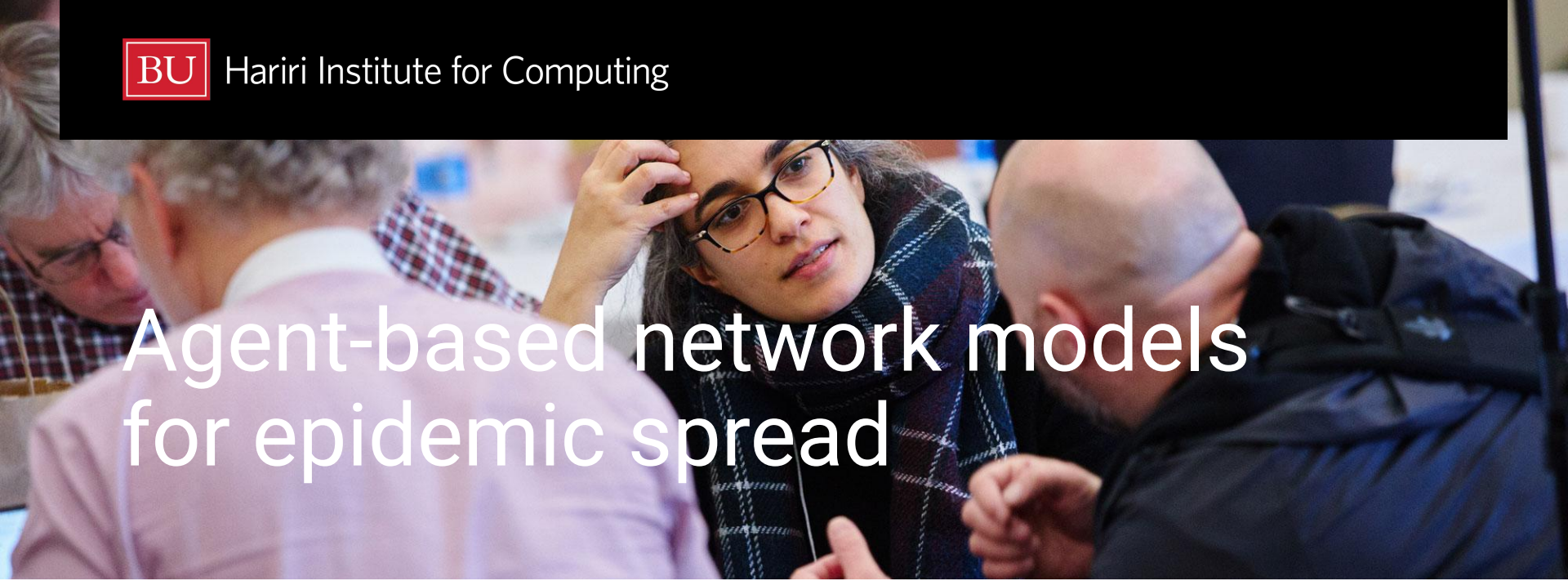
BU Hariri Institute for Computing



Agent-based network models for epidemic spread

Eric Kolaczyk, Hariri Institute for Computing (Director)
Faculty for Computing & Data Science
Mathematics & Statistics
Email: kolaczyk@bu.edu / Twitter: @EKolaczyk

CS591 Guest Lecture, November 24, 2020

Point of Departure

- Unit-level (stochastic) simulation of complex systems is widespread
- Standard representation is as a network and a process on that network
- Primary example dominating the last 6+ months is epidemic spread of a virus on a network of contacts among individuals in a population.
- Other examples include spread of rumors, innovation, and product adoption.

Bottom line: Generally computationally intensive but also lends itself well to parallelization.

Topics for Today

- High-level summary of
  - Standard SIR epidemic model
  - Network-based extension
- A brief look at the underlying algorithms for simulation
- Case study: BU Covid modeling exercise this past summer

Standard and Network-based SIR Models

Traditional SIR Model

We consider the *general epidemic model*, a particularly simple version of the class of *susceptible-infected-removed (SIR)* models.

NOTE: There is no network inherent in this model.

In a population of $N + 1$ elements (e.g., people, computers, etc.), we picture the random triple $(N_S(t), N_I(t), N_R(t))$ evolving in time t, where

- $N_S(t) = \#$ susceptible to infection at time t
- $N_I(t) = \#$ infected at time t
- $N_R(t) = \#$ removed/recovered by time t

Evolution of an SIR Process

We specify the evolution of this stochastic process through *instantaneous transition probabilities*

$$\mathbb{P}\left(N_S(t+\delta t) = s-1\,,\, N_I(t+\delta t) = i+1 \,|\, N_S(t) = s\,,\, N_I(t) = i \right) \approx \beta s i \delta t$$

$$\mathbb{P}\left(N_S(t+\delta t) = s\,,\, N_I(t+\delta t) = i-1 \,|\, N_S(t) = s\,,\, N_I(t) = i \right) \approx \gamma i \delta t$$

$$\mathbb{P}\left(N_S(t+\delta t) = s\,,\, N_I(t+\delta t) = i \,|\, N_S(t) = s\,,\, N_I(t) = i \right) \approx 1 - (\beta s + \gamma) i \delta t \;\;,$$

where

- δt refers to an infinitesimal time-step and
- $N_R(t)$ is omitted due to the constraint $N_S(t) + N_I(t) + N_R(t) = N + 1$.

Interpretation

The model says that, at any given time t,

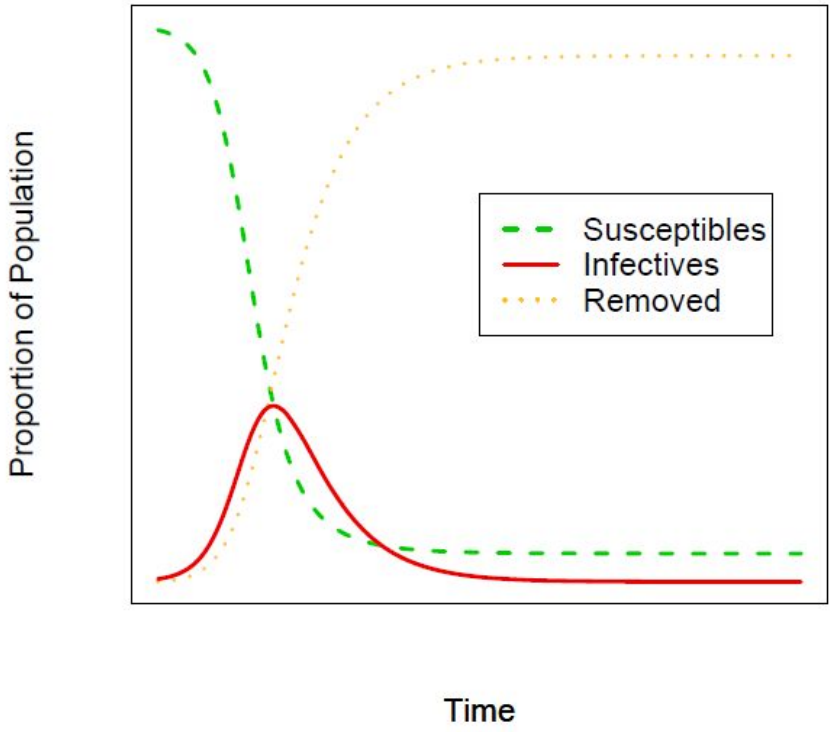
- sucesptibles become infected with probability proportional to the *product* of their numbers (i.e., $s \times i$), while
- infectives recover with probability proportional to their number (i.e., i)

The product form corresponds to an assumption of 'homogeneous mixing' (or 'mass-action,' in chemistry) among members of the population, which asserts that the population is

1. homogeneous and
2. well mixed, in the sense that any pair of members are equally likely to interact with each other.

It is this assumption that changes in network-based epidemic modelin

Schematic Characterization



Network-based Epidemic Modeling

The assumption of homogeneous mixing (aka 'mass action') typically is suspect.

Focus instead is more on 'structured population models', wherein contacts are constrained by some structure(s) of interest within the population.

Such structure can arise from, e.g.,

- spatial proximity (e.g., diseases of plants)
- social contact (e.g., sexual contact in the transmission of AIDS), or
- demographics (e.g., households, age brackets, etc.).

Often convenient to represent the underlying contact structure as a graph $G = (V, E)$, where

- V represent elements of the population and
- $\{i, j\} \in E$ indicates contact between elements i and j.

Modifying the SIR Model

A network-based modification of the general epidemic SIR model assumes that infection can only arise, at some rate, through contact with neighbors.

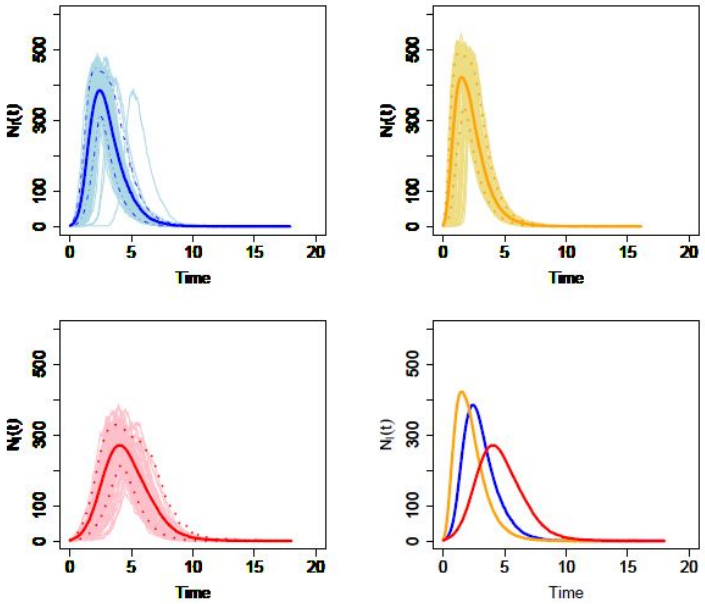
Mathematically this is expressed as

$$\mathbb{P}\left(\mathbf{X}(t+\delta t)=\mathbf{x}' \,|\, \mathbf{X}(t)=\mathbf{x}\right) \approx \begin{cases} \beta M_i(\mathbf{x})\delta t \text{ , if } x_i = 0 \text{ and } x_i' = 1 \quad , \\ \gamma \delta t \text{ , if } x_i = 1 \text{ and } x_i' = 2 \ , \\ 1 - [\beta M_i(\mathbf{x}) + \gamma]\, \delta t \text{ , if } x_i = 2 \text{ and } x_i' = 2 \ , \end{cases}$$

where

$$M_i(\mathbf{x}) = \ \# \text{ of neighbors } j \in \mathcal{N}_i \text{ for which } x_j = 1 \ .$$

Impact of Including Network Structure



Network-based SIR processes generated on ER (blue), BA (yellow) and small-world (red) random graphs.

Stochastic Simulation of SIR Dynamics

Computing SIR Dynamics

Numerical methods are used to compute curves like the ones you've seen. Such methods fall into two large classes, broadly speaking

- Numerical solvers for ordinary differential equations (ODEs)
- Stochastic simulation (i.e., Monte Carlo methods)

The first are appropriate when systems of (nonstochastic) differential equations are used. (e.g., see here for a gentle intro and python code).

Since our versions above are stochastic, we will use simulation. What follows is a very brief peek, based on Appendix A of the book Mathematics of Epidemics on Networks.

Gillespie Algorithm for SIR Simulation

Input: Network G, per-edge transmission rate τ, recovery rate γ, set of index node(s) initial_infecteds, maximum time t_{max}.
Output: Lists times, S, I, and R giving number in each state at each time.

function Gillespie_network_epidemic(G, τ, γ, initial_infections, t_{max})
    times, S, I, R ← [0], [|G|-len(initial_infections)], [len(initial_infections)], [0]
    infected_nodes ← initial_infections
    at_risk_nodes ← uninfected nodes with infected neighbours
    for each node u in at_risk_nodes **do**
        infection_rate[u] = $\tau \times$ number of infected neighbours
    total_infection_rate ← $\sum_{u \in \text{at\_risk\_nodes}}$ infection_rate[u],
    total_recovery_rate ← $\gamma \times$ len(infected_nodes)
    total_rate ← total_transmission_rate + total_recovery_rate
    time ← exponential_variate(total_rate)
    while time< t_{max} and total_rate> 0 **do**

Gillespie Algorithm (cont)

```
while time< t_max and total_rate> 0 do
    r = uniform_random(0,total_rate)
    if r <total_recovery_rate then
        u = random.choice(infected_nodes)
        remove u from infected_nodes
        reduce infection_rate[v] for u's susceptible neighbours v
    else
        choose u from at_risk_nodes with probability infection_rate[u] / total_infection_rate
        remove u from at_risk_nodes
        add u to infected_nodes
        for susceptible neighbours v of u do
            if v not in at_risk_nodes then
                add v to at_risk_nodes
            update infection_rate[v]
    update times, S, I, and R
    update total_recovery_rate, total_infection_rate, and total_rate
    time ← time + exponential_variate(total_rate)
return times, S, I, R
```

Improving on the Gillespie Algorithm

The slowest step is determining which nodes become infected (due to acceptance-rejection sampling)

Speed ups can be obtained using a so-called event-driven approach, wherein upon infection of a node we calculate immediately when it recovers and when it transmits to its neighbors. Priority queues are then used to track, ordered by event time..

An Event-driven Algorithm

Input: Network G, per-edge transmission rate τ, recovery rate γ, set of index node(s) initial_infecteds, and maximum time t_{max}.
Output: Lists times, S, I, and R giving number in each state at each time.

```
function fast_SIR(G,τ, γ, initial_infecteds, t_max)
    times, S, I, R ← [0], [|G|], [0], [0]
    Q ← empty priority queue
    for u in G.nodes do
        u.status ← susceptible
        u.pred_inf_time ← ∞
    for u in initial_infecteds do
        Event ← {node: u, time: 0, action: transmit}
        u.pred_inf_time ← 0
        add Event to Q                                  ▷ ordered by time
    while Q is not empty do
        Event ← earliest remaining event in Q
        if Event.action is transmit then
            if Event.node.status is susceptible then
                process_trans_SIR(G, Event.node, Event.time, τ, γ, times, S, I, R, Q, t_max)
        else
            process_rec_SIR(Event.node, Event.time, times, S, I, R)
    return times, S, I, R
```

BU COVID-19 Modeling this Past Summer

Primary contributors

Biostatistics: Laura White

Institutional Research: Linette Decarie, Mike McDevitt

MAG Testing Group: David Hamer, Chris Gill, Helen Jenkins, Judy Platt

RCS / IS&T: Katia Bulekova, Brian Gregor

Statistics/Networks: Eric Kolaczyk, Wenrui Li

Informed by regular discussions with university leadership / committees

Our goals

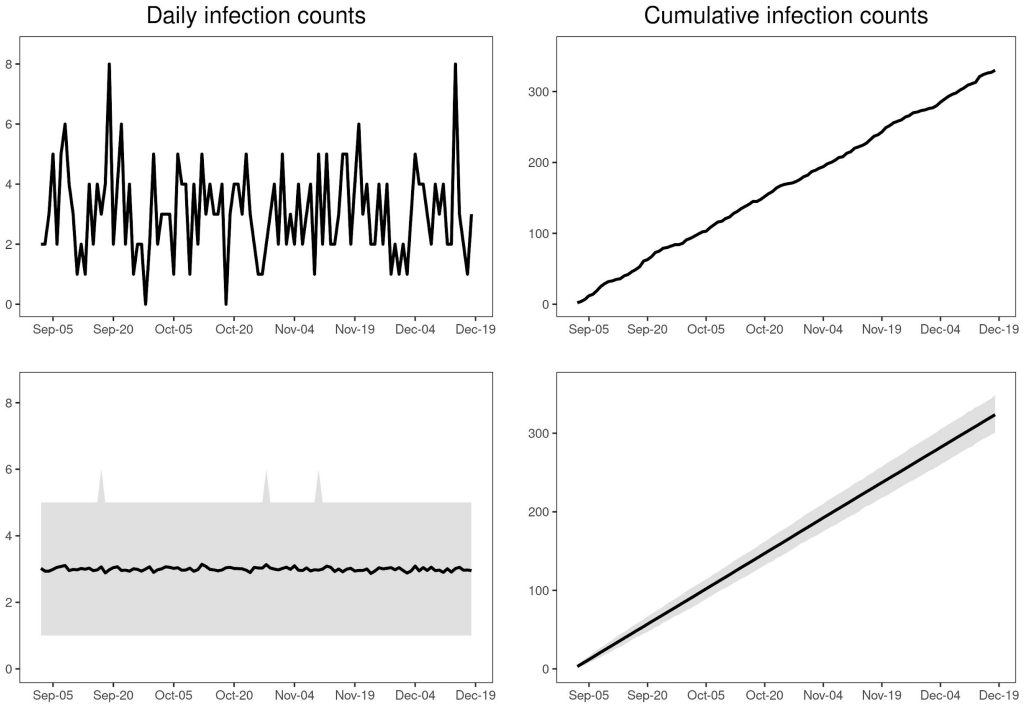
Use mathematical modeling to provide **qualitative insight** into the **expected relative efficacy** of potential interventions taken around a Fall 2020 BU re-opening.

Do so in a manner informed as much as **reasonably possible** by BU data.

Focus primarily on the dynamics around (i) **classroom instruction**, and (ii) **residential housing**.

"All models are wrong. Some are useful." George E.P. Box

Illustration: a relevant ideal



Key Point: Rather than zero infections, a more realistic goal is to *keep the BU rate of infections proportional to a constant background rate* (or linear cumulative rate) in the Boston area as a whole.

Modeling BU's reality

Exogenous Factors: Captured as a single, aggregate external input.

Contact Networks: Interactions between individuals that can cause infection.

Modified SEIR model: Dictates the progression of disease for individuals.

Interventions: Classroom masks and distancing, contact tracing, isolation, quarantine and self-attestation strategies are implemented.

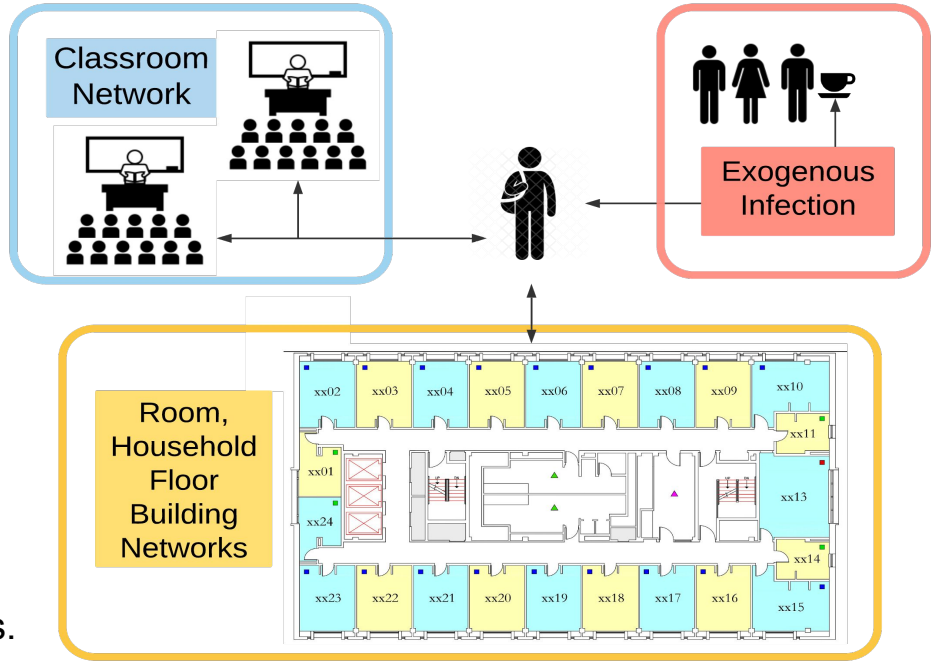
Model parameters set using current best knowledge.

Multiple scenarios run where choices and/or uncertainty dictate.

Note: Simulations run through adaptation of the Covasim package, from the Institute for Disease Modeling (Global Good Fund).
Our software pipeline, inclusive of model parameters, is available open source at https://github.com/bu-rcs/BU-COVID .

Multiple sources of interactions

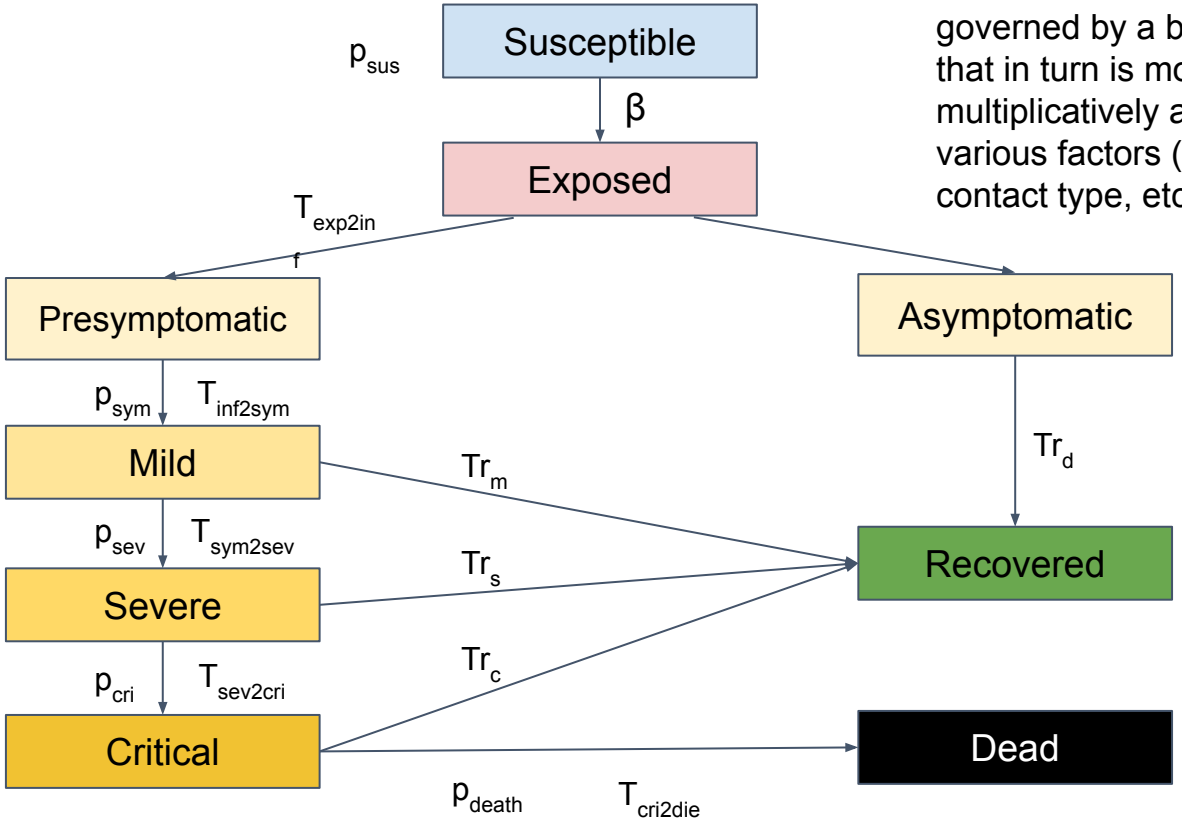
Classrooms Contacts modeled as those seated adjacent. Transmission probabilities are multiplicative with more shared classes.



Exogenous All external sources captured through a single, aggregate "source" per person.

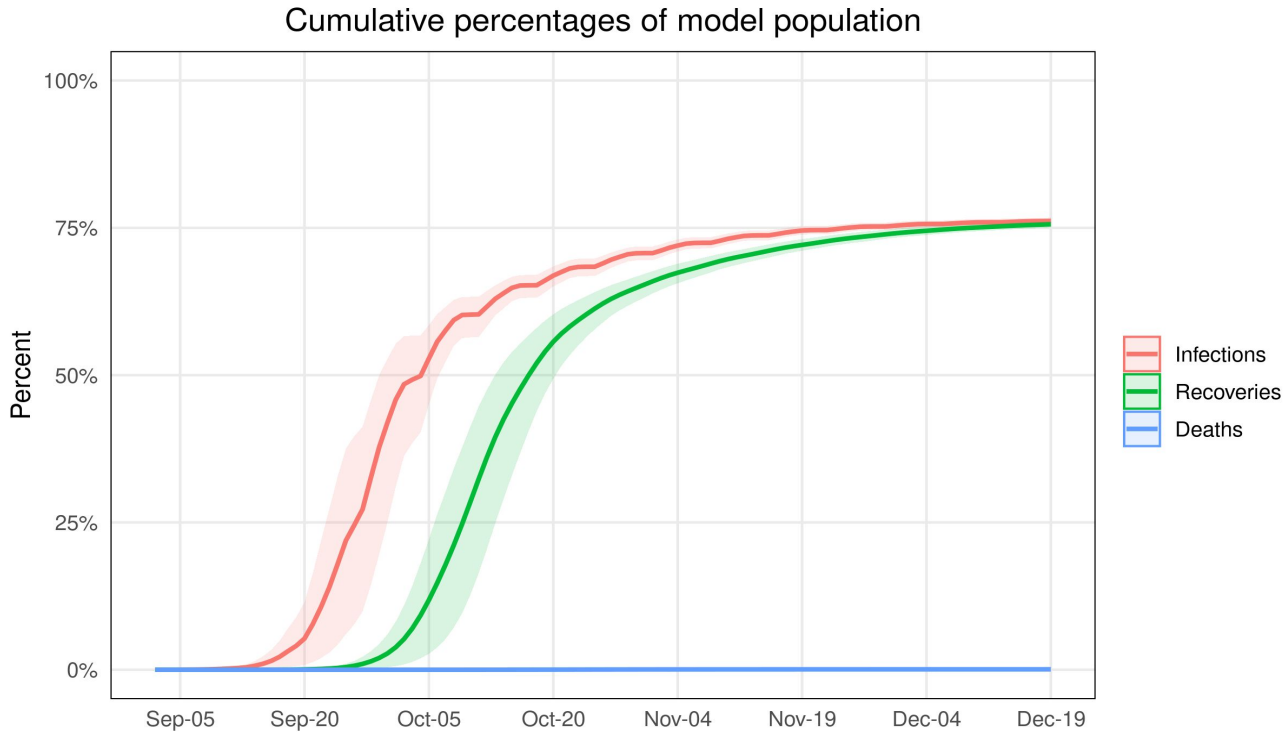
BU Residences Nested subnetworks with different transmission probabilities between connected individuals.

Modified SEIR model

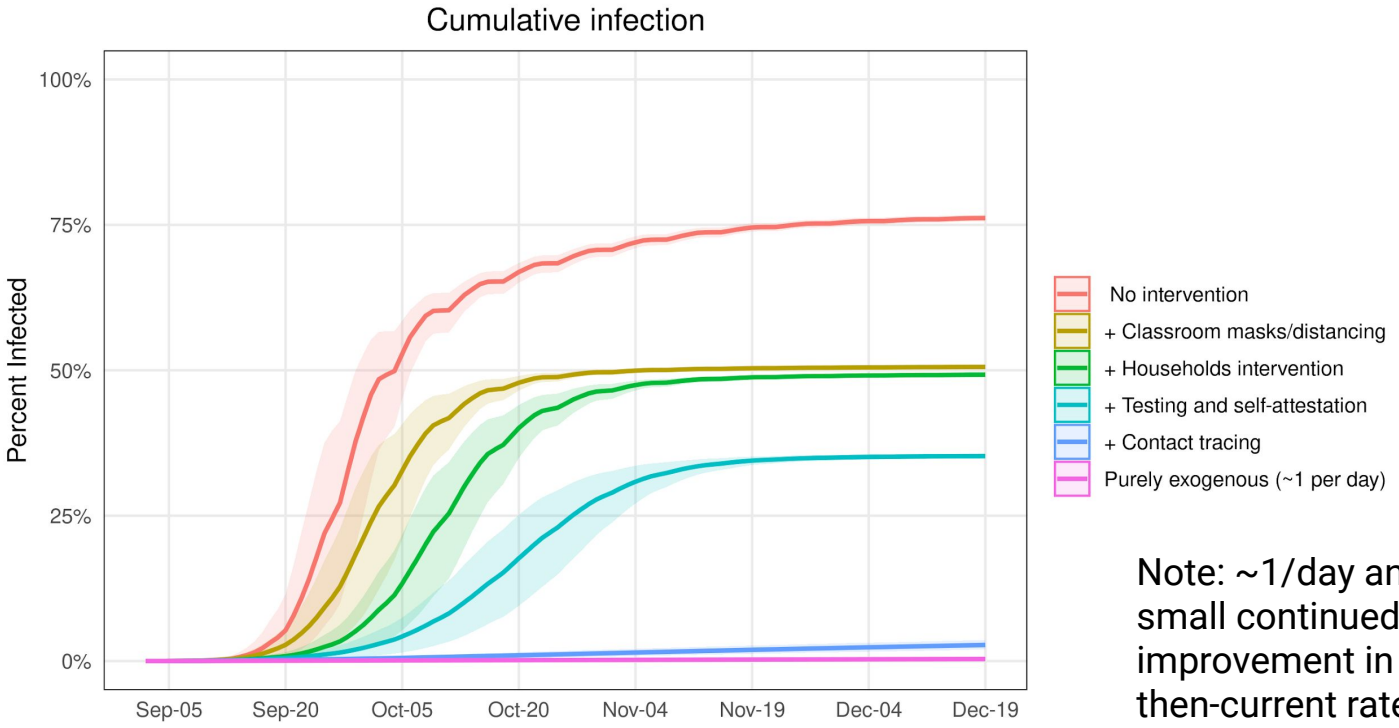


NOTE: Transmission is governed by a baseline rate β that in turn is modified multiplicatively as a function of various factors (e.g., age, contact type, etc.).

What if we did nothing?



Cumulative impact of interventions



Note: ~1/day anticipated a small continued improvement in then-current rates in Suffolk county.

Sensitivity of outcomes to assumptions

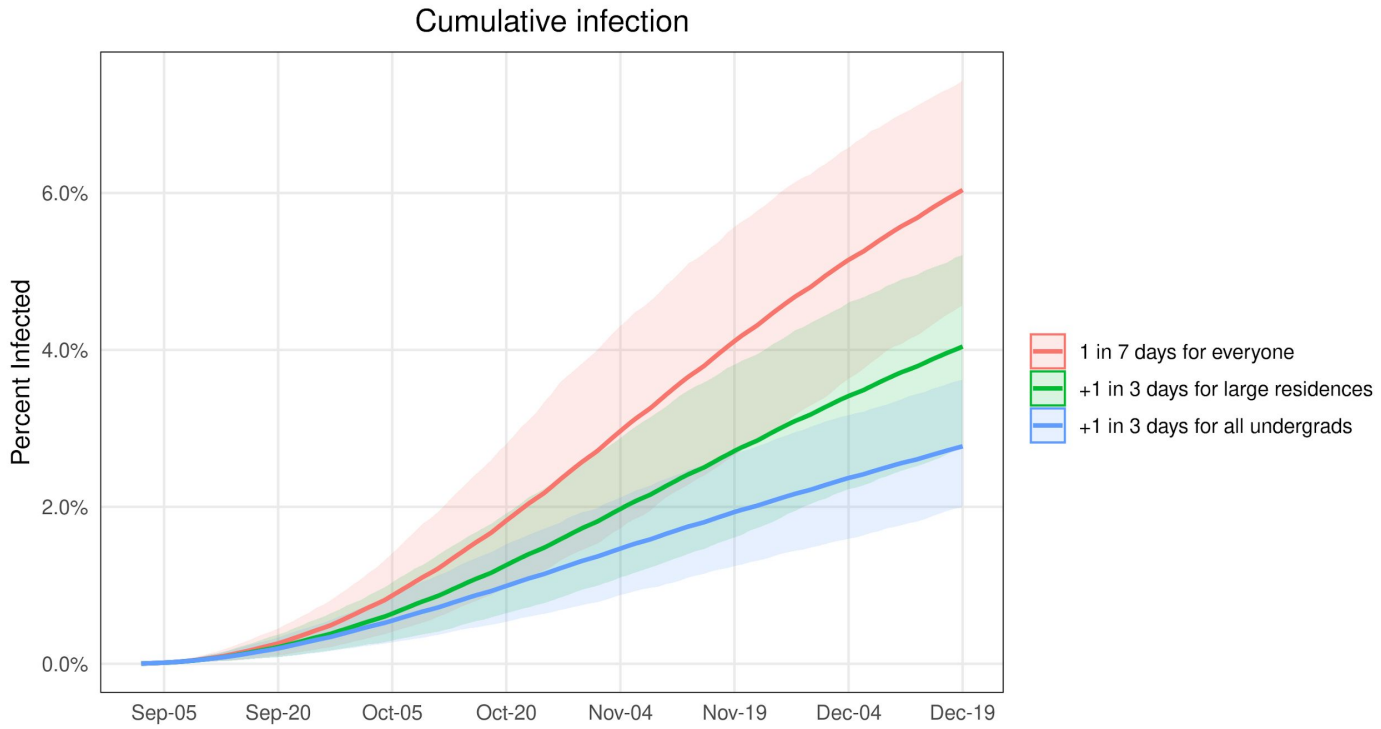
Assumptions come with uncertainty.

Comparing the outcomes under various assumptions provides a sense of the **sensitivity** of our insights to these assumptions and of our strategies to elements we cannot control.

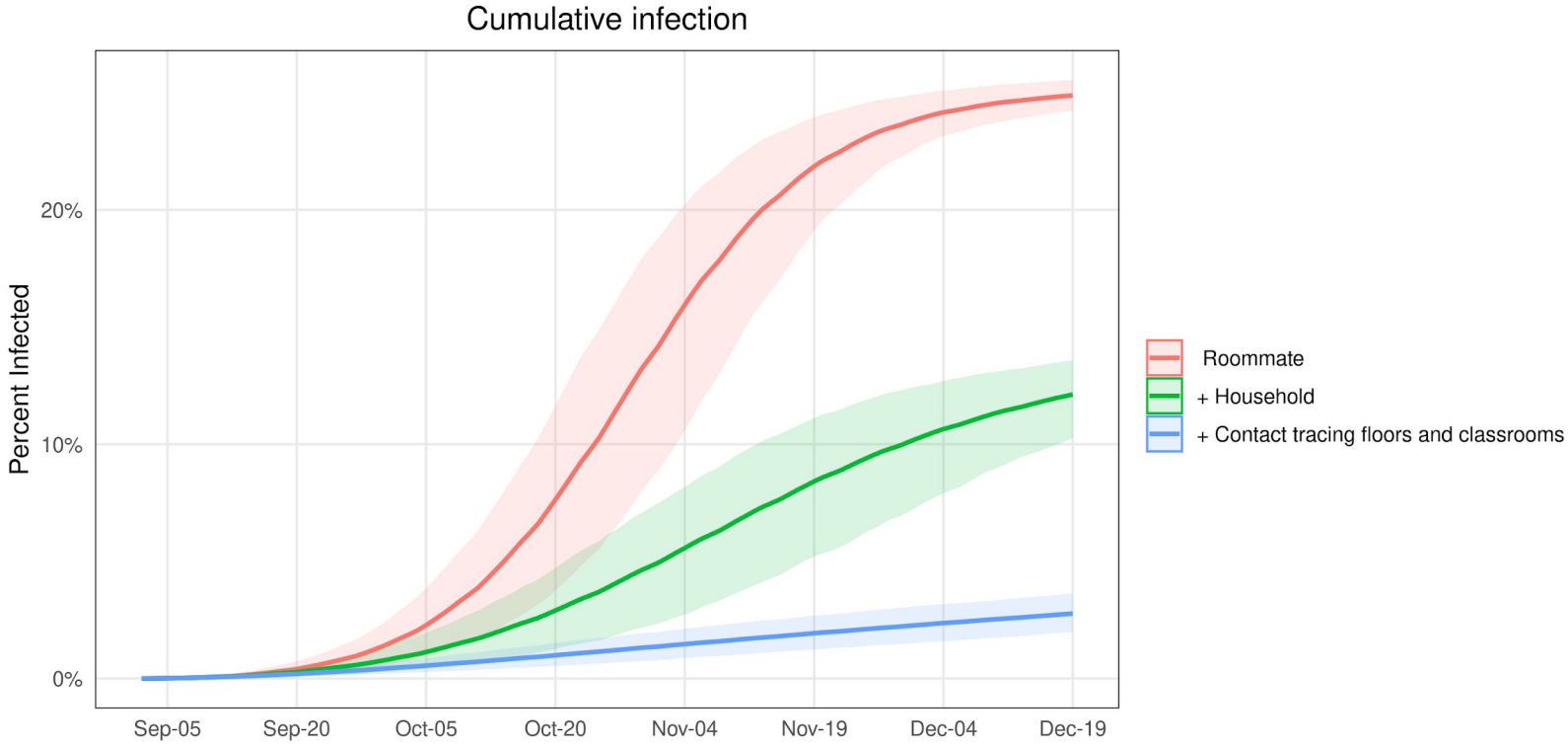
For BU planning purposes, assumptions of particular interest center around

- Testing (and isolation)
- Contact tracing (and quarantine)
- Boston area infectivity

Choice of testing protocol matters



Choice of contact tracing strategy matters



High-level take-home points

- Modeling suggested keeping Covid-19 infections at BU near the “linear regime” was a feasible goal if
  - start with a “screened and cleaned” population;
  - use the full combination of interventions and measures;
  - enforce strong compliance to all protocols;
  - Boston infections remain constant at current rates.

Take-home points (continued)

- Frequency, pattern and accuracy of testing, as well as extent and accuracy of contact tracing, are key to success of BU's strategy -- both individually and in tandem with each other.

- There is some robustness expected to Boston area infectivity levels.

- Compliance to protocols across the board is **absolutely critical**.

Questions?