Epidemic Modeling using the SIR Model and Graph Network

Mentor: Bedřich Sousedík

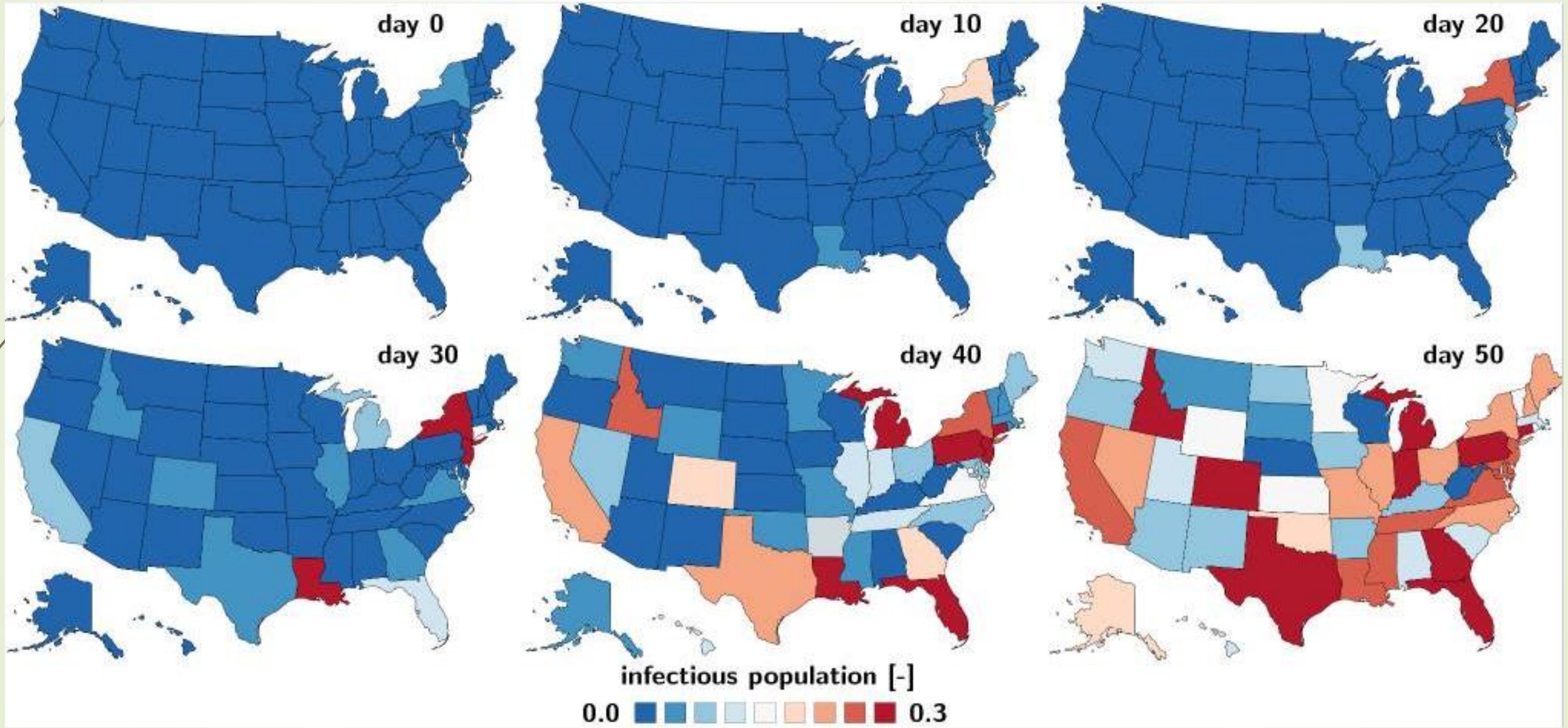
Author: Alex Flitter, Nathan Tamiru

Date: 5/12/2023

What is Mathematic Epidemiology?

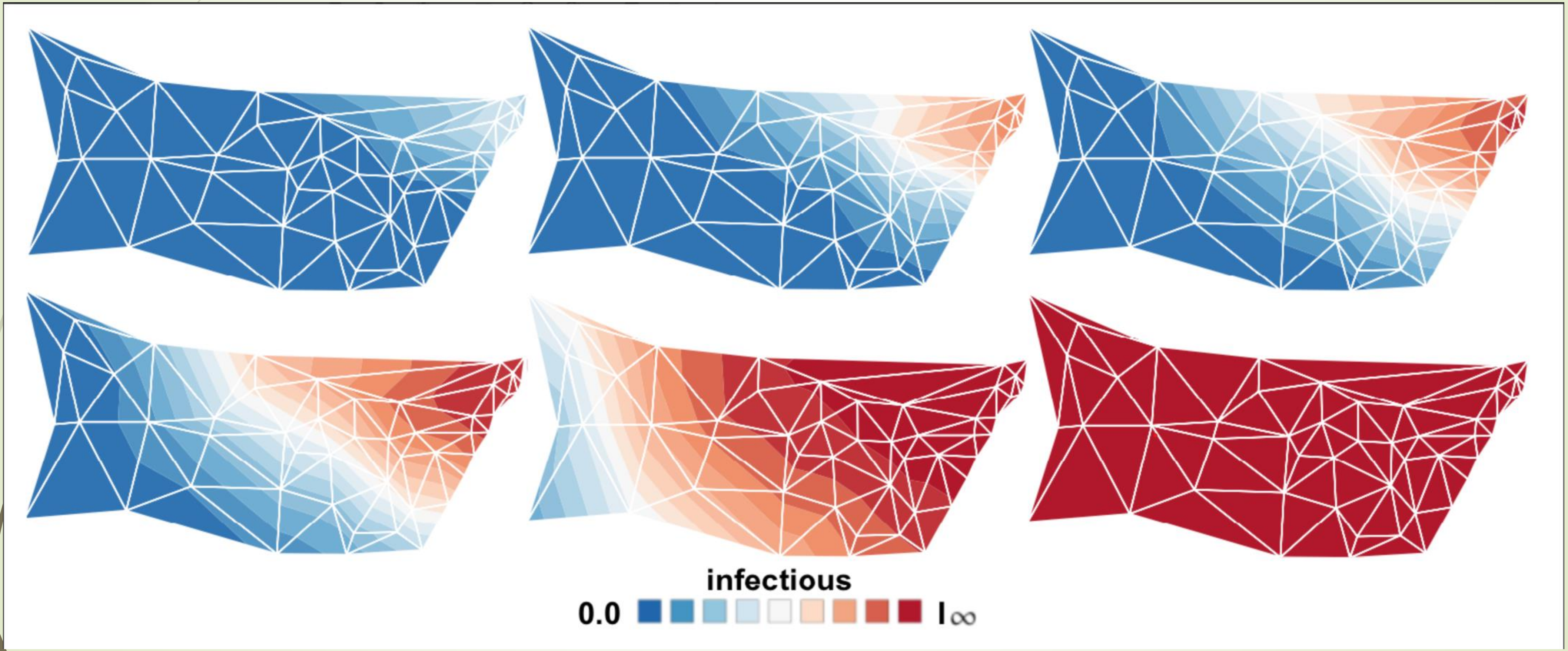
- Mathematic epidemiology is the science of understanding the cause of a disease, predicting its outbreak dynamics, and developing strategies to control it.
- Today's focus is epidemic modeling
  - Using mathematical formulas to anticipate the spread of diseases
    - Calculating maximum number of people infected
    - When can we expect the maximum number of people infected

Why is epidemic modeling important



Peirlinck, M., Linka, K., Sahli Costabal, F., & Kuhl, E. (2020). Outbreak dynamics of COVID-19 in China and the United States. Biomechanics and modeling in mechanobiology, 19(6), 2179–2193. https://doi.org/10.1007/s10237-020-01332-5

Heat Equation for Epidemic Modeling



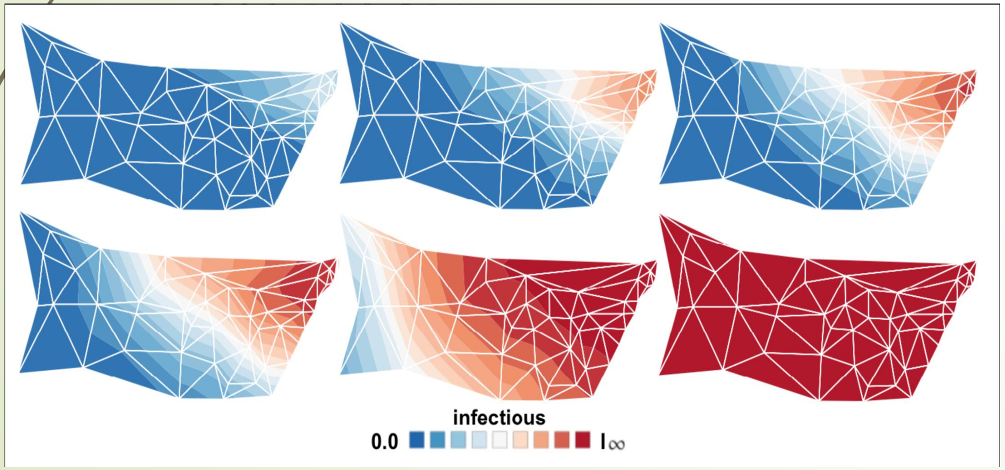
Computational Epidemiology Data-Driven Modeling of COVID-19(pg 189), by E. Kuhl

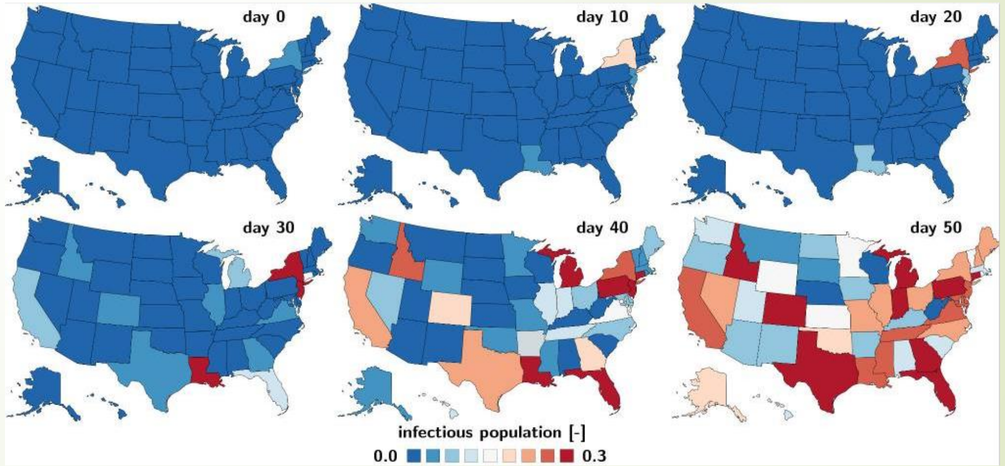
Downsides of using the heat equation

- The heat equation assumes whatever is spread is contiguous
  - If we look at the map of how covid spread through out the united states, we can see that it is not contiguous

$$\dot{I} = \mathrm{div}(D \cdot \nabla I) + \beta I[I_\infty - I]$$

Where $I_\infty = 1 - \frac{\gamma}{\beta}$





Agent based epidemic modeling

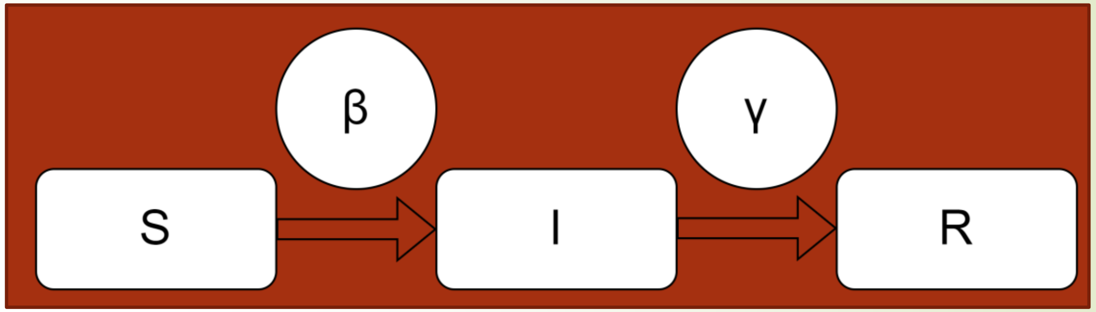
- Dr. Sanjeev Seahra and The Black Arcs
  - https://blackarcs.org/
- Uses locations and schedules

Downsides of using an Agent system

- Agent based models are computationally intensive
  - Only works for small populations
  - Time
  - Memory
- Initial cost to set up is high
  - Survey of people's schedule
    - Inaccurate data

SIR model Variables

- S: number of people who are susceptible
- I: number of people who are infected
- R: number of people who are recovered
- β: infection rate
- γ: recovery rate
- $\dot{S}$ is the rate which susceptible people changes
- $\dot{I}$ is the rate which infected people changes
- $\dot{R}$ is the rate which recovered people changes
- $R_0 = {}^{\beta}/_{\gamma}$: basic reproduction number
- Slow dynamic: $R_0 \approx 2$
- Fast dynamic: $R_0 \geq 4$



SIR explicit model

- $\dot{S} = -\beta SI$
- $\dot{I} = +\beta SI - \gamma I$ (1)
- $\dot{R} = \quad +\gamma I$

We can use forward Euler method with a discrete time step of $\Delta t = t_{n+1} - t_n$ to say

- $\dot{S} = \frac{S_{n+1} - S_n}{\Delta t}$, $\dot{I} = \frac{I_{n+1} - I_n}{\Delta t}$, $\dot{R} = \frac{R_{n+1} - R_n}{\Delta t}$ (2)

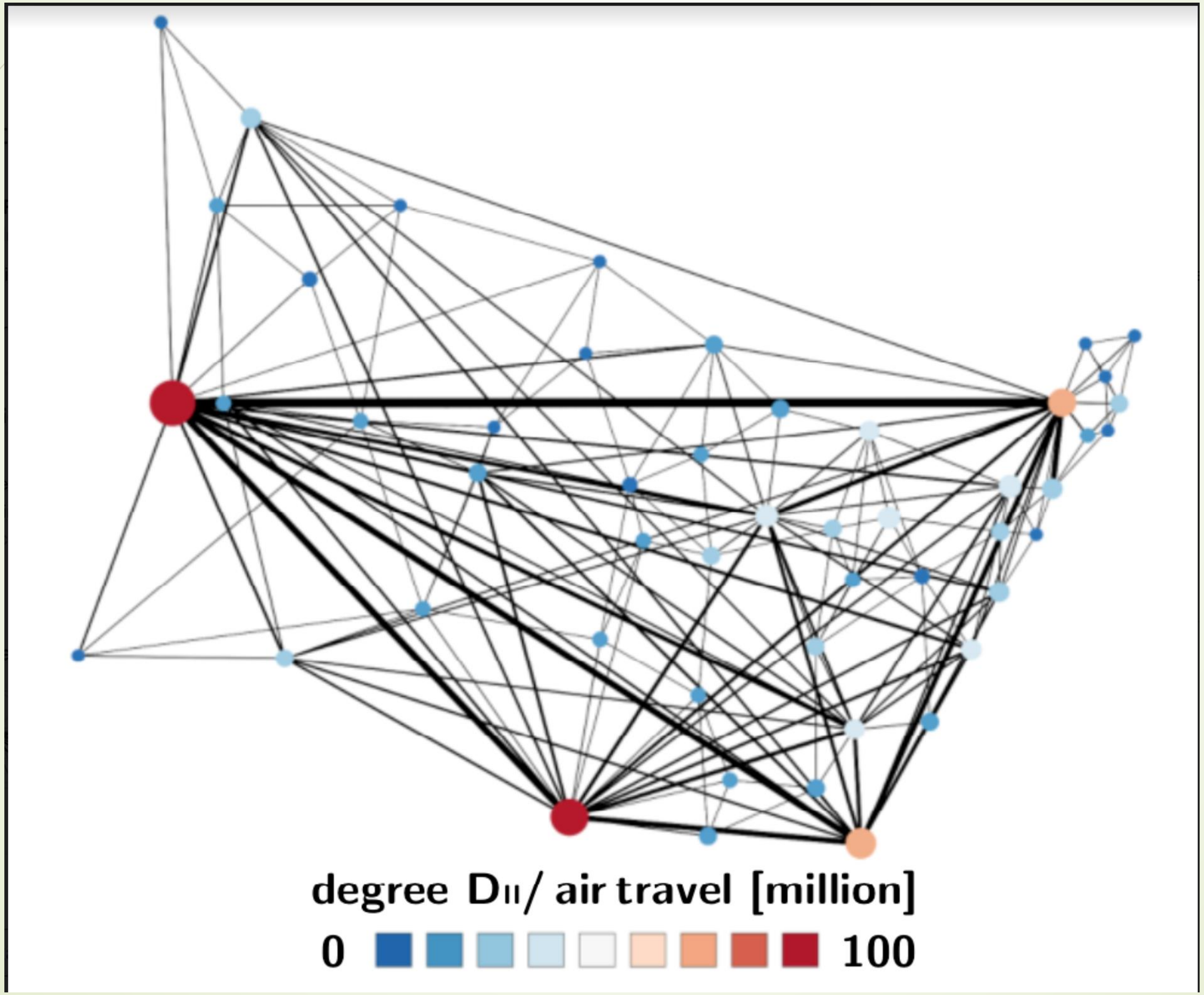
By substituting the list of equations above into the original SIR model equations we get

- $\frac{S_{n+1} - S_n}{\Delta t} = -\beta SI$
- $\frac{I_{n+1} - I_n}{\Delta t} = +\beta SI - \gamma I$ (3)
- $\frac{R_{n+1} - R_n}{\Delta t} = \quad +\gamma I$

And now we can solve for the unknowns S_{n+1}, I_{n+1}, R_{n+1} and get

- $S_{n+1} = S_n - \beta S_n I_n \Delta t$
- $I_{n+1} = I_n + \beta S_n I_n \Delta t - \gamma I_n$ (4)
- $R_{n+1} = R_n \quad + \gamma I_n$

Network diffusion method



Computational Epidemiology Data-Driven Modeling of COVID-19(pg 177), by E. Kuhl

Network diffusion method

- A: $n_{nd} \times n_{nd}$ adjacency matrix for the graph G
- D: $n_{nd} \times n_{nd}$ degree matrix
  - $D_{II} = Sum\ of\ all\ entries\ in\ column\ I$
- Graph Laplacian $n_{nd} \times n_{nd}$, where $L_{IJ} = D_{IJ} - A_{IJ}$

Number of People on Flights per year	California	New York	Arizona
California	0	10,000	2,000
New York	10,000	0	500
Arizona	2,000	500	0

A	California	New York	Arizona
California	0	1	0.2
New York	1	0	0.05
Arizona	0.2	0.05	0

D	California	New York	Arizona
California	1.2	0	0
New York	0	1.05	0
Arizona	0	0	0.25

One Node vs Multi Node Systems

- $\dot{S} = -\beta SI$
- $\dot{I} = +\beta SI - \gamma I$ (1)
- $\dot{R} = \quad +\gamma I$

We can use forward Euler method with a discrete time step of $\Delta t = t_{n+1} - t_n$ to say

- $\dot{S} = \frac{S_{n+1} - S_n}{\Delta t}$, $\dot{I} = \frac{I_{n+1} - I_n}{\Delta t}$, $\dot{R} = \frac{R_{n+1} - R_n}{\Delta t}$ (2)

By substituting the list of equations above into the original SIR model equations we get

- $\frac{S_{n+1} - S_n}{\Delta t} = -\beta SI$
- $\frac{I_{n+1} - I_n}{\Delta t} = +\beta SI - \gamma I$ (3)
- $\frac{R_{n+1} - R_n}{\Delta t} = \quad +\gamma I$

- $\dot{S}_I = -\kappa_S \sum_{J=1}^{n_{nd}} L_{IJ} S_J - \beta S_I I_I$
- $\dot{I}_I = -\kappa_I \sum_{J=1}^{n_{nd}} L_{IJ} I_J + \beta S_I I_I - \gamma I_I$ (1)
- $\dot{R}_I = -\kappa_R \sum_{J=1}^{n_{nd}} L_{IJ} R_J \quad + \gamma I_I$

We can use forward Euler method with a discrete time step of $\Delta t = t_{n+1} - t_n$ to say

- $\dot{S}_I = \frac{S_{I,n+1} - S_{I,n}}{\Delta t}$, $\dot{I}_I = \frac{I_{I,n+1} - I_{I,n}}{\Delta t}$, $\dot{R}_I = \frac{R_{I,n+1} - R_{I,n}}{\Delta t}$ (2)

By substituting the list of equations above into the original SIR model equations we get

- $\frac{S_{I,n+1} - S_{I,n}}{\Delta t} = -\kappa_S \sum_{J=1}^{n_{nd}} L_{IJ} S_{J,n} - \beta S_{I,n} I_{I,n}$
- $\frac{I_{I,n+1} - I_{I,n}}{\Delta t} = -\kappa_I \sum_{J=1}^{n_{nd}} L_{IJ} I_{J,n} + \beta S_{I,n} I_{I,n} - \gamma I_{I,n}$ (3)
- $\frac{R_{I,n+1} - R_{I,n}}{\Delta t} = -\kappa_R \sum_{J=1}^{n_{nd}} L_{IJ} R_{J,n} \quad + \gamma I_{I,n}$

One Node vs Multi Node Systems Part 2

And now we can solve for the unknowns S_{n+1}, I_{n+1}, R_{n+1} and get

- $S_{n+1} = S_n - \beta S_n I_n \Delta t$
- $I_{n+1} = I_n + \beta S_n I_n \Delta t - \gamma I_n$ (4)
- $R_{n+1} = R_n + \gamma I_n$

And now we can solve for the unknowns S_{n+1}, I_{n+1}, R_{n+1} and get

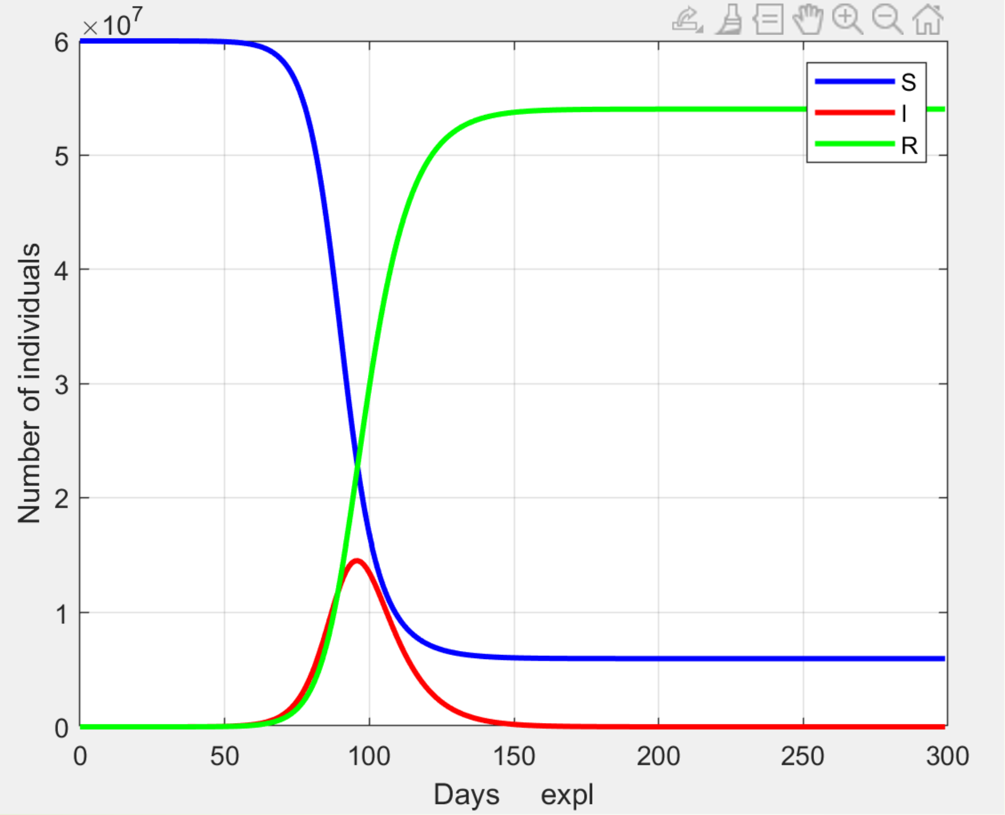
- $S_{I,n+1} = S_{I,n} - \kappa_S \sum_{J=1}^{n_{nd}} L_{IJ} S_{J,n} \Delta t - \beta S_{I,n} I_{I,n} \Delta t$
- $I_{I,n+1} = I_{I,n} - \kappa_I \sum_{J=1}^{n_{nd}} L_{IJ} I_{J,n} \Delta t + \beta S_{I,n} I_{I,n} \Delta t - \gamma I_{I,n} \Delta t$ (4)
- $R_{I,n+1} = R_{I,n} - \kappa_R \sum_{J=1}^{n_{nd}} L_{IJ} R_{J,n} \Delta t + \gamma I_{I,n} \Delta t$

The Mobility Term

- $-\kappa_S \sum_{J=1}^{n_{nd}} L_{IJ} S_J$
- κ_S is a mobility constant
- Let S be [1, .8, 1] and I be the California node
- Then our mobility term looks like
- $-\kappa_S\big((1.2 * 1) + (-1 * .8) + (-.2 * 1)\big) = -.2\kappa_S$

L	California	New York	Arizona
California	1.2	-1	-0.2
New York	-1	1.05	-0.05
Arizona	-0.2	-0.05	.25

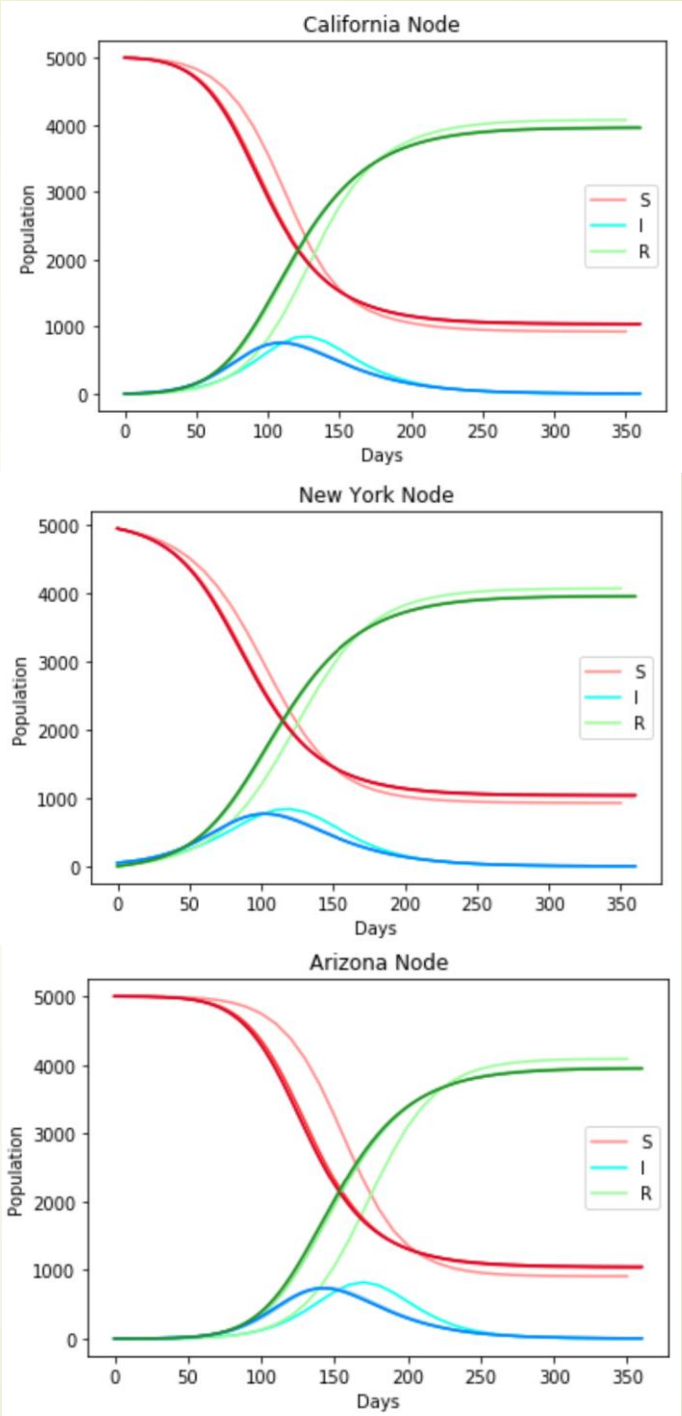
One Node SIR Graph



Slow Dynamic Example

I_max(Peak, Day)	California	New York	Arizona
Δt = 10	(850, 130)	(839, 120)	(821, 170)
Δt = 1	(766, 111)	(775, 104)	(743, 145)
Real solution (ODE45)	(758, 110)	(767, 102)	(735, 142)

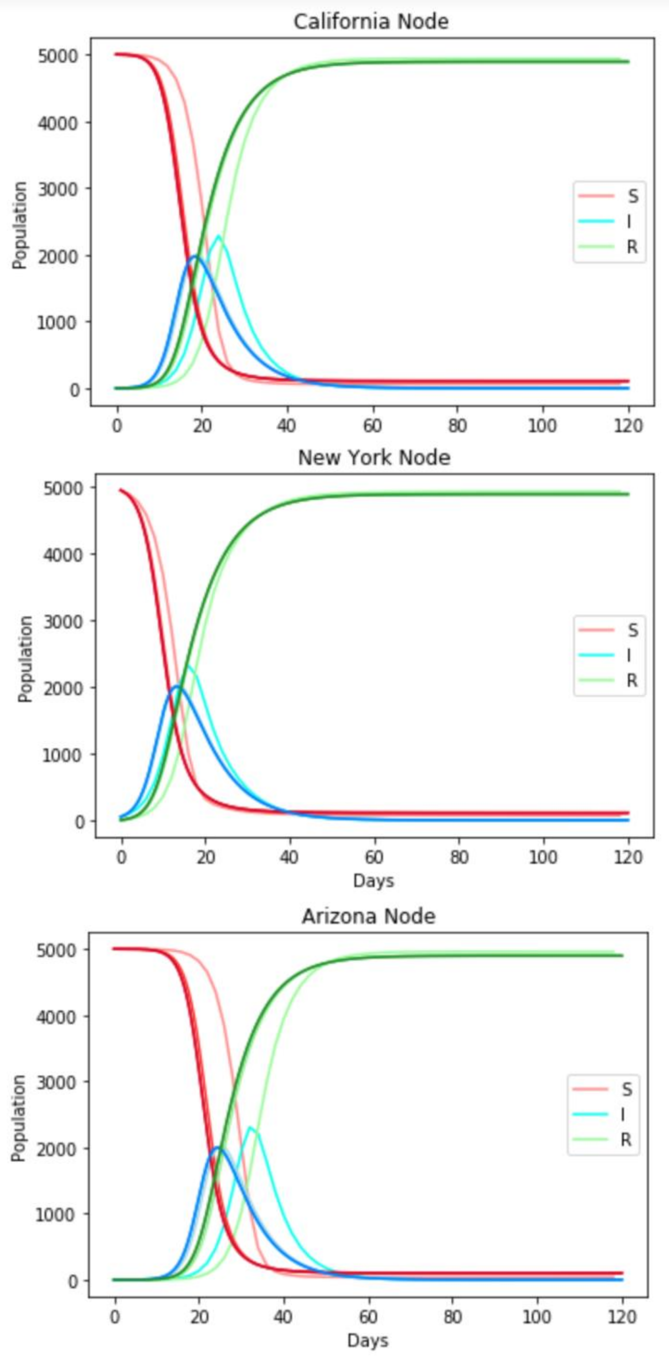
I_max Error	California	New York	Arizona
Δt = 10	12%	9%	12%
Δt = 1	1%	1%	1%



Fast Dynamic Example

I_max(Peak, Day)	California	New York	Arizona
Δt = 2	(2287, 24)	(2324, 16)	(2304, 32)
Δt = 1	(2115, 21)	(2145, 15)	(2139, 29)
Δt = .2	(2000, 19)	(2030, 14)	(2024, 25)
Real solution (ODE45)	(1973, 18)	(2006, 13)	(1998, 24)

I_max Error	California	New York	Arizona
Δt = 2	16%	16%	15%
Δt = 1	7%	7%	7%
Δt = .2	1%	1%	1%



Single Node Implicit System

- $\dot{S} = -\beta SI$
- $\dot{I} = +\beta SI - \gamma I$
- $\dot{R} = \quad +\gamma I$

Again, we can use Implicit/backwards Euler method with a discrete time step of $\Delta t = t_{n+1} - t_n$ to say

- $\dot{S} = \frac{S_{n+1} - S_n}{\Delta t}$, $\dot{I} = \frac{I_{n+1} - I_n}{\Delta t}$, $\dot{R} = \frac{R_{n+1} - R_n}{\Delta t}$

By substituting the list of equations above into the original SIR model equations we get

$$\frac{S_{n+1} - S_n}{\Delta t} = -\beta S_{n+1} I_{n+1}$$

$$\frac{I_{n+1} - I_n}{\Delta t} = +\beta S_{n+1} I_{n+1} - \gamma I_{n+1}$$

$$\frac{R_{n+1} - R_n}{\Delta t} = +\gamma I_{n+1}$$

$$\Longrightarrow$$

$$\frac{S_{n+1} - S_n}{\Delta t} + \beta S_{n+1} I_{n+1} = 0$$

$$\frac{I_{n+1} - I_n}{\Delta t} - \beta S_{n+1} I_{n+1} + \gamma I_{n+1} = 0$$

$$\frac{R_{n+1} - R_n}{\Delta t} - \gamma I_{n+1} = 0$$

Single Node Implicit System

For every time step, use Newton's Method to solve the implicit model:
Calculate the residual:

$$Res = \begin{bmatrix} \frac{S_{n+1}-S_n}{\Delta t} + \beta S_{n+1} I_{n+1} \\ \frac{I_{n+1}-I_n}{\Delta t} - \beta S_{n+1} I_{n+1} + \gamma I_{n+1} \\ \frac{R_{n+1}-R_n}{\Delta t} - \gamma I_{n+1} \end{bmatrix} \tag{1}$$

Calculate the Jacobian:

$$DRes = \begin{bmatrix} \frac{1}{\Delta t} + \beta I_{n+1} & \beta S_{n+1} & 0 \\ -\beta I_{n+1} & \frac{1}{\Delta t} - \beta S_{n+1} + \gamma & 0 \\ 0 & -\gamma & \frac{1}{\Delta t} \end{bmatrix} \tag{2}$$

Solve for update "s":

$$\begin{aligned} DRes(\alpha_k)s &= -Res(\alpha_k) \\ \alpha_{k+1} &= \alpha_k + s \end{aligned} \qquad \alpha_k = (S, I, R) \tag{3}$$

Update model until convergence to a set tolerance level or for a set number of times:

Multi-Node Implicit System

- In every time step, for every node we will solve for the system:

$$\frac{S_{I,n+1} - S_{I,n}}{\Delta t} = -\kappa_S \sum_{J=1}^{n_d} L_{I,J} S_{J,n+1} \ - \beta S_{I,n+1} I_{I,n+1}$$

$$\frac{I_{I,n+1} - I_{I,n}}{\Delta t} = -\kappa_I \sum_{J=1}^{n_d} L_{I,J} I_{J,n+1} \ + \beta S_{I,n+1} I_{I,n+1} - \gamma I_{I,n+1} \Delta t$$

$$\frac{R_{I,n+1} - R_{I,n}}{\Delta t} = -\kappa_R \sum_{J=1}^{n_d} L_{I,J} R_{J,n+1} \ + \gamma I_{I,n+1}$$

- We will obtain a solution of the form:

$$\begin{pmatrix} S_1 \\ I_1 \\ R_1 \\ \vdots \\ S_{n_{nd}} \\ I_{n_{nd}} \\ R_{n_{nd}} \end{pmatrix}$$

Multi-Node Implicit System

For every time step, use Newton's Method to solve the implicit model:
Calculate the residual of the multi-node system:

$$Res = \begin{pmatrix} \frac{S_{1,n+1}-S_{I,n}}{\Delta t} + \kappa_S \sum_{J=1}^{n_d} L_{I,J} S_{J,n+1} \ + \beta S_{1,n+1} I_{1,n+1} \\ \frac{I_{1,n+1}-I_{1,n}}{\Delta t} + \kappa_I \sum_{J=1}^{n_d} L_{I,J} I_{J,n+1} \ - \beta S_{1,n+1} I_{1,n+1} + \gamma I_{1,n+1} \Delta t \\ \frac{R_{1,n+1}-R_{1,n}}{\Delta t} + \kappa_R \sum_{J=1}^{n_d} L_{I,J} R_{J,n+1} \ - \gamma I_{1,n+1} \\ \vdots \\ \frac{S_{n_{nd},n+1}-S_{n_{nd},n}}{\Delta t} + \kappa_S \sum_{J=1}^{n_d} L_{I,J} S_{J,n+1} \ + \beta S_{n_{nd},n+1} I_{n_{nd},n+1} \\ \frac{I_{n_{nd},n+1}-I_{n_{nd},n}}{\Delta t} + \kappa_I \sum_{J=1}^{n_d} L_{I,J} I_{J,n+1} \ - \beta S_{n_{nd},n+1} I_{n_{nd},n+1} + \gamma I_{n_{nd},n+1} \Delta t \\ \frac{R_{n_{nd},n+1}-R_{n_{nd},n}}{\Delta t} + \kappa_R \sum_{J=1}^{n_d} L_{I,J} R_{J,n+1} \ - \gamma I_{n_{nd},n+1} \end{pmatrix}$$

We obtain a vector in $\mathbb{R}^{3\times n_{nd}}$

Multi-Node Implicit System

Calculate the Jacobian using the Kronecker Product to expand our dimensions:

$$DRes = \begin{bmatrix} \frac{1}{\Delta t} + \beta I_{n+1} & \beta S_{n+1} & 0 \\ -\beta I_{n+1} & \frac{1}{\Delta t} - \beta S_{n+1} + \gamma & 0 \\ 0 & -\gamma & \frac{1}{\Delta t} \end{bmatrix} \otimes I_{n_{nd}} + \begin{bmatrix} \kappa_S & 0 & 0 \\ 0 & \kappa_I & 0 \\ 0 & 0 & \kappa_R \end{bmatrix} \otimes L_{I,J}$$

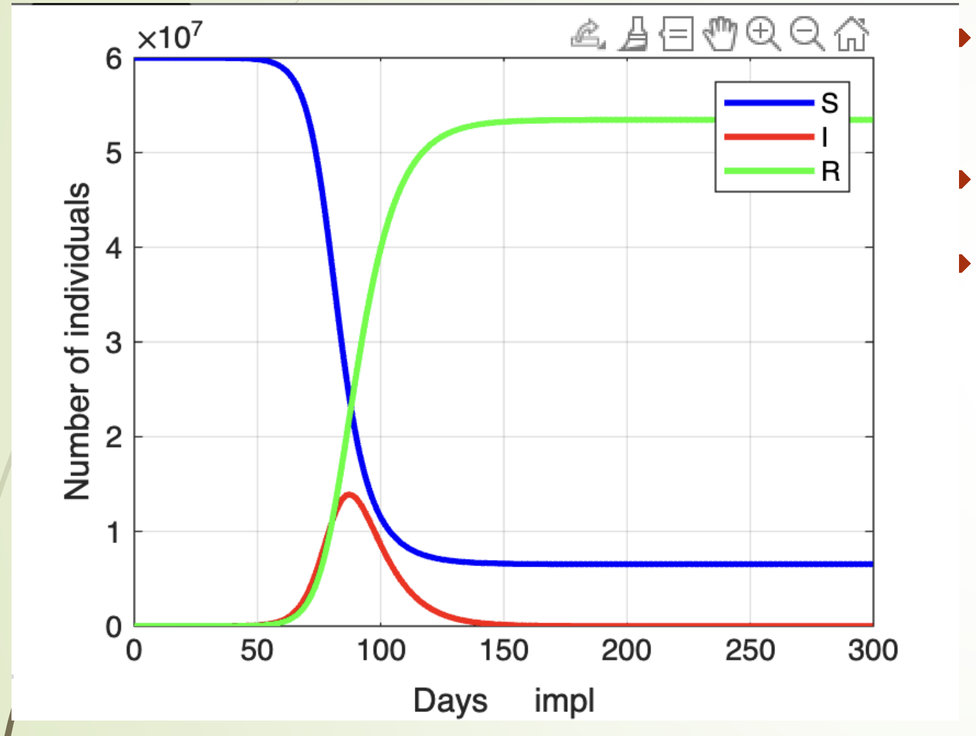
We will obtain a $3n_{nd} \times 3n_{nd}$ matrix

Solve for update "s":

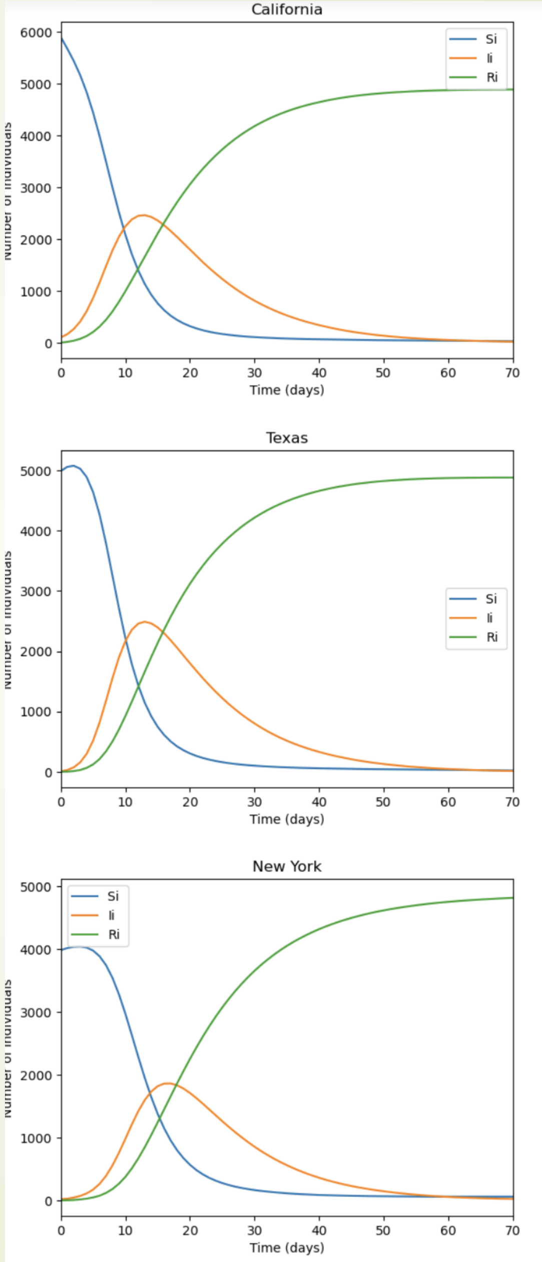
$$DRes(\alpha_k)s = -Res(\alpha_k)$$
$$\alpha_{k+1} = \alpha_k + s$$
$$\alpha_k = (S, I, R)$$

Update for a set number of iterations or until convergence to a set tolerance level.

Implicit Method Graphs



- B = .0001, γ = .1, S0 = [6000,5000,4000]
- I0 = [100,10,20]
- R0 = [0,0,0]



Features of the Implementations

	Explicit/Forward	Implicit/ Backward
Accuracy	May result in inaccurate solutions especially with large time steps	Generally more accurate
Computational Cost	Efficient/ Simple Iterations/ fast convergence	Computationally expensive
Stability	Oscillations or divergent solutions may occur	More stable
Stiffness	Stiffness may cause numerical instability or slow convergence	Handles Stiffness well

Conclusion

- Fast dynamic
  - $\Delta t > 1$
    - If you are fine with an error of around 7% use explicit method
    - Otherwise use implicit method to reduce the error
  - $\Delta t \leq 1$
    - Use explicit method
- Slow dynamic
  - $\Delta t > 1$
    - If you are fine with an error of around 7% use explicit method
    - Otherwise use implicit method to reduce the error
  - $\Delta t \leq 5$
    - Use explicit method