Principled Approaches for Managing Emergency Response in Smart and Connected communities

Geoffrey Pettet[1], Ayan Mukhopadhay[2], Eugene Voroybeychik[3], Abhishek Dubey[1]

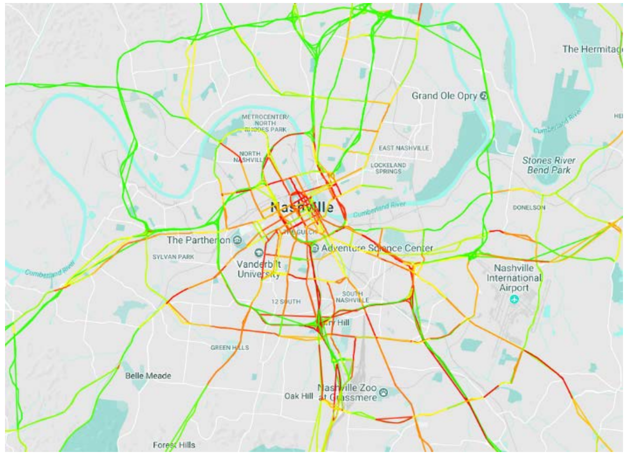
[1]Vanderbilt University

[2]Stanford University

[3]Washongton University in St Louis

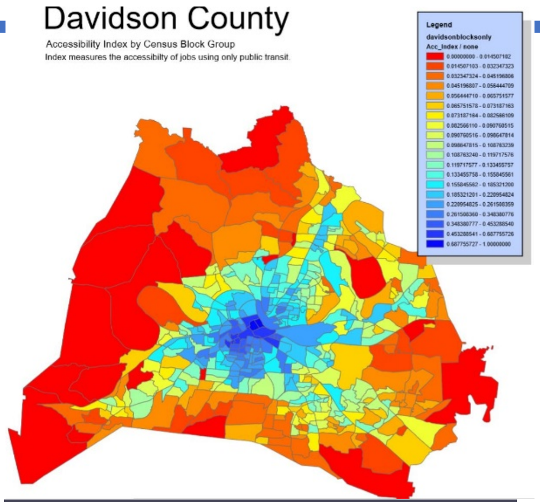
Sponsored by National Science Foundation

Our communities are being stressed



Congestion

Air Quality



Gentrification

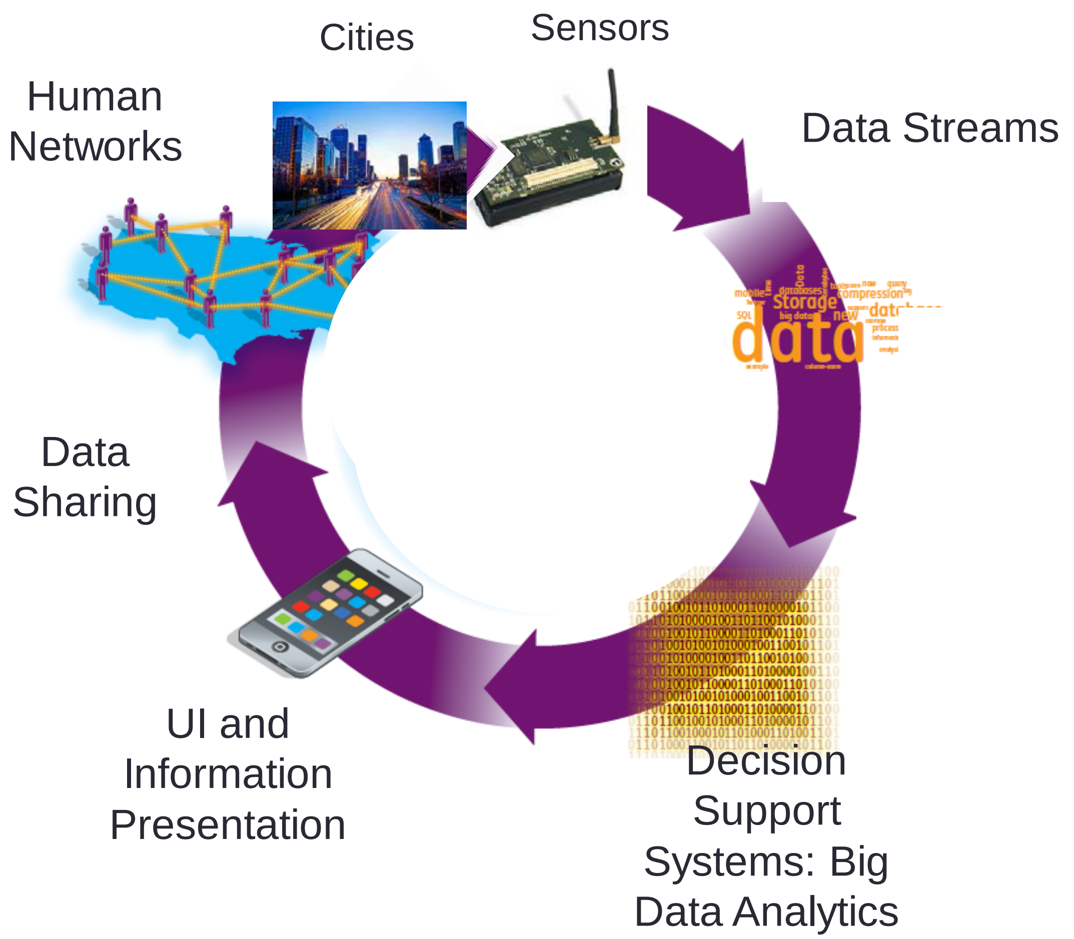
Energy & Water

Workforce

Safety

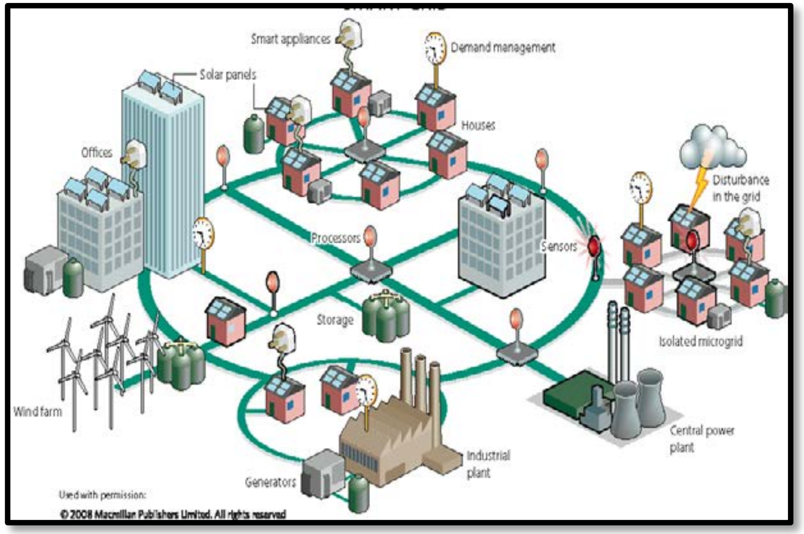
Cyber-Physical System Science as a Solution

CPS = Computation Software + Physical system + Networks + Human + Closed-loop

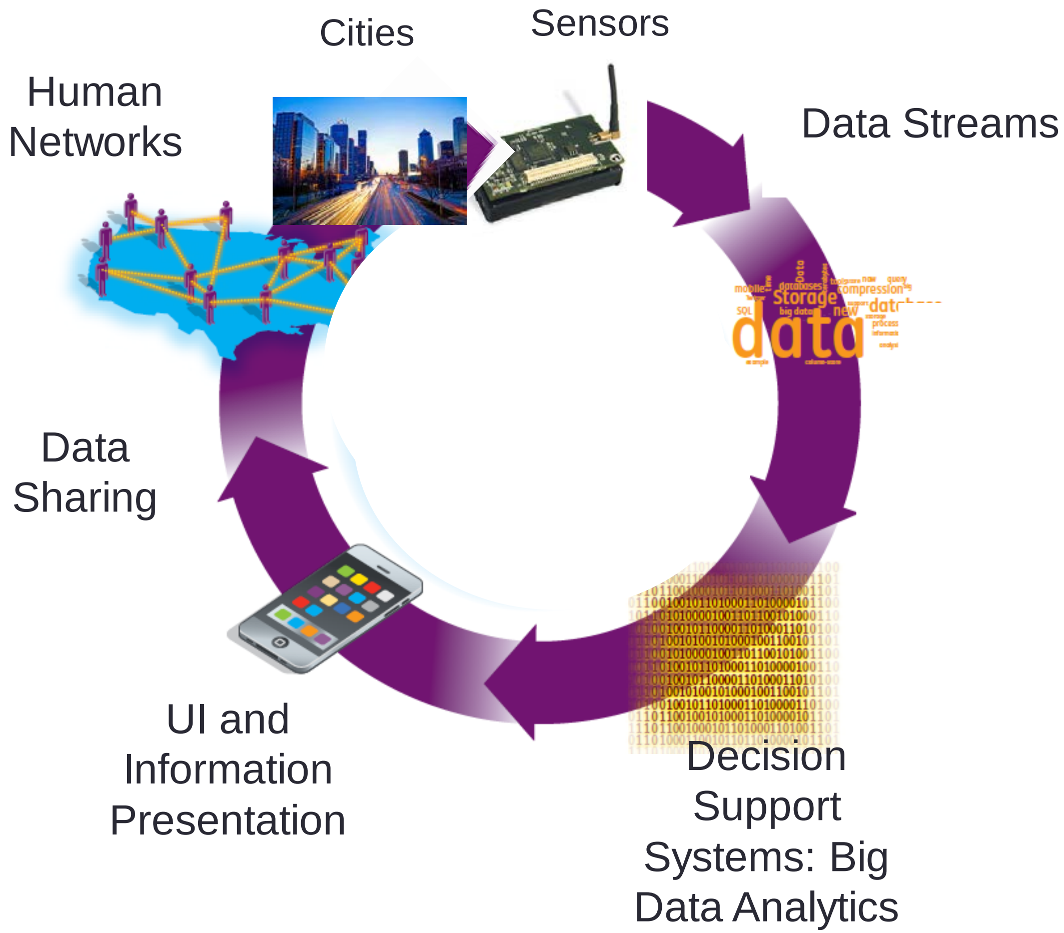


Cyber-Physical System Science as a Solution

CPS = Computation Software + Physical system + Networks + Human + Closed-loop

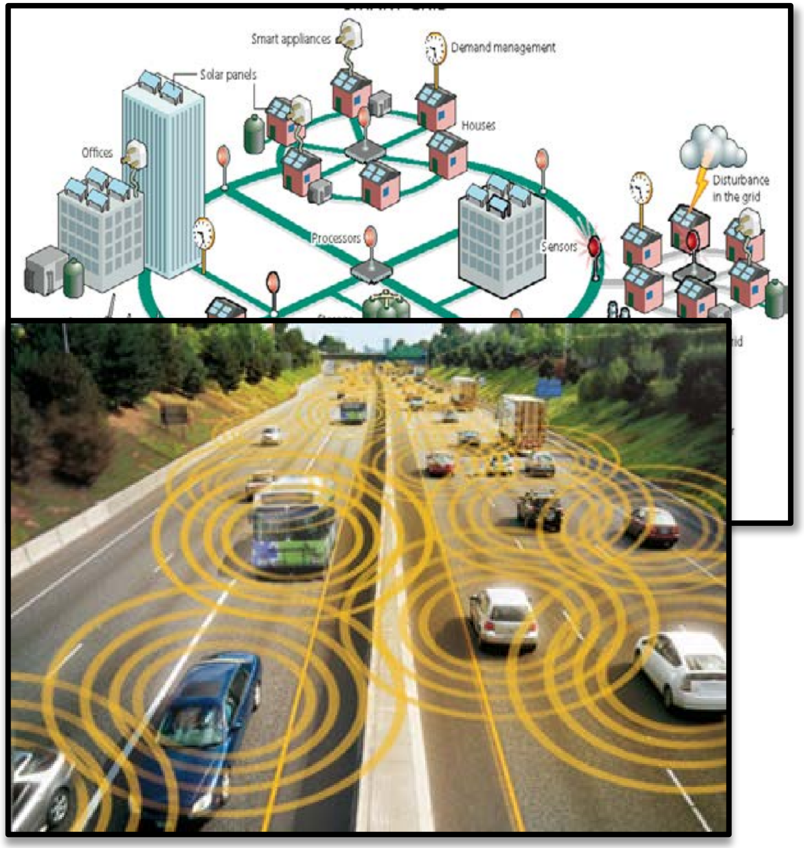


Smart Grid



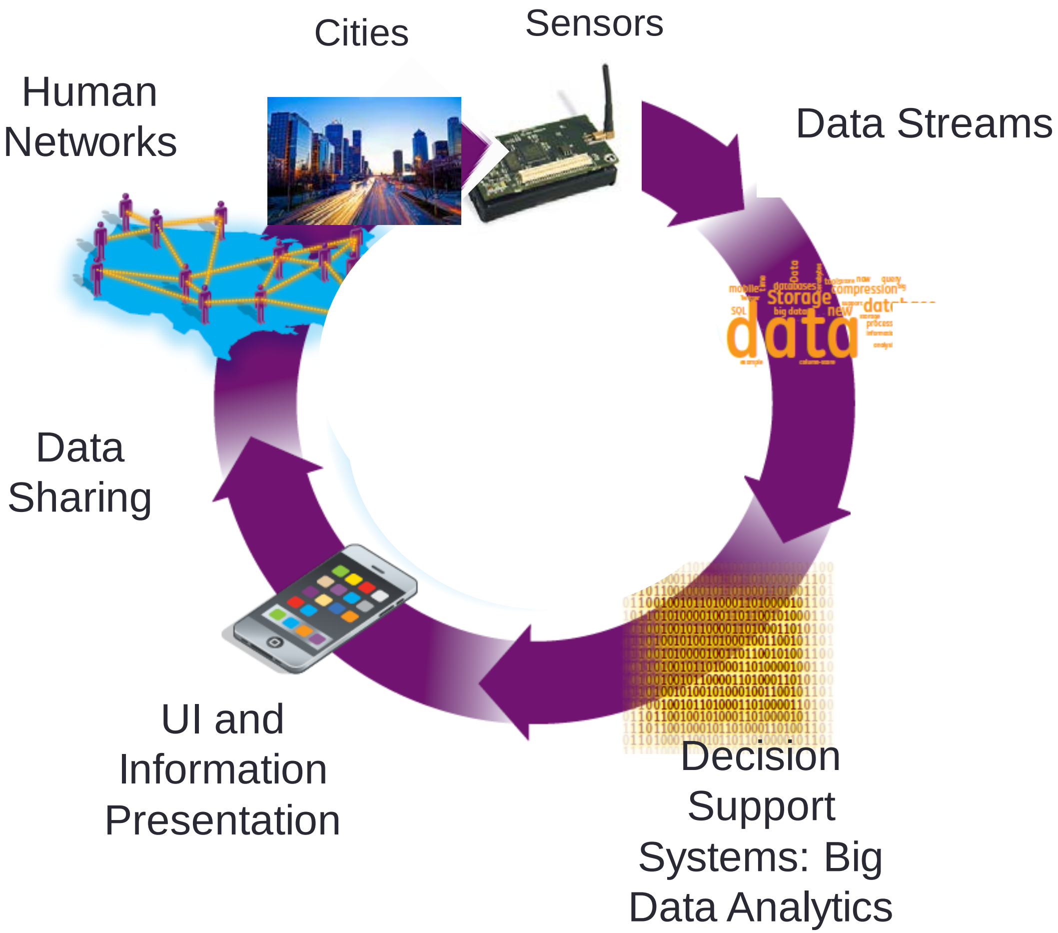
Cyber-Physical System Science as a Solution

<u>CPS = Computation Software + Physical system + Networks + Human + Closed-loop</u>



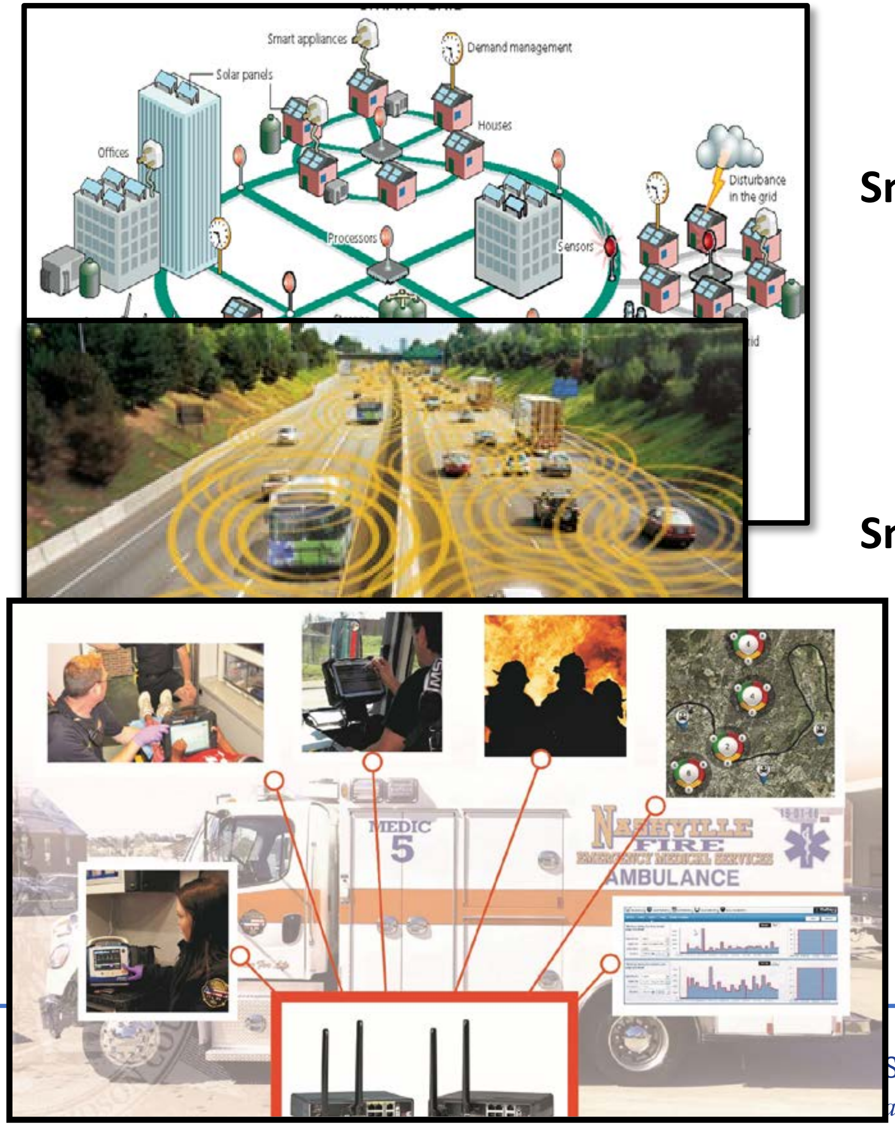
Smart Grid

Smart Transportation



Cyber-Physical System Science as a Solution

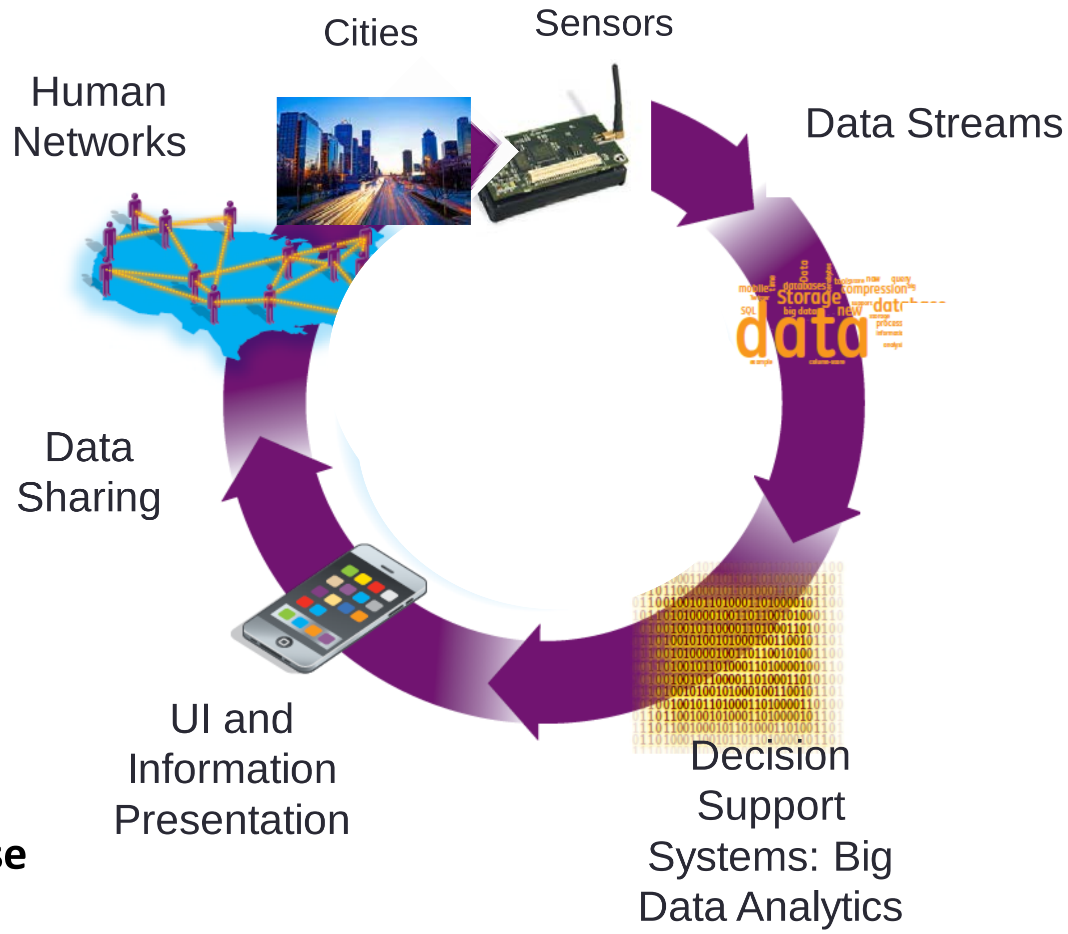
CPS = Computation Software + Physical system + Networks + Human + Closed-loop



Smart Grid

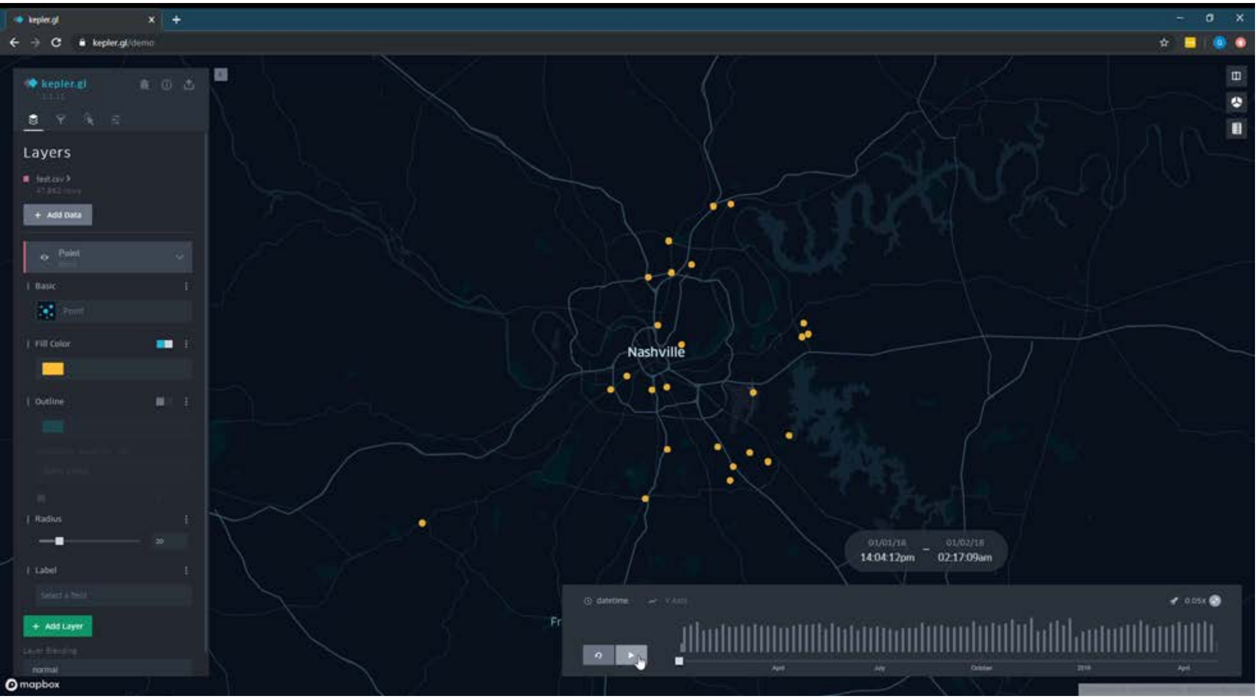
Smart Transportation

Smart Emergency Response



The Emergency Response Problem

The emergency response problem



This is all traffic incidents occurring in Davidson County In January 2018, with a sliding window of ~12 hours worth of incidents shown at once.

The problem: Respond Efficiently to all incidents spread over a large geographical area with limited resources.

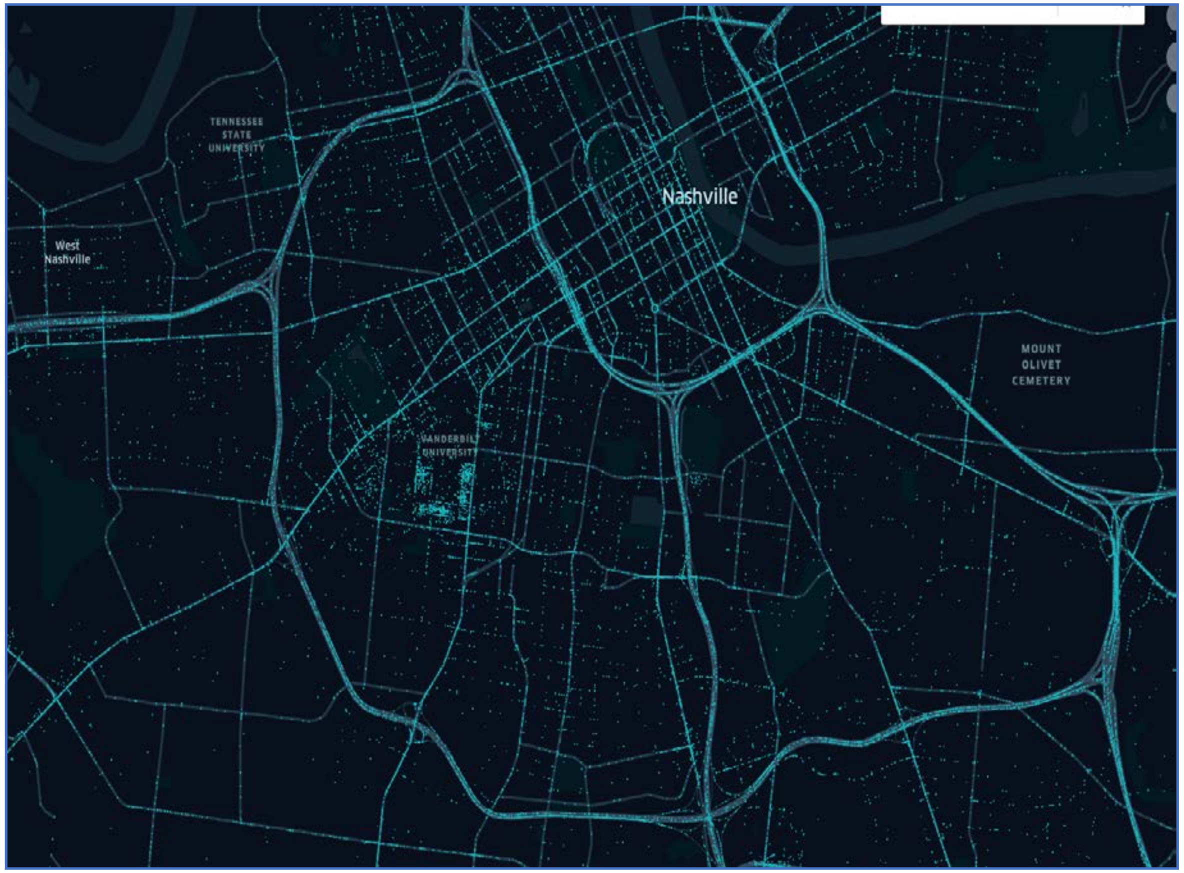
The emergency response problem

Receive 911 call

Dispatch Nearest Responder by Euclidean Distance

Service Incident (Possibly go to Hospital)

Responder Returns to Assigned Station – Now Available for Dispatch Again



Accidents over 5 year period

Current State of the art is **reactive**. Respond when the call arrives.

The Proactive Approach to Emergency Response Problem

Develop Online Models to Estimate Demand

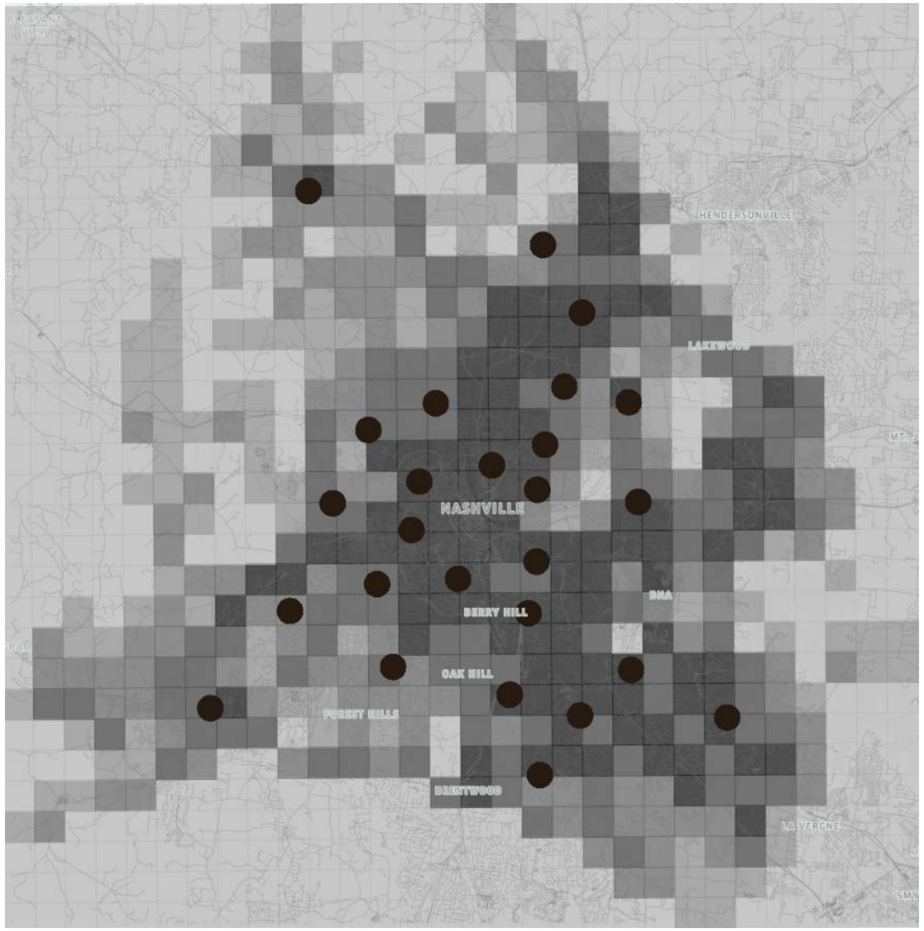
Anticipatory Stationing of Resources

Optimal Dispatch

Active Learning and Improvement Mechanisms

Anticipating Demand

- **GOAL** : Learn a probability distribution *f(t|w)*
- <u>Given</u>: a finite set of grids over a geographical region, and a dataset *D* of time-stamped incidents.
- <u>D</u> : $\{\{x_1,w_1\}, \{x_2,w_2\},\ldots, \{x_n,w_n\}\}$
  - where <u>x is time of occurrence, and w_i is a set of features associated with the i^{th} incident</u>
  - **Features**: Past rate of incidents, weather condition in the area, speed limit etc.



Nashville depots overlaid on incident density map

Online Incident Likelihood Estimation

- We use **survival analysis** - a class of methods to find *inter-arrival* times.
- **Inter-arrival time**: $t_i = x_i - x_{i-1}$

$$\log(t_i) = \sum_j \beta_j w_j + \epsilon$$

Online Incident Likelihood Estimation

- We use **survival analysis** - a class of methods to find *inter-arrival* times.
- **Inter-arrival time**: $t_i = x_i - x_{i-1}$
- We use **Maximum Likelihood Estimation** to estimate parameters.

$$\log(t_i) = \sum_j \beta_j w_j + \epsilon$$

$$L = \prod_i h(\log(t_i) - \bar{\beta} W)$$

Online Incident Likelihood Estimation

- We use **survival analysis** - a class of methods to find *inter-arrival* times.
- **Inter-arrival time**: $t_i = x_i - x_{i-1}$
- We use **Maximum Likelihood Estimation** to estimate parameters.
- However, accidents often cascade and the survival model has to be updated online.

$$\log(t_i) = \sum_j \beta_j w_j + \epsilon$$

$$L = \prod_i h(\log(t_i) - \bar{\beta}W)$$

Probabilistic Model for Incident Prediction

Online Incident Likelihood Estimation

- We use **survival analysis** - a class of methods to find *inter-arrival* times.
- **Inter-arrival time**: $t_i = x_i - x_{i-1}$
- We use **Maximum Likelihood Estimation** to estimate parameters.
- However, accidents often cascade and the survival model has to be updated online.
  - Let *D'* represent a stream of new incidents.
  - Assume that β^p is already known.
  - Our goal is to update β^p to β^{p+1} without re-learning the entire model.
  - We take gradient steps for each parameter based on *D'*

$$\log(t_i) = \sum_j \beta_j w_j + \epsilon$$

$$L = \prod_i h(\log(t_i) - \bar{\beta} W)$$

Probabilistic Model for Incident Prediction

$$\beta^{p+1} = \beta^p + \alpha \nabla L(\beta^p, D^{'})$$

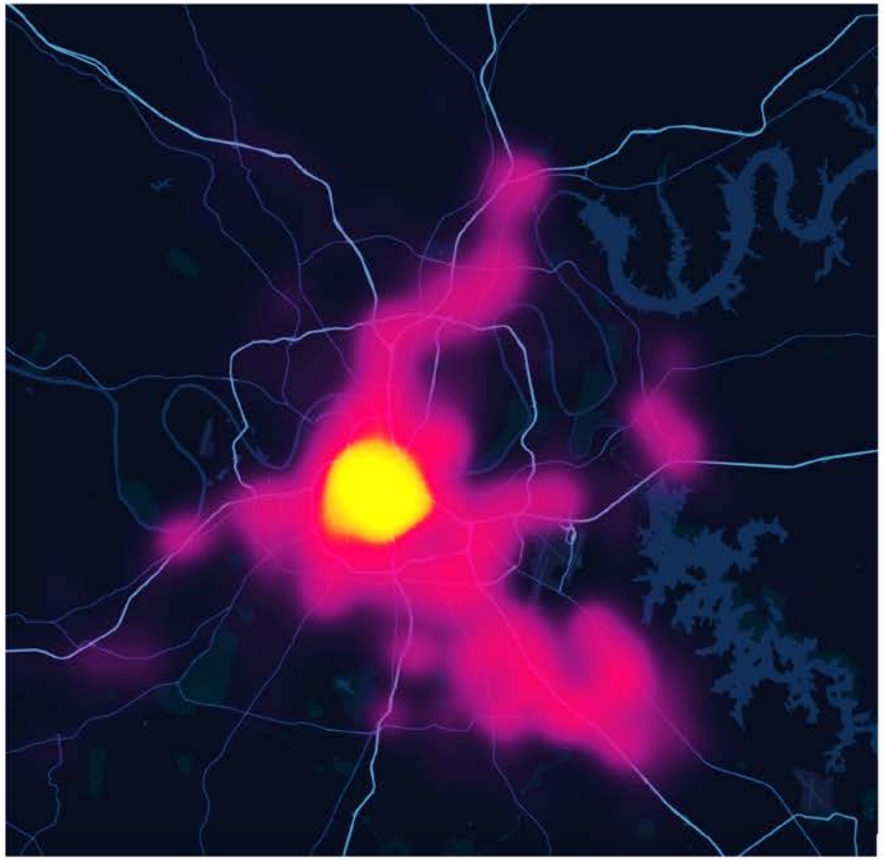
Online Update of Coefficients

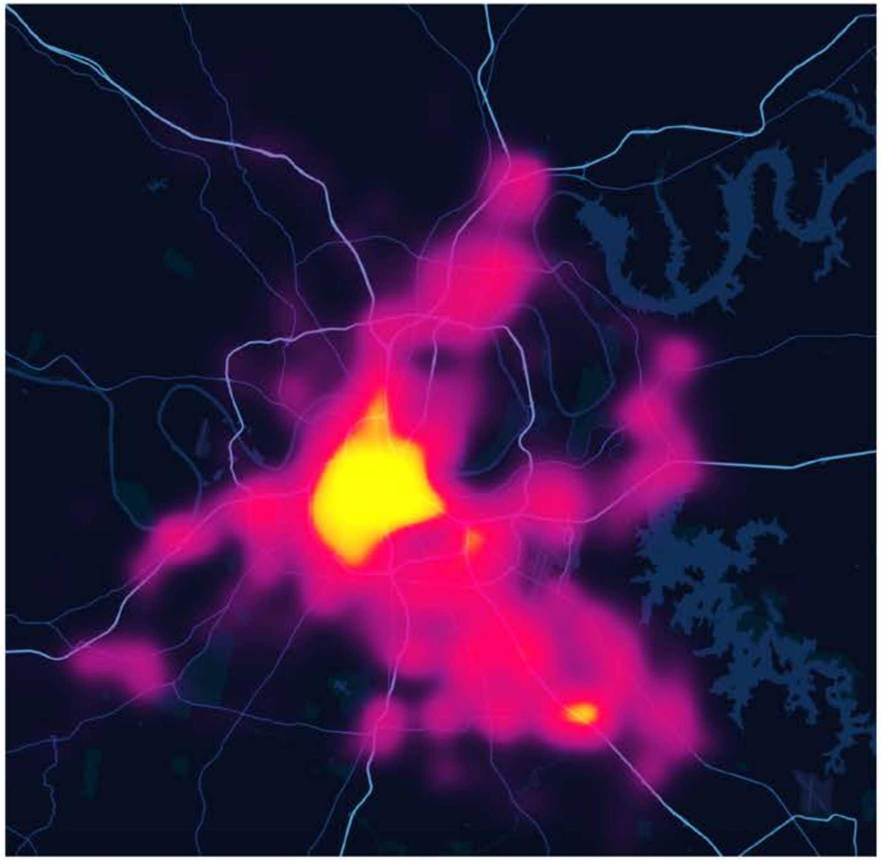
$$\frac{\partial L}{\partial \beta_j} = \sum_{i=1}^{k} -w_{ij} + w_{ij}\{e^{(\log \tau_i - \beta^* w_i)}\}$$

Gradient Calculation

Prediction Example

- Comparison of (1) incidents predicted by model (left), and (2) real incident distribution (right) over January 2019

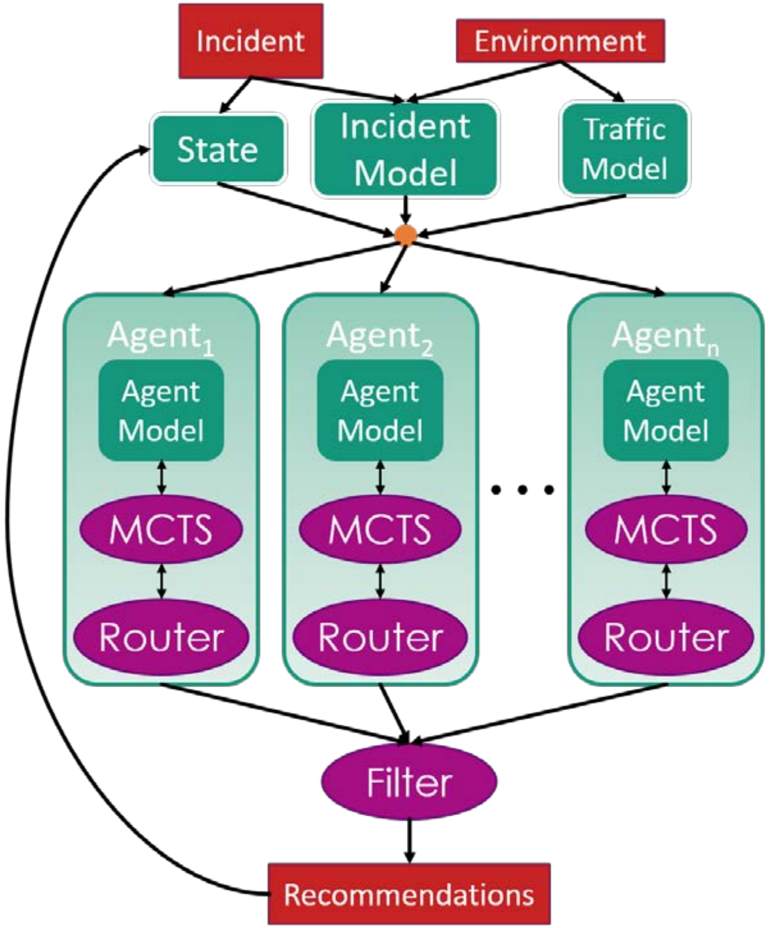




Resource Assignment

- Goal: allocate EMS resources to optimize total response times to incidents
- Considerations:
  - Decision must be made quickly at the time of an incident
  - Optimizing over responder distribution and response as a multi-objective optimization problem is typically computationally infeasible.
    - **Example**: let the number of responders r=20, and the number of possible depot locations be d=30. Possible actions for dispatching is the number of responders -> **20**
    - Possible actions for allocation is P(d, r) = 30!/10!

Partially Decentralized Decision Process

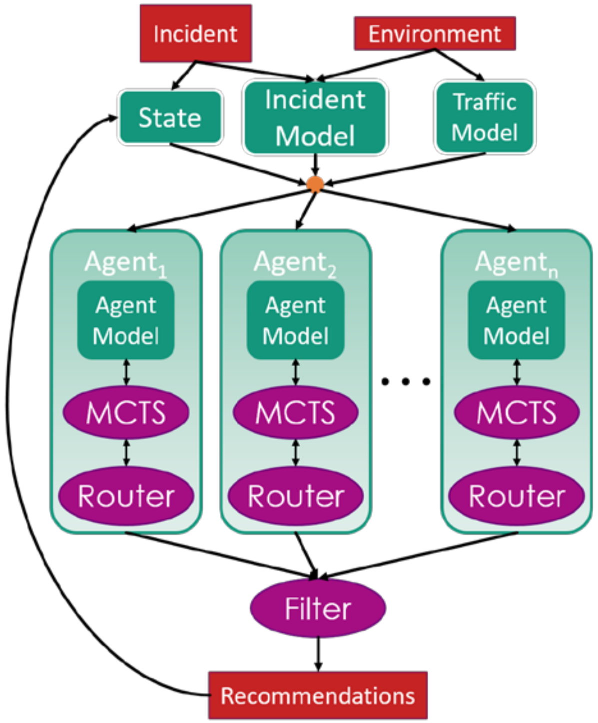


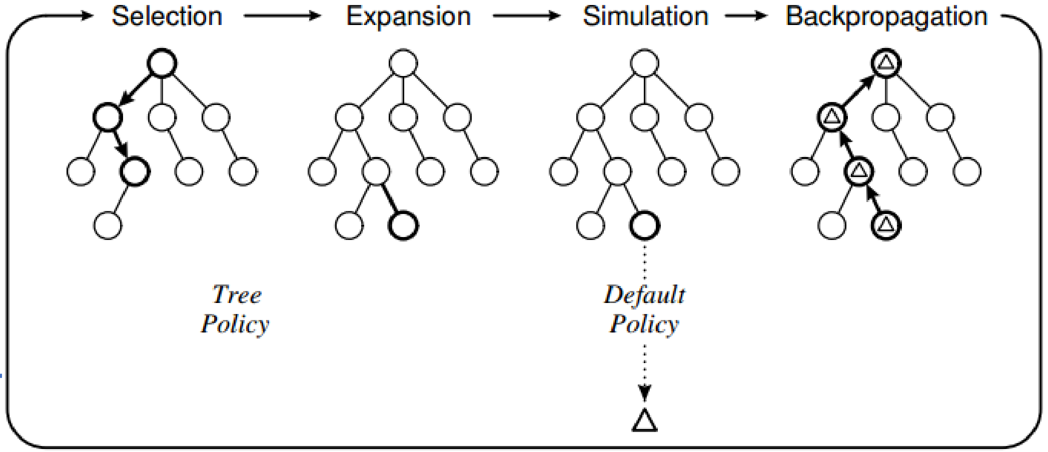
Rather than building a monolithic, large search tree exploring all possible system states, each agent builds an individual tree focusing on the subset of actions relevant to them – i.e. their rebalancing action

Resource Assignment

- Each agent ***a*** builds own general Monte Carlo Search Tree with a few extensions
  - Expansion:
    - Action space includes all relevant actions for ***a*** (***a*** responding to an incident, moving to a new station, etc.)
- Other agents' actions are assumed to follow some static policy to reduce action space. Examples include:
    - Naïve - agents are stationary
    - Greedily follow heuristic (M/M/c queue model response time, etc.)

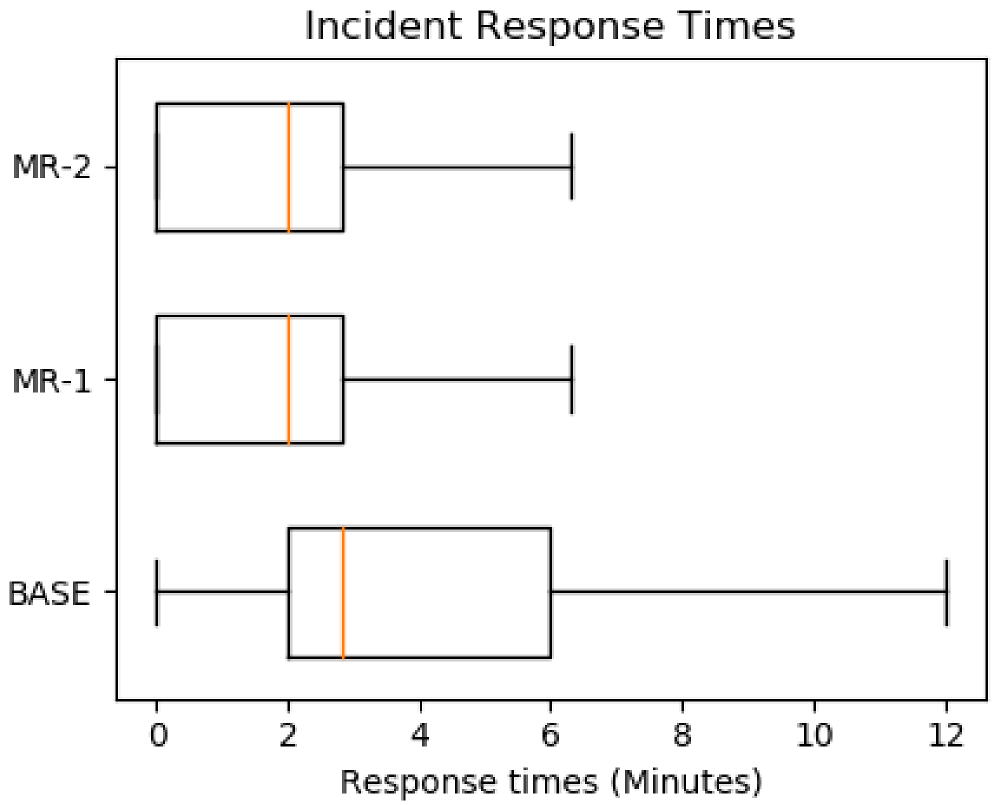
Partially Decentralized Decision Process

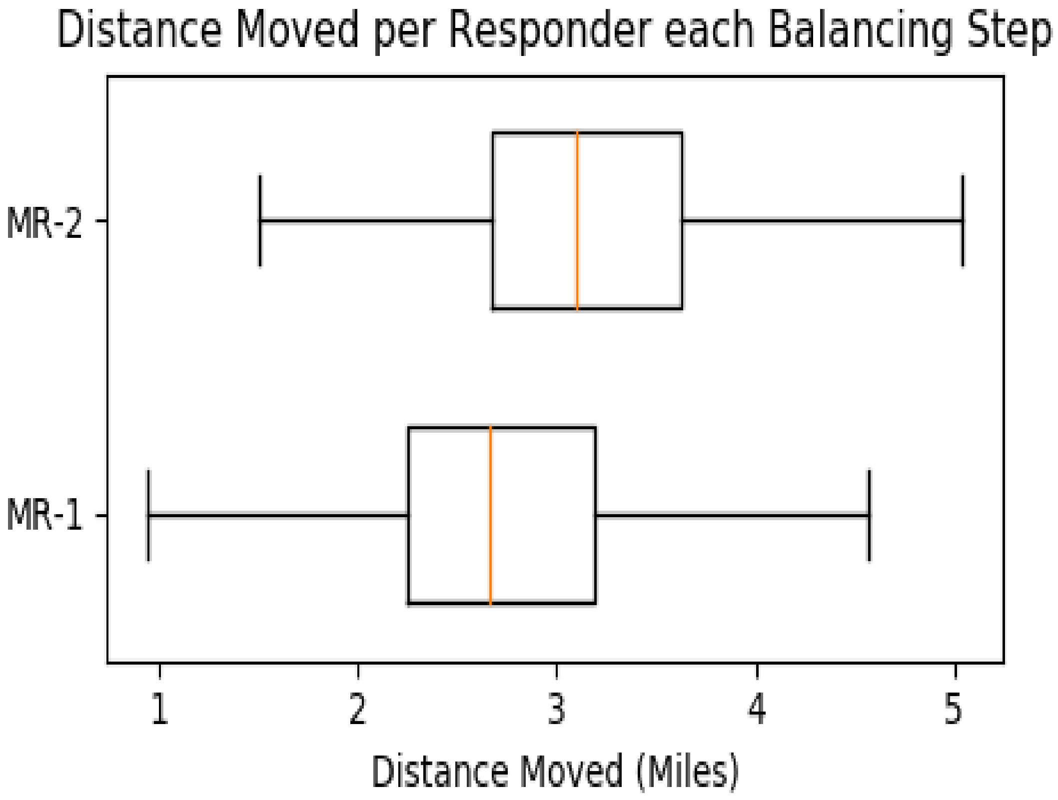




Results

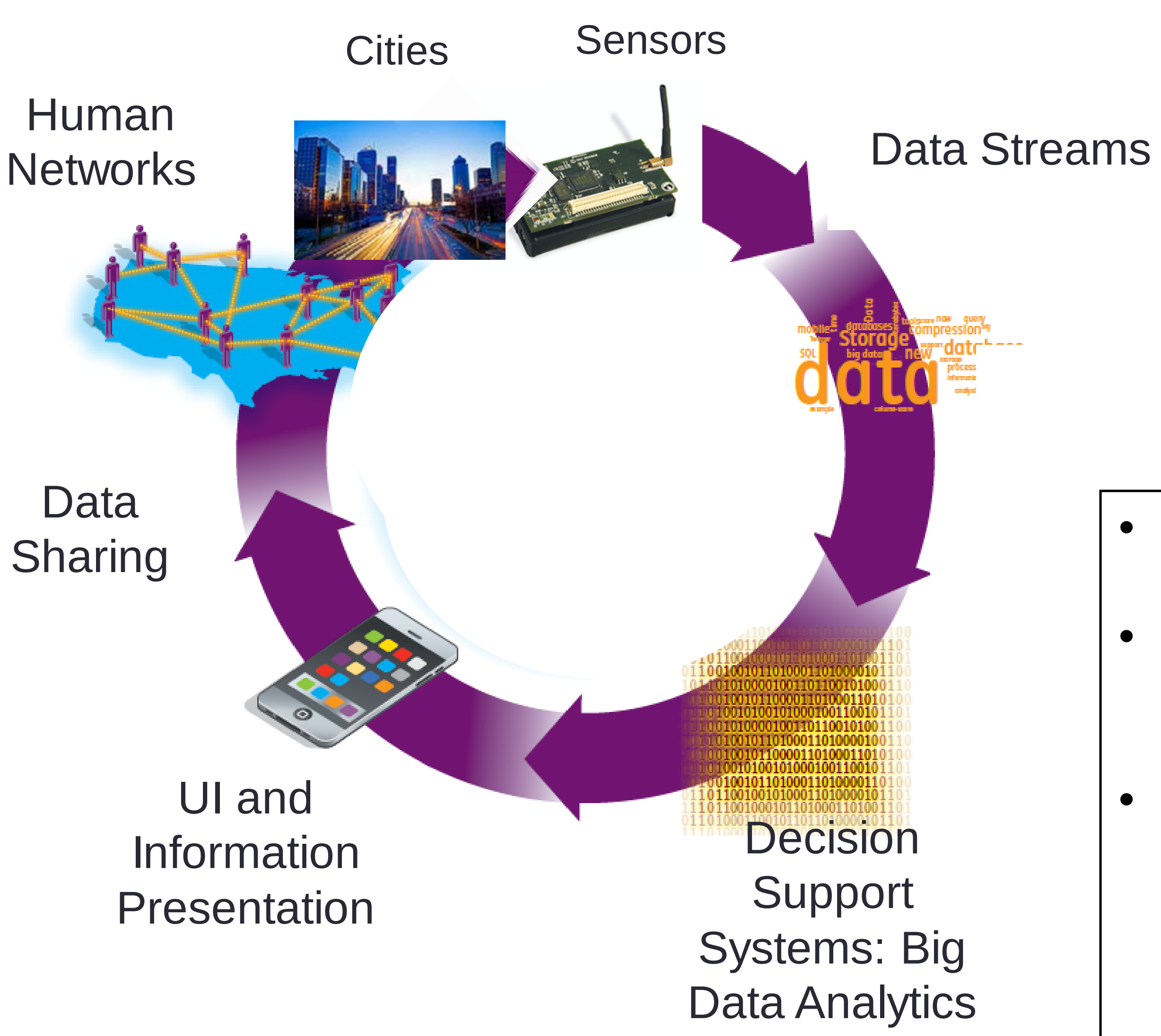
Key

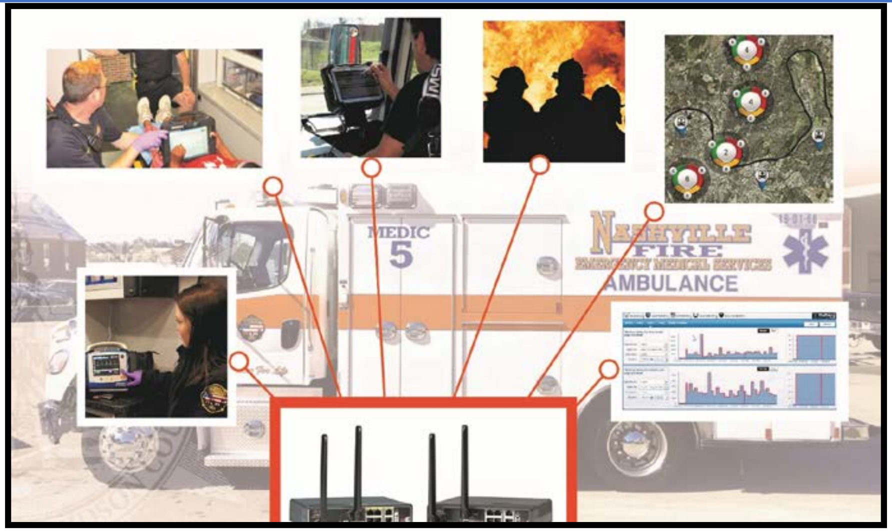




MR-1 and MR-2 are two separate Hyper-parameter strategies. MR-1 is discounts the distance moved. The results show almost 3 minutes of response time saving.

Conclusion





- We discussed mechanisms to anticipate demand and then rebalance resources
- This problem is not unique to emergency response and applies to other transportation systems **such as public transit and micro-transit.**
- The challenges still exist.
  - Is the data we are learning from correct?
  - How should we handle concept drift?
  - Do the cities have enough computation power to handle these big data driven processes?