



Universität Stuttgart

# Agenten-basierte Modelle

Eric  
Sîrbu

# Überblick

- Motivation
- Modell des Modells
- Beispiel Covid-19
- Beispiel eigene Implementierung

# Motivation

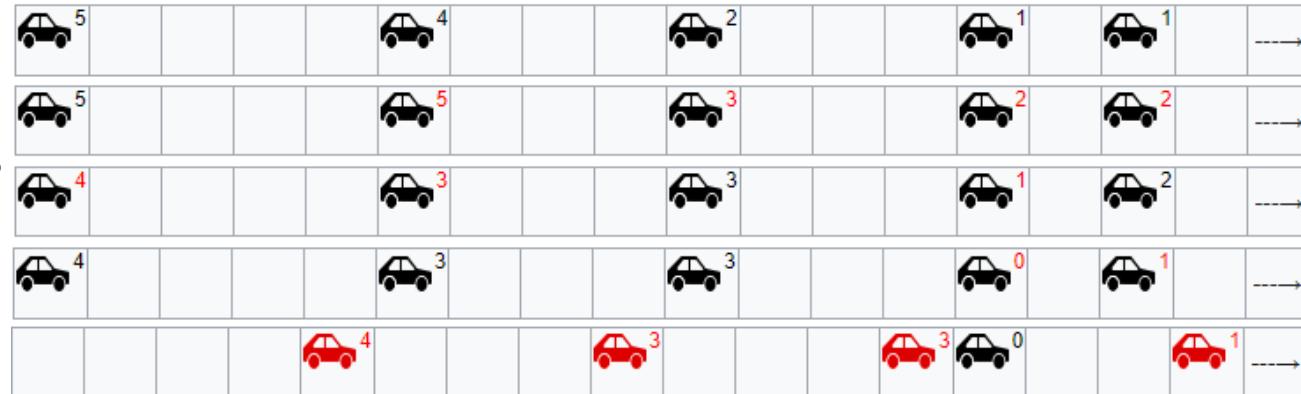
- System aus Agenten
- Nicht-linear zusammengesetzt
- Agenten im Vergleich simpel
- Low-level Auswertung
  
- Anwendungen:
  - Aktienmarkt
  - Physischer Fluss
  - Soziale Netzwerke
  - Epidemien

# Modell des Modells

- Agenten
  - Eigenschaften und Verhalten
- Umgebung
  - Interaktion unter Agenten

# Modell des Modells

- Agenten
  - Eigenschaften und Verhalten
- Umgebung
  - Interaktion unter Agenten
- Nagel-Schreckenberg
  - Entstehung von Staus
  - Trödeln als Erklärung
  - Erweiterung möglich



[https://de.wikipedia.org/wiki/Nagel-Schreckenberg-Modell#Beispiel\\_f%C3%BCr\\_den\\_Ablauf\\_einer\\_Runde](https://de.wikipedia.org/wiki/Nagel-Schreckenberg-Modell#Beispiel_f%C3%BCr_den_Ablauf_einer_Runde)

# Beispiel Covid-19

## Motivation

- Pandemie ist schlecht
- Auswirkungen lindern ist gut
- Kompliziertes Zusammenspiel
- Bewertung und Prognose nötig
- Interdisziplinäre Kommunikation
- Menschen als Agenten
- Infektionsradius als Nachbarschaft

# Beispiel Covid-19

## Annahmen über Krankheit

- Neue Krankheit (mehr oder weniger)
- Ausgangssituation bekannt
- Ansteckrate?
- Infektionsdauer?
- Todesrate?
- Immunität?

# Beispiel Covid-19

## Annahmen über Menschen

- Mehrere mögliche Zustände
  - Anfällig
  - Infiziert
  - Erholt
  - (Tot)
- Bewegung durch die Simulation
- Bevölkerungsverteilung
  - Alter
  - Gesundheit



# Beispiel Covid-19

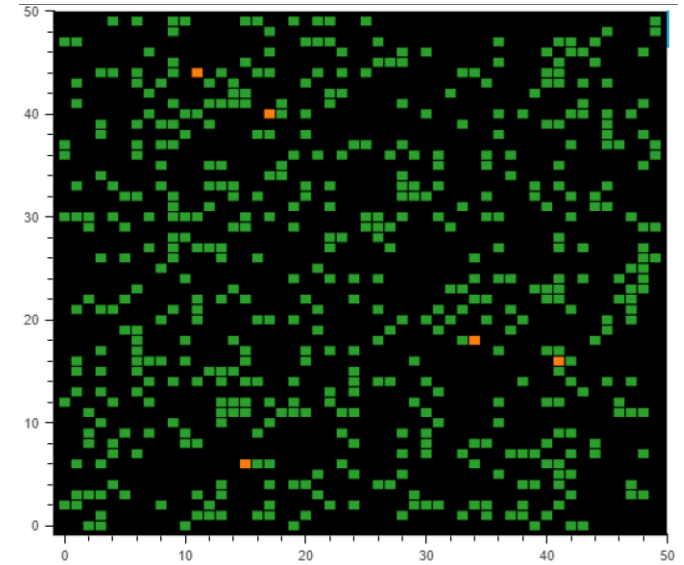
## Annahmen über Maßnahmen

- Quarantäne der (Teil-)Bevölkerung
- Social Distancing der (Teil-)Bevölkerung
- Schließung von Schulen
- Reiseeinschränkungen
  
- Unbekannte Infektionen?
- Einhaltung?
- Verhaltensänderungen?

# Beispiel Covid-19

## Abstraktion

- Genauigkeit vs. Simplizität
  - Auswirkung auf Bewertung
- Simulationsfläche
  - Kleines Quadrat ausreichend?
- Menschen
  - Bewegung
  - Beziehungen
  - Mehr Zustände
    - Asymptomatische Fälle



# Beispiel Covid-19

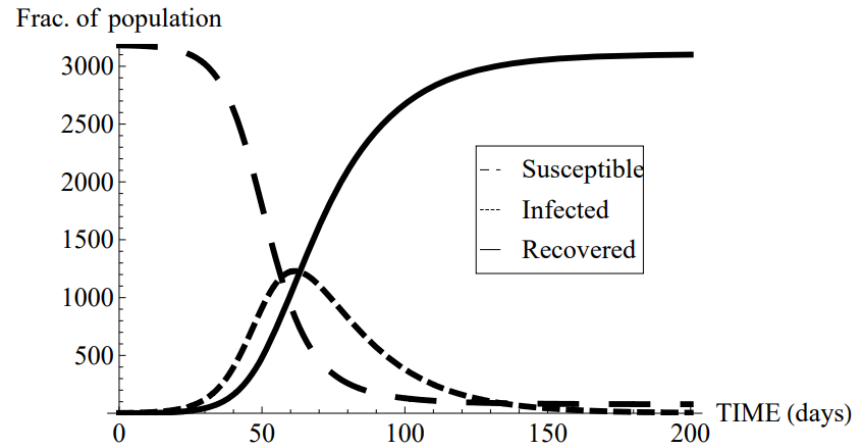
## Ergebnisse

- Prognose
  - Todesfälle
  - Pandemiedauer
  - Benötigte Fachkräfte
- Kombinationen testen
- Abläufe beobachten
- Weiter Testen

# Beispiel Covid-19

## Ergebnisse

- Prognose
  - Todesfälle
  - Pandemiedauer
  - Benötigte Fachkräfte
- Kombinationen testen
- Abläufe beobachten
- Weiter Testen
- Validierung
  - SIR-Modell



# Beispiel einfache Implementierung

# Beispiel einfache Implementierung

## Initialisierung

```
1 def __init__(self, N=500, width=50, height=50, infected_start=0.01, ...):
2     for i in range(N):
3         a = MyAgent(i, self)
4         self.schedule.add(a)
5         while len(self.grid.get_cell_list_contents([(x,y)])) != 0:
6             x = self.random.randrange(self.grid.width)
7             y = self.random.randrange(self.grid.height)
8         self.grid.place_agent(a, (x,y))
9         if i < N * infected_start:
10             a.state = State.INFECTED
11             a.recovery_time = self.get_recovery_time()
```

# Beispiel einfache Implementierung

## Status

```
1 if self.state == State.INFECTED:
2     dead = self.model.death_rate > random.random()
3     if dead:
4         self.model.dead_agents += 1
5         self.model.schedule.remove(self)
6         self.model.grid.remove_agent(self)
7     elif self.model.schedule.time - self.infection_time >= self.recovery_time:
8         self.state = State.RECOVERED
```

# Beispiel einfache Implementierung

## Bewegung

```
1 neighborhood = self.model.grid.get_neighborhood(self.pos, ...)
2 new_position = random.choice(neighborhood)
3 if len(self.model.grid.get_cell_list_contents([new_position])) == 0:
4     self.model.grid.move_agent(self, new_position)
```



# Beispiel einfache Implementierung

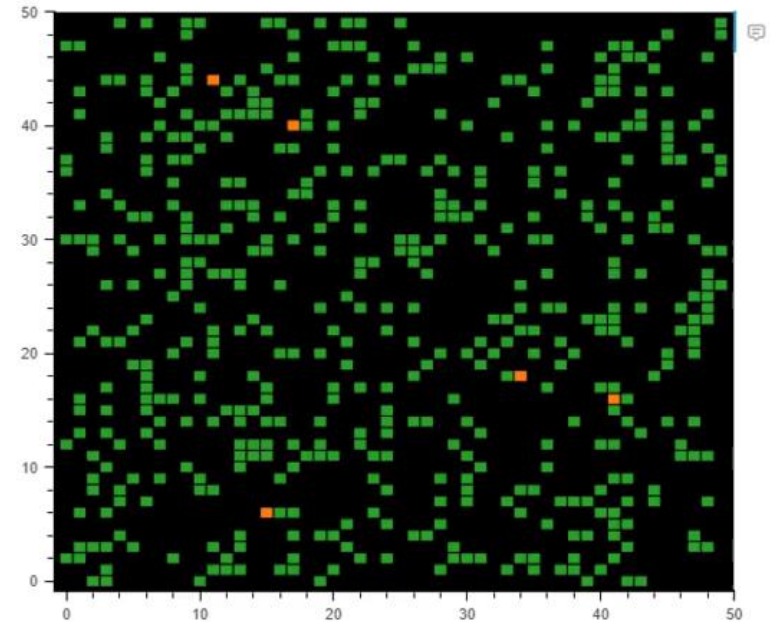
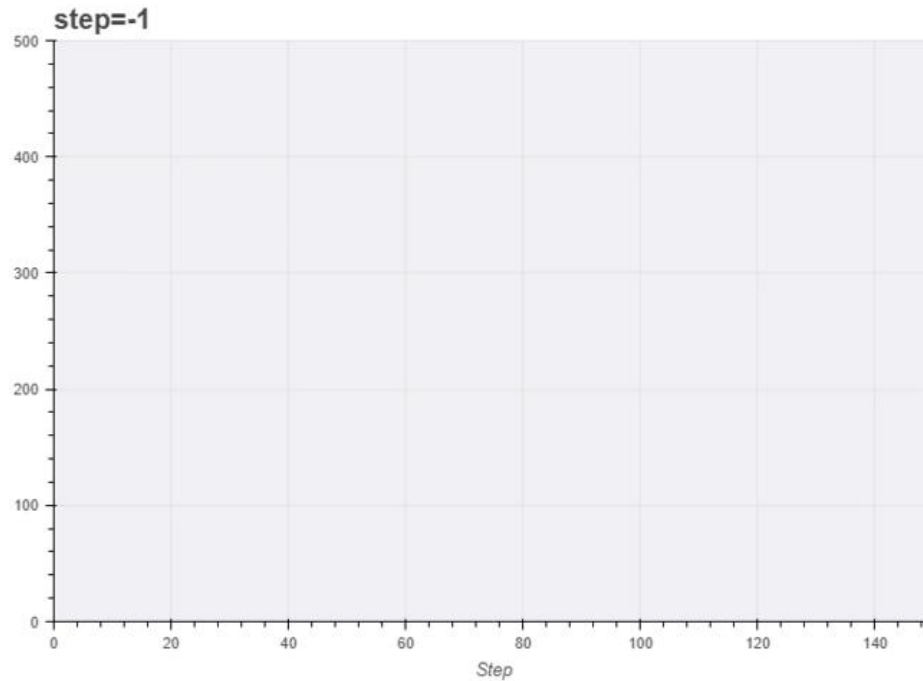
## Bewegung und Kontakt

```
1 neighborhood = self.model.grid.get_neighborhood(self.pos, ...)
2 new_position = random.choice(neighborhood)
3 if len(self.model.grid.get_cell_list_contents([new_position])) == 0:
4     self.model.grid.move_agent(self, new_position)
```

```
1 if self.state == State.INFECTED:
2     neighborhood = self.model.grid.get_neighborhood(self.pos, ...)
3     neighbours = self.model.grid.get_cell_list_contents(neighborhood)
4     for other in neighbours:
5         if other.state is State.SUSCEPTIBLE:
6             if self.model.p_transmission > random.random():
7                 other.state = State.INFECTED
8                 other.infection_time = self.model.schedule.time
9                 other.recovery_time = self.model.get_recovery_time()
```

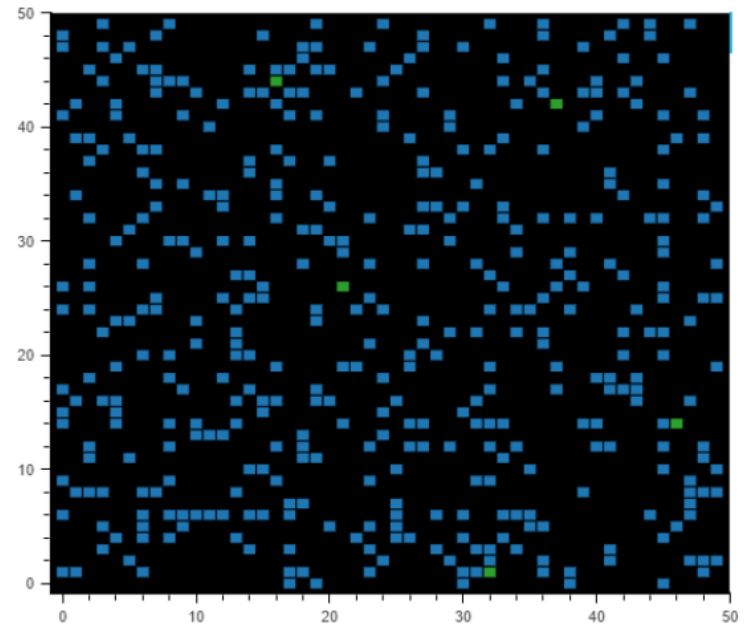
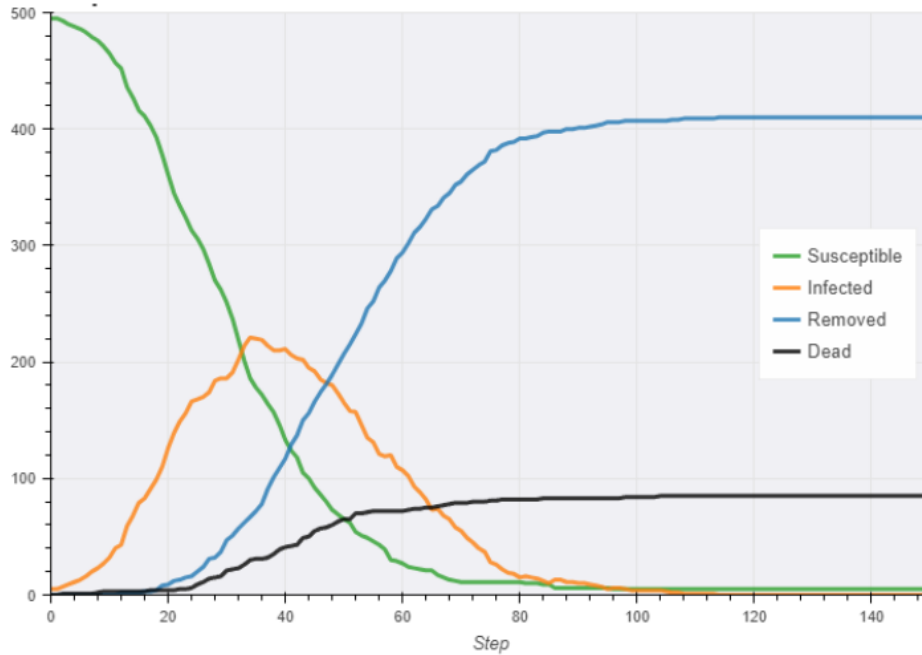
# Beispiel einfache Implementierung

## Basis



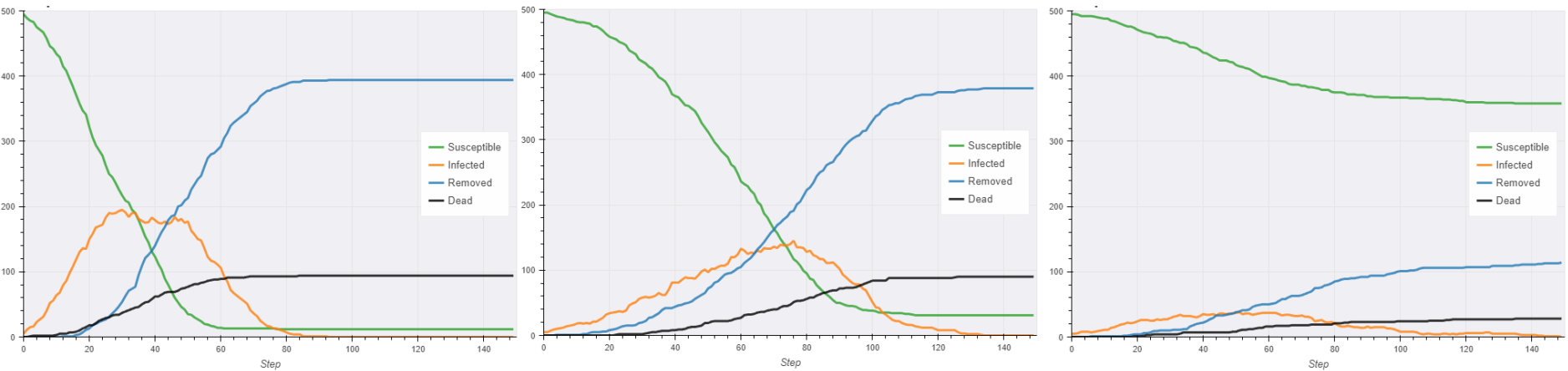
# Beispiel einfache Implementierung

## Basis



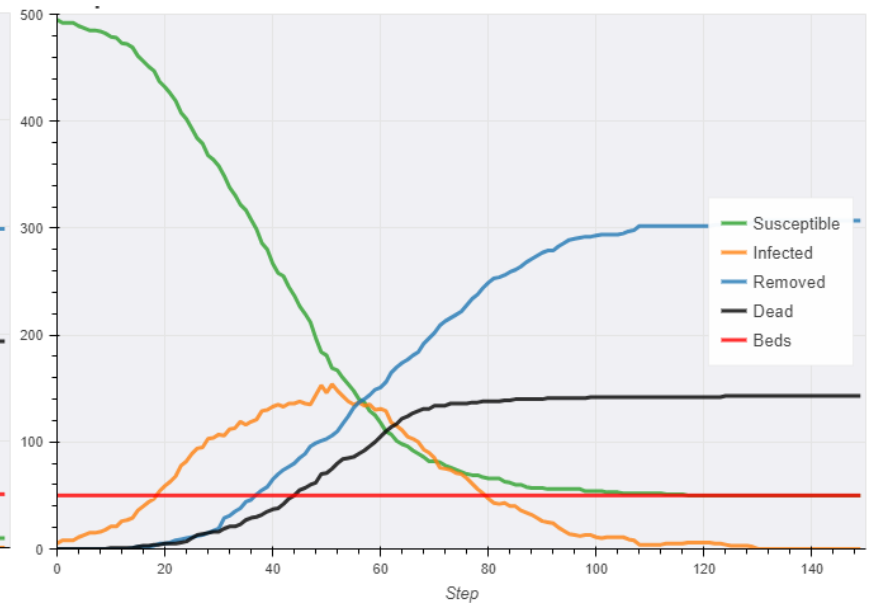
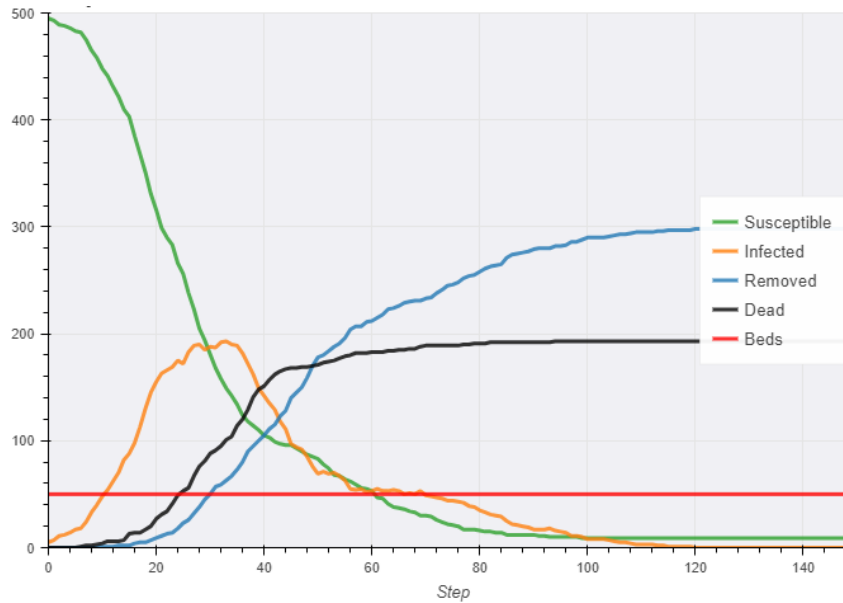
# Beispiel einfache Implementierung

## Unterschiedliche Infektionsraten



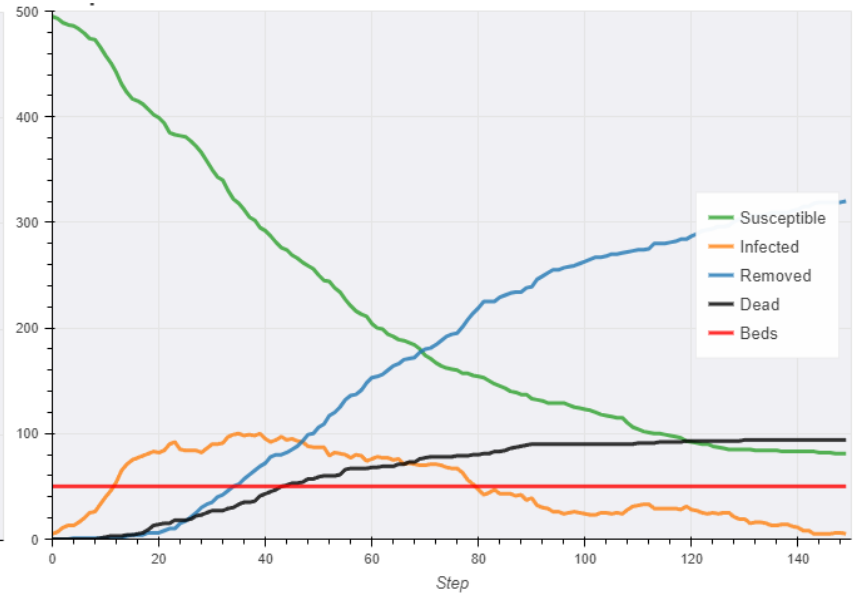
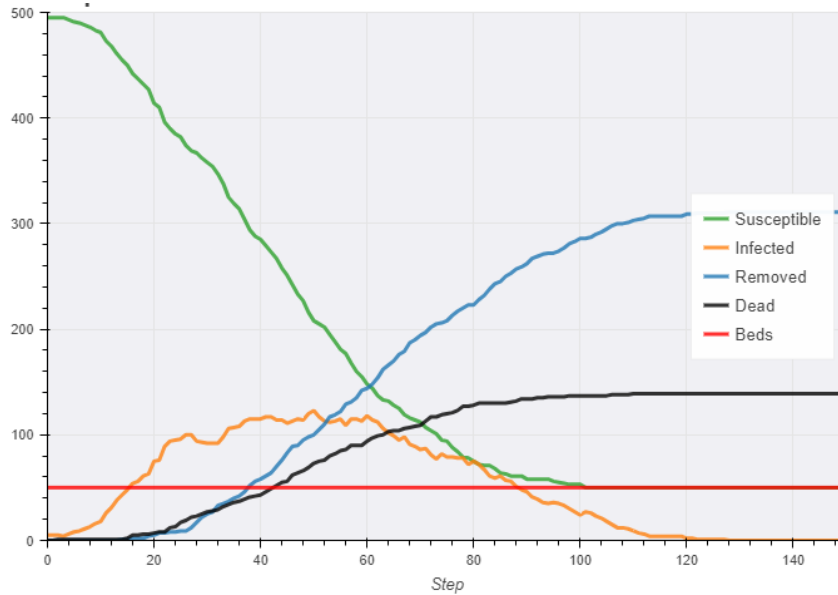
# Beispiel einfache Implementierung

## Verfeinerung



# Beispiel einfache Implementierung

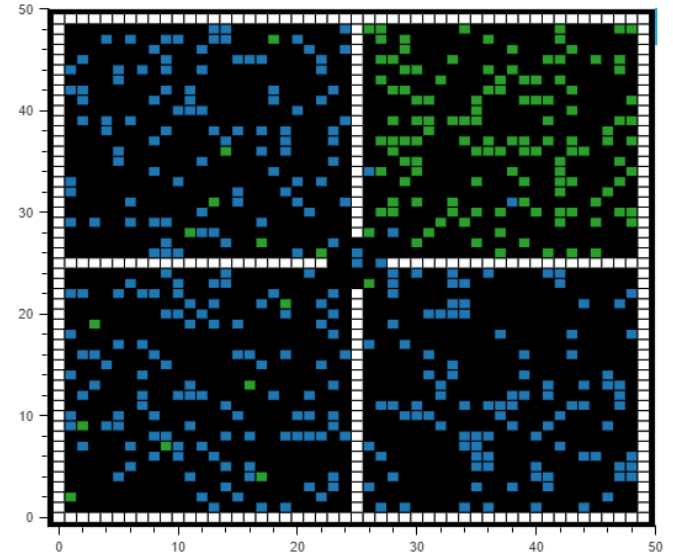
## Reiseeinschränkungen?



# Beispiel einfache Implementierung

## Diskussion

- Einfach zu erstellen
- Leicht zu verstehen
- Praktisch zu erweitern
- Sieht korrekt aus



# Beispiel einfache Implementierung

## Diskussion

- Einfach zu erstellen
  - Leicht zu verstehen
  - Praktisch zu erweitern
  - Sieht korrekt aus
- 
- Nur 500 Agenten
  - Sehr Quadratisch
  - Homogene Agenten
  - Willkürliche Werte



# Zusammenfassung

- ABM sind simpel
- Können komplizierte Systeme darstellen
  - Heterogene Akteure
  - Wechselndes Verhalten
  - Keine globalen Entscheidungen
- Leicht erweiterbar
- Variable Abstraktion
  - Richtiges Niveau treffen