Agent Modeling of a Sarin Attack in Manhattan

Venkatesh Mysore, Giuseppe Narzisi, Lewis Nelson, Dianne Rekow, Marc Triola, Alan Shapiro, Clare Coleman, Ofer Gill, Raoul-Sam Daruwala & Bud Mishra
Talk outline

- New York University’s Center for Catastrophe Preparedness & Response (NYU CCPR)
  - Large Scale Emergency Readiness (LaSER-MED)
- Agent Based Models
  - Why ABM ?
  - System Design and Implementation
- Outbreak-Model
  - Manhattan topology and transportation
  - Agents parameters, behavior and route computation
  - Health time-course
  - Experimental results
    - Sarin attack scenario
Center for Catastrophe Preparedness & Response (CCPR)

- Founded in response to the events of September 11, 2001
- A university-wide, cross-disciplinary center to improve preparedness and response capabilities to terrorist threats and catastrophic events.
- Addresses diverse issues:
  - Medical capacity during crises
  - Legal issues relating to security
  - First-responder trauma response
  - State-of-the-art training for first-responders
- A partnership with the Department of Homeland Security and its Office for Domestic Preparedness
- 113 University Place, 9th Floor
  New York, NY. 10003
  212.998.2183
  center.for.cpr@nyu.edu
Large Scale Emergency Readiness (LaSER-MED)

- Composed of five related sub-projects:
  - ... 
  - Computational Modeling of Large Magnitude Casualty
  - ...
  - **Goal**: Develop and evaluate *scalable, robust* mathematical models
    - *Large* magnitude (1 million) casualty scenarios
    - Three different agents: chemical, biological and radiological
    - Range of input variables of hosts (population) and environments, e.g., Manhattan.
Agent Based Models

- Simulations based on the *global* consequences of *local* interactions of members of a population
- *Environment* or framework
- Some number of individuals defined in terms of their *behaviors* (procedural rules) and characteristic *parameters*.

- **ABM** - a powerful simulation modeling technique that has seen an increasing number of applications in the last few years
- **Success in real-world problems** & phenomena belonging to
  - social science
  - economy
  - robotics
What is an Agent?

- **Autonomous**: capable of acting independently, exhibiting control over their internal state
- **Capable of autonomous action in some environment in order to meet its design objectives**
Why ABM?

- ABM provides a *natural* way to describe a system:
  - Suited to systems in which the overall dynamics can be described as the result of the behavior of populations of autonomous components
  - Fixed set of rules based on local information rather than on central control

- ABM is able to capture *emergent phenomena*:
  - *Whole system is more than the sum of its parts*

- ABM is *flexible* and *intuitive*:
  - Very simple to add or remove agents to the model
  - Easier transfer of domain specific knowledge into a model
Right level of description in order to produce a system that serves its purpose correctly
- Remains an art more than a science

Avoid modeler bias and unnecessary complexity.

Repast 3.1
- Java-Based software toolkit

ProActive
- software for parallel and distributed concurrent computing on clusters of workstations
Topological & Transportation Constraints

- The Geographic Information Systems (GIS) street map:
  - New York City Department of Information Technology and Telecommunications (DoITT).

- Converted into a non-planar graph with 104,730 nodes (including 167 subway stops) under the following assumptions:
  - Each node represents a location
  - Each edge represents a straight-line segment of any walkway or a subway
  - People & Vehicles: constrained to move only along the edges
  - Edges are assumed to be bidirectional
LRTA* with Ignore-List for Route Finding

- The Learning Real-Time (LRTA) algorithm, proposed by Korf in 1990, interleaves planning and execution in an on-line decision-making setting.
- When the *heuristic function is admissible*, the LRTA* algorithm enjoys the property that the agent *never fails to reach the goal in a single run*.
  - **Default:**
    - If all neighbors of the current node $i$ are in the ignore list, pick one randomly.
  - **Else:**
    - **Look-Ahead:** Calculate $f(j) = k(i,j) + h(j)$ for each neighbor $j$ of the current node $i$ that is not in the ignore-list. Here, $h(j)$ is the agent's current estimate of the minimal time-cost required to reach the goal node from $j$, and $k(i,j)$ is the link time-cost from $i$ to $j$.
    - **Update:** Update the estimate of node $i$ as follows:
      $$ h(i) = \max\left\{h(i), \min_{j \in \text{Near}(i)} f(j)\right\} $$
    - **Action Selection:** Move towards the neighbor $j$ that has the minimum $f(j)$ value.
Person Agent

Selfish and bounded rational, with stochastic personality traits emulating panic behavior:

- **State:**
  - headed to original destination or to a hospital;

- **Facts:**
  - current health level ($H_l$),
  - current “amount” of medication / treatment,
  - access to a communication device,
  - probability of communication;

- **Knowledge:**
  - location and current capacities of known hospitals,
  - time of last-update of this information,
  - tables of the LRTA estimates,
  - list of $N$ most recently visited nodes;

- **Personality:**
  - degree of worry ($W_l$),
  - level of obedience ($O_l$),
  - level of distress ($D = W_l 	imes (1 - H_l)$).

```java
if(U(0,1) < Obedience) {
    if (health < unsafe health level)
        Head to a hospital
} else if (U(0,1) < distress level))
    Head to a hospital

if(U(0,1) < distress level) {
    Find nearest hospital
} else {
    Find nearest hospital in available mode
}
```
Hospital Agent

- **Stationary agent**
- Abstraction of any medical facility.

- **State:**
  - available,
  - critical or
  - full;

- **Facts:**
  - resource level,
  - reliability of communication device;

- **Knowledge:**
  - locations and current capacities of known hospitals;

- **Triage Behavior:**
  - health-levels below which a person is considered critical, non-critical or dischargeable.

Treat all admitted patients
for all persons inside the hospital{
  if (health >= dischargeable health level)
    Discharge person
  else if(person is waiting for admission) {
    if(hospital is in available mode)
      Admit and treat the person
    else if(hospital is in critical mode &&
            health < critical health level)
      Admit and treat the person
  }
  if (person is waiting &&
      health < critical health level)
    Add to critical list
  if (person is admitted &&
      health > non-critical health level)
    Add to non-critical list
}
Discharge non-critical patients, admit critically ill
On-Site Responder Agents

- **On-Site Responders:**
  - *Small mobile hospitals.*
  - After receiving notification of the disaster, they move towards the catastrophe site.
  - Knowledge:
    - starting location
    - time of dispatch
    - locations and current capacities of known hospitals
  - The behavior is exactly the same as a hospital in *critical* mode
Disease modeling:

Time-course of Deterioration and Recovery

- Three step probabilistic function such that the healthier the person is, the more likely that health will improve rather than deteriorate:
  - if \((U(0,1) < \text{health})\)
    \[
    \text{health} = \text{health} + U(0, \text{treatment} + \text{maximum untreated recovery});
    \]
  - Else
    \[
    \text{worsening} = (\text{health} > \text{dangerous health level})?\]
    \[
    \text{maximum worsening: } ((\text{health} > \text{critical health level})?\]
    \[
    \text{maximum dangerous worsening: } \]
    \[
    \text{maximum critical worsening})
    \]
    \[
    \text{health} = \text{health} - U(0,(1 - \text{treatment})*\text{worsening});
    \]

<table>
<thead>
<tr>
<th>Exposure level</th>
<th>Health range</th>
<th>People Exposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>High (lethal injuries)</td>
<td>(0.0, 0.2)</td>
<td>5%</td>
</tr>
<tr>
<td>Intermediate (severe injuries)</td>
<td>(0.2, 0.5)</td>
<td>25%</td>
</tr>
<tr>
<td>Low (light injuries)</td>
<td>(0.5, 0.8)</td>
<td>35%</td>
</tr>
<tr>
<td>No symptoms</td>
<td>(0.8, 1.0)</td>
<td>35%</td>
</tr>
</tbody>
</table>
## Model Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of iterations</td>
<td>[0, ∞)</td>
</tr>
<tr>
<td>Number of agents (person, hospitals, On-site responders and ambulances)</td>
<td>[0, ∞)</td>
</tr>
<tr>
<td>Alert time (in minutes for On-site responders and ambulances)</td>
<td>[0, ∞)</td>
</tr>
<tr>
<td>dangerous health level</td>
<td>[0, 1]</td>
</tr>
<tr>
<td>Critical health Level</td>
<td>[0, 1]</td>
</tr>
<tr>
<td>Non-critical health level</td>
<td>[0, 1]</td>
</tr>
<tr>
<td>Unsafe health level</td>
<td>[0, 1]</td>
</tr>
<tr>
<td>On-site responder dischargeable health level</td>
<td>[0, 1]</td>
</tr>
<tr>
<td>Hospital dischargeable health level</td>
<td>[0, 1]</td>
</tr>
<tr>
<td>Probability to have a communication device (phone, radio, ecc)</td>
<td>[0, 1]</td>
</tr>
<tr>
<td>Phone update probability</td>
<td>[0, 1]</td>
</tr>
<tr>
<td>Hospital low resource level (percentage of)</td>
<td>[0, 100]</td>
</tr>
<tr>
<td>Hospital very low resource level (percentage of)</td>
<td>[0, 100]</td>
</tr>
<tr>
<td>Hospital low beds level (percentage of)</td>
<td>[0, 100]</td>
</tr>
<tr>
<td>Percentage of lethal, severe, light injuries</td>
<td>[0, 100]</td>
</tr>
</tbody>
</table>

- All experimental results involve 1,000 people, 22 hospitals, and 5 on-site responder teams.
- Simulation stopped after 3000 ticks = 3 days.
- Every point that is plotted is the average of 10 independent runs.
Unsafe Health Level

- When the unsafe health level is too low (< 0.2), people have been instructed to wait so much that their condition turns fatal.

- When unsafe health level is high (> 0.2), people who do not require much emergency treatment end up consuming a share of the available resources, which would have been better spent on the sicker people already at the hospital or on persons who are still on their way to the hospital.

- Number of deaths due to crowding is dramatically mitigated if there are on-site treatment units.
Dischargeable Health Level

- When the dischargeable health level is too low, the person dies after being discharged prematurely.
- When it is too high, the person is given more medical attention than necessary and effectively decreases the chances of survival of sicker persons.
Worry vs Obedience

- Disobedient worrying persons will head to the nearest hospital too early, thus crowding the most critical resource.
- Obedient people who are not worried choose to go to a hospital only when they are really sick, and also distribute themselves between the different hospitals.
Beyond the 10 responders that seem to be required, the effect on the improvement in the number of survivors is less evident.

Bound on the number of dying people that can actually be saved causes this surface attending.

Plot also shows a near-linear dependence of the system on the alert time.
Some drawbacks...

- The modeler wants to produce a simulated system which behavior satisfies *different and often conflicting criteria*:
  - number of fatalities
  - waiting time
  - average health of population
  - average time taken to die
- *Extensive number of parameters* that must be "tuned" or "calibrated"
- *Non-linear interactions* between parameters.
- Possible parameters *constraints*:
  - Ex 1: $0 < \text{critical health level} < \text{non-critical health level} < \text{dischargeable health level} < 1$
  - Ex 2: $0 < \text{unsafe health level} < \text{dischargeable health level} < 1$
Parameter space of Agent-Based Models

- Lots of parameters…
  - Some can be extracted from the knowledge of the field (either experimental or theoretical) and can be associated to *fixed values*
  - Other parameters have to be kept *variable* for different reasons:
    - Knowledge of the field is generally not exhaustive (which is the reason why we build a model and simulate it)
    - Knowledge may not be directly compatible with the model – need to sample parameter space and evaluate
  - Parameters belong to different domains:
    - *real*, *integer*, *boolean*, etc.
Thank You.