BMD Agents: An Agent-Based Framework to Model Ballistic Missile Defense Strategies

Duminda Wijesekera¹, J. Bret Michael² and Anil Nerode³
George Mason University¹, Naval Postgraduate School² and Cornell University³

This is a work of the U.S. government and is in the public domain. It may be freely distributed and copied, but it is requested that the author be acknowledged.
Disclaimer

The views and conclusions contained in this presentation are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government.
Acknowledgements

This work is sponsored by the Missile Defense Agency
The environment

BMD modeling requirements

Must account for
- Deliberate planning
- Crisis action planning

Individuals follow a *kill chain*
- Map the individual duties to agents

Optimize QoS Measures
- MoP: Measure of Performance
- MoE: Measure of Effectiveness
Using a collection of agents based on the roles they play in the missile defense environment:

- **Strategic Command Agents**
  - Directs high-level strategies among many regions

- **Regional Command Agents**
  - Coordinates regions consisting of multiple theaters

- **Theater Command Agents**
  - Directs theater-level actions
In pictures

SCA
We lead from here

RCA
Launch

SCA

RCA
The operating environment

SensorNet
- Information gathered (using sensors) about flying objects of interests are broadcasted here

WeaponsNet
- Operational status about weapons systems are broadcasted here
Modeling details

- Build using Event-Condition-Action (ECA) rules
- Agents behavior depends on modes
  - Peacetime
  - Pre-hostilities
  - Hostilities
  - Post hostilities
- Duty cycle
  - Acquire target and lock on
  - Launch, wait, cancel
  - Assess
Designing agents 1: SCAs

- Obtain information from SensorNet, WeaponsNet, AND friends, and
- Assigns tasks with timing constraints to subordinates consisting of
  - Changing modes (peace, war, pre-war, post-war)
  - Computing regional objectives of tracking, and destroying flying objects
  - Altering and/or canceling current objectives
- Informs friends as necessary
Designing agents 2: RCAs

- On receiving directives from superiors
  - Get data from SensorNet, WeaponsNet and
  - Assign time-constrained tasks to TCAs consisting of
  - Pass on changing mode commands (war, pre-war, post-war) to subordinates, and change own mode.
  - Computing regional firing, holding (fire) and canceling fire orders and assign them to TCAs
- Send feedback acknowledgements (about their ability to comply with orders) to superiors
Designing agents 3: TCAs

On receiving directives from superiors

- Get data from SensorNet, WeaponsNet and
- Change mode on command (war, pre-war, post-war)
- Execute the duty cycle of acquiring/locking on/firing/assessing damage to the target
- On command, recompute firing/reload/holdfire/cancel schedules per weapon under own command

Send feedback acknowledgements (about their ability to comply with orders) to superiors
Inform sensorNet an weaponsNet about changes to tracked targets and weapons status
Designing agent communities

- Need to design command, control and communication (C^3I) structure for agents to model BMD functionality

- Use real-life examples
Command structure 1: Hierarchical
Command structure 2: Partially flattened

RCAs removed

Diagram showing the command structure with TCA and SCA nodes.
Command structure 3: Flattened

TCAs work autonomously
Composing agents: $C^2$ structures

- A tree consisting of at most 3 levels
- Every level has at most *one type* of agents
- Agents listed in the SCA/RCA/TCA order
- Every agent knows its superiors/subordinates
- Every SCA knows all of its *friends*

**Lemma:** A simple static analysis algorithm can detect if any collection of agents is a $C^2$ structure

**Limitation:** Does not account for duty polymorphism (i.e. SCA’s doing RCA’s work)
Analysis objectives

- Can the threat missiles be destroyed before it hits or scatters debris over intended target?
  - Missiles entering airspace need to be identified and categorized as threat, potential threat, or benign
  - Targets and travel trajectories/times be computed and all fragments tracked and destroyed in threat missiles
  - Commanders need to obtain authority to aim at missiles
    - This authority need to propagate through the command chain
    - Takes time to lock on and fire
    - Do follow-up shots destroy the threat missile?
    - If object is reclassified as benign, need to cancel/delay firing
Preliminary results

Compute periods for duty cycles of agents using
- Worst-case estimates for command execution times
- Performance delays of weapon systems

Compute command propagation times through statically-composed C2 structures using
- Worse-case communication delays
- Computed duty cycle periods

Using these estimates, one can compute if a properly identified threat missile can be intercepted with a particular weapon
Limitations

- Need to account for
  - Hit/destroy probabilities
  - Reclassification of missile status and the ability to recall/re-target missiles

- Need to incorporate measures
Measures

 Measures of Effectiveness, such as

- Can launches (or repeat launches) destroy threat missiles?
- Does the system hold fire if missile status is reclassified?

 Measures of Performance, such as

- How much above ground are they destroyed?
- Delay in reacting to reclassification
Related work

- Many approaches
  - Force-structure-based
  - Strategy-based

- Some examples:
  - Michael, Pace, Shin, Tummala, Weller, Miklakski, Babbit: Test and evaluation of BMD systems, NPS TR-CS-03-007, 2003
Summary

- Presented a preliminary ECA rule-based agent framework to capture BMD C2 requirements where
  - Strategy and policy are written as BMD rules
- A preliminary formulation of a well-formed agent society for BMD C2
- A back-of-the-envelope timing calculation
Ongoing work

- Experimenting with a model that uses probabilistic temporal reasoning
  - (Probabilistic Temporal Agents of Kraus et al.)
- Using rules to code policies and strategies
- Formulating a framework for both
  - Hierarchically building the MoEs and MoPs
  - Computing
    - Probability of achieving the numbers
    - Schedules for launches