Creating, Updating and Validating Simulations in a Dynamic, Data-Driven Application System

Tim Schoenharl

Notre Dame CSE

June 5, 2007

Overview

DDDAS: Putting Simulations in the Loop

Dynamic, Data-Driven Application Systems are coupled systems of sensors and simulations. The motivation of DDDAS is as follows:

- "Simulation output can be improved when simulations are created and updated with streaming data from sensors" [Darema 2004]
- Better response to time-sensitive events
- More efficient use of simulation time, less wasted results

Challenges to the DDDAS Approach

The proceedings of the DDDAS grant recipient's workshop lists several challenges to the approach that must be overcome: [1]

- Simulations must be designed to accept data at execution time (Creating)
- Simulations must be able to accept streaming data (Updating)
- Online validation is an open problem [Davis 1998] (Validating)
- Applications must be able to select from competing models at runtime

[1] For further reading, see the Online Proceedings at `http://www.dddas.org/NSFworkshop2006.html`

Research Questions And Approach

- What design criteria do we use to develop online simulations?
  - Balance modularity with runtime performance. Demonstrate through application of Design Patterns and Pattern Oriented Modeling

Research Questions And Approach

- What design criteria do we use to develop online simulations?
  - Balance modularity with runtime performance. Demonstrate through application of Design Patterns and Pattern Oriented Modeling
- How do we create and update simulations with streaming data?
  - Aggregate data at tower level, apply aggregate data to simulation with random variation

Research Questions And Approach

- What design criteria do we use to develop online simulations?
  - Balance modularity with runtime performance. Demonstrate through application of Design Patterns and Pattern Oriented Modeling
- How do we create and update simulations with streaming data?
  - Aggregate data at tower level, apply aggregate data to simulation with random variation
- How do we validate simulations against streaming data?
  - Treat validation as model selection process. Create ensemble of models, rank simulations based on distance measure.

Research Overview

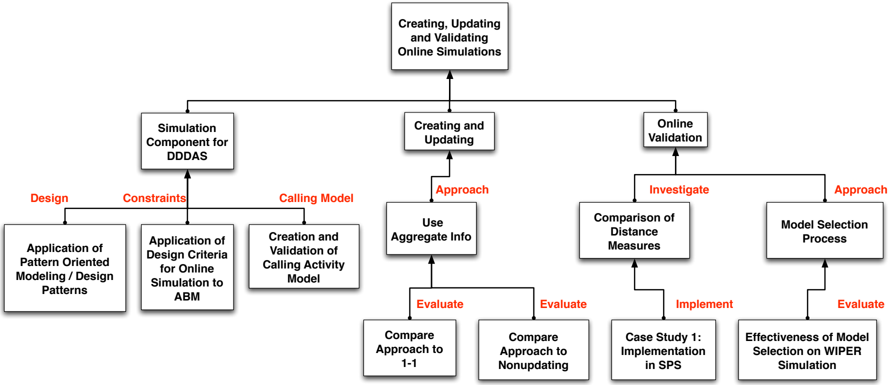


Figure: Contributions of this research.

Research Context: The WIPER System

- The WIPER System is a project to build a DDDAS that receives dynamic, streaming data from a cell phone provider, detects anomalies in calling activity, runs ensembles of simulations to explore the anomaly and provides results to end users through a web-based console.
- WIPER is an NSF-funded research project, CISE-CNS Award #0540348

The WIPER Scenario

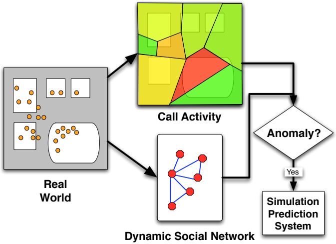


Figure: The WIPER Scenario

WIPER Scenario

- Detect Anomalies from streaming data
- Run simulations to understand crisis events
- Output results to web console

Overview of the WIPER System

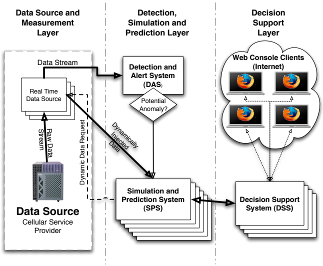


Figure: The WIPER system

The WIPER System Components

- Real Time Data Source
- Detection and Alert System
- Simulation Prediction System
- Decision Support System

Data Visualization

Figure: Cell Phone activity aggregated at tower level and overlaid on a satellite image. Satellite imagery courtesy Google Earth.

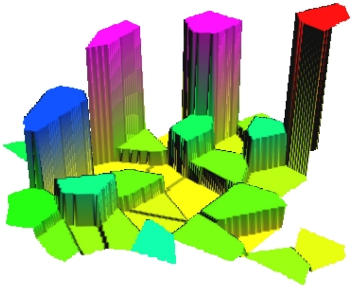


Figure: Call activity in 3D. Calling activity represented by cell height.

WIPER Simulation: Goals

The WIPER simulation is designed according to the following goals:

- Simulations must implement multiple crisis scenarios, as the validation framework relies upon selecting from among multiple models
- Simulations should generate reasonable behavior for calling activity and movement
- Simulations must run in faster than real time

Approach

- Manage tradeoff between modularity and performance
  - Apply Design Patterns (Singleton, Strategy, Flyweight) [Gamma et al 1995]

Approach

- Manage tradeoff between modularity and performance
  - Apply Design Patterns (Singleton, Strategy, Flyweight) [Gamma et al 1995]
- Present multiple, interchangeable models of crisis scenarios
  - Pattern Oriented Modeling [Grimm et al 2005]
  - Extends ideas from Design Patterns to model development
  - Models are composed of patterns, allows for configurable scenarios

Approach

- Manage tradeoff between modularity and performance
  - Apply Design Patterns (Singleton, Strategy, Flyweight) [Gamma et al 1995]
- Present multiple, interchangeable models of crisis scenarios
  - Pattern Oriented Modeling [Grimm et al 2005]
  - Extends ideas from Design Patterns to model development
  - Models are composed of patterns, allows for configurable scenarios
- Models consist of movement and calling activity
  - Calling activity is produced from an empirical distribution
  - Movement activity generated by one of 5 movement types

Design and Implementation of WIPER Simulation

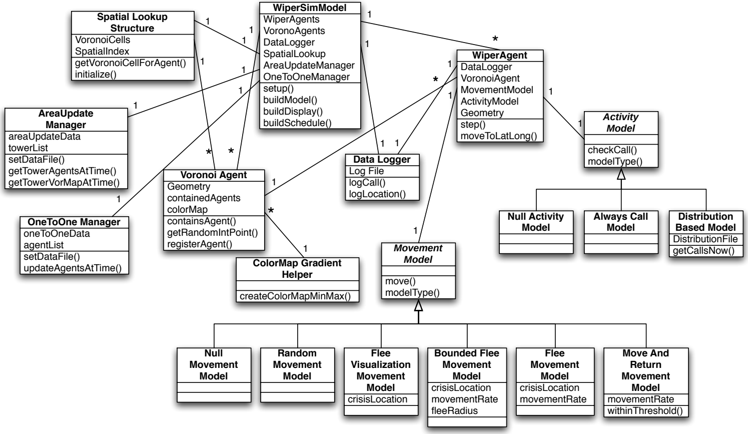


Figure: The WIPER Simulation

Crisis Behavior Taxonomy

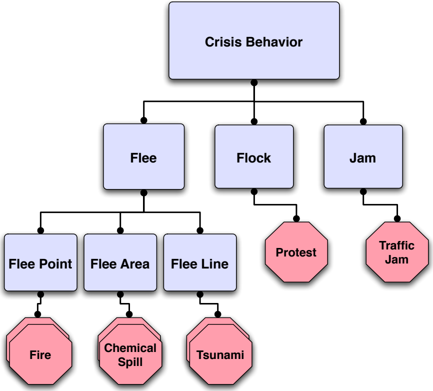


Figure: A Taxonomy of Crisis Scenarios.

Crisis Taxonomy and Pattern Oriented Modeling

- Simulation behavior generated by model composition (Pattern Oriented Modeling)
- Crisis Events - Composition of Movement and Activity
- Simulation can generate normal behavior or crisis based on composition of models

Movement Models

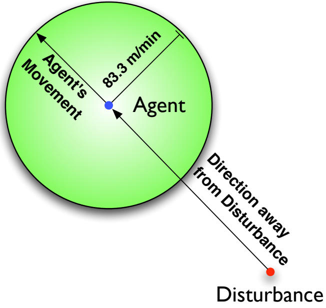


Figure: Basic Flee Action

Movement Model Explanation

- Agent calculates new location based on direction to disturbance
- Random variation in terms of movement speed and direction are introduced

Activity Models

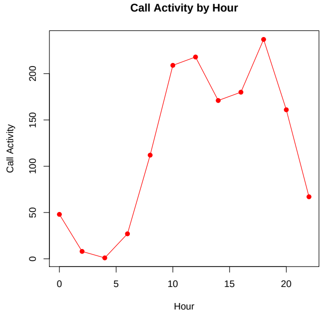


Figure: Empirical Distribution of Call Activity

Activity Model Explanation

- Activity models used to determine when agents should place calls
- Distribution Based Activity - Agent calling activity generated from empirical distribution

Distribution Based Activity Model

Distribution Based Model

- Samples from an empirical distribution
- Empirical data from WIPER cellular CDR data

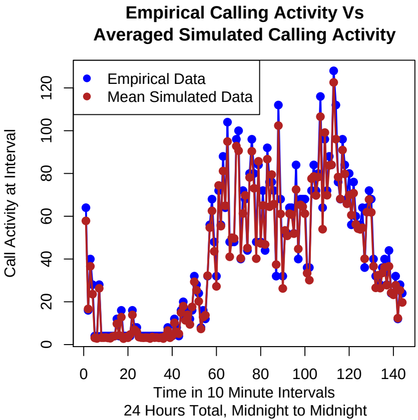
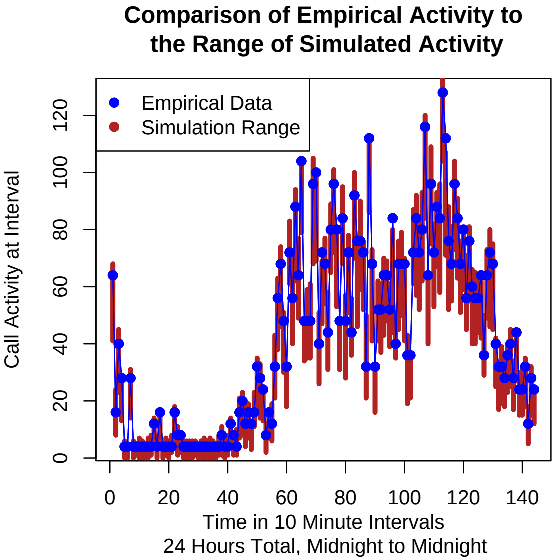


Figure: Empirical data plotted with mean of 100 runs of simulated data

Distribution Based Activity Model



Offline Validation of Activity Model

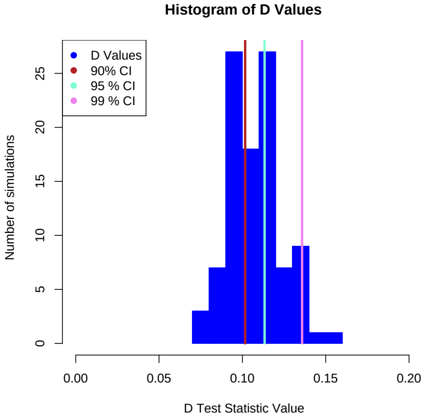


Figure: Histogram of the KS test values for 100 simulations, bars indicate $\alpha = 0.10$, $\alpha = 0.05$ and $\alpha = 0.01$ levels.

Call Activity Data Validation

- This validation work is part of canonical model development process
- Model is validated against empirical data using Kolmogorov-Smirnov test, H_0: Simulations generate calls according to same distribution
- For most simulations, H_0 is not rejected at $\alpha = 0.01$ or $\alpha = 0.05$.

Simulation Scalability

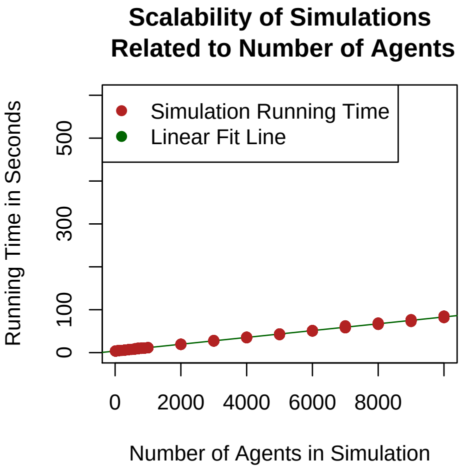


Figure: Simulation scalability with respect to number of agents. Simulations run for 1 hour of simulated time.

Simulation Scalability

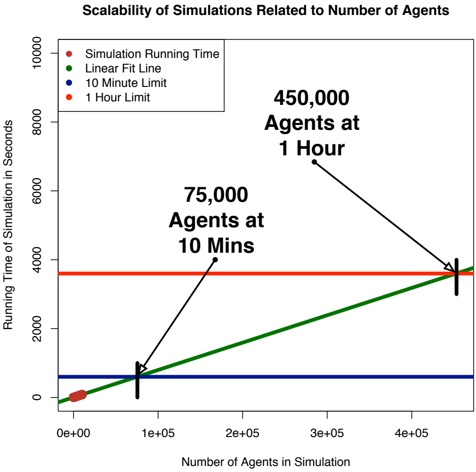


Figure: Simulation scalability with respect to number of agents. Shown on the graphic agent population size that can be simulated for a given amount of time.

Simulation Scalability

Simulation Runtime Characteristics

- Simulation displays linear scaling with respect to number of agents
- According to scalability results, simulation can handle up to 75,000 agents in faster than real time when run at ten minute intervals
- Simulation can handle up to 450,000 agents in faster than real time when run at 1 hour intervals
- Simulation has not been optimized for speed, so performance improvements may exist

Online Validation: Approach

- How do we validate simulations against streaming data?
  - Treat validation as model selection process. Create ensemble of models, rank simulations based on distance measure.

Online Validation: Approach

- How do we validate simulations against streaming data?
  - Treat validation as model selection process. Create ensemble of models, rank simulations based on distance measure.
- How do we rank models?
  - Simulations generate output as a vector of towers, with agents in tower as the value at each position

Online Validation: Approach

- How do we validate simulations against streaming data?
  - Treat validation as model selection process. Create ensemble of models, rank simulations based on distance measure.
- How do we rank models?
  - Simulations generate output as a vector of towers, with agents in tower as the value at each position
  - Rank simulation output by measuring distance from simulation output vector to target

Online Validation

Validation on Movement Models

- Treat the output from simulation, list of towers with numbers of active agents, as a vector
- Use distance measure to choose closest match
- Validation work demonstrates this approach, evaluates measures for their applicability
- This approach is related to Input-Output Validation [Balci 98]

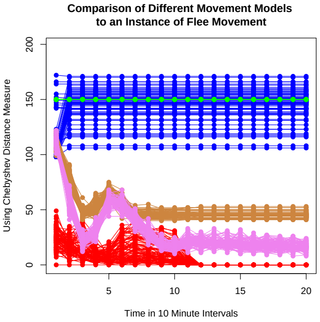


Figure: Validation on agent movement models. Target Flee, metric L_∞

Distance Measures

We consider the following distance measures for their applicability to online model validation. For each measure, $\overline{p} = (p_1, p_2, \cdots p_n), \overline{q} = (q_1, q_2, \cdots q_n)$

- Euclidean $d(\overline{p}, \overline{q}) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$
- Manhattan $d(\overline{p}, \overline{q}) = \sum_{i=1}^{n} |p_i - q_i|$
- Chebyshev

  $$d(\overline{p}, \overline{q}) = \max_i(|p_i - q_i|) = \lim_{k \to \infty} \left(\sum_{i=1}^{n} |p_i - q_i|^k \right)^{1/k}$$
- Binary $d(\overline{p}, \overline{q}) = \frac{\sum_{i=1}^{n} p_i \text{ xor } q_i}{\sum_{i=1}^{n} p_i \text{ or } q_i}$ where $p_i = \left\{ \begin{array}{c} 0 \text{ if } p_i = 0 \\ 1 \text{ otherwise} \end{array} \right\}$
- Canberra $d(\overline{p}, \overline{q}) = \sum_{i=1}^{n} \frac{|p_i - q_i|}{|p_i + q_i|}$

Validating ABM

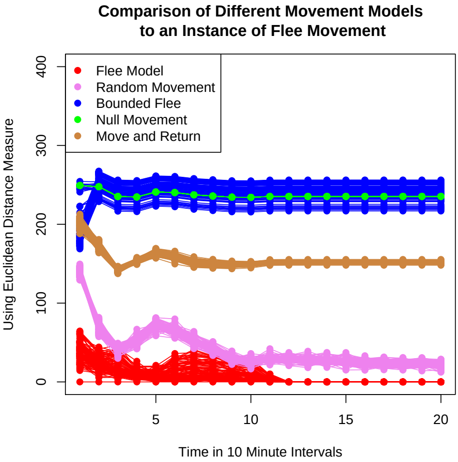


Figure: Plot of the distances of multiple simulation runs of various models on a Flee target using the Euclidean Distance metric, 10 minute intervals.

Validating ABM

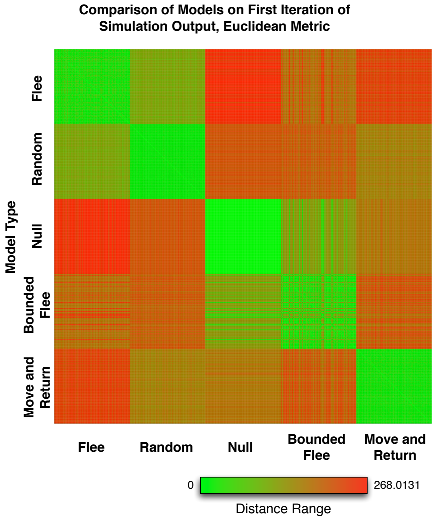


Figure: Plot of distance values between simulation instances.

Ranking Models

- Use distance measures for ranking simulations
- Evaluate the effectiveness: CMC curve

Ranking

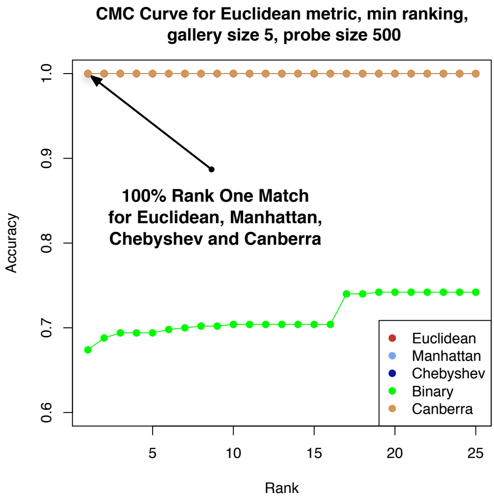


Figure: CMC Curve displaying Rank 1-25 matches for all of the 5 distance metrics.

Ranking

Measure	First True	First False	F-T Dist
Euclidean	1	8.627	7.627
Manhattan	1	7.285	6.285
Chebyshev	1	9.143	8.143
Canberra	1	5.250	4.250

Table: Summary of average distances to first true and false matches, showing value of measures for classification. All of the measures from the L family display good characteristics.

Online Validation Contributions

- Method: Online Validation as Model Selection
- Demonstrate 100% Accuracy on Matching Model Type
- Several measures in L family work well

Creating and Updating ABM: Approach

- How do we create simulations from streaming data?
  - Treat creation as a special case of updating
  - Aggregate agent locations at tower, distribute agents randomly within cell

Creating and Updating ABM: Approach

- How do we create simulations from streaming data?
  - Treat creation as a special case of updating
  - Aggregate agent locations at tower, distribute agents randomly within cell
- How do we update simulations against streaming data?
  - Redistribute agents within cells to match aggregate data

Creating and Updating ABM: Approach

- How do we create simulations from streaming data?
  - Treat creation as a special case of updating
  - Aggregate agent locations at tower, distribute agents randomly within cell
- How do we update simulations against streaming data?
  - Redistribute agents within cells to match aggregate data
- Evaluate approach
  - Compare aggregate updating to naive 1-1 updating

Updating Vs Reparameterizing

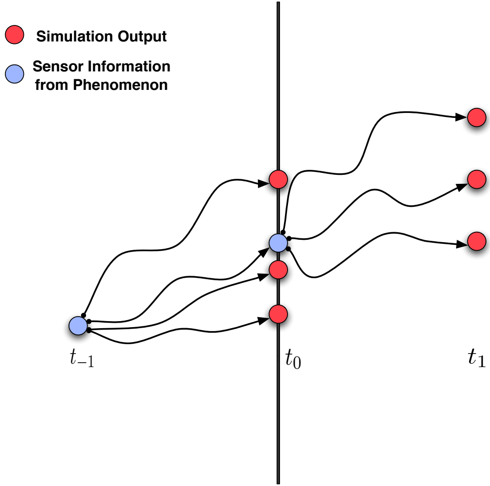


Figure: Updating simulations and the effects on simulation trajectory. Adapted from [Davis 98].

Updating Vs Reparameterizing

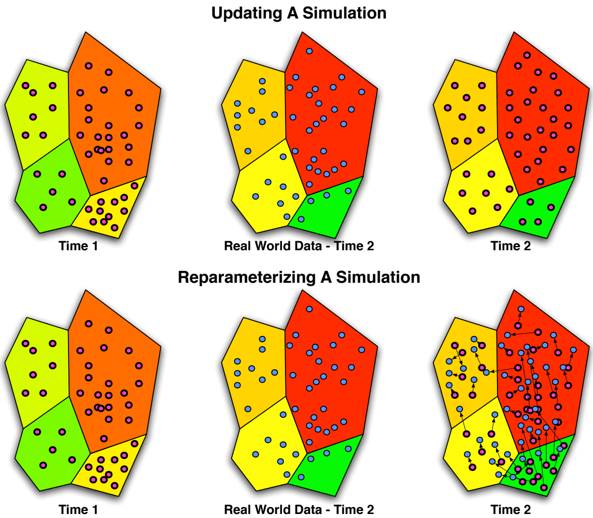


Figure: Comparison of updating a simulation to reparameterizing.

Updating Vs Reparameterizing

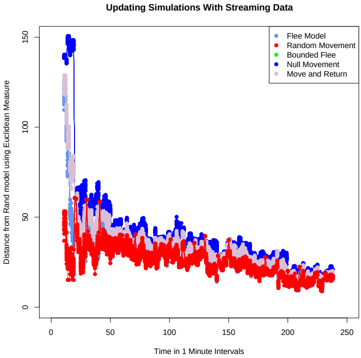


Figure: Updating

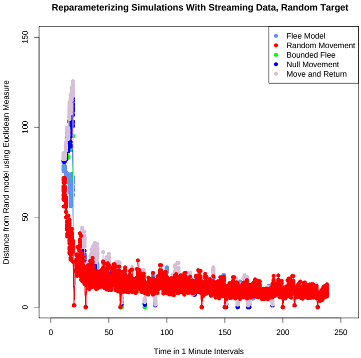


Figure: Reparameterizing

Updating Vs Reparameterizing

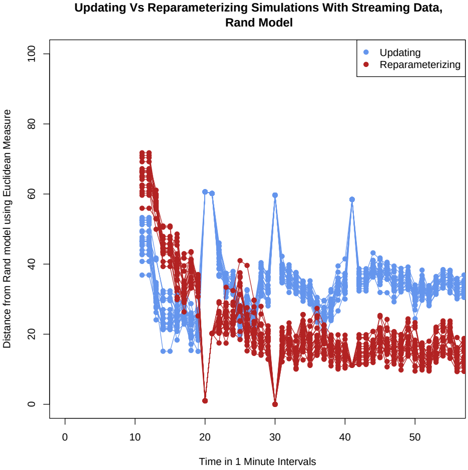


Figure: Comparison of the effectiveness of updating simulations vs reparameterizing. Results shown with 20 simulation runs for each.

Updating Vs Reparameterizing

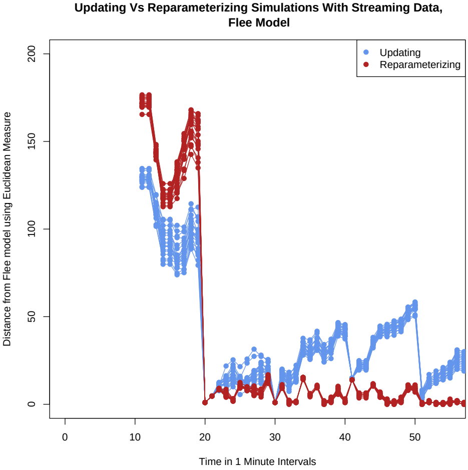


Figure: Comparison of the effectiveness of updating simulations vs reparameterizing. Results shown with 20 simulation runs for each.

Updating Vs Reparameterizing

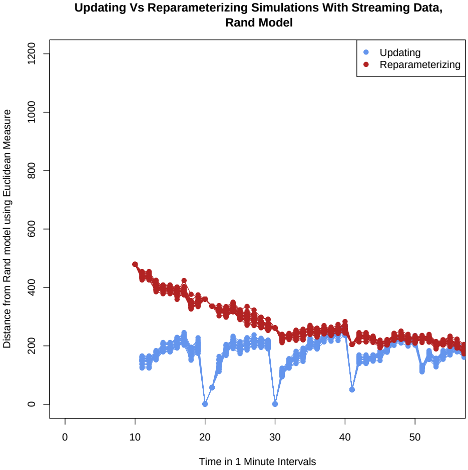


Figure: Comparison of the effectiveness of updating simulations vs reparameterizing. Results shown with 20 simulation runs for each, agent population size

Creating and Updating Contributions

- Method: Aggregate data for creating/updating
- Comparison: Aggregate updating outperforms 1-1 reparameterizing
- Results from Updating vs Reparameterizing show evidence for Naive Realism

Application of Research: The WIPER Simulation Prediction System

Results from this work have been applied to the Simulation Prediction System

- Creates ensemble of simulations from streaming data
- Updates using aggregate method
- Performs online validation using Euclidean metric
- Sends results to DSS console

Movie of Simulations

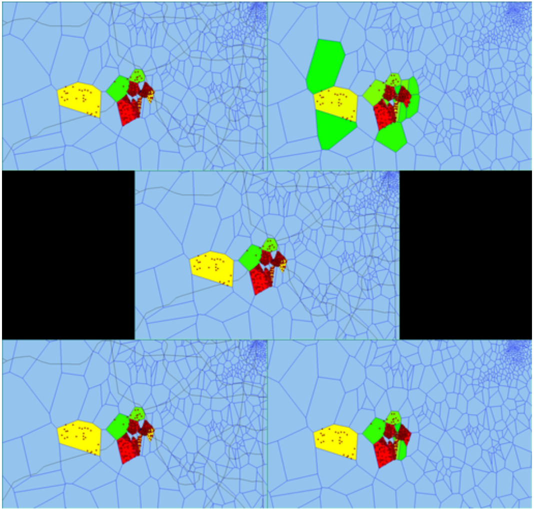


Figure: Movie of simulations, Flee target, 4 other models. Simulations initialized with identical input.

Contributions

- Creation of simulation criteria for use in DDDAS
- Application of simulation criteria: Design and implementation of WIPER simulation

Contributions

- Creation of simulation criteria for use in DDDAS
- Application of simulation criteria: Design and implementation of WIPER simulation
- Presentation of the aggregate approach to creating and updating simulations
- Experiments demonstrating effectiveness of aggregate approach to updating simulations

Contributions

- Creation of simulation criteria for use in DDDAS
- Application of simulation criteria: Design and implementation of WIPER simulation
- Presentation of the aggregate approach to creating and updating simulations
- Experiments demonstrating effectiveness of aggregate approach to updating simulations
- Presentation of selection method for online validation of simulations
- Evaluation of distance measures for ranking models in online validation

Contributions

- Creation of simulation criteria for use in DDDAS
- Application of simulation criteria: Design and implementation of WIPER simulation
- Presentation of the aggregate approach to creating and updating simulations
- Experiments demonstrating effectiveness of aggregate approach to updating simulations
- Presentation of selection method for online validation of simulations
- Evaluation of distance measures for ranking models in online validation
- Design and implementation of Simulation Prediction System, incorporating above research results

Future Work

- Revise movement models to reflect real-world movement studies
- Apply results of validation study to predict optimal ensemble size
- Explore effects of time step length

Publications in Press

[1] Gregory R. Madey, Albert-László Barabási, Nitesh V. Chawla, Marta Gonzalez, David Hachen, Brett Lantz, Alec Pawling, Timothy Schoenharl, Gábor Szabó, Pu Wang, and Ping Yan.

Enhanced situational awareness: Application of DDDAS concepts to emergency and disaster management.

In Y. Shi, G. D. van Albada, J. Dongarra, and P. M. A. Sloot, editors, Lecture Notes in Computer Science (LNCS 4487), pages 1090–1097. Springer, May 2007.

[2] Tim Schoenharl, Ryan Bravo, and Greg Madey.

WIPER: Leveraging the cell phone network for emergency response.

International Journal of Intelligent Control and Systems, 11(4), December 2006.

[3] Tim Schoenharl, Greg Madey, Gábor Szabó, and Albert-László Barabási.

WIPER: A multi-agent system for emergency response.

In Proceedings of the Third International ISCRAM Conference, May 2006.

Publications Under Review / Proposed

- Chapter 4, WIPER Simulation Description under review at the Journal of Defense Modeling and Simulation, Special Issue on Modeling and Simulation in Homeland Security
- Chapter 5, Creating and Updating Simulations from Streaming Data, to be submitted to the International Journal of Modelling and Simulation pending revision
- Chapter 6, Online Validation of ABM to be submitted to the journal SCS Simulation pending revision

Questions

Questions?

Measurement Images

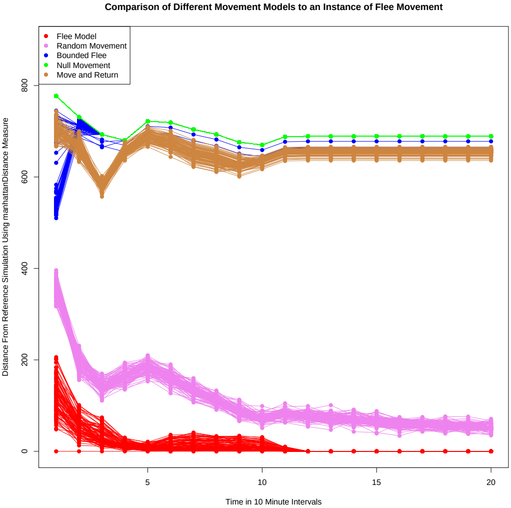


Figure:

Measurement Images

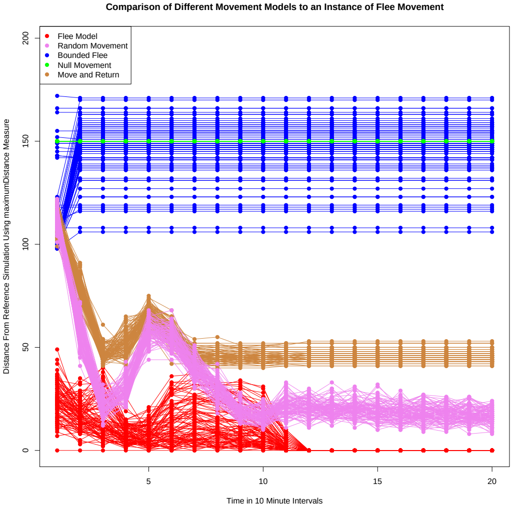


Figure:

Measurement Images

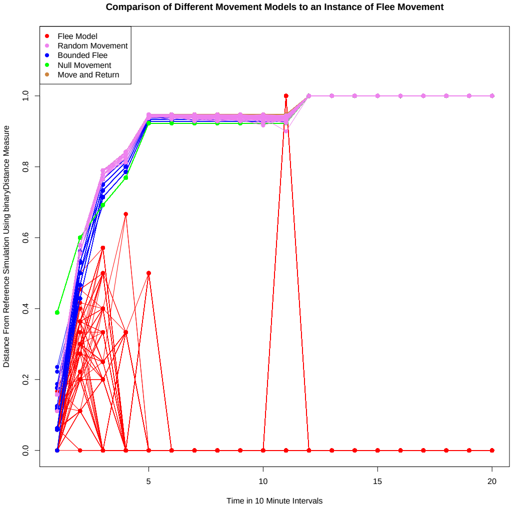


Figure:

Measurement Images

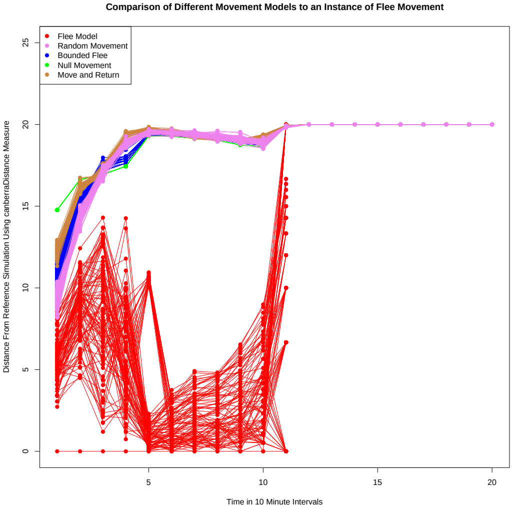


Figure:

Measurement Images

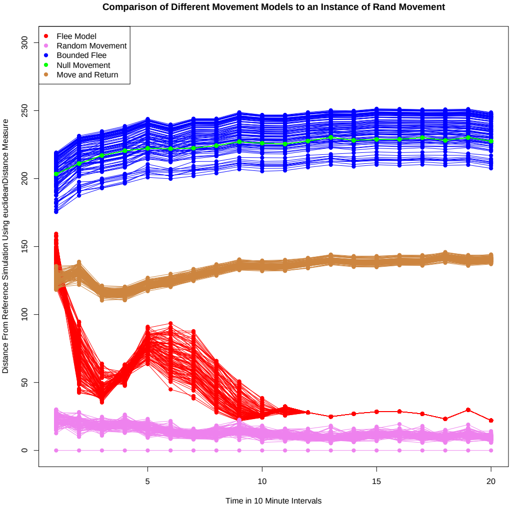


Figure:

Measurement Images

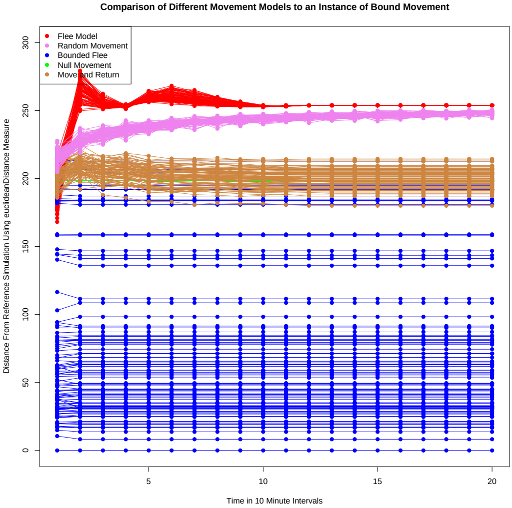


Figure:

Measurement Images

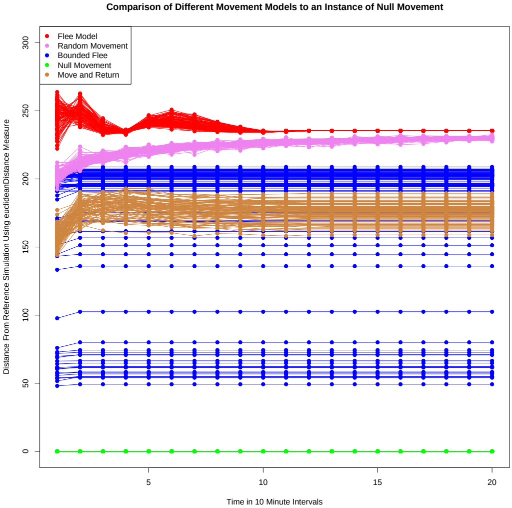


Figure:

Measurement Images

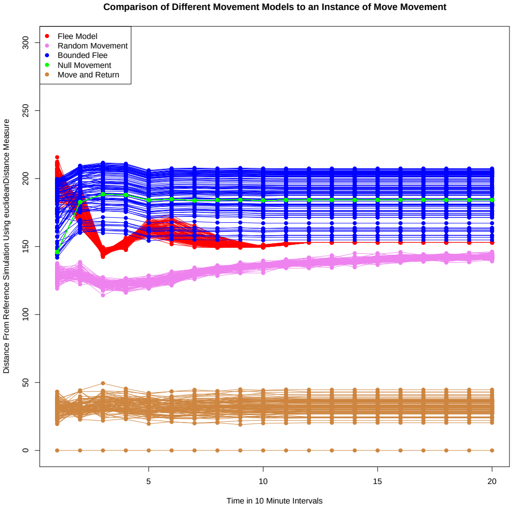
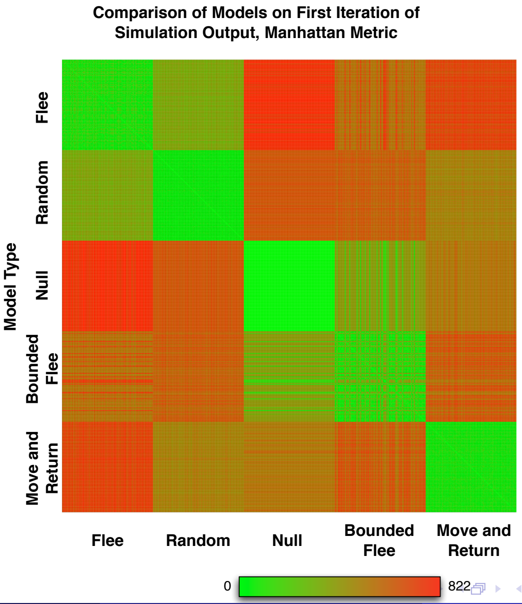
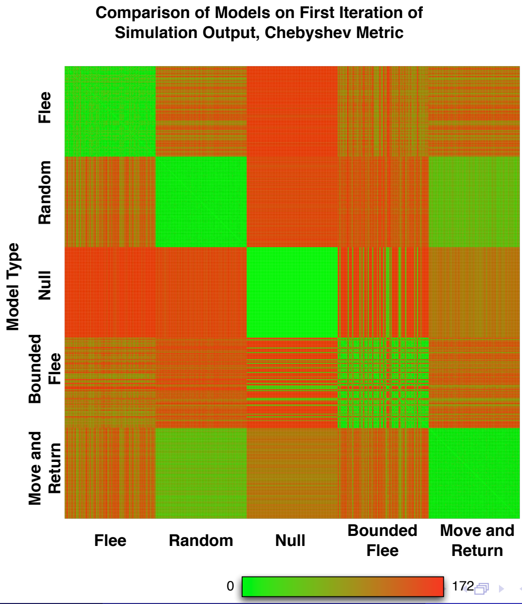


Figure:

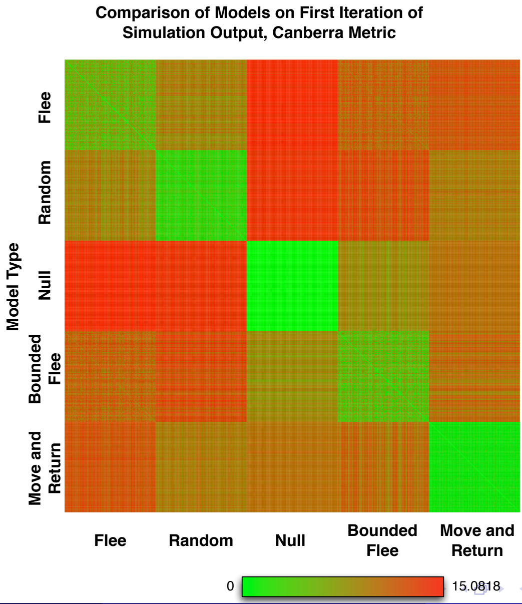
Measurement Images



Measurement Images



Measurement Images



Measurement Images