach is useful when movement between locations is being modeled explicitly

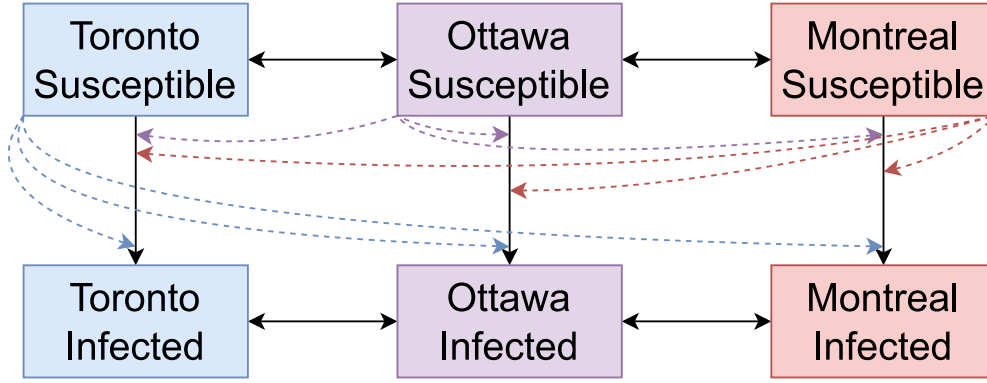


Fig. 6 The modified product of an SI model with location model including Toronto, Ottawa, and Montreal. Notice that the size infectious population at each location influences the force of infection at every location. This approach is useful when movement between location is modeled implicitly, for example through non-zero contact rates between populations at different locations

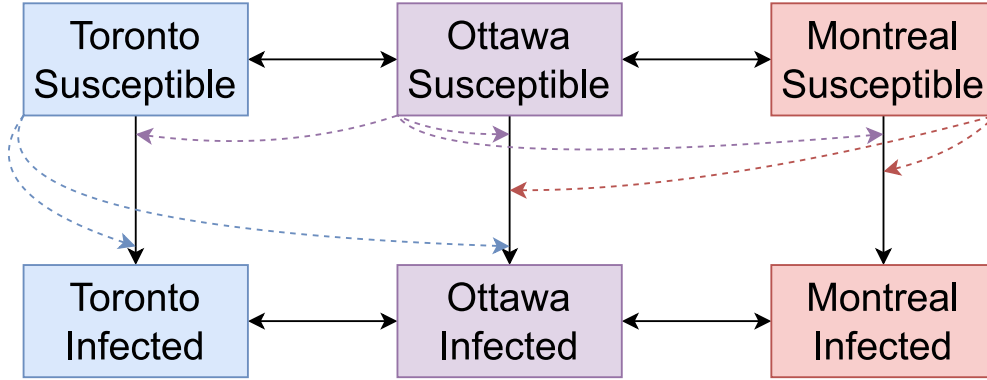


Fig. 7 A generalized product of an SI model with location model including Toronto, Ottawa, and Montreal. Notice that the force of infection at a given location depends on the size of the infectious population at the same location *and* at neighboring locations. This approach has a wide variety of applications, for example when there is significant variance in the distance between different locations

5 Uncooperative Examples

While the operations defined above allow us to construct a wide range of compartmental models by taking products of simpler factor models, they cannot account for every possible model. In this section we discuss a number of examples where products alone, as they appear in this article, are insufficient.

5.1 Models with alternate functional forms

Although we have striven in this article to make the functional forms we use as general as possible there are still certain approaches that do not fit neatly into the formalism

we have developed. Fortunately the difficulties that arise from this can typically be overcome by designing a new model product that possesses the desired properties. As we stated in Section 1 in this paper we have followed the approach of Worden and Porco when computing the magnitude of flows in a product model. That is, we individually compute the contribution from every stratum and compute the sum. In an alternate approach, we instead take a weighted average of the compartment populations in each stratum and use this new average as the input to the flow rate function. This approach is particularly useful when incorporating inhibitory influences in a model. For example during an epidemic will be more careful if they know hospitals are at capacity than they would be when there are ample medical resources available. We introduce the following product definition:

Definition 6 (Weighted States Product). Let $D^a = (V^a, E^a, F^a)$ and $D^b = (V^b, E^b, F^b)$ be two compartmental models and let V , E , E^c , and E^d be as in Definition 3. As in Definition 2 we know that every edge $e \in E$ is associated with an edge in one factor model and a vertex in the other; we use $p(e)$ to denote the associated factor model edge and $s(e)$ to denote the associated factor model vertex. For every $e \in E^a$ there are functions $f_e^a \in F^a$ and $f_{r,e} \in \mathcal{F}(|V^a|)$, as well as a filtering function $\tau_e^{\text{from}} \in \Upsilon(|V^a|, 1)$ such that

$$f_e^a(\vec{\beta}, \vec{x}) = \tau_e^{\text{from}}(\vec{x}) \cdot f_{r,e}(\vec{\beta}, \vec{x}). \quad (16)$$

For every $v^b \in V^b$ let $\tau_{v^b}^{\text{arg}}$ be the filtering function from V to $V^a \times v^b$. For every $e \in E^c$ and $\vec{x} \in \mathbb{R}_{\geq 0}^{|V|}$ we define $f_e : \mathbb{R}^{\mathfrak{p}(f_{p(e)}^a) \times |V_{v^b}|} \times \mathbb{R}_{\geq 0}^{|V|} \rightarrow \mathbb{R}_{\geq 0}$ by

$$f_e(\vec{\beta}, \vec{x}) = \tau_{p(e)}^{\text{from}}(\tau_{s(e)}^{\text{arg}}(\vec{x})) f_{r,p(e)} \left(\vec{\beta}_{s(e)}, \sum_{v^b \in V^b} w_{v^b} \tau_{v^b}^{\text{arg}}(\vec{x}) \right) \quad (17)$$

where for every $v^b \in V^b$, $w_{v^b} \in \mathbb{R}_{\geq 0}$ denotes the weight to be given to the v^b strata in the weighted average. We define $F^c = \bigcup_{e \in E^c} f_e$. A symmetric definition can be given for $e \in E^d$ yielding $F^d = \bigcup_{e \in E^d} f_e$. We define $F = F^c \cup F^d$ and we say $D = (V, E, F)$. We define

$$D^a \triangle D^b = D. \quad (18)$$

We call \triangle **the weighted states product**.

Alternatively, one might notice that if a flow rate function involves normalization, when computing the contribution to the flow of each stratum individually the normalization is done with respect to the population of the individual strata rather than the total population of the model. One example of this can be seen in Equations 11 and 12. To state this difference more clearly we introduce a new function N which simply sums the population of every compartment in a state vector. So for $\vec{x} \in \mathbb{R}_{\geq 0}^n$,

$$N(\vec{x}) = \sum_{i=0}^{n-1} x_i \quad (19)$$

Recalling Definition 1 many, but not necessarily all, flow functions in a model will have the form

$$f(\vec{\beta}, \vec{x}) = \tau(\vec{x}) \cdot f_r(\vec{\beta}, \vec{x}) = \tau(\vec{x}) \cdot \frac{g(\vec{\beta}, \vec{x})}{N(\vec{x})} \quad (20)$$

where $g \in \mathcal{F}$. If a factor model has flow functions with the above form then, if that model is used in a modified product for example, the form of the related flow function in the product model which is given by Equation 7 will be

$$f_e(\vec{\beta}, \vec{x}) = \tau_{p(e)}^{\text{from}}(\tau_{s(e)}^{\text{arg}}(\vec{x})) \sum_{v^b \in V^b} \frac{g(\vec{\beta}_{v^b}, \tau_{v^b}^{\text{arg}}(\vec{x}))}{N(\tau_{v^b}^{\text{arg}}(\vec{x}))} \quad (21)$$

where in fact, depending on the specific context the desired outcome may be

$$f_e(\vec{\beta}, \vec{x}) = \frac{\tau_{p(e)}^{\text{from}}(\tau_{s(e)}^{\text{arg}}(\vec{x}))}{N(\vec{x})} \sum_{v^b \in V^b} g(\vec{\beta}_{v^b}, \tau_{v^b}^{\text{arg}}(\vec{x})) \quad (22)$$

This difference suggests yet another type of model product, although a rigorous definition would require altering Definition 1 to explicitly include a denominator in the form of a flow function. (DF: That is not exactly straightforward to do either because not every flow in a model will necessarily have such a denominator.)

Another example where the functional form used in Definition 1 may require amendment is when investigating models with non-linear incidence rates. In such cases flow functions related to the force of infection in a model may have an additional factor associated with them yielding the form:

$$f(\vec{\beta}, \vec{x}) = \tau^{\text{from}}(\vec{x}) \cdot f_r(\vec{\beta}, \vec{x}) = \frac{\tau^{\text{from}}(\vec{x})^\zeta}{N(\vec{x})^{\zeta+1}} \cdot g(\vec{\beta}, \vec{x}) \quad (23)$$

Defining a space of flow functions which allows some, but not all, of the flows in a model to employ these variations in form is a non-trivial challenge and a significant reason why it is difficult to build a simulation engine for a product model without the supervision of a trained programmer. BB: unclear? DF: I tried to clarify. The point here is that we cant just say all flows in a model have this form because some may not. The flow from I to R for example typically doesn't have explicit normalization since that gets incorporated into the parameter value. Similarly flows specified in absolute terms won't be multiplied by the population of a from compartment. So each of these possible variations in form must be always allowed but never required. It's tricky to formulate a definition that does that. BB: I think this is OK but if someone wants to wordsmith a little bit ... ??

5.2 Models with Testing

One such example (where model products alone cannot produce the desired result) involves modeling the effects of testing for infection, inspired by the dynamics of testing during the COVID-19 pandemic. One example of a model that includes the effects of testing can be found in Gharouni et al (2022). Consider the epidemiological model in

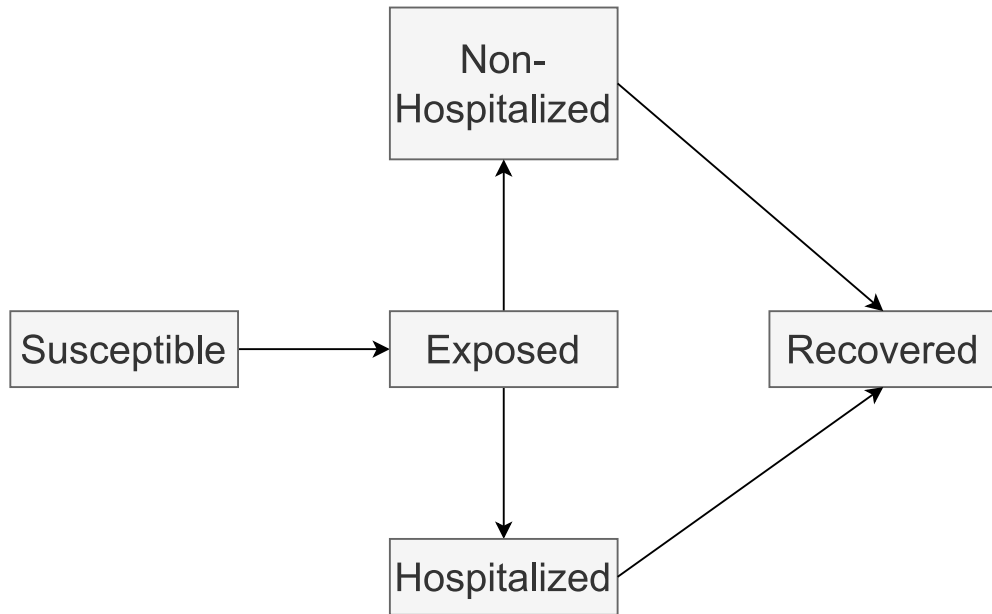


Fig. 8 A simple epidemiological model that we will expand to include testing. In this model, some exposed individuals will develop asymptomatic or mild illness, in which case they stay in the community during their infectious period (and potentially transmit to others); those who instead develop severe illness will be hospitalized. (This model allows neither for within-hospital transmission nor for disease-induced mortality either inside or outside the hospital.)

Figure 8 and the testing process depicted in Figure 9. The modified product of these two models includes a compartment for untested individuals at the hospital. However, this product is not what we want (Figure 10). The key difference is that untested individuals entering the hospital are typically tested (i.e., moved from “untested” to “awaiting results”); our model world assumes that they always are. Therefore, the “untested hospitalized” compartment in product model is empty and should be eliminated; the flow that goes to that compartment should instead be directed to the “hospitalized/awaiting test result” compartment. Constructing the desired model would thus require an extra step to remove the superfluous compartment.

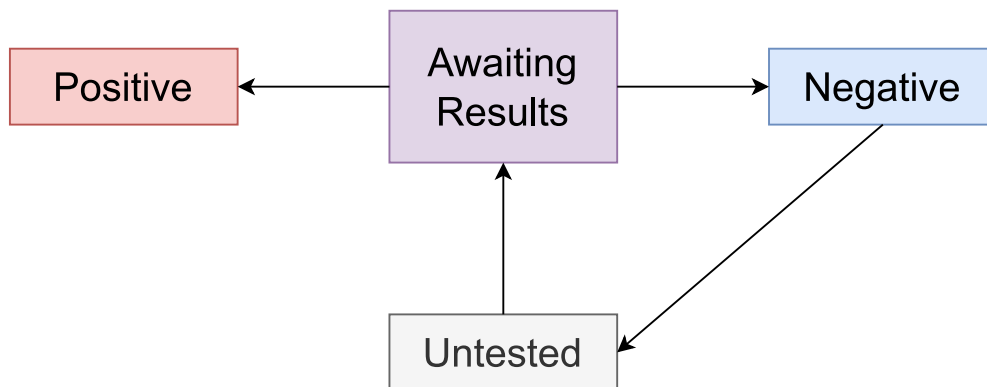


Fig. 9 A simple testing model. Individuals who test negative will, over time, revert back to the “untested” status. This is not the case for those that test “positive”; at least during the early stages of the COVID-19 pandemic, someone who had tested positive for COVID-19 would assume that they were immune and would not be re-tested even if they developed COVID-like symptoms.

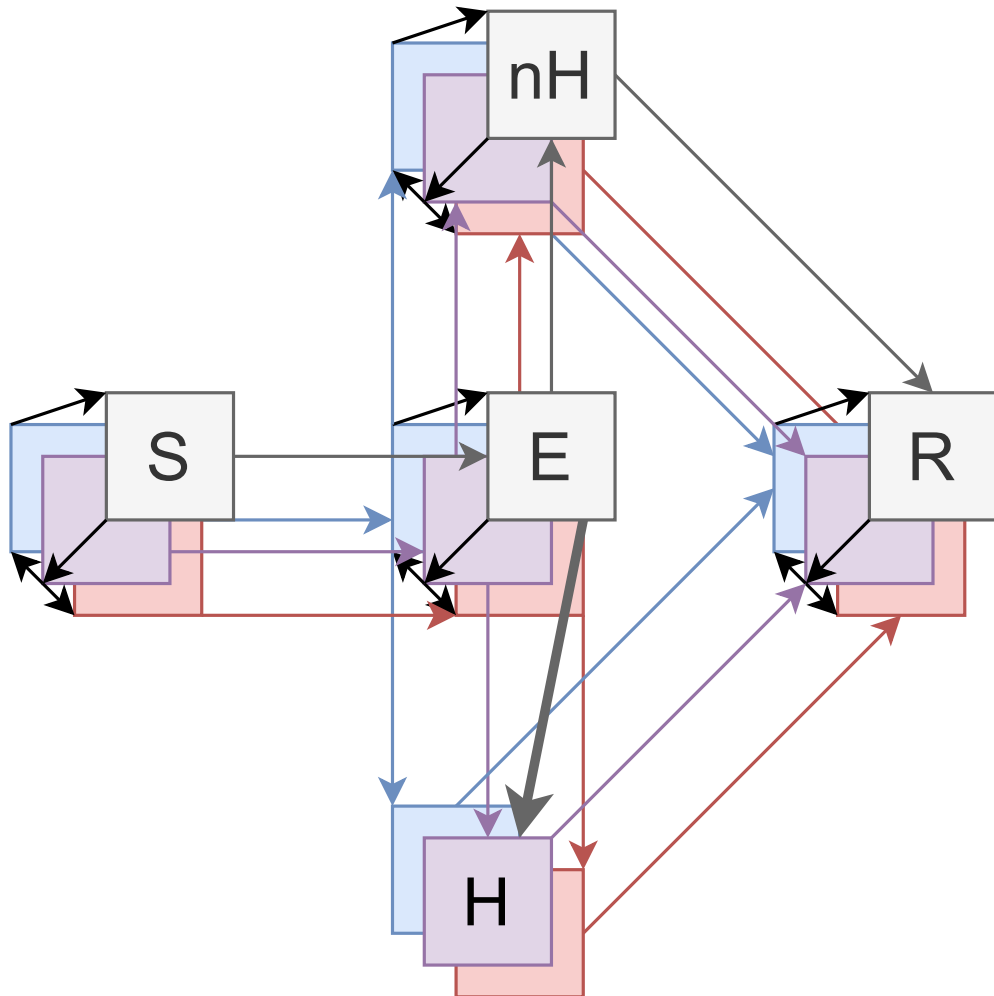


Fig. 10 The desired result of combining Figure 8 with Figure 9. Note the missing grey “untested” box associated with the hospital location; exposed individuals going into the hospital (enlarged, grey downward arrow starting at *E*) flow into the purple “awaiting results” subcompartment.

5.3 Multistrain Models and a Weak Product

Many epidemics involve multiple co-circulating strains of the same pathogen (Gog and Grenfell, 2002; Williams et al, 2021). In the case of COVID-19 such variants have significant implications for the efficacy of vaccines (Abu-Raddad et al, 2021; Koyama et al, 2020) and diagnostic tests (Vasireddy et al, 2021). In more complex models, including multiple strains rapidly inflates the size of both the state space and the parameter space (Kryazhimskiy et al, 2007). One way to limit the size of these unwieldy models while continuing to include the effects of multiple strains in our model is to disallow the possibility of *superinfection* (i.e. an individual being infected with multiple strains at the same time). It would therefore be useful to define a *weak product* similar to the operations proposed by Worden and Porco (2017) but which excludes all states corresponding to a superinfected status. Below we propose such a product which is satisfactory for producing two-strain models but, as we shall see, fails for more than two strains.

Definition 7 (Weak product). Suppose A and B are two compartmental models; let $S_a \subset V(A)$ and $S_b \subset V(B)$ denote the set of all vertices with either no inflows (i.e. *sources*) or no outflows (i.e. *sinks*) in A and B respectively. Let $I_a = V(A) \setminus S_a$ and $I_b = V(B) \setminus S_b$. We define

$$V_c = (S_a \times S_b) \cup (S_a \times I_b) \cup (S_b \times I_a). \quad (24)$$

We define

$$E_c^a = \{((w, x), (y, x)) \in V_c \times V_c \mid (w, y) \in E(A)\}, \quad (25)$$

$$E_c^b = \{((w, x), (w, z)) \in V_c \times V_c \mid (x, z) \in E(B)\}, \quad (26)$$

and

$$E_c = E_c^a \cup E_c^b. \quad (27)$$

For every $e \in E_c^a$, let $p(e) = (w, y) \in E(A)$. For every $e \in E(A)$ there are functions $f_e^a \in F(A)$ and $f_{r,e} \in \mathcal{F}(|V(A)|)$, as well as a filtering function $\tau_e^{\text{from}} \in \Upsilon(|V(A)|, 1)$ such that

$$f_e^a(\vec{\beta}, \vec{x}) = \tau_e^{\text{from}}(\vec{x}) \cdot f_{r,e}(\vec{\beta}, \vec{x}) \quad (28)$$

For every, $v_b \in S_b$ let $\tau_{v_b}^{\text{arg}}$ be the filtering function from V to $V(A) \times v_b$. For every $e \in E_c^a$ we define

$$f_e(\vec{\beta}, \vec{x}) = \sum_{v_b \in S_b} \tau_{p(e)}^{\text{from}}(\tau_{v_b}^{\text{arg}}(\vec{x})) f_{p(e)}^a(\vec{\beta}_{v_b}, \tau_{v_b}^{\text{arg}}(\vec{x})) \quad (29)$$

We define $F_c^a = \bigcup_{e \in E_c^a} f_e$. A symmetric definition can be given for $e \in E_c^b$ yielding $F_c^b = \bigcup_{e \in E_c^b} f_e$. We define $F_c = F_c^a \cup F_c^b$ and we say $C = (V_c, E_c, F_c)$. We define

$$A \boxplus B = C \quad (30)$$

We call \boxplus **the weak product**.

Figure 11 depicts a two-strain SIR model without super-infection; this corresponds to the weak product of two SIR models. Figures 12 and 13 depict two different results for the weak product of three SIR models; note that the difference between them results from changing the order in which products are done. If, for example, we denote the models of the red, yellow, and purple strains by A , B , and C respectively then Figure 12 depicts $(A \boxplus B) \boxplus C$ and Figure 13 depicts $A \boxplus (B \boxplus C)$. Figure 14 depicts the desired result for a three-strain SIR model with no super-infection. It is possible to create a version of the weak product defined above that will produce the model shown in Figure 14; however it requires us to distinguish between compartments that are global sources or sinks and compartments that are sources or sinks with respect to one of the three strains specifically. That is to say, while a global sink must have no outflows, a weaker condition says that a compartment is a sink with respect to a specific pathogen if every compartment that can be reached via the outflow has the same infection status with respect to that pathogen as the original compartment. Programmatically we achieve this by introducing a concept of ‘labeled partitions’ which separates the vertices of the model into disjoint sets corresponding to the vertices status with respect to a specific pathogen. Each dimension of stratification in the model corresponds to a different labeled partition with each stratum corresponding to a different disjoint set. In this way we can define sources and sinks with respect to a specific set of labels rather than globally. For example we can say a compartment A is a sink with respect to a specific labelled partition if every compartment that can be reached after being in A exists in the same disjoint set as A . Figure 15 outlines a compartmental model with one source compartment but two sink compartments and Figure 16 shows the weak product of two such models. An interesting complication arises in this product model as several of the compartments can only be reached by individuals after they are already dead (!). Of course if there are relatively few such compartments a modeler may choose simply to leave them in the model and treat them all as a single compartment. But if there are many such “zombie compartments” or if computational efficiency is a pressing concern it would be necessary to remove them from the model.

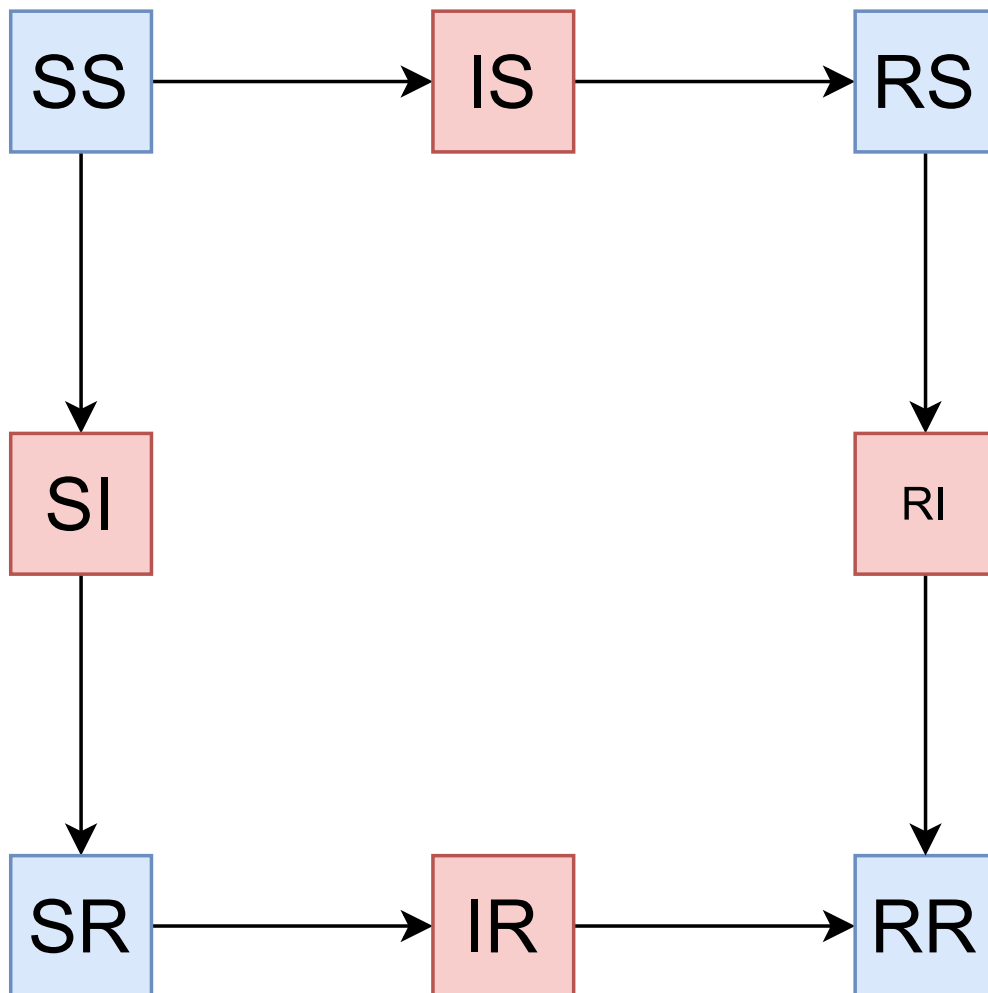
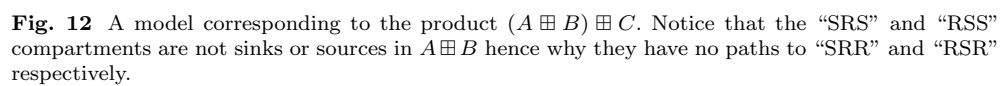


Fig. 11 A two-strain SIR model admitting no superinfection. Red Compartments indicate an infectious population whereas the population in blue compartments are not infectious



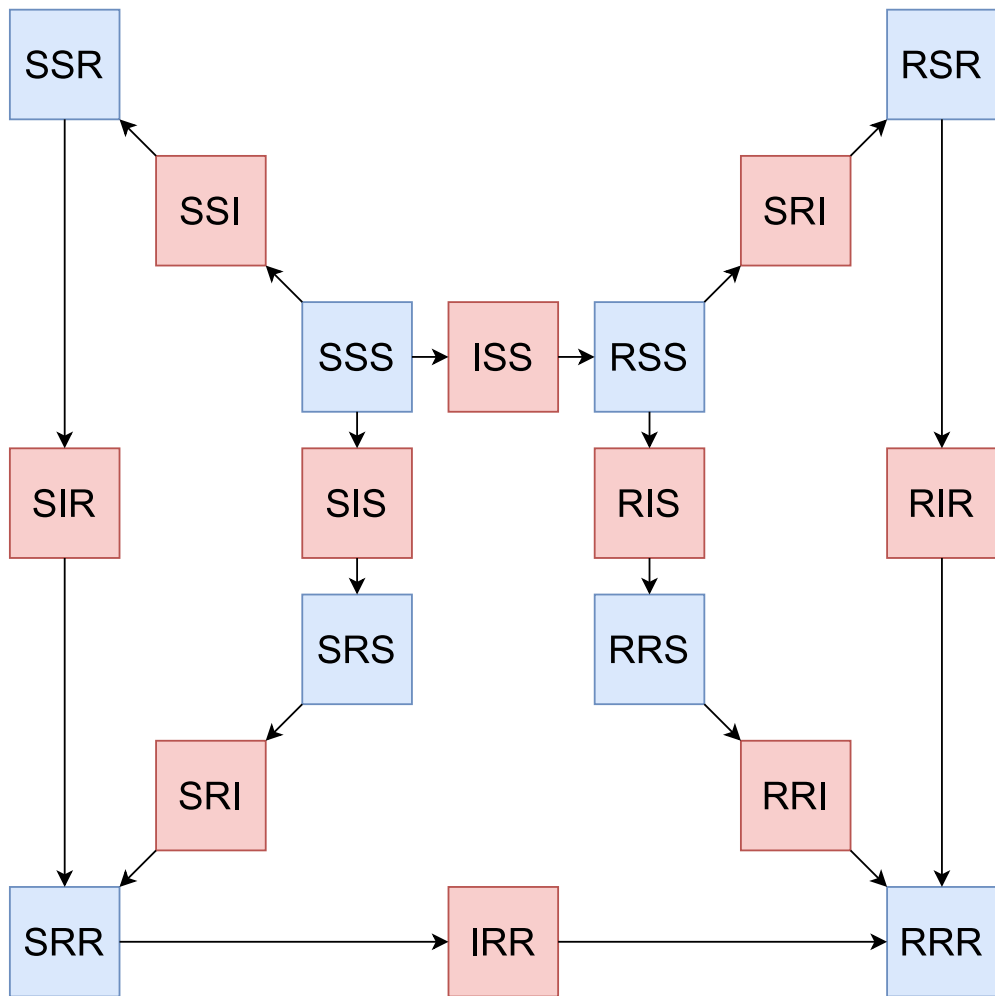


Fig. 13 A model corresponding to the product $A \boxplus (B \boxplus C)$. Notice that “SRS” and “SSR” compartments are not sources or sinks in $(B \boxplus C)$ hence why they have no paths to “RRS” and “RSR” respectively

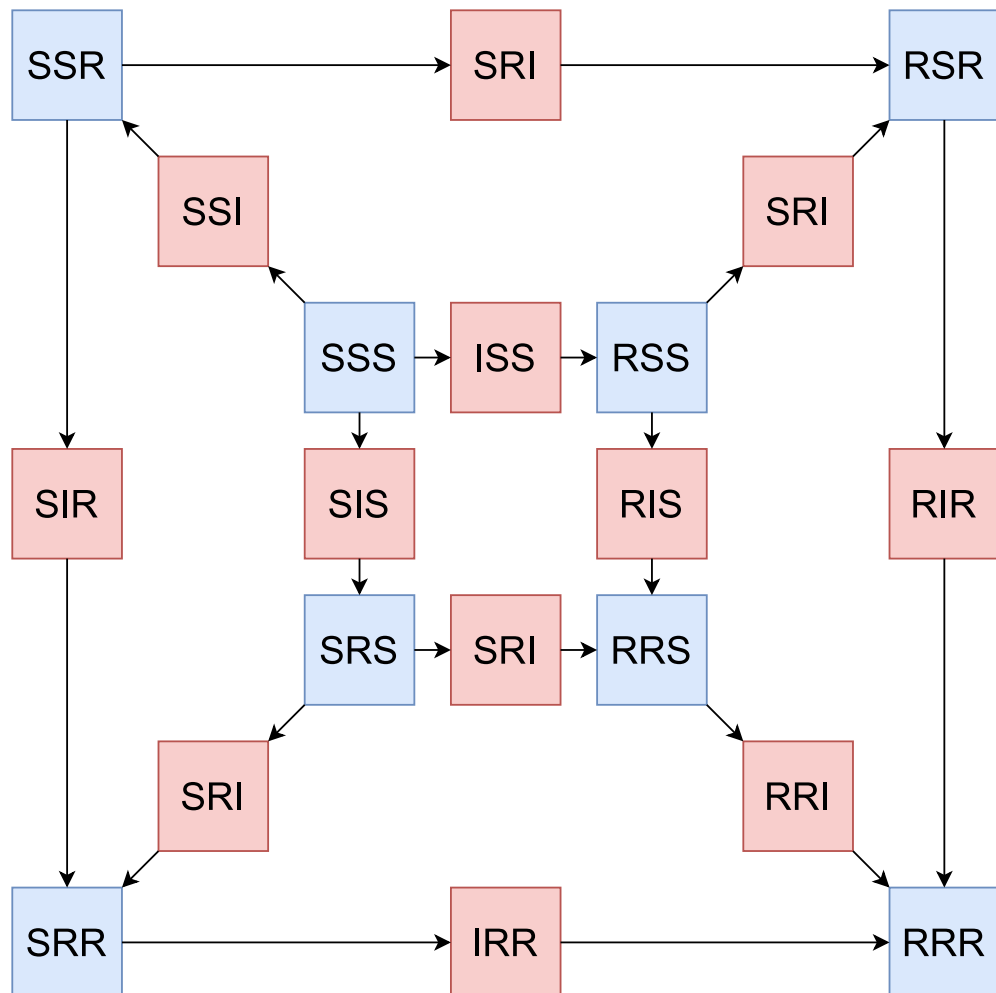


Fig. 14 A three strain SIR model admitting no superinfection. This model cannot be constructed using only the products defined in this article

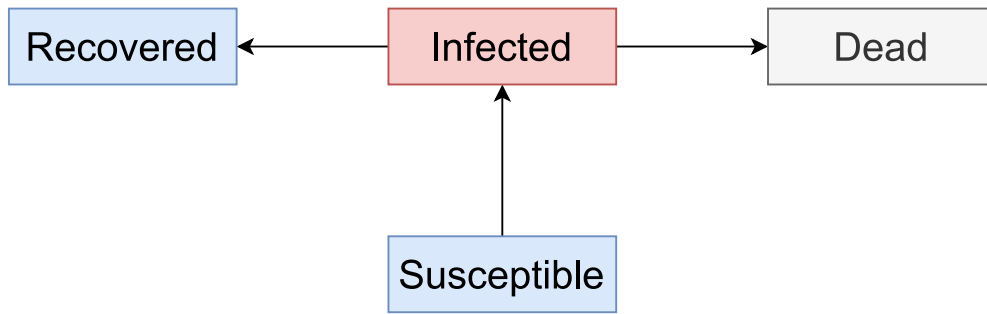


Fig. 15 A single strain model with two sinks and one source

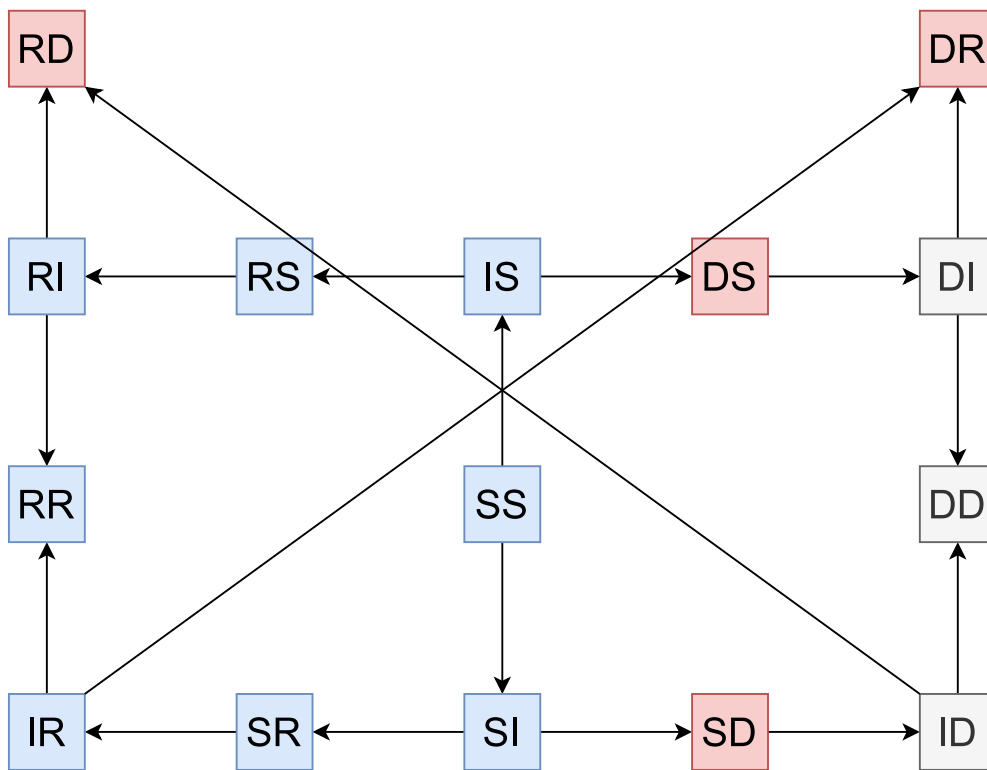


Fig. 16 The weak product of two of the single strain models depicted in Figure 15. Notice that the grey compartments are superfluous as they correspond to changes in infection status occurring after death

6 Conclusion

Adding new strata to simple epidemiological models is closely related to taking the Cartesian product of digraphs. Modellers who want to combine sets of simple models into a single large stratified model would benefit from a toolkit based on well-defined mathematical operations. This toolkit must contain a variety of operations representing a useful subset of the numerous ways that separate strata in a model can interact. We have developed a mathematical formalism for defining such operations and used it to restate two previously proposed model operations, the naive and modified products, which represent extremes of a spectrum of interactions between strata. The naive product corresponds to the case where different strata never interact, while the modified product corresponds to scenarios where any stratum can interact with any other stratum. We generalize these previously proposed operations to a third operation that allows any level of interaction between model strata, for example to construct geographically stratified models where interactions can occur within a single location and its neighbours but not more distantly.

Modelers employ a wide variety of functional forms; it is not always clear how to adjust them to accommodate new levels of stratification. BB: I still don't fully understand the normalization stuff (i.e., use cases) DF: As far as I can tell the different use cases are going to come down to what information a modeler has available to them. Steve and I have been over this a couple times and I recall at least one heated discussion with JD. We still can't seem to reach any kind of consensus about which way is more correct/useful so I've defaulted to including both possibilities. An example of this is when flow functions in a factor model involve normalization by the total population size. When generalizing to a product model, modelers must choose whether that normalization occurs with respect to the total population of the model or just the population of each individual stratum. Either option could be considered correct depending on the nature of the mechanisms being modelled. It seems likely that these issues arising from the abundance of potentially useful functional forms could be resolved by modifying Definition 1. For example one could explicitly include numerator and denominator terms or even an arbitrary number of terms each potentially drawing its arguments from a different subset of the state space. We have refrained from exploring such possibilities in this paper in an effort to avoid unnecessary complexity and potential confusion.

Several challenges remain for anyone wishing to further develop a model construction toolkit. We have paid little attention to parameter space and the question of how to use knowledge about factor model parameters to draw conclusions about product model parameters. It would be convenient to have a catalog of the most common ways of generalizing factor model parameters to product models so that modelers aren't required to reinvent the procedure every time. Many models (e.g. models with infection status testing) also have asymmetries in their structure that cannot be reproduced with Cartesian-like products. This suggests the need for addition-like operations to supplement the multiplication-like operations discussed in this paper. In fact such operations already exist in the category theoretic approach to model operations, which is one reason why it could be a worthwhile project to unite the category theory and graph theory approaches. In our understanding this would involve finding so-called

“type-graphs” that cause the category theoretic operations known as “pull-backs” and “push-outs” to reproduce the results of graph theory operations.

We are heavily motivated by the desire to develop software to facilitate model construction. A key insight of our investigations is the utility of a system of so-called “labeled partitions”, which divide the compartments of a model into mutually exclusive groups BB: this seems to be the first time we’ve emphasized that this is important, or why??? I see that it’s mentioned as part of the notation, but do we ever talk about them or how they’re useful? Introudcing this in the penultimate paragraph seems weird. DF: Labeled partitions offer a solution to the issues discussed in the multistrain/weak product section. I have expanded the discussion there to try to make the connection more explicit. Each group in such a division will contain all compartments that are in the same level of some dimension of stratification and the groups can be labelled accordingly. By applying several such divisions to a model, one for each dimension of stratification, it becomes possible to specify important subsets of the model compartments by referring to the relevant labels. Using this system of labels and partitions provides an easy way to address issues like the non-commutativity of the “weak product” and the presence of “zombie compartments” discussed in Section 5.3.

Although there remain theoretical and practical challenges with the application of binary operations on model space, this approach represents a powerful toolkit for the construction of compartmental models.

7 Acknowledgements

This project was supported by the Canandian Network for Modelling Infectious Diseases (CANMOD), which is funded through the Emerging Infectious Disease Modelling programme of the Natural Sciences and Engineering Research Council of Canada (NSERC).

8 Declarations

The authors declare that they have no competing interests.

References

- About-Ismael A (2020) Compartmental models of the COVID-19 pandemic for physicians and physician-scientists. *SN Comprehensive Clinical Medicine* 2(7):852–858
- Abu-Raddad LJ, Chemaitelly H, Butt AA (2021) Effectiveness of the BNT162b2 COVID-19 vaccine against the B.1.1.7 and B.1.351 variants. *New England Journal of Medicine* 385(2):187–189
- Adam D (2020) Special report: The simulations driving the world’s response to COVID-19. *Nature* 580(7802):316–319

- Anderson RM, May RM (1985) Age-related changes in the rate of disease transmission: implications for the design of vaccination programmes. *Journal of Hygiene (Cambridge)* 94:365–436
- Anderson RM, May RM (1992) *Infectious Diseases of Humans: Dynamics and Control*. Oxford University Press
- Baez J, Li X, Libkind S, et al (2022) Compositional modeling with stock and flow diagrams. *arXiv preprint arXiv:220508373*
- Baez JC, Pollard BS (2017) A compositional framework for reaction networks. *Reviews in Mathematical Physics* 29(09):1750028
- Balabdaoui F, Mohr D (2020) Age-stratified discrete compartment model of the COVID-19 epidemic with application to Switzerland. *Scientific reports* 10(1):1–12
- Chang YC, Liu CT (2022) A stochastic multi-strain SIR model with two-dose vaccination rate. *Mathematics* 10(11):1804
- Currie CS, Fowler JW, Kotiadis K, et al (2020) How simulation modelling can help reduce the impact of COVID-19. *Journal of Simulation* 14(2):83–97
- Dietz K, Sattenspiel L (1995) A structured epidemic model incorporating geographic mobility among regions. *Mathematical Biosciences* 128(1-2):71–91. [https://doi.org/10.1016/0025-5564\(94\)00068-B](https://doi.org/10.1016/0025-5564(94)00068-B)
- Enserink M, Kupferschmidt K (2020) With COVID-19, modeling takes on life and death importance
- Fields R, Humphrey L, Flynn-Primrose D, et al (2021) Age-stratified transmission model of COVID-19 in Ontario with human mobility during pandemic’s first wave. *Heliyon* 7(9):e07905
- Fong B, Spivak DI (2018) Seven sketches in compositionality: An invitation to applied category theory. *arXiv preprint arXiv:180305316*
- Friston KJ, Parr T, Zeidman P, et al (2020) Dynamic causal modelling of COVID-19. *Wellcome open research* 5
- Gharouni A, Abdelmalek FM, Earn DJD, et al (2022) Testing and Isolation Efficacy: Insights from a Simple Epidemic Model. *Bulletin of Mathematical Biology* 84(6):66. <https://doi.org/10.1007/s11538-022-01018-2>, URL <https://link.springer.com/10.1007/s11538-022-01018-2>
- Gog JR, Grenfell BT (2002) Dynamics and selection of many-strain pathogens. *Proceedings of the National Academy of Sciences* 99(26):17209–17214

- Grenfell BT, Anderson RM (1985) The estimation of age-related rates of infection from case notifications and serological data. *Journal of Hygiene (Cambridge)* 95:419–436
- Halter M, Baas A, Patterson E, et al (2022) Algebraicjulia/algebraicpetri.jl: v0.7.2. <https://doi.org/10.5281/zenodo.6928476>, URL <https://doi.org/10.5281/zenodo.6928476>
- Koyama T, Weeraratne D, Snowden JL, et al (2020) Emergence of drift variants that may affect COVID-19 vaccine development and antibody treatment. *Pathogens* 9(5):324
- Kryazhimskiy S, Dieckmann U, Levin SA, et al (2007) On state-space reduction in multi-strain pathogen models, with an application to antigenic drift in influenza A. *PLoS Computational Biology* 3(8):e159
- Lavielle M, Faron M, Zeitoun JD, et al (2020) Extension of a SIR model for modelling the propagation of COVID-19 in several countries. *medRxiv*
- Leontitsis A, Senok A, Alsheikh-Ali A, et al (2021) SEAHIR: A specialized compartmental model for COVID-19. *International journal of environmental research and public health* 18(5):2667
- Libkind S, Baas A, Patterson E, et al (2021) Operadic modeling of dynamical systems: mathematics and computation. *arXiv preprint arXiv:210512282*
- Libkind S, Baas A, Halter M, et al (2022) An algebraic framework for structured epidemic modeling. *arXiv preprint arXiv:220316345* <https://doi.org/10.48550/ARXIV.2203.16345>, URL <https://arxiv.org/abs/2203.16345>
- Lofgren ET, Halloran ME, Rivers CM, et al (2014) Mathematical models: A key tool for outbreak response. *Proceedings of the National Academy of Sciences* 111(51):18095–18096
- Massonis G, Banga JR, Villaverde AF (2021) Structural identifiability and observability of compartmental models of the COVID-19 pandemic. *Annual reviews in control* 51:441–459
- McBryde ES, Meehan MT, Adegboye OA, et al (2020) Role of modelling in COVID-19 policy development. *Paediatric respiratory reviews* 35:57–60
- Mohammadi Z, Cojocaru M, Arino J, et al (2023) Importation models for travel-related SARS-CoV-2 cases reported in Newfoundland and Labrador during the COVID-19 pandemic. <https://doi.org/10.1101/2023.06.08.23291136>, URL <https://www.medrxiv.org/content/10.1101/2023.06.08.23291136v1>, pages: 2023.06.08.23291136
- Roberts FS, Tesman B (2009) *Applied Combinatorics*, 2nd edn. Taylor and Francis

- Savageau MA (1988) Introduction to S-systems and the underlying power-law formalism. *Mathematical and Computer Modelling* 11:546–551
- Vasireddy D, Vanaparthi R, Mohan G, et al (2021) Review of COVID-19 variants and COVID-19 vaccine efficacy: what the clinician should know? *Journal of Clinical Medicine Research* 13(6):317
- Voit E (1990) S-system modelling of endemic infections. *Computers & Mathematics with Applications* 20(4-6):161–173
- Voit EO (1988) Recasting nonlinear models as S-systems. *Mathematical and Computer Modelling* 11:140–145
- Walter GG, Contreras M (1999) *Compartmental modeling with networks*. Springer Science & Business Media
- Williams BJ, St-Onge G, Hébert-Dufresne L (2021) Localization, epidemic transitions, and unpredictability of multistrain epidemics with an underlying genotype network. *PLoS Computational Biology* 17(2):e1008606
- Worden L, Porco TC (2017) Products of compartmental models in epidemiology. *Computational and Mathematical Methods in Medicine* 2017:e8613878. <https://doi.org/10.1155/2017/8613878>