Informing Joint C2 System-of-Systems Engineering with Agent-Based Modeling: An Analysis and Case Study

Chuck Lutz
Mitchell Kerman
Greg Schow
Mike DiMario
Ambrose Kam
20 June 2007

System of Systems Engineering

- **Military acquisitions process shift from “systems” to “capabilities”**
- **US DoD Joint Vision 2010 and 2020 spells out necessity for:**
  - Multinational and US interagency cooperation
  - Interdependent, joint C2 for full capability range
  - Optimal integration of joint forces
- **Complexity of interdependent capabilities gives rise to SoS and SoSE**
- **Individual entities remain independent, yet act in concert**
  - Systems are dynamically formed and dissolved
  - SoS has greater capabilities than the sum of the individual systems
  - SoS typically exhibit “emergent behavior”
- **Systems Engineering is evolving to address SoS problems of ambiguity, multiple contexts, and nonlinear inter-system relationships**

SoS and Agent-Based Modeling

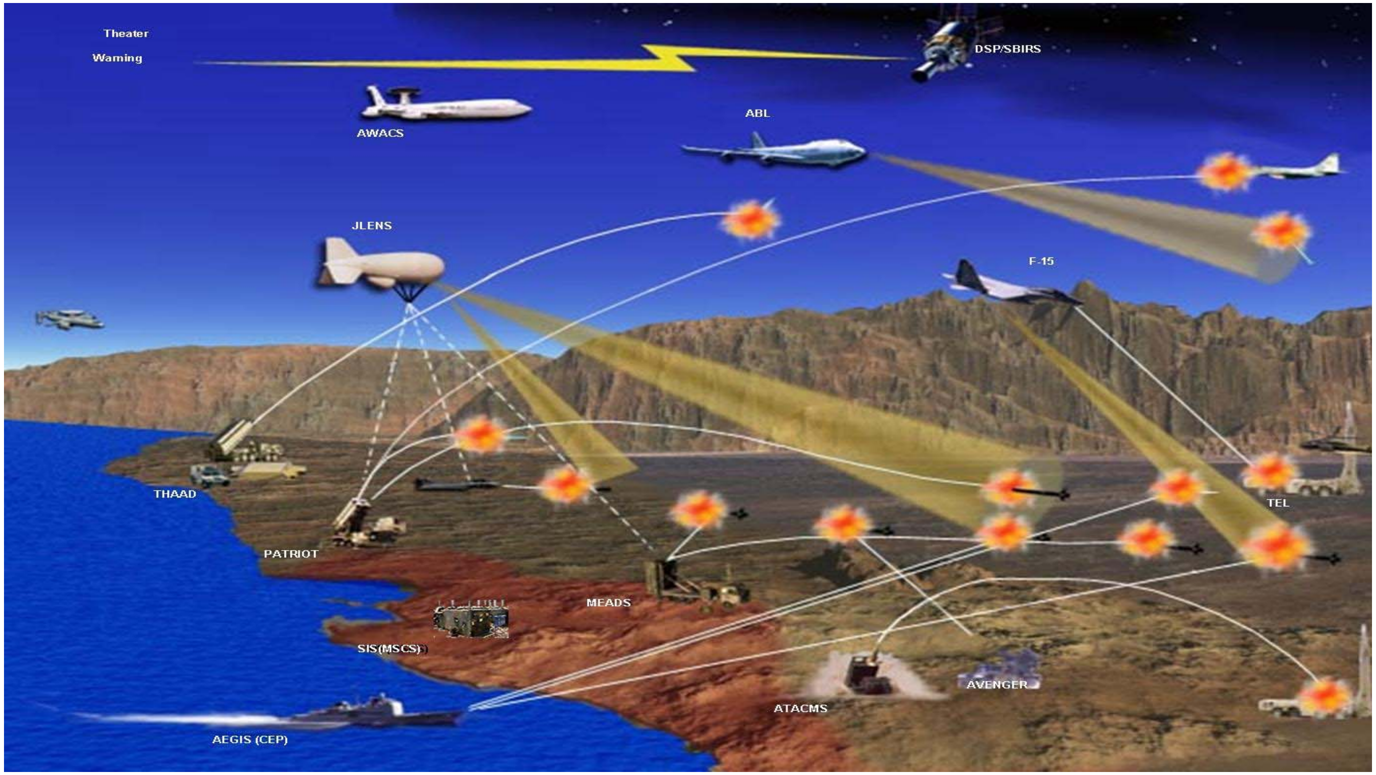
- **Complex Adaptive Systems (CAS) dynamics apply to SoS**
  - Autonomous agents interact with each other and their environment, exhibiting stimulus-response behavior
  - Properties: nonlinearity, difficult or impossible long-term prediction, dynamic group formation, and complex inter-agent communication
  - Agents adapt their behavior and may spawn new species
- **Agent-Based Modeling is a good platform for capturing CAS models**
  - Model consists of agents situated in some contextual world
  - Agents and their environment (the world) have properties of interest
  - Agents are programmed with simple rules for sensing and interacting with the environment, including other agents
  - Agents can include learning and memory
- **ABM approach is gaining popularity in many fields (finance, biology, physics, national defense, etc.)**
- **We use it to create and validate models of SoS architectures and explore alternative solutions**

Conventional vs. Agent-Based M&S

- **Much M&S work uses conventional discrete-event simulations**
  - Event-driven execution
  - Entities have planned behavior patterns – “scripted” predefined operating areas and command structures
  - Typically don’t provide ability to program adaptive behavior
  - Offer sensor and weapon models of varying fidelity
  - Offer predefined C2 rules for situational awareness and reactions
- **ABM captures CAS properties that conventional M&S does not**
  - Complex macroscopic behavior patterns from simple individual behavior
  - Impossible to “script” ahead of time – “surprise” factor
- **ABM execution model differences**
  - Time vs. event-driven
  - Parallel agent behavior processing requires lower fidelity sensors etc.
  - Rule-based, reactive behavior is primary “mode”
  - Agent interactions typically more “dense” in time vs. conventional

Combat Identification

- Combat ID research tries to find better, more reliable means of accurately identifying objects and acting accordingly



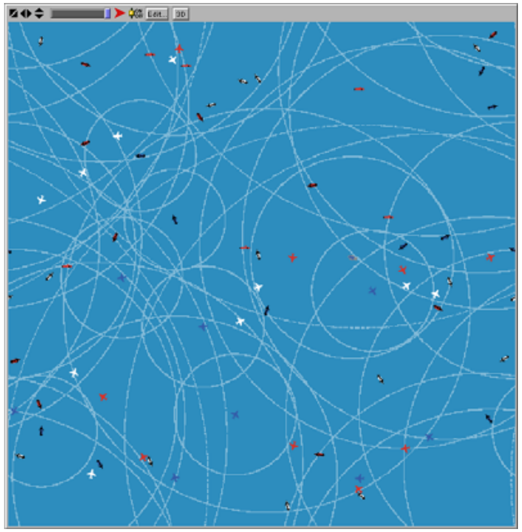
CCID/CRA & SIAP Overview

- **Research in automated combat identification (CID)**
  - Interest in decreasing fratricide due to misidentification in the battlespace
  - Current activities to form a cohesive operating picture are manual
  - Reconciliation of several forms of data is required
- **ONR's Composite Combat Identification Common Reasoning Algorithm**
  - Dempster-Shaefer evidential reasoning approach to automatic CID
  - Given multiple sensor and other input, makes CID recommendations
- **Part of a larger effort**
  - JSSEO: Joint SIAP Systems Engineering Organization
  - SIAP: Single Integrated Air Picture
  - Automatic Combat ID deployment is one focus
  - Enhanced comms and networking is also a focus
  - IABM: Integrated Architecture Behavior Model – caputure requirements for fully networked sensors and data available across integrated forces

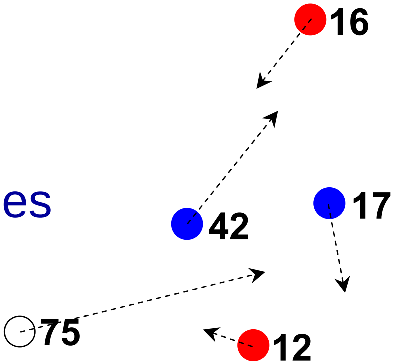
Study Overview / Problem Statement

- **CCID/CRA architecture and algorithms were still under development**
  - Decided to apply an exploratory modeling approach
  - Model runs consititute experiment runs
- **Our focus is distributed CCID/CRA**
  - Existing CCID/CRA studies considered only a single CRA node
  - Study the impact of network etc. effects on coordinated, distributed CRA
  - Assume CRA deployed on air and surface platforms
  - Assume SIAP environment: Full sharing of sensor and track data
- **Network Delays**
  - Expected even under the best bandwidth and QoS conditions
  - Inorganic networked sensor data arrives to a CRA node delayed
  - **What is the effect on the processing?**
- **Resolution of CRA recommendation per-track "disagreements"**
  - Both sensor data and CRA recommendations are broadcast to the force
  - Due to network delays, different CRA nodes across the force produce different CID recommendations for the same common track
  - **How should these be resolved to form a "final" agreed CID for the track?**

Case Study: Model Design

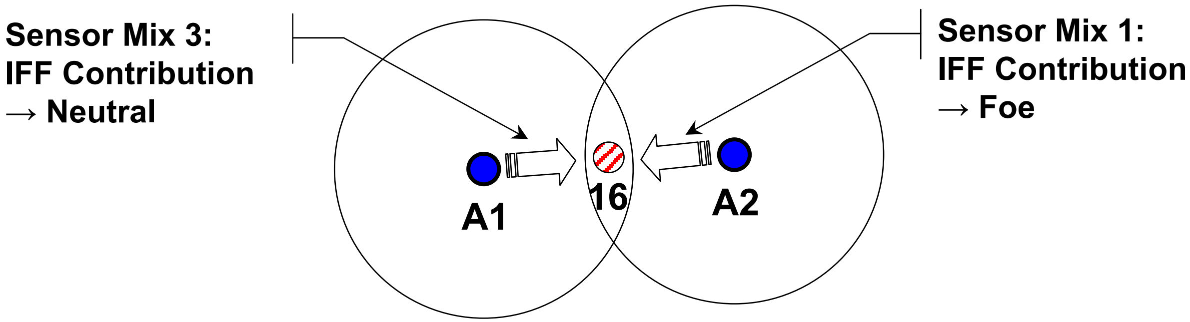


- **Implemented in an ABM M&S tool**
- **Air, surface, and subsurface platforms**
  - Agents instances of platform types
  - Simple radial sensing
  - Simulated CRA processing
  - Simple movement
  - Each has a ground truth track ID number
- **Agents communicate "full mesh" with delays**
  - Delay based on a simple distance calculation
  - Each agent has a FIFO queue that holds received messages with timestamps indicating when the messages should "arrive"
- **Agents have ground-truth Combat ID (CID) attributes:**
  - Allegiance: FRIEND, FOE, or NEUTRAL
  - Nationality: Generic friendly, neutral, or hostile nations
  - Type: Various known and invented generic platform types



Case Study: Model Design

- **CRA calculates probabalistic “belief values” based on attributes of tracks under consideration and produces CID recommendations based on the resultant belief values**
- **Different sensor types could contribute differently to CRA belief values**
  - Toward this end, we created eight generic organic sensor types
  - Each has a weighting indicating its contribution to belief values
  - Simple sensor model: P(d) = 1.0 within sensor’s range
  - Each Blue platform type has specific sensor mix with differing ranges
  - Allows to model varying effects of input sensor data
  - Example: IFF (Identify Friend or Foe) has greater impact on Allegiance or Nationality recommendations than on Type.



Case Study: Model Design

- **Time advancement occurs in "ticks"**
  - A global tick counter keeps track of the time value of "now"
- **Sensing, CRA processing, and reporting**
  - All Blue agents sense and report tracks to all Blue surface agents
  - All Blue surface agents perform CRA processing and report their recommendations to all other Blue surface agents
  - Conflicting recommendations must be resolved
- **Track report**
  - [<track_ID> <alleg_sens_contrib> <natl_sens_contrib> <type_sense_contrib>]
- **CRA report**
  - [<track_ID> <allg_rec> <allg_bv> <natl_rec> <natl_bv> <type_rec> <type_bv>]
- **Each agent keeps a FIFO queue for Track and CRA reports**
  - Incoming messages can be consumed when "timestamp" = "now"
  - Multiple track reports for a given track are fused simply
- **CRA belief values are floats between 0 and 1 representing confidence**

Case Study: Model Design

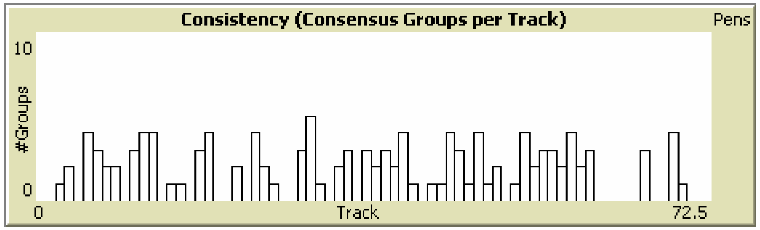
- **Each time tick constitutes one increment of processing**
- **Agent Event Loop (per tick)**
  - (surface/air) Detect all agents within the detection radius
  - (surface/air) Produce organic track report for each detected agent
  - (surface/air) Publish organic track reports to the network
  - (surface) Retrieve all non-organic track reports from the network queue to obtain a full set of tracks
  - (surface) Fuse reports for identical tracks to obtain minimized track set
  - (surface) Generate organic CRA report for each track
  - (surface) Publish organic CRA reports to the network
  - (globally) Arbitrate on the network-wide set of CRA reports to obtain a globally agreed final CRA report for each track
- **During the processing, various metric data are collected (more later)**

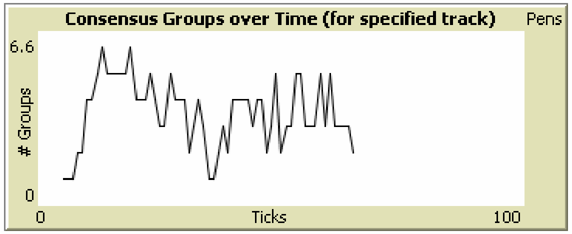
Case Study: Arbitration Concepts

- **<u>Consensus Group:</u> Given N CRA-enabled nodes that have processed a SIAP track, we will be able to divide the N nodes according to their CCID recommendations into a number of groups G < N, where all the nodes in a group have produced the same recommendation (i.e. they "agree"). Ideally, G = 1 and all nodes agree.**
- **<u>Arbitration Scheme:</u> When G > 1, the CRA nodes require some way of resolving the differences and coming to agreement on a final CCID recommendation. We call the method by which they do this an *arbitration scheme*.**
- **We nominated four arbitration schemes:**
  - <u>Majority Voting:</u> At iteration end, the number of CRA report "votes" of different kinds per track is counted and the report with the most "votes" is the "winner"
  - <u>Maximum:</u> Per track, the belief values for each CCID attribute are compared across all CRA reports; the final accepted CRA report is a hybrid, consisting of the maximum belief values per attribute type across all reports
  - <u>Naïve Bayesian:</u> The track's final CRA report is based on the median of the CRA report belief values per attribute across all reports for the track
  - <u>Weighted Bayesian:</u> The track's final report is based on a weighted combination of CRA report belief values across the report group

Case Study: Hypothesis

- **We expected to find that as network transmission delays increased, the ability to maintain a single global, shared track picture would be diminished.**
  - With no time delay, all agents would receive track and CRA report information simultaneously, so the number of consensus groups would be one
  - The number of consensus groups per-track would increase with delay
  - The number of consensus groups formed would depend on the particular geometry of the agents' positions at any given moment
- **We were also interested in evaluating the performance of our nominated arbitration schemes compared to CID ground truth**
- **Additionally, we were curious as to the effects of blue/red/white agent population mix variations**





Case Study: Experiment Plan

- We created two groups of heterogeneous sets of agents, varying the number of blue forces in the first group and the number of red and white forces in the second group

Run Matrix One

Blue	Red	White	Total
2	25	25	52
5	25	25	55
10	25	25	60
15	25	25	65
20	25	25	70

Run Matrix Two

Blue	Red	White	Total
10	5	5	20
10	10	10	30
10	15	15	40
10	20	20	50
10	25	25	60
10	30	30	70

- We ran two sets of simulation runs, one for each agent population matrix. Other dimensions were maximum communications time delay and arbitration scheme.

	Run Matrix One	Run Matrix Two
Number of Population Sets	5	6
Number of Maximum Time Delay Settings	4	4
Number of Arbitration Schemes	4	4
Matrix Size (Runs)	80	96

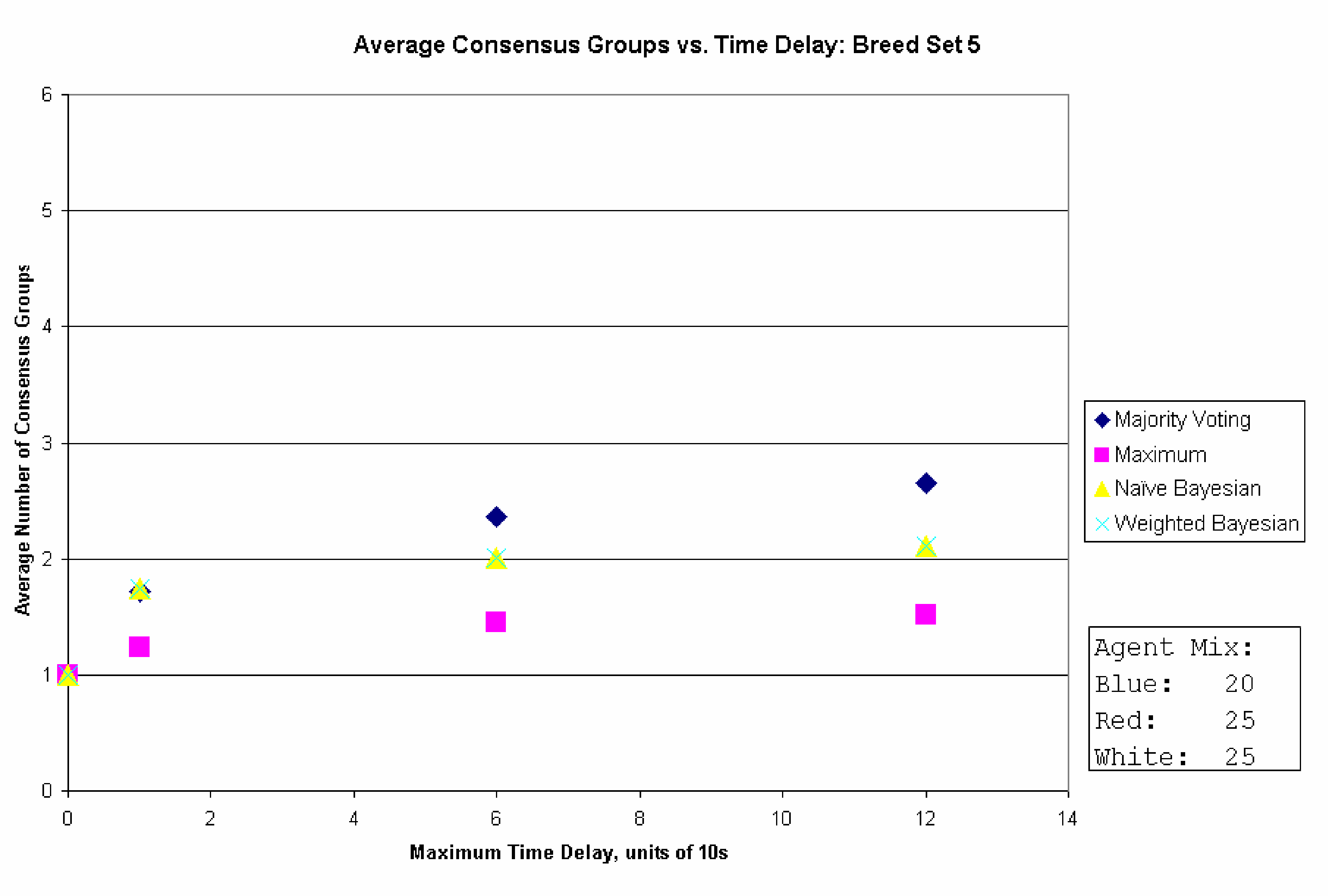
Case Study: Experiment Metrics

- **We collected the following metrics data with each run:**
  - Average, median, and ceiling of the number of consensus groups across all tracks for the entire run
  - Overall correctness (percentage) compared to ground truth track data
  - Percentage of reports that had each individual CID recommendation attribute (Allegiance, Nationality, and Type) correct
  - Percentage of reports that had 0, 1, 2, or 3 out of 3 attributes correct
- **Overall Correctness**
  - Overall correctness is given by

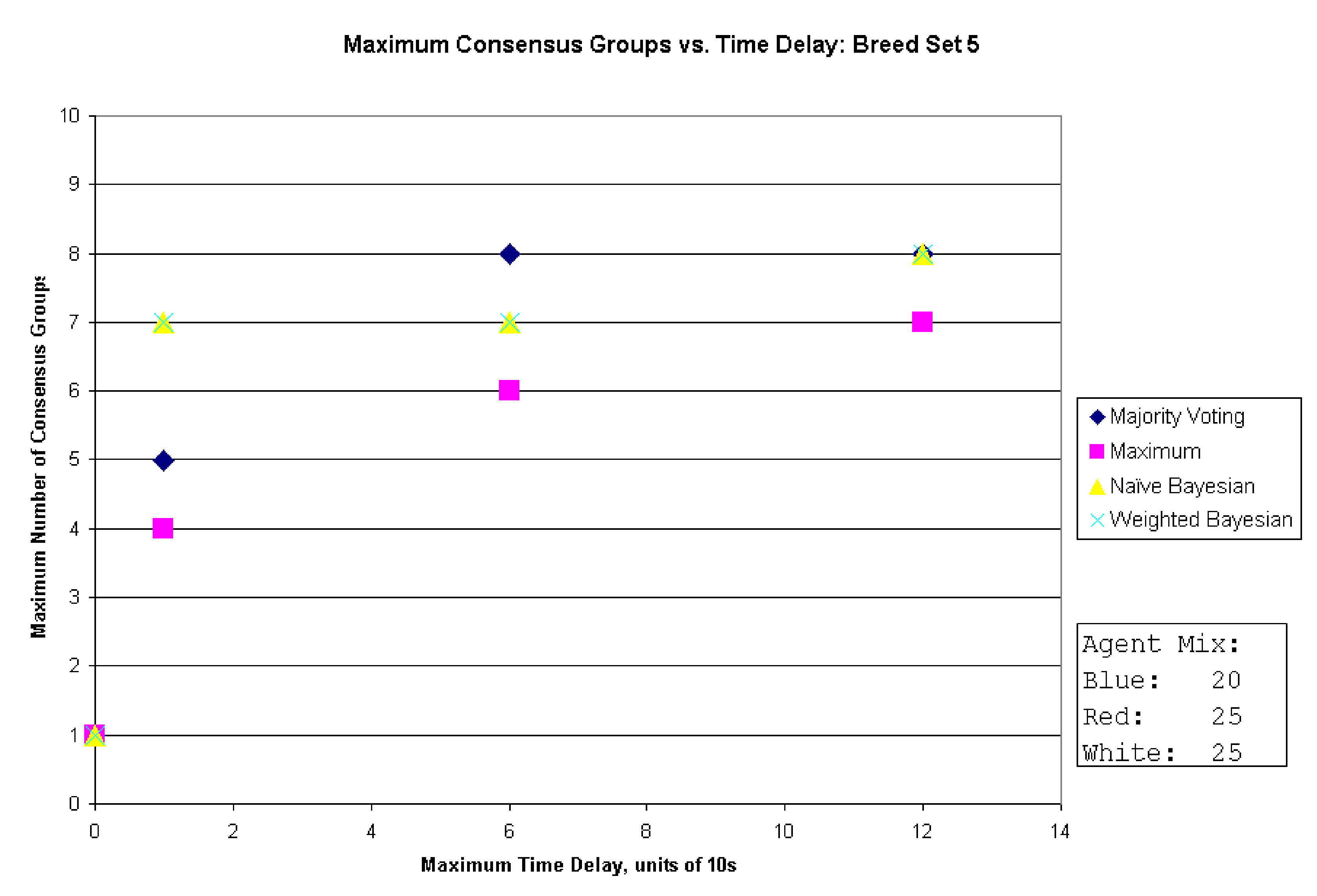
$$OverallCorrectness = \frac{\sum_{i=1}^{R} score_i}{R * 3},$$

… where R is the number of final CRA recommendation reports generated throughout the run (there were three attributes per report). Per-attribute correctness was calculated similarly.

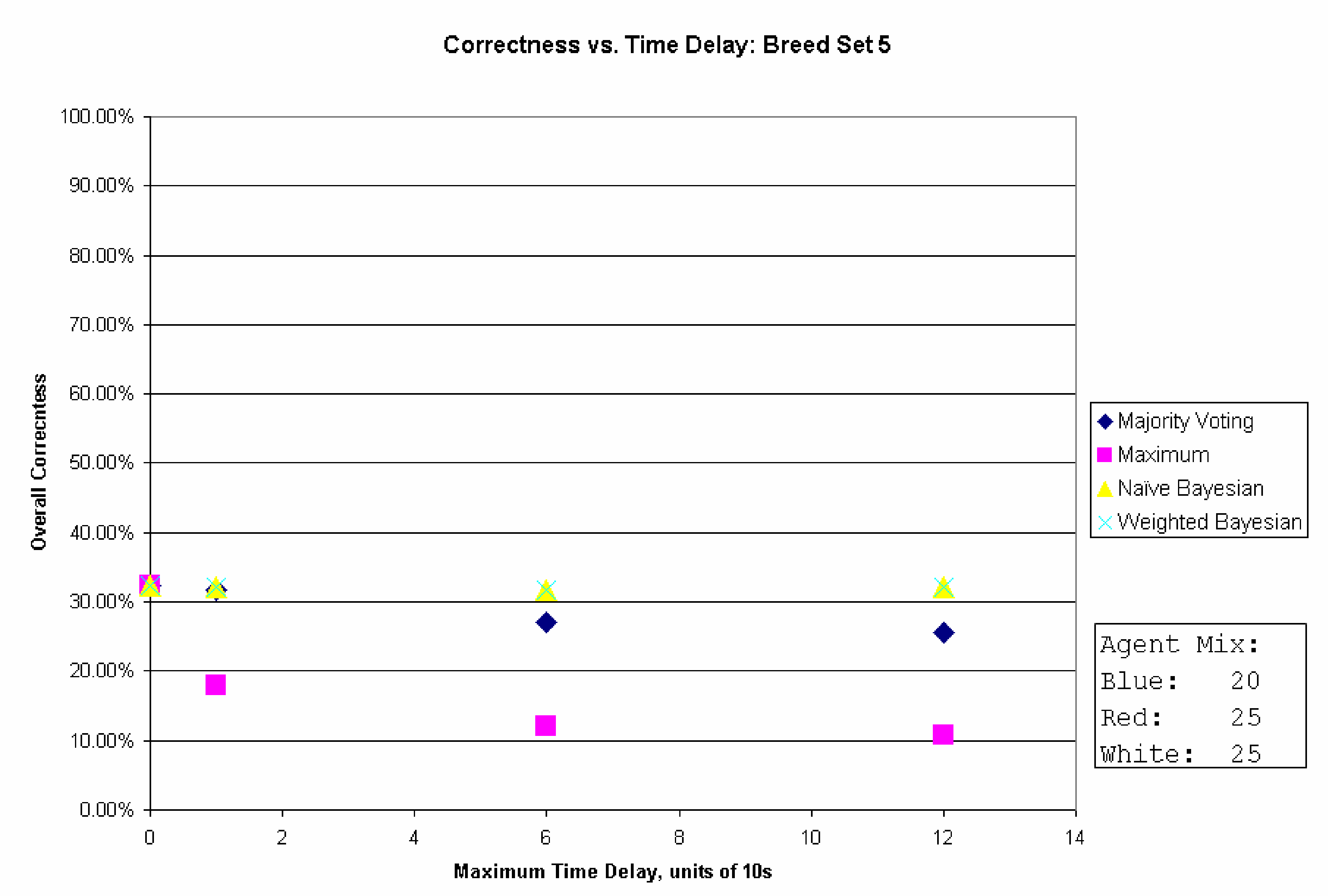
Case Study: Results



Case Study: Results



Case Study: Results



Case Study: Results

- Tabulated results for correctness, normalized over all runs

Normalized Correctness over All Runs		
Scheme	**Overall**	**Rank**
Naïve Bayesian	1.000	1
Weighted Bayesian	0.997	2
Majority Voting	0.996	3
Maximum	0.746	4

Case Study: Conclusion

- **Utility of the Approach**
  – Using an ABM approach allows performing "bottom up" vs. "top down" analyses offered by conventional simulations
  – ABM lends itself to the investigation of distributed coordination problems and for identifying inconsistencies in the distributed architecture
  – By exploring the effects of aggregate behavior, the model allowed us to look for unanticipated results

- **Areas for Further Research**
  – Further enhance the use of ABM in SoS Engineering to provide insight into hard-to-capture qualities of SoS
  – Incorporate more agent learning and adapability and increase the fidelity of sensor and effector modeling
  – Use ABM as a "light weight" simulator to identify "main effects" within a run matrix to prune dimensional values for further study in higher-fidelity simulations (e.g. Taguchi method)
  – Integrate ABM model with other models and simulations that provide higher fidelity for environmental and other aspects