FUNMANAbstraction

Dan Bryce

February 3, 2025

1 Introduction

We describe methods to stratify and abstract (de-stratify) Petrinets for compartmental models. The motivation for abstracting a Petrinet is to reduce its size, which becomes exponential in the number of stratified variables. Reducing the size of the model can have a significant impact on runtime, and it is possible to answer useful queries with the abstract model. The following sections include a background (defining the models), a description of abstraction, an approach needed to bound the abstracted models (or to perform parameter synthesis directly in a simulator), and then a comparison of simulation results for several variations of a baseline model that applies stratification, bounding, and abstraction.

2 Background

Definition 1 *A Petrinet Ω is a directed graph (V, E) with vertices $V = (V_x, V_z)$ partitioned into sets V_x of state vertices and V_z of transition vertices, and edges $E = (E_{out}, E_{in})$ partitioned into collections E_{out} of flow-out and E_{in} flow-in edges (relative to state vertices).*

Definition 2 *A flow-out edge $e \in E_{out}$ comprises a pair of vertices (v_x, v_z), where $v_x \in V_x$ is a state vertex, $v_z \in V_z$ is a transition vertex, and the flow is directed from v_x to v_z.*

Definition 3 *A flow-in edge $e \in E_{in}$ comprises a pair of vertices (v_z, v_x), similar to a flow-out edge, except that the flow is directed from v_z to v_x.*

Example 1 *The SIR model that stratifies the S state variable into S_1 and S_2 for two susceptible populations and defines Ω by:*

$$\begin{aligned}
V_x &= \{v_{S_1}, v_{S_2}, v_I, v_R\} \\
V_z &= \{v_{inf_1}, v_{inf_2}, v_{rec}\} \\
E_{in} &= ((v_{inf_1}, v_{S_1}), (v_{inf_1}, v_I), (v_{inf_1}, v_I), (v_{inf_2}, v_{S_2}), (v_{inf_2}, v_I), (v_{inf_2}, v_I), (v_{rec}, v_R)) \\
E_{out} &= ((v_{S_1}, v_{inf_1}), (v_{S_2}, v_{inf_2}), (v_I, v_{inf_1}), (v_I, v_{rec}))
\end{aligned}$$

Definition 4 *The ODE semantics Θ of the Petrinet Ω defines a tuple $(P, X, Z, \mathcal{I}, \mathcal{P}, \mathcal{X}, \mathcal{Z}, \mathcal{R})$ where*

- *P is a set of parameters;*
- *X is a set of state variables;*
- *Z is a set of transitions;*
- *$\mathcal{I} : S \to \mathbb{R}$ assigns the initial value of state variables to a real number;*
- *$\mathcal{P} : P \to \mathbb{R} \cup \mathbb{R} \times \mathbb{R}$ assigns parameters to a real number, or a pair of real numbers defining an interval;*

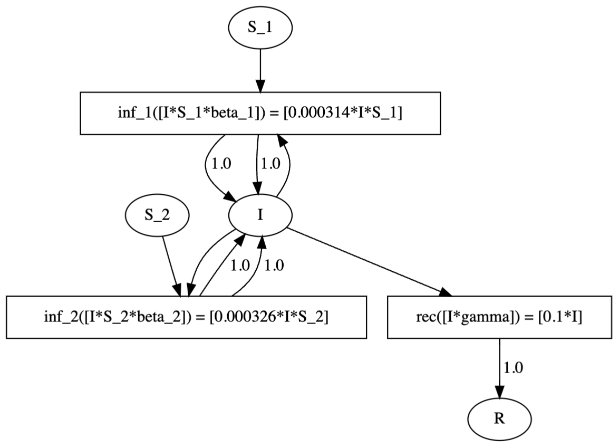

Figure 1: SIR model stratified with two populations in the S state (S_1 and S_2), each with a unique β parameter (β_1 and β_2).

- $\mathcal{X} : X \rightarrow V_x$ *assigns state variables to state vertices;*
- $\mathcal{Z} : Z \rightarrow V_z$ *assigns transtions to transition vertices; and*
- $\mathcal{R} : \mathbf{P} \times \mathbf{X} \times Z \rightarrow \mathbb{R}$ *defines the rate of each transition* $z \in Z$ *in terms of the set of parameter vectors* $\mathbf{P}$ *and state variable vectors* $\mathbf{X}$.

The elements of the Petrinet Ω *and semantics* Θ *define the partial derivative* $\frac{d\mathbf{x}}{dt}$*, so that for each state variable* $x \in X$*:*

$$\frac{dx}{dt} = \sum_{z \in Z^{in(x)}} \mathcal{R}(\mathbf{p}, \mathbf{x}, z) - \sum_{z \in Z^{out(x)}} \mathcal{R}(\mathbf{p}, \mathbf{x}, z) \tag{1}$$

where $Z^{in(x)} = \{z \in Z | (z, x) \in E_{in}\}$ *and* $z^{out(x)} = \{z \in Z | (x, z) \in E_{out}\}$ *are the transition vertices that flow in and out of the vertex* v_x*, respectively. We denote by* $\nabla_{\Omega,\Theta}(\mathbf{p}, \mathbf{x}, t) = (\frac{dx_1}{dt}, \frac{dx_2}{dt}, \ldots)^T$*, the gradient comprised of components defined in Equation* (1).

In the following, we simplify the definition of a Petrinet and associated semantics because the graph vertices and the semantic elements are one to one. We drop the vertex terminology by assuming the following:

- States: given X, V_x and $\mathcal{X}$, referring to a state variable $x \in X$ is synonymous with v_x because $\mathcal{X}(x) = v_x$.
- Transitions: given Z, V_z and $\mathcal{Z}$, referring to a state variable $z \in Z$ is synonymous with v_z because $\mathcal{Z}(z) = v_z$.
- Edges: Each edge $e \in E$ corresponds to a pair of vertices (v_x, v_z) or (v_z, v_x), and it is (respectively) synonymous to pairs (x, z) or (z, x).

Example 2 *The stratified SIR model defines Θ (dropping $\mathcal{X}$ and $\mathcal{Z}$ per above) by:*

$$
\begin{aligned}
P &= \{\beta_1, \beta_2, \gamma\} \\
X &= \{S_1, S_2, I, R\} \\
Z &= \{inf_1, inf_2, rec\} \\
\mathcal{I} &= \begin{cases} 0.45 & : S_1 \\ 0.45 & : S_2 \\ 0.1 & : I \\ 0.0 & : R \end{cases} \\
\mathcal{P} &= \begin{cases} 1e{-7} & : \beta_1 \\ 2e{-7} & : \beta_2 \\ 1e{-5} & : \gamma \end{cases} \\
\mathcal{R} &= \begin{cases} \beta_1 S_1 I & : z_{inf_1} \\ \beta_2 S_2 I & : z_{inf_2} \\ \gamma I & : z_{rec} \end{cases}
\end{aligned}
$$

Using the partial derivatives defined by the Petrinet graph and semantics, we can define the state vector at given time $t + dt$ with the forward Euler method as:

$$
\begin{aligned}
\frac{d\mathbf{x}}{dt} &= \nabla_{\Omega,\Theta}(\mathbf{p}, \mathbf{x}, t) \\
\frac{\mathbf{x}(t+dt) - \mathbf{x}(t)}{dt} &= \nabla_{\Omega,\Theta}(\mathbf{p}, \mathbf{x}, t) \\
\mathbf{x}(t+dt) &= \nabla_{\Omega,\Theta}(\mathbf{p}, \mathbf{x}, t)dt + \mathbf{x}(t)
\end{aligned}
$$

3 Abstraction

Definition 5 *An abstraction (Θ^A, Ω^A) of a Petrinet and the associated semantics (Θ, Ω) that is produced by the abstraction operator A has the following properties:*

- *State: For each $x \in X$, there exists an $x' \in X^A$ where $A(x) = x'$. For each abstract state variable $x' \in X^A$, the initial value is the sum of the initial values of state variables mapped to x' by A, so that $\mathcal{I}^A(x') = \sum_{x \in X : A(x) = x'} \mathcal{I}(x)$.*
- *Parameters: For each $p \in P$, there exists a $p' \in P^A$ where $A(p) = p'$. For each abstract parameter $p' \in P^A$, the value (or interval) is the sum of all parameters mapped to p' by A, so that $\mathcal{P}^A(p') = \sum_{p \in P : A(p) = p'} \mathcal{P}(p)$.*
- *Transitions: For each $z \in Z$, there exists a $z' \in Z^A$ where $A(z) = z'$.*
- *In Edges: For each edge $(z, x) \in E_{in}$, there exists a $(z', x') \in E^A_{in}$, where $A((z, x)) = (z', x')$, $A(x) = x'$, and $A(z) = z'$.*
- *Out Edges: For each edge $(x, z) \in E_{out}$, there exists a $(x', z') \in E^A_{out}$, where $A((x, z)) = (x', z')$, $A(x) = x'$, and $A(z) = z'$.*
- *Transition Rates: For each $z' \in Z^A$,*

$$
\mathcal{R}^A(\mathbf{p}', \mathbf{x}', z') = \sum_{z \in Z : A(z) = z'} \mathcal{R}(\mathbf{p}, \mathbf{x}, z) \tag{2}
$$

Example 3 *The abstraction (Θ^A, Ω^A) of the stratified SIR model defines (with the changed elements highlighted by "*"):*

$$
\begin{array}{rcl}
A & = & \left\{ \begin{array}{lll}
S & : S_1 & * \\
S & : S_2 & * \\
I & : I & \\
R & : R & \\
\beta & : \beta_1 & * \\
\beta & : \beta_2 & * \\
\gamma & : \gamma & \\
inf & : inf_1 & * \\
inf & : inf_2 & * \\
rec & : rec &
\end{array} \right. \\
\mathcal{R}^A & = & \left\{ \begin{array}{lll}
\beta_1 S_1 I + \beta_2 S_2 I & : z_{inf} & * \\
\gamma I & : z_{rec} &
\end{array} \right.
\end{array}
$$

In Example 3, the abstraction (Θ^A, Ω^A) maps the S_1 and S_2 state variables to the S state variable (de-stratifying the Petrinet). In combining the state variables, the abstract Petrinet consolidates the transitions inf_1 and inf_2 and associated rates from susceptible to infected.

Like the base model, the abstraction (Θ^A, Ω^A) defines a gradient $\nabla_{\Omega^A, \Theta^A}(\mathbf{p}^A, \mathbf{x}^A, t) = (\frac{dx'_1}{dt}, \frac{dx'_2}{dt}, \ldots)^T$, in terms of Equation 1. Via Equation 2, the abstraction thus expresses the gradient by aggregating terms from the base Petrinet and semantics. It preserves the flow on consolidated transitions, but expresses the transition rates in terms of the base states. As such, the abstraction compresses the Petrinet graph structure, but at the cost of expanding the expressions for transition rates. Moreover, the transition rates refer to state variables and parameters (e.g., β_1, β_2, S_1, and S_2) that are not expressed directly by the abstract Petrinet and semantics (e.g., as β and S), and by extension, the gradient. We address this in the next section.

4 Bounding Petrinets

We transform the abstraction in what we call a *bounded Petrinet*, so that it can refer to the abstract, and not the base, Petrinet and semantics. The same bounding can be applied to any Petrinet by controlling which parameter bounds and abstracted variables are used in constructing the bounded Petrinet. Applying bounding to abstract Petrinets allows us to summarize multiple related parameters (e.g., stratified parameters) with an interval containing the parameter values. The degenerate case of bounding a Petrinet is when the lower and upper bound on each parameter is identical, and thus the lower and upper bounds for each state variable at each time are equal.

Independent of abstraction, relaxing the parameter bounds so that the lower and upper bound are not equal allows us to perform parameter synthesis. For example, by computing the upper bound I^{ub} on the number of infected I given that $\beta \in [\beta^{lb}, \beta^{ub}]$ will allow us to conclude that any value of β within the bounds will satisfy $I^{ub} < c$ for some threshold c. FUNMAN evaluates the same type of queries by showing their negation $I \geq c$ is unsatisfiable. In constructing a modestly-larger (polynomial in the original Petrinet), bounded Petrinet, we can use off the shelf simulators to synthesize parameters.

4.1 Bounding Transformation

We present a general definition of how to bound a Petrinet and identify the special case where the Petrinet was constructed by abstraction. The key to bounding a Petrinet is to map each state variable x to a pair of state variables $[x^{lb}, x^{ub}]$. We ensure that $x^{lb}(t) \leq x(t)$ for all time points t by developing a lower bound on each contribution to the value of x, and similarly for upper bounds. The contributors to the value of x are based upon transitions $z \in Z$, where there is an in-edge $(z, x) \in E_{in}$ or out-edge $(x, z) \in E_{out}$.

From Equation 1, the in-edges define positive contributions (i.e., increase the value of x) and the out-edges define negative contributions (i.e., decrease the value of x). To formulate a lower bound on x, we define the derivative $\frac{dx^{lb}}{dt}$ so that $\frac{dx^{lb}}{dt} \leq \frac{dx}{dt}$, and similarly for upper bounds (i.e., $\frac{dx}{dt} \leq \frac{dx^{ub}}{dt}$). It relies upon finding a substitution σ of the form $[x_1^*/x_1, \ldots, x_n^*/x_n, p_1^*/p_1, \ldots, p_m^*/p_m]$ where $* \in \{lb, ub\}$ and may be different for each term.

Definition 6 *A bounded Petrinet (Θ^B, Ω^B) transforms Petrinet (Θ, Ω) by replacing each element of (Θ, Ω) by a pair of elements denoting the lower and upper bound of that element (and referred to with the "lb" and "ub" superscripts). The bound transformation defines (Θ^B, Ω^B) so Theta:*

- *State: For each $x \in X$, there exists $x^{lb}, x^{ub} \in X^B$. For each $x^{lb}, x^{ub} \in X^B$, $\mathcal{I}^B(x^{lb}) = \mathcal{I}^B(x^{ub}) = \mathcal{I}(x)$.*
- *Parameters: For each $p \in P$, let $\mathcal{P}^B(p^{lb}) = \mathcal{P}^B(p^{ub}) = \mathcal{P}(p)$ if $\mathcal{P}(p)$ is a single value, or $\mathcal{P}^B(p^{lb}) = a$, $\mathcal{P}^B(p^{ub}) = b$, if $\mathcal{P}(p)$ is a pair that signifies an interval $[a, b]$.*
- *Transitions: For each transition $z \in Z$ and state variable $x \in X$, we define up to four transitions $z^{in}_{x^{lb}}$, $z^{in}_{x^{ub}}$, $z^{out}_{x^{lb}}$, and $z^{out}_{x^{ub}}$, so that Z^B is defined as follows:*

$$\begin{aligned} Z^B &= \{z^{in}_{x^{lb}}, z^{in}_{x^{ub}} | (z, x) \in E_{in}\} \cup \\ &\quad \{z^{out}_{x^{lb}}, z^{out}_{x^{ub}} | (x, z) \in E_{out}\} \end{aligned}$$

- *In Edges: For each edge $(z, x) \in E_{in}$, there is a pair of transitions $(z^{in}_{x^{lb}}, x^{lb}), (z^{in}_{x^{ub}}, x^{ub}) \in E^B_{in}$.*
- *Out Edges: For each edge $(x, z) \in E_{out}$,there is a pair of transitions $(x^{lb}, z^{out}_{x^{lb}}), (x^{ub}, z^{out}_{x^{ub}}) \in E^B_{in}$.*
- *Transition Rates: For each transition in $z \in Z^B$, the rate $\mathcal{R}^B(\mathbf{p}^B, \mathbf{x}^B, z)$ defines the rate of z, expressed in terms of a vector of parameters $\mathbf{p}^B$ (corresponding to P^B) and a vector of state variables $\mathbf{x}^B$ (corresponding to X^B). The rate for each transition in Z^B depends on finding a substitution σ (mapping $\mathbf{p}$ to $\mathbf{p}^B$ and $\mathbf{x}$ to $\mathbf{x}^B$) that obeys the following cases:*

$$\begin{aligned} \mathcal{R}^B(\mathbf{p}^B, \mathbf{x}^B, z^{in}_{x^{lb}}) &= \arg\min_\sigma(\sigma \circ ([x^{lb}/x] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, z))) \\ \mathcal{R}^B(\mathbf{p}^B, \mathbf{x}^B, z^{out}_{x^{lb}}) &= \arg\max_\sigma(\sigma \circ ([x^{lb}/x] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, z))) \\ \mathcal{R}^B(\mathbf{p}^B, \mathbf{x}^B, z^{in}_{x^{ub}}) &= \arg\max_\sigma(\sigma \circ ([x^{ub}/x] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, z))) \\ \mathcal{R}^B(\mathbf{p}^B, \mathbf{x}^B, z^{out}_{x^{ub}}) &= \arg\min_\sigma(\sigma \circ ([x^{ub}/x] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, z))) \end{aligned}$$

  The $\circ$ operator applies a substitution to an expression and returns the resulting expression. In the rates, we first substitute the lower bound x^{lb} for x, and find a substitution σ for the remaining state variables and parameters, and similarly for upper bounds.

  The rates capture how each transition either effects x positively by flowing in, or negatively by flowing out. Propagating a lower bound involves minimizing the positive flow in, and maximizing the negative flow out. Propagating upper bounds follows similar rationale, maximizing flow in, and minimizing flow out.

Equations 3 to 8 show how we derive the lower bound $\frac{dx^{lb}}{dt}$. The primary intuition is that there exists a substitution σ that minimizes (or maximizes) each expression in brackets, and that by moving the choice of substitution inward we allow each sub-expression a different choice of σ. This is a type of independence assumption that allows us to symbolically minimize (or maximize) each rate by choosing an appropriate substitution.

$$
\begin{aligned}
\frac{dx}{dt} &= \sum_{z \in Z^{in(x)}} \mathcal{R}(\mathbf{p}, \mathbf{x}, z) - \sum_{z \in Z^{out(x)}} \mathcal{R}(\mathbf{p}, \mathbf{x}, z) && (3)\\
&\geq \arg\min_{\sigma} \sigma \circ \left[\sum_{z \in Z^{in(x)}} \mathcal{R}(\mathbf{p}, \mathbf{x}, z) - \sum_{z \in Z^{out(x)}} \mathcal{R}(\mathbf{p}, \mathbf{x}, z) \right] && (4)\\
&\geq \arg\min_{\sigma} \sigma \circ \left[\sum_{z \in Z^{in(x)}} \mathcal{R}(\mathbf{p}, \mathbf{x}, z) \right] - \arg\max_{\sigma} \sigma \circ \left[\sum_{z \in Z^{out(x)}} \mathcal{R}(\mathbf{p}, \mathbf{x}, z) \right] && (5)\\
&\geq \sum_{z \in Z^{in(x)}} \arg\min_{\sigma} \sigma \circ [\mathcal{R}(\mathbf{p}, \mathbf{x}, z)] - \sum_{z \in Z^{out(x)}} \arg\max_{\sigma} \sigma \circ [\mathcal{R}(\mathbf{p}, \mathbf{x}, z)] && (6)\\
&\geq \sum_{z \in Z^{in(x)}} \arg\min_{\sigma} \sigma \circ \big[[x^{lb}/x] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, z)\big] - \sum_{z \in Z^{out(x)}} \arg\max_{\sigma} \sigma \circ \big[[x^{lb}/x] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, z)\big] && (7)\\
&= \frac{dx^{lb}}{dt} && (8)
\end{aligned}
$$

For example, in the stratified SIR model from Example 2, we can derive a lower bound for I, as listed in Equations 9 to 13, and the choice of substitutions listed in Equations 14 to 16. As in σ_{rec}, the substitutions must include I^{lb}/I whenever I is a rate term because we are defining $\frac{dI^{lb}}{dt}$.

$$
\begin{aligned}
\frac{dI}{dt} &= \mathcal{R}(\mathbf{p}, \mathbf{x}, inf_1) + \mathcal{R}(\mathbf{p}, \mathbf{x}, inf_2) - \mathcal{R}(\mathbf{p}, \mathbf{x}, rec) && (9)\\
&\geq \sigma_1 \circ ([I^{lb}/I] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, inf_{1I^{lb}})) + && (10)\\
&\quad \sigma_2 \circ ([I^{lb}/I] \circ \mathcal{R}(p, \mathbf{x}, inf_{2I^{lb}})) - && (11)\\
&\quad \sigma_3 \circ ([I^{lb}/I] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, rec_{I^{lb}})) && (12)\\
&= \frac{dx^{lb}}{dt} && (13)
\end{aligned}
$$

$$
\begin{aligned}
\sigma_1 &= [S_1^{lb}/S_1, \beta_1^{lb}/\beta_1] && (14)\\
\sigma_2 &= [S_2^{lb}/S_2, \beta_2^{lb}/\beta_2] && (15)\\
\sigma_3 &= [\gamma^{ub}/\gamma] && (16)
\end{aligned}
$$

Applying the substitutions to the rate terms, $\frac{dI}{dt}$ and $\frac{dI^{lb}}{dt}$ are simplified to the expressions in Equations 17 to 22.

$$
\begin{aligned}
\frac{dI}{dt} &= \mathcal{R}(\mathbf{p}, \mathbf{x}, inf_1) + \mathcal{R}(\mathbf{p}, \mathbf{x}, inf_2) - \mathcal{R}(\mathbf{p}, \mathbf{x}, rec) && (17)\\
&= S_1 I \beta_1 + S_2 I \beta_2 - I\gamma && (18)\\
\frac{dI^{lb}}{dt} &= [S_1^{lb}/S_1, \beta_1^{lb}/\beta_1] \circ ([I^{lb}/I] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, inf_{1I^{lb}})) + && (19)\\
&\quad [S_2^{lb}/S_2, \beta_2^{lb}/\beta_2] \circ ([I^{lb}/I] \circ \mathcal{R}(p, \mathbf{x}, inf_{2I^{lb}})) - && (20)\\
&\quad [\gamma^{ub}/\gamma] \circ ([I^{lb}/I] \circ \mathcal{R}(\mathbf{p}, \mathbf{x}, rec_{I^{lb}})) && (21)\\
&= S_1^{lb} I^{lb} \beta_1^{lb} + S_2^{lb} I^{lb} \beta_2^{lb} - I^{lb}\gamma^{ub} && (22)
\end{aligned}
$$

4.2 Bounding Abstracted Petrinets

We previously described how abstracted Petrinets contain state variables and parameters from the previously stratified Petrinet in the transition rates. For example, if $A(S_1) = S$ and $A(S_2) = S$ (S_1 and S_2 are stratified variables represented by S in the abstraction), the transition rate associated with the inf transition is

$$\mathcal{R}^A(\mathbf{p}^A, \mathbf{x}^A, inf) = \beta_1 S_1 I + \beta_2 S_2 I$$

By construction, we know that $S_1 + S_2 = S$. Applying this equivalence to the rate law as a substitution results in:

$$\begin{aligned}
[S - S_1/S_2] \circ \mathcal{R}^A(\mathbf{p}^A, \mathbf{x}^A, inf) &= \beta_1 S_1 I + \beta_2 (S - S_1) I \\
&= \beta_1 S_1 I + \beta_2 S I - \beta_2 S_1 I
\end{aligned}$$

Furthermore, when bounding this rate to determine $\mathcal{R}^B(\mathbf{p}^B, \mathbf{x}^B, inf^{in}_{I_{lb}})$, it is possible to substitute the same term $\min(\beta_1, \beta_2)$ for both parameters β_1 and β_2:

$$\begin{aligned}
\mathcal{R}^B(\mathbf{p}^B, \mathbf{x}^B, inf^{out}_{I_{lb}}) &= [\min(\beta_1, \beta_2)/\beta_1, \min(\beta_1, \beta_2)/\beta_2, S^{lb}/S, I/I^{lb}] \circ ([S - S_1/S_2] \circ \mathcal{R}^A(\mathbf{p}^A, \mathbf{x}^A, inf)) \\
&= \min(\beta_1, \beta_2) S_1 I + \min(\beta_1, \beta_2) S^{lb} I^{lb} - \min(\beta_1, \beta_2) S_1 I \\
&= \min(\beta_1, \beta_2) S^{lb} I^{lb} \\
&= \beta^{lb} S^{lb} I^{lb}
\end{aligned}$$

where we use a new parameter β^{lb} and define $\mathcal{P}^B(\beta^{lb}) = \min(\beta_1, \beta_2)$. A similar argument can be made for the upper bound using $\max(\beta_1, \beta_2)$.

By introducing the bounded parameters, we no longer rely upon the base state variables or parameters. However, in tracking the effect of the bounded parameters, the bounded abstraction must also track bounded rates and bounded state variables. The resulting bounded abstraction thus over-approximates the abstraction and base model, wherein we can derive bounds on the state variables at each time, which may correspond to a larger (hence over-approximation) set of state trajectories.

4.3 Bounded Petrinet Example

Example 4 *The bounded abstraction (Θ^B, Ω^B) of the stratified SIR model defines:*

$$
\begin{aligned}
E^B &= ((S^{lb}, inf^{out}_{S_{lb}}), (S^{ub}, inf^{out}_{S_{ub}}), (I^{lb}, inf^{out}_{I_{lb}}), (I^{ub}, inf^{out}_{I_{ub}}), (inf^{in}_{I_{lb}} I^{lb}), (inf^{in}_{I_{ub}}, I^{ub}),\\
&\quad (I^{lb}, rec^{out}_{I_{lb}}), (I^{ub}, rec^{out}_{I_{ub}}), (rec^{in}_{R_{lb}}, R^{lb}), (rec^{in}_{R_{ub}}, R^{ub}))\\
P^B &= \{\beta^{lb}, \beta^{ub}, \gamma^{lb}, \gamma^{ub}\}\\
X^B &= \{S^{lb}, S^{ub}, I^{lb}, I^{ub}, R^{lb}, R^{ub}\}\\
Z^B &= \{inf^{out}_{S_{lb}}, inf^{out}_{S_{ub}}, inf^{out}_{I_{lb}}, inf^{out}_{I_{ub}}, inf^{in}_{I_{lb}}, inf^{in}_{I_{ub}}, rec^{out}_{I_{lb}}, rec^{out}_{I_{ub}}, rec^{in}_{R_{lb}}, rec^{in}_{R_{ub}}\}\\
\mathcal{I}^B &= \begin{cases} 0.9 & : S^{lb}\\ 0.9 & : S^{ub}\\ 0.1 & : I^{lb}\\ 0.1 & : I^{ub}\\ 0.0 & : R^{lb}\\ 0.0 & : R^{ub} \end{cases}\\
\mathcal{P}^B &= \begin{cases} 1e{-}7 & : \beta^{lb}\\ 2e{-}7 & : \beta^{ub}\\ 1e{-}5 & : \gamma^{lb}\\ 1e{-}5 & : \gamma^{ub} \end{cases}\\
\mathcal{R}^B &= \begin{cases} \beta^{ub} S^{lb} I^{ub} & : inf^{out}_{S_{lb}}\\ \beta^{lb} S^{ub} I^{lb} & : inf^{out}_{S_{ub}}\\ \beta^{ub} S^{ub} I^{lb} & : inf^{out}_{I_{lb}}\\ \beta^{lb} S^{lb} I^{ub} & : inf^{out}_{I_{ub}}\\ \beta^{lb} S^{lb} I^{lb} & : inf^{in}_{I_{lb}}\\ \beta^{ub} S^{ub} I^{ub} & : inf^{in}_{I_{ub}}\\ \gamma^{ub} I^{lb} & : rec^{out}_{I_{lb}}\\ \gamma^{lb} I^{ub} & : rec^{out}_{I_{ub}}\\ \gamma^{lb} I^{lb} & : rec^{in}_{R_{lb}}\\ \gamma^{ub} I^{ub} & : rec^{in}_{R_{ub}} \end{cases}
\end{aligned}
$$

5 SIR Example Results

Results are available in this notebook.

6 SIRHD Example Results

We measured the time to simulate various formulations of the SIRHD model to highlight the effects of stratification, abstraction, and bounding. We defined a series of stratification, abstraction, and bounding operations as follows. Starting with the base model (Ω, Θ), we either bound the model $(\Omega^B, \Theta^B) = \text{Bound}(\Omega, \Theta)$, or stratify the model $(\Omega^S, \Theta^S) = \text{Stratify}(\Omega, \Theta)$. Each stratified model (Ω^S, Θ^S) is stratified again differently $(\Omega^{S'}, \Theta^{S'}) = \text{Stratify}(\Omega^S, \Theta^S)$, bounded $(\Omega^B, \Theta^B) = \text{Bound}(\Omega^S, \Theta^S)$, or abstracted $(\Omega^A, \Theta^A) = \text{Abstract}(\Omega^A, \Theta^A)$. Each abstracted model (Ω^A, Θ^A) is bounded $(\Omega^B, \Theta^B) = \text{Bound}((\Omega^A, \Theta^A)$, or abstracted again $(\Omega^{A'}, \Theta^{A'}) = \text{Abstract}(\Omega^A, \Theta^A)$. While it is also possible to stratify an abstracted model, we don't explore this operation here.

Figure 2 describes how we developed models from the SIRHD base model. We used several stratifications, and then applied abstractions that reversed the stratifications. Each abstraction was bounded so that we could simulate the model. We organized the models in the fashion so that we could demonstrate the time to

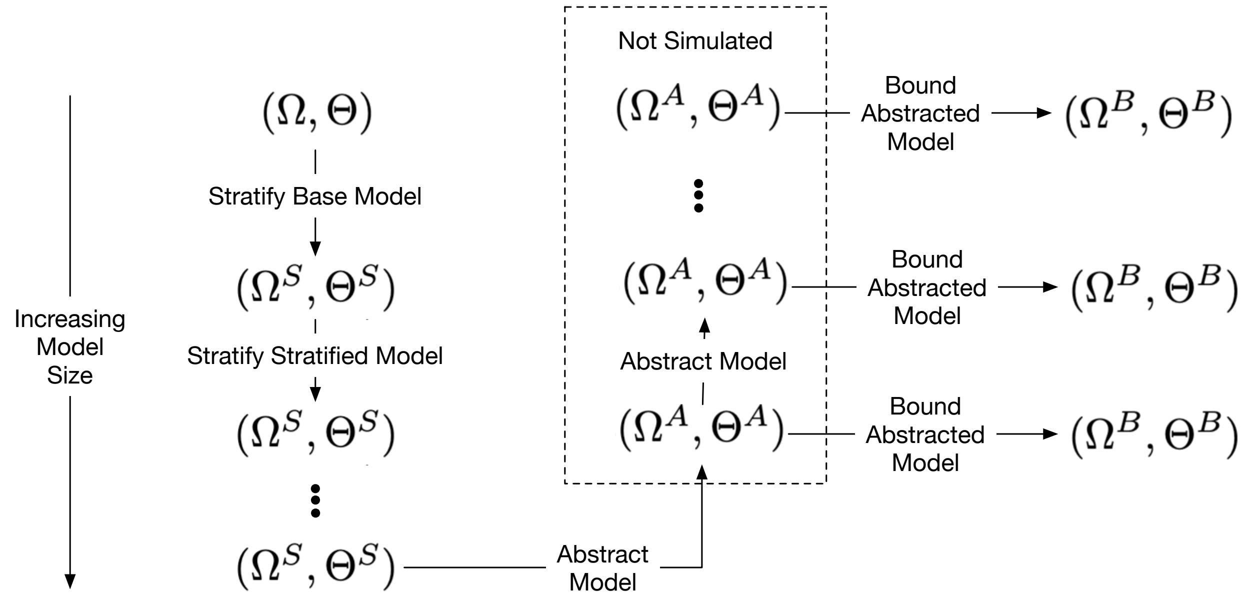

Figure 2: Conceptual relationships between model formulations. The base model can be stratified a number of times. The final stratified model is abstracted to reverse the stratifications. Each abstracted model is bounded so that it can be simulated. The dashed box indicates which model are not simulated.

simulate each model. In the figure, the models are aligned vertically to indicate which models represent the same level of detail. For example, the base model and the most-abstract abstraction have the same number of states, transitions, and parameters. Likewise, reversing the last stratification with the first abstraction results in a model that is the same size as the one prior to the last stratification. Bounding an abstracted model will increase its size polynomially, whereas stratifying it will increase its size exponentially. The runtime results tables arrange the models in a similar fashion. The first column of results increases the model size by stratification from the top to the bottom, and the second column decreases the model size from the top to the bottom.

Table 1 lists runtime results in seconds to simulate each SIRHD model variation. Each instance seeks to check a constraint that assesses whether over 200 days the number of infected is no more than 30% of the population ($N = 1.5e9$). We use 15 stratifications on the model, three for each of the state variables S, I, R, H and D. The three stratifications involve stratifying each state variable into vaccinated and unvaccinated groups (e.g., S_{vac} and S_{unvac}), the vaccinated group into age groups (e.g., $S_{vac,0}$, $S_{vac,1}$, $S_{vac,2}$, ...), and the unvaccinated group into age groups (e.g., $S_{unvac,0}$, $S_{unvac,1}$, $S_{unvac,2}$, ...). We report results for either three or five age groups. The model index 0-15 refers to successive stratifications of the base model 0 in the Stratified column. Similarly, the Bounded Abstracted column refers to successive abstractions of the models, where model 14 is the abstraction of model 15 in the Stratified column, and model 13 is the abstraction of model 14 in the Bounded Abstracted column.

We construct the most abstract model 0 from a series of stratifications and abstractions. It is an over-approximation of the most stratified model that may allow us to prove the constraint is satisfied. The abstract bounded model represents lower and upper bounds on the number of infected, and if the upper bound is less than 30% of N (checked through simulation) then we do not need to simulate the most stratified model 15 (our reference model formulation). The bounded abstract model 0 can be inconclusive if 30% of N falls between the lower and upper bounds on the number of infected. By considering successively less abstract models (i.e., proceeding from model 0 to model 1), we expect the lower and upper bounds to tighten. The bolded time in the Bounded Abstracted model column is the runtime of the first model formulation (starting from model 0) where we can prove that the upper bound on the number of infected is less than 30% of

	3 Age Levels		5 Age Levels	
Model	Stratified	Bounded Abstracted	Stratified	Bounded Abstracted
0	0.57	5.26	0.54	5.35
1	0.76	7.18	0.69	6.93
2	1.33	5.92	2.86	7.43
3	2.18	**9.93**	6.29	**22.98**
4	3.34	9.50	6.17	27.63
5	3.88	14.25	11.46	52.81
6	5.36	22.84	19.47	108.55
7	7.31	23.12	22.97	114.84
8	7.13	28.87	26.41	143.46
9	8.14	34.59	39.39	200.15
10	9.25	32.94	36.55	209.17
11	12.36	41.55	49.47	245.51
12	12.96	50.17	58.79	319.67
13	11.49	56.79	61.23	292.44
14	13.73	67.72	66.92	361.31
15	**15.17**	-	**83.80**	-

Table 1: Runtime in seconds to simulate each model formulation. Model 0 in the Stratified column is the base model. Models 1-15 in the Stratified column are successive stratifications. From model 14 to 0 in the Bounded Abstracted column each is a successive abstraction. The bolded number in the Stratified column is the fully stratified model that we assume is the starting point for analysis. The bolded number in the Bounded Abstracted column is the smallest model that allows us to confirm the model constraint is satisfied.

N. Therefore, if we need to simulate Bounded Abstracted models 0 through 3, we require 28.29 and 42.69 seconds, respectively, for either three or five age groups. Compared to the time to simulate the largest Stratified model 15, it takes 15.17 or 82.80 seconds respectively. The savings due to abstraction reduces runtime by one half when there are five age groups, but doubles the runtime when there are three age groups. This illustrates the trade-off between simulating several abstract models versus one large stratified model. When the abstract models can capture the important model dynamics using bounds and the number of collapsed stratification levels is large, then abstraction is a win.