SYSTEMS ENGINEERING RESEARCH CENTER

USC Viterbi School of Engineering Systems Architecting and Engineering

Next Generation Adaptive Cyber-Physical-Human Systems

Sponsor: OUSD(R&E) | CCDC

By
Dr. Azad Madni
11th Annual SERC Sponsor Research Review
November 19, 2019
FHI 360 CONFERENCE CENTER
1825 Connecticut Avenue NW, 8th Floor
Washington, DC 20009

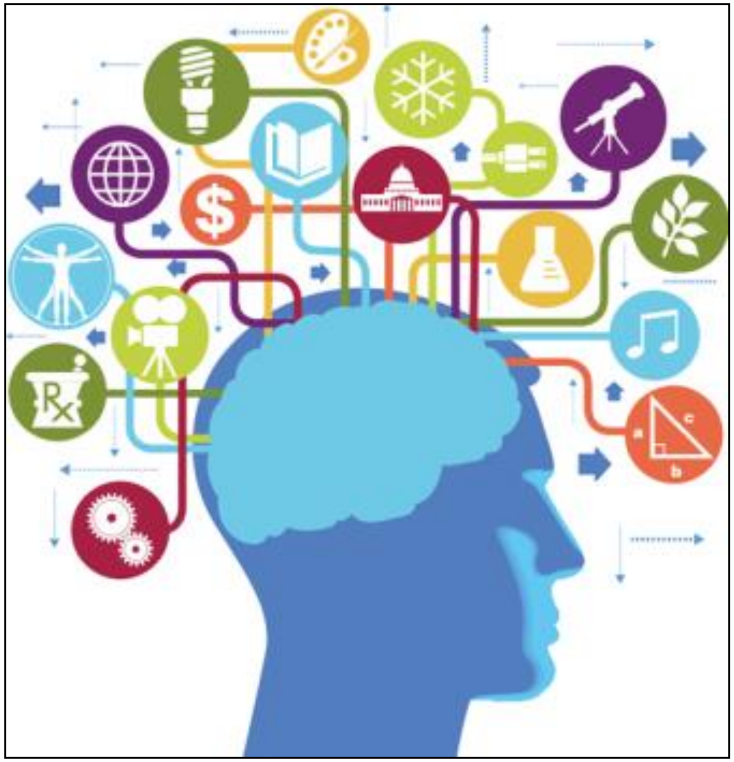
www.sercuarc.org

Project Team

- Prof. Azad Madni, Principal Investigator
- Prof. Dan Erwin, Co-Investigator
- Dr. Ayesha Madni, Project Manager
- Edwin Ordoukhanian, Research Assistant
- Parisa Pouya, Research Assistant
- Shatad Purohit, Research Assistant

Outline

- Research Objectives
- Accomplishments Summary
- Technical Approach
- Prototype System
- Findings and Lessons Learned
- Technology Transition



Research Objectives

- Investigate innovative approaches for developing next generation adaptive CPHS in which human(s) and cyber-physical (CP) elements collaborate in joint task performance and adapt as needed to respond to operational contingencies and disruptions

- **Illustrative Application:** Perimeter security of C-130 aircraft parked on a landing strip and secured by fixed and mobile collection assets

21st Century DoD Systems

- High complexity (hyper-connectivity, interdependencies)
- Need to operate safely for extended periods in dynamic, uncertain environments subject to disruptions
- Long-lived (> 20 years)
- Likely to be extended / adapted over lifetime
- Stringent physical and cyber security requirements
- Adaptive and distributed autonomy

Need new modeling methods and tools

Cyber-Physical-Human Systems
(Madni et al., 2018)

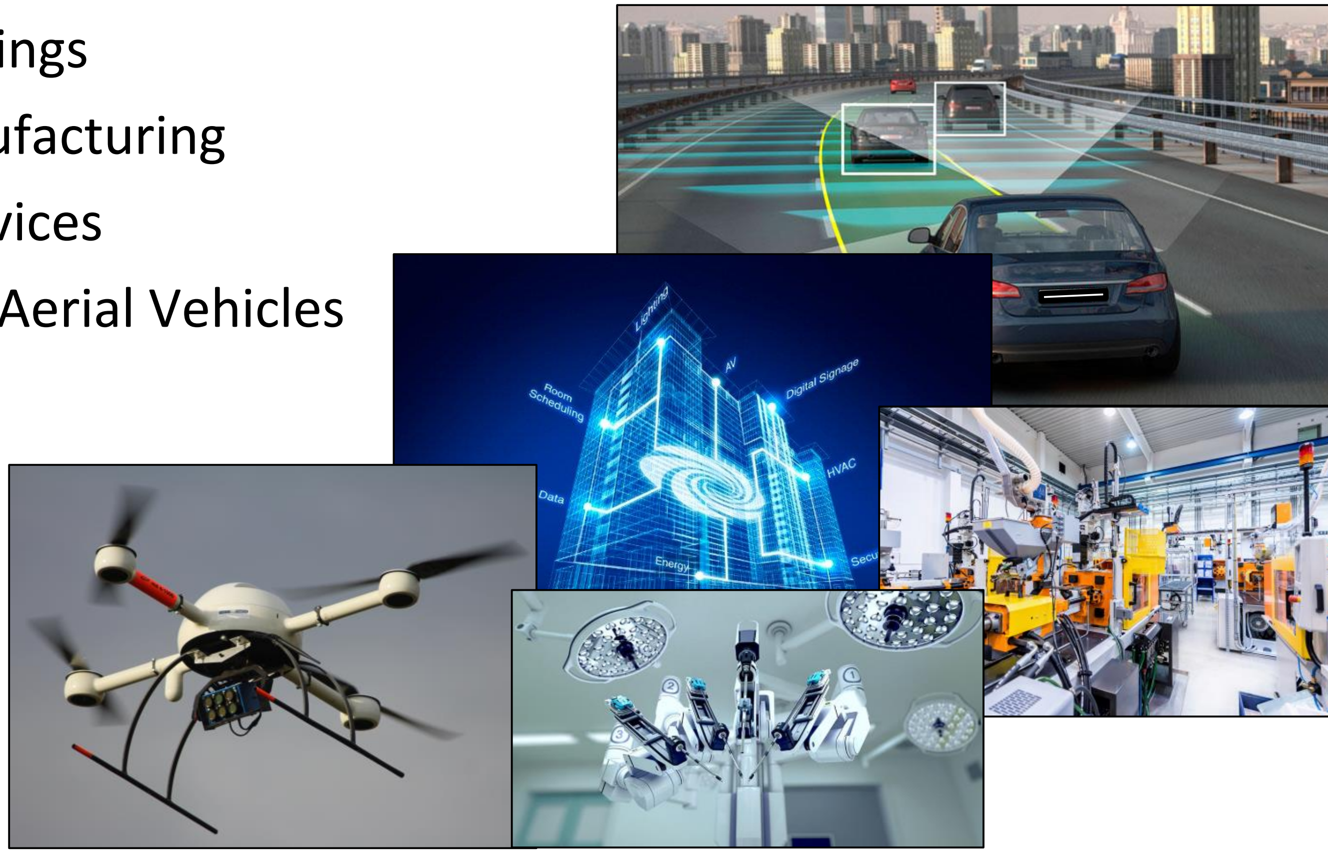
- A class of safety-critical socio-technical systems in which interactions between *physical system* and *cyber elements* that control its operation are influenced by *human agent(s)*

- System objectives achieved through interactions between:
  - —**Physical system** (or process) to be controlled
  - —**Cyber elements** (i.e., communication links and software)
  - —**Human agents** who monitor and influence cyber-physical system operation

- **Distinguishing Feature:** Human (agents) intervene to:
  - —redirect cyber-physical elements or supply needed information
  - —.....not just to exercise manual over-ride or assume full control

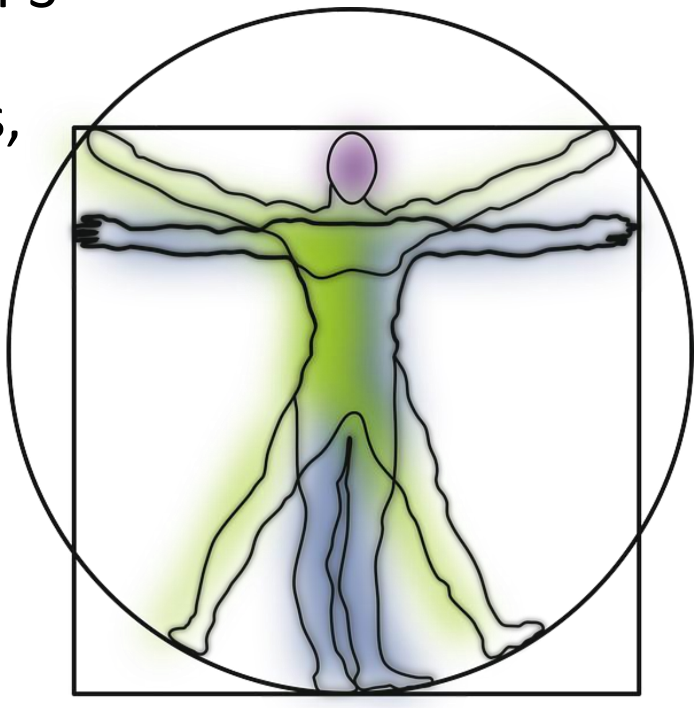
Exemplar CPHS

- Safety-critical systems - range from small devices to SoS
  - Self-Driving Vehicles
  - Smart Buildings
  - Smart Manufacturing
  - Medical Devices
  - Unmanned Aerial Vehicles



Adaptive CPHS

- **Respond** to disruptions and changes in context
- **Exploit synergy** between humans and CPS
- **Capitalize on** unique human capabilities, while circumventing human limitations
- **Leverage** CPS strengths while circumventing CPS limitations
- **Learn** from experience (observations, outcomes) using ML

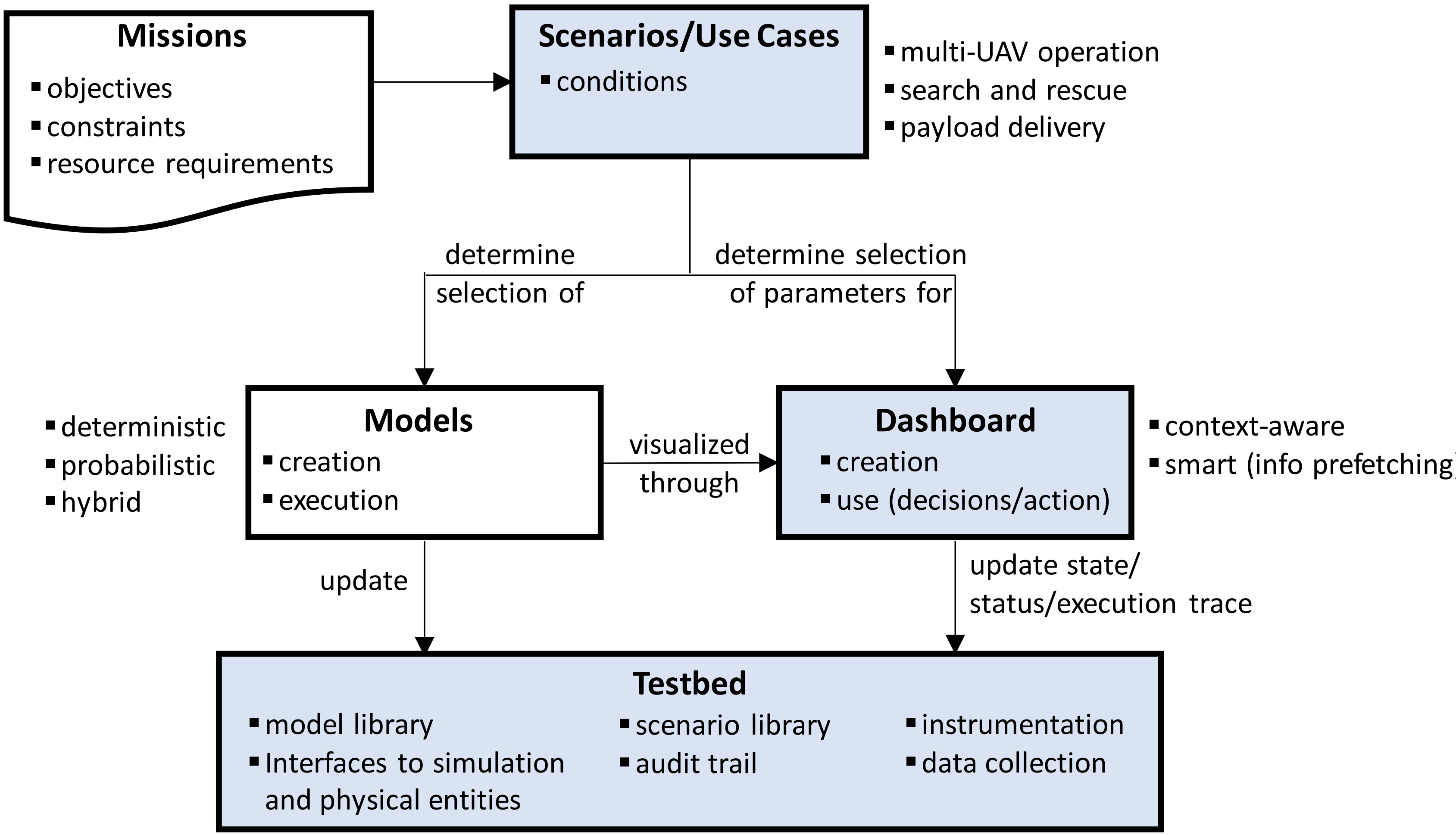


Deficiencies in Existing Modeling Methods and Tools

- **Methods:** Ill-suited for tightly-coupled, sociotechnical learning systems – do not have:
  - —semantics of time
  - —ability to improve with use
  - —flexible representation of human behavior
  - —learning ability (offline, in-situ)
- **Tools:** reflect methodological deficiencies
  - —address cyber, physical, and human elements in isolation
  - —focus primarily on subsystems, not their interactions and dependencies
    and synchronization constraints
  - —“build-time” approaches -- no “run-time” learning

Technical Approach

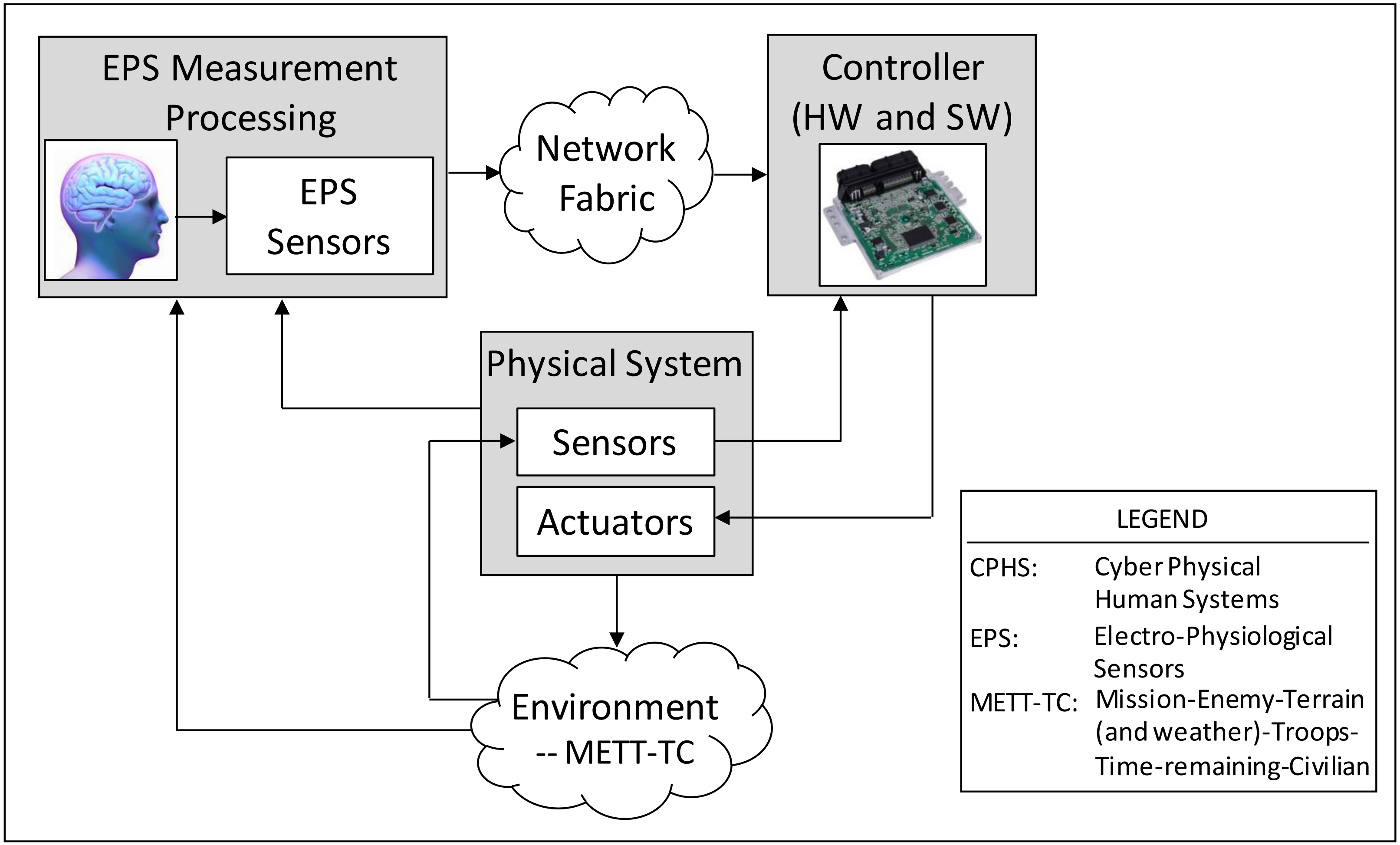
Conceptual Framework



Approach Highlights

- Leverage models from RT-210
  - formal and probabilistic modeling
  - machine learning
- Adaptive CPHS Research Focus
  - interactive planning and decision making
  - supervisory and autonomous control
  - geographic region coverage optimization
  - human behavior modeling
- Context-aware ("smart") dashboard
  - context defined by a formal ontology (METT-TC)
  - multi-perspective, multilevel, with visual cueing
- Testbed Capabilities
  - support adaptive CPHS research focus areas
  - support data collection and maintain audit trail
  - control both virtual simulation models and physical systems

Adaptive CPHS System Concept



Human Roles in Adaptive CPHS

- **Monitor/Supervisor**: outside the control loop
  - monitor and interact with environment (CPS unaware of this interaction)
  - assess correctness of operation of CPS; approve CPS decision
  - intervene at appropriate level in control loop (context: CPS requests take over; incorrect or error-prone CPS behavior; over-ride erroneous CPS decision)
  - re-allocate tasks (context: cognitive overload/fatigue; CPS request)
- **Controller**: within the control loop
  - intervene at appropriate level in control loop (context: have new / missing info)
  - e.g., redirect sensors / collection assets; supply missing information
  - e.g., modify actuator inputs based on info unavailable to controller
- **Backup**: within the control loop
  - assume CPS control function (context: when CPS malfunctions, or CPS requests human takeover, or CPS fails to respond in allotted time)

Exemplar Adaptations

Adaptation Type	Triggering Criteria	Desired Outcome
Re-allocation of Task(s) from Human to Machine	Human Cognitive load exceeds threshold; Fatigue; Human error rate exceeds threshold	Manageable human cognitive load; Acceptable error rate
Re-allocation of Task(s) from Machine to Human	Novel situation (unrecognizable by CPS); CPS request; CPS malfunction	Proper handling of novel situations/contingencies
Machine Adapts to Human	Change in human preference structure and information seeking policy	Increased S/N ratio information delivered to human especially under time-stress
Human Adapts to Machine	Machine request to transfer control; change of context requires transfer of control	Superior ability to deal with operational tasks and situation

Human Behavioral Modeling

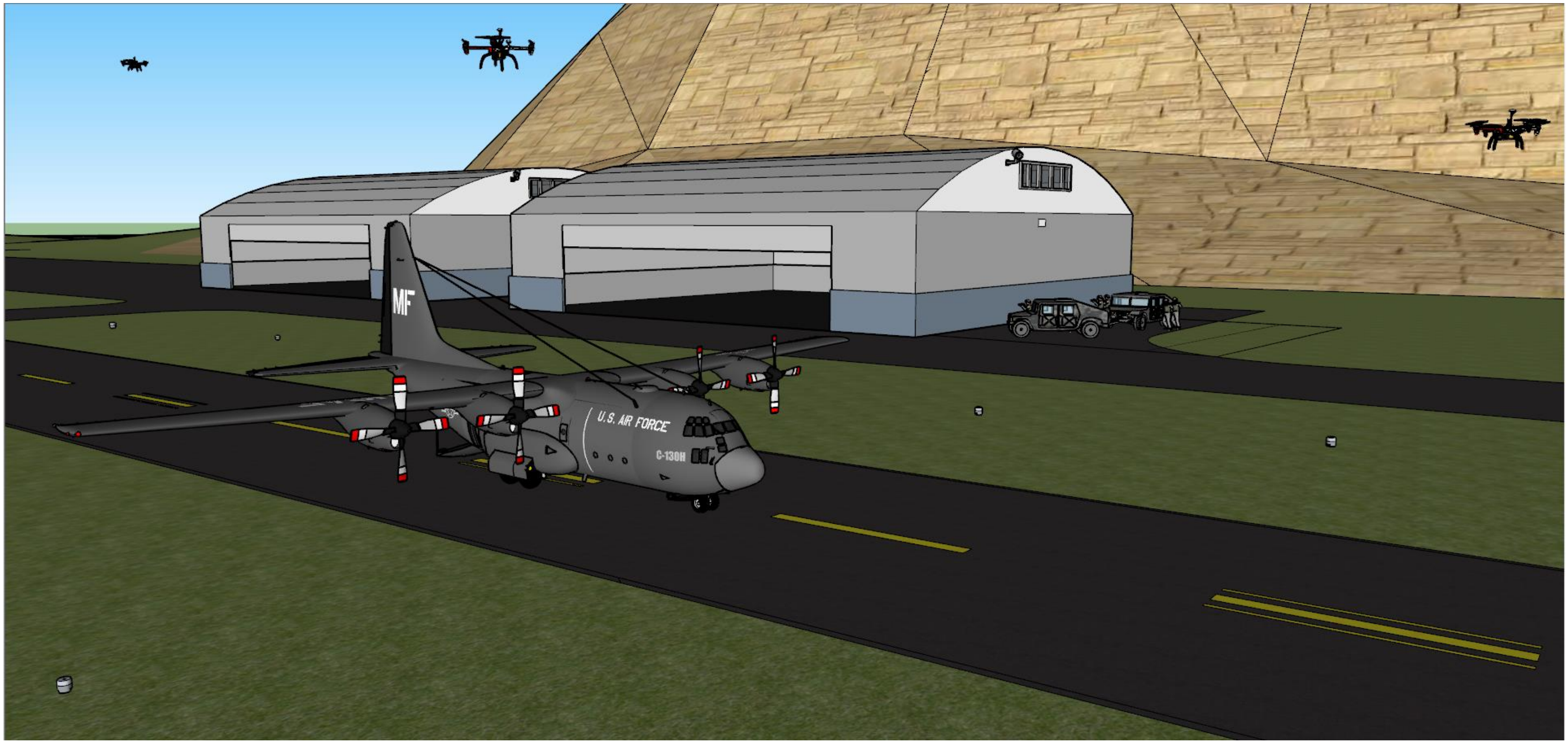
- Scope is a function of human roles in the adaptive CPHS
- Need to ensure that the adaptive CPHS is operating within human cognitive constraints while capitalizing on human strengths
  - effects of cognitive load, fatigue, and attention level on error rates
- Key research questions:
  - What aspects of humans to represent for specific problem contexts?
  - Is there a methodological basis to determine an appropriate sparse representation of a human?
  - At what level should human (model) be incorporated in feedback loop (e.g., on-the-loop, in-the-loop, inside controller, inside system model)?
  - What modeling approach (e.g., HMM, MAU decision models, optimal control model) best fits a particular problem context?

Machine Learning

- Different ML techniques for different uses in Adaptive CPHS
- **Reinforcement Learning:** Discover unidentified environment states from observations during mission execution
- **Supervised Learning:** Capture human preferences offline from simulated task performance in different contexts
- **Unsupervised Learning:** Discover behavior patterns from data in different contexts

Prototype System Implementation

Illustrative Scenario: Perimeter Security of C-130 Aircraft



Perimeter Security of C-130 Aircraft

- Multiple QCs with downward-facing video cameras
- Building-mounted video and Long Wave Infrared (LWIR) cameras
- QCs change and hold position and altitude that maximizes a collective fitness function (FF)
  - FF reflects perimeter coverage
  - QCs can change position and altitude to maximize FF
- **Contingencies1:** low battery causing QC to land; loss of QC

Resilience responses: reposition remaining QCs to restore coverage; launch backup QC if repositioning does not work

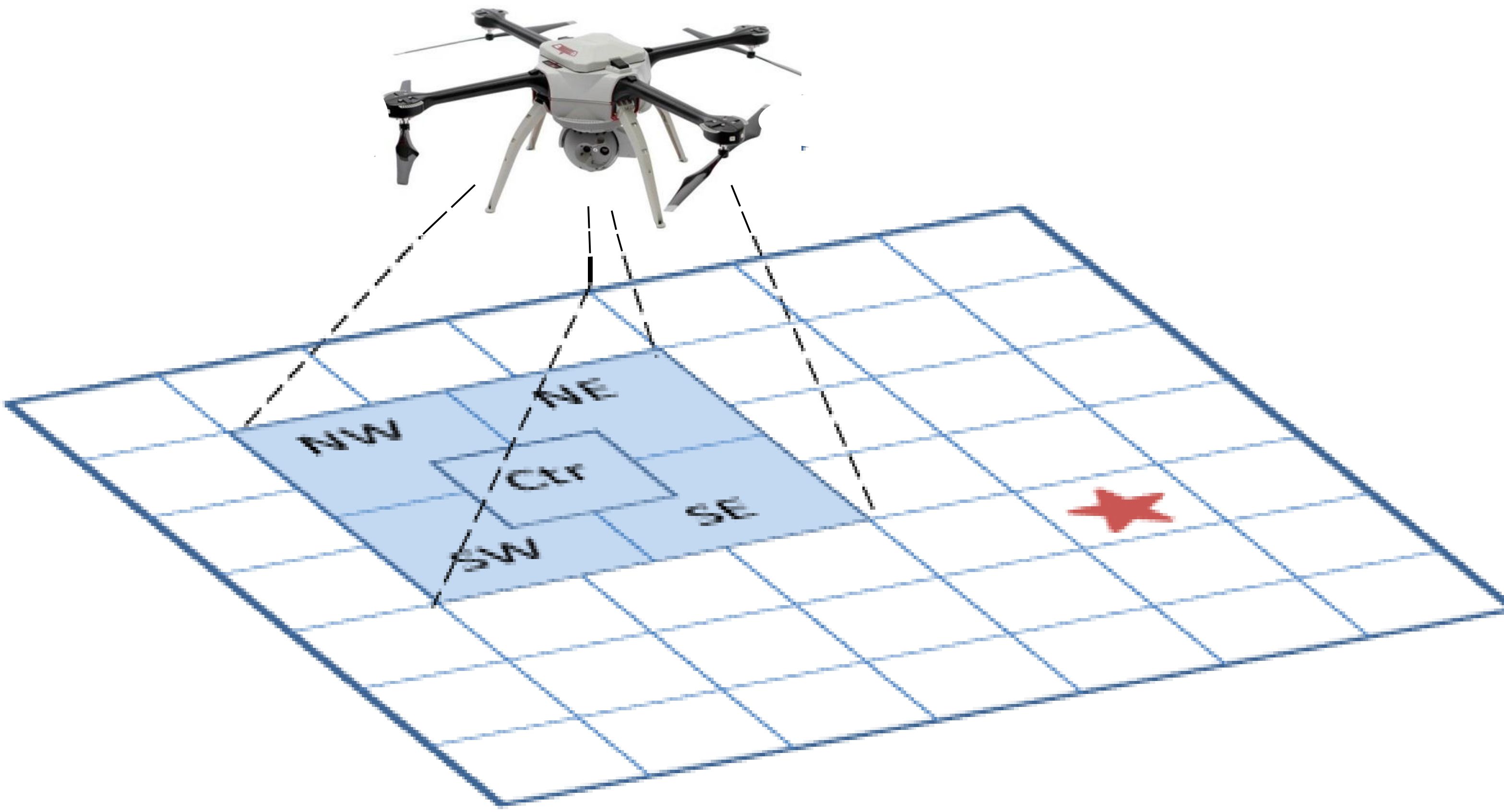
- **Contingencies2:** Intruder in the secured field

Resilience responses: collect motion data and extract features; use an ML technique to classify foes from friends; respond autonomously while keeping commander in the loop, or request commander intervention to respond

Scenario Segments

- **Segment #1:** Navigate to target area with partial observability
  - —account for uncertainty and adjust route with observations
  - —monitor system health during route to target area
- **Segment #2:** Maximize perimeter coverage with available static and mobile sensors
  - —detect intrusion and notify commander (intrusion location, action)
  - —Request commander to confirm intruder (if ambiguous to autonomous agent)
  - —tune algorithm parameters based on human’s response
  - —continually adjust location and altitude of remaining QCs to restore perimeter coverage upon loss of QC
  - —if coverage cannot be restored, request launch of backup QC

Segment #1: Navigating to Target Area



- QC Position relative to a reconnaissance target (red star) and FOV (blue)
- Employ appropriate models to cope with partial observability

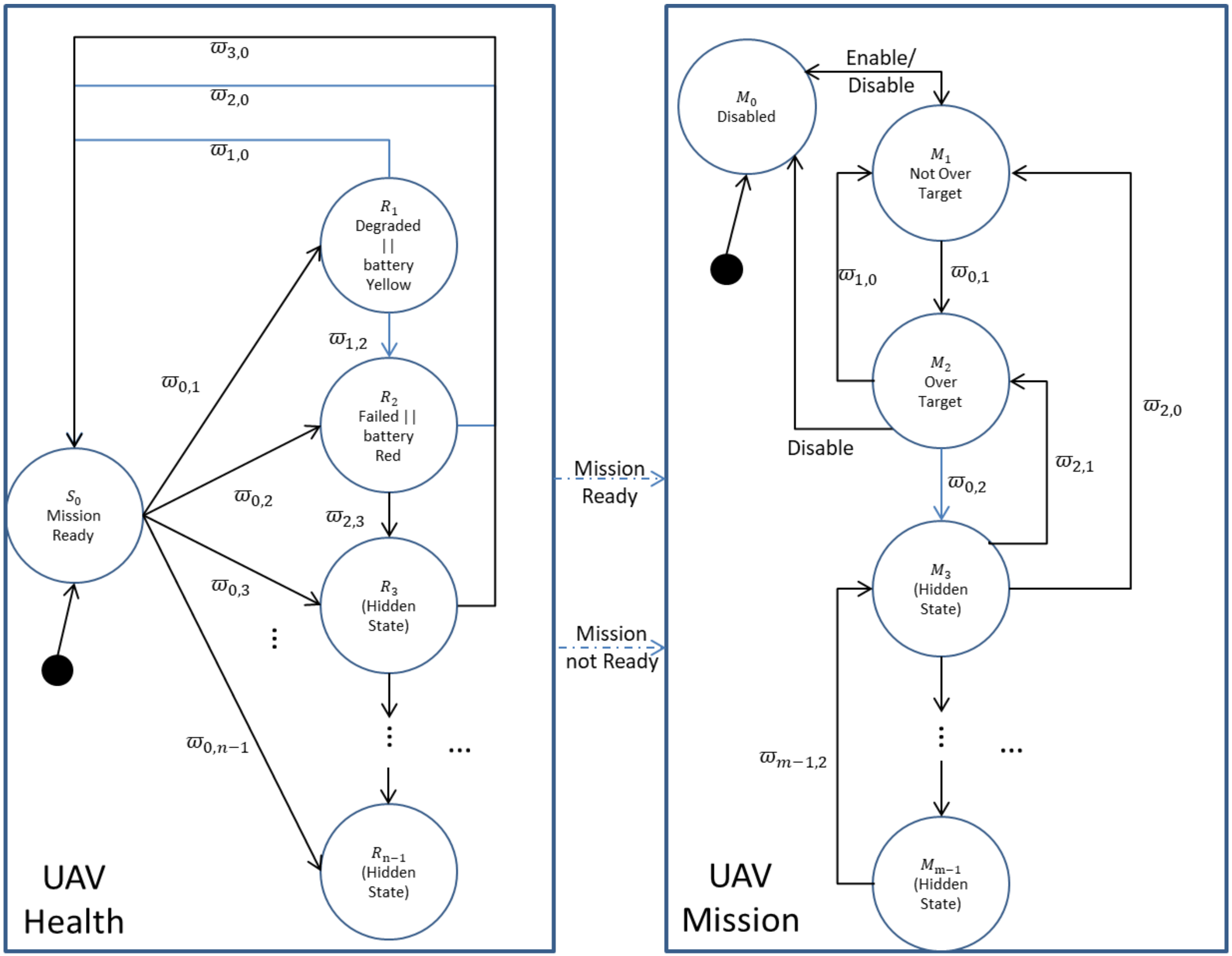
SYSTEMS ENGINEERING RESEARCH CENTER

Exemplar Contracts

USC Viterbi School of Engineering Systems Architecting and Engineering

1. ¬overTarget && healthy && batteryGreen → move_to_target
2. ¬batteryRed && degraded || batteryYellow → move_to_base
3. batteryRed || failed → land
4. unknownHealth || unknownBattery → move_to_base
5. overTarget && CTR && healthy → takeImages & hover
6. overTarget && NW && healthy → takeImages & move SE
7. overTarget && NE && healthy → takeImages & move SW
8. overTarget && SW && healthy → takeImages & move NE
9. overTarget && SE && healthy → takeImages & move NW

Simplified POMDPs: Health and Mission Models



Segment #2: Maintain Perimeter Security

- Assure coordinated response by team members
  - —Human-in-the-loop response for previously unseen situations
  - —preplanned protocols between QCs for known patterns
- Continually adapt coverage in the face of disruptions
  - —monitor and share health status of QCs (battery, comm links)
  - —monitor disruptions (e.g., loss of a QC due to malfunction, low battery)
  - —respond to disruptions (e.g., adjust locations and altitudes to restore coverage, request backup)

Fitness Function to Maximize Coverage

- Discretize perimeter area into tiles
  - goal: one or two cameras observing each tile (more than two is redundant and should not be rewarded)
  - closer coverage (higher resolution of imaging) is better
- Simple algorithm: for each tile and each camera
  - if tile is visible from camera, sum up 1/(distance to camera)
  - cap each tile sum to avoid rewarding redundant coverage
- Future improvements to fitness function
  - reward views from widely separate camera locations to maximize available information e.g. stereo
  - account for different camera capabilities e.g. higher resolution on fixed building cameras

Multi-Level Coverage Algorithm

- Multi-agent control
  - multiple QCs move independently to maximize their contributions to the fitness function
  - resulting cooperative motion works to increase fitness

- Adaptation to changing circumstances
  - e.g., one QC crashes, or has low battery power and needs to land
  - other QCs move to restore loss of coverage

- Human-in-the-loop
  - if multi-agent control proves to be insufficient to provide adequate coverage, human intervention is requested
  - it is up to the human to act, e.g. launch additional QC, or request help from higher headquarters
  - if CPS cannot distinguish intruders from friendly troops, human intervention is requested

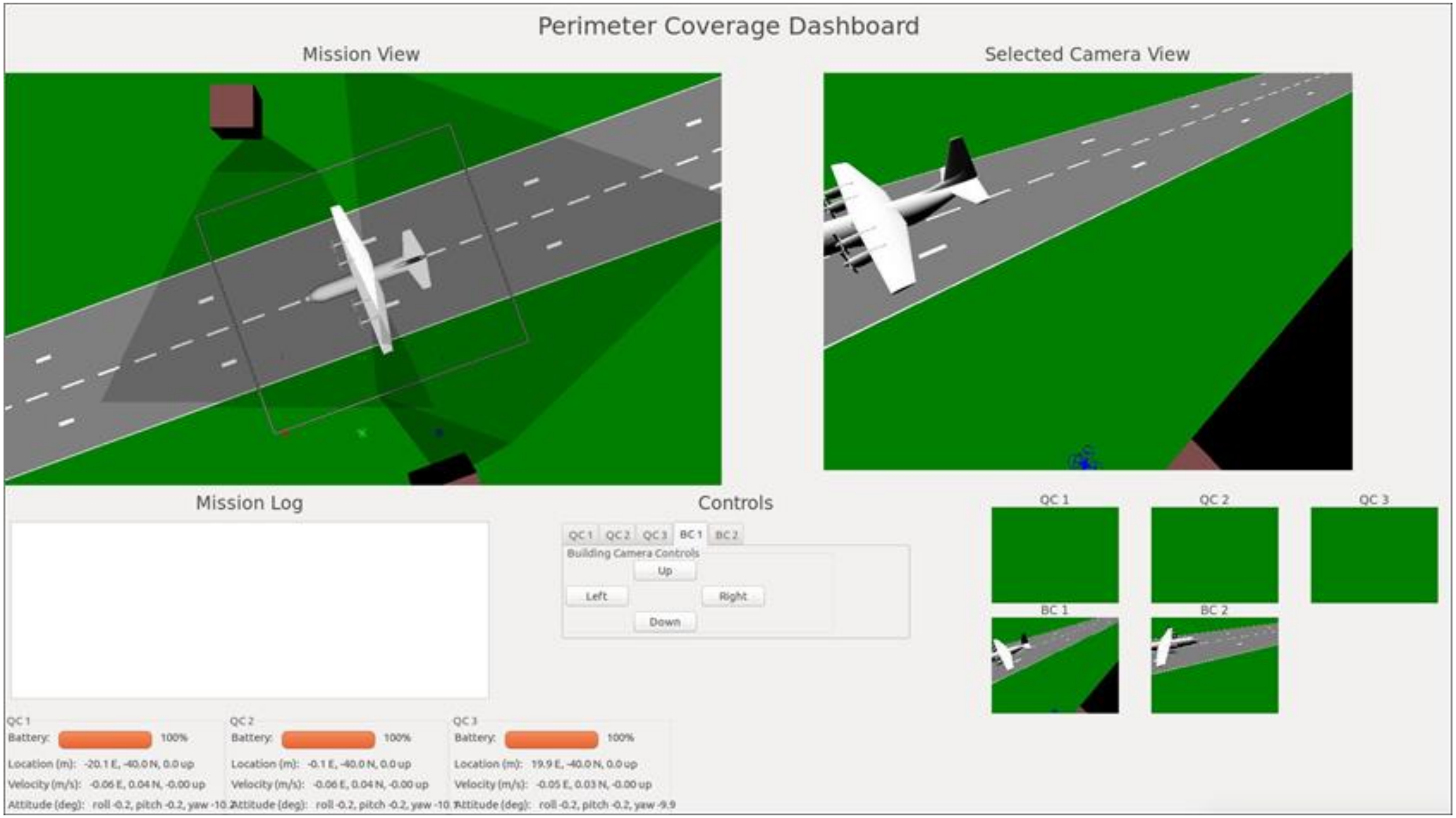
Smart Dashboard

- Content and composition based on METT-TC ontology
  - Concepts and attributes of mission, enemy, troops, terrain (and weather), time available, and civilian population
  - Relationships between concepts
- Enables scenario setup, execution monitoring, visualization, resource allocation, control and supports supervised machine learning
  - Intrusion detection: detection of threats approaching aircraft/airfield perimeter
  - Monitoring of enemy combatants and / or unidentified moving objects (e.g., animals)
  - Threat tracking using available cameras (mobile, building-mounted)
  - Motion tracking and feature extraction of any moving objects
  - Agent-in-the-loop learning from human supervisor

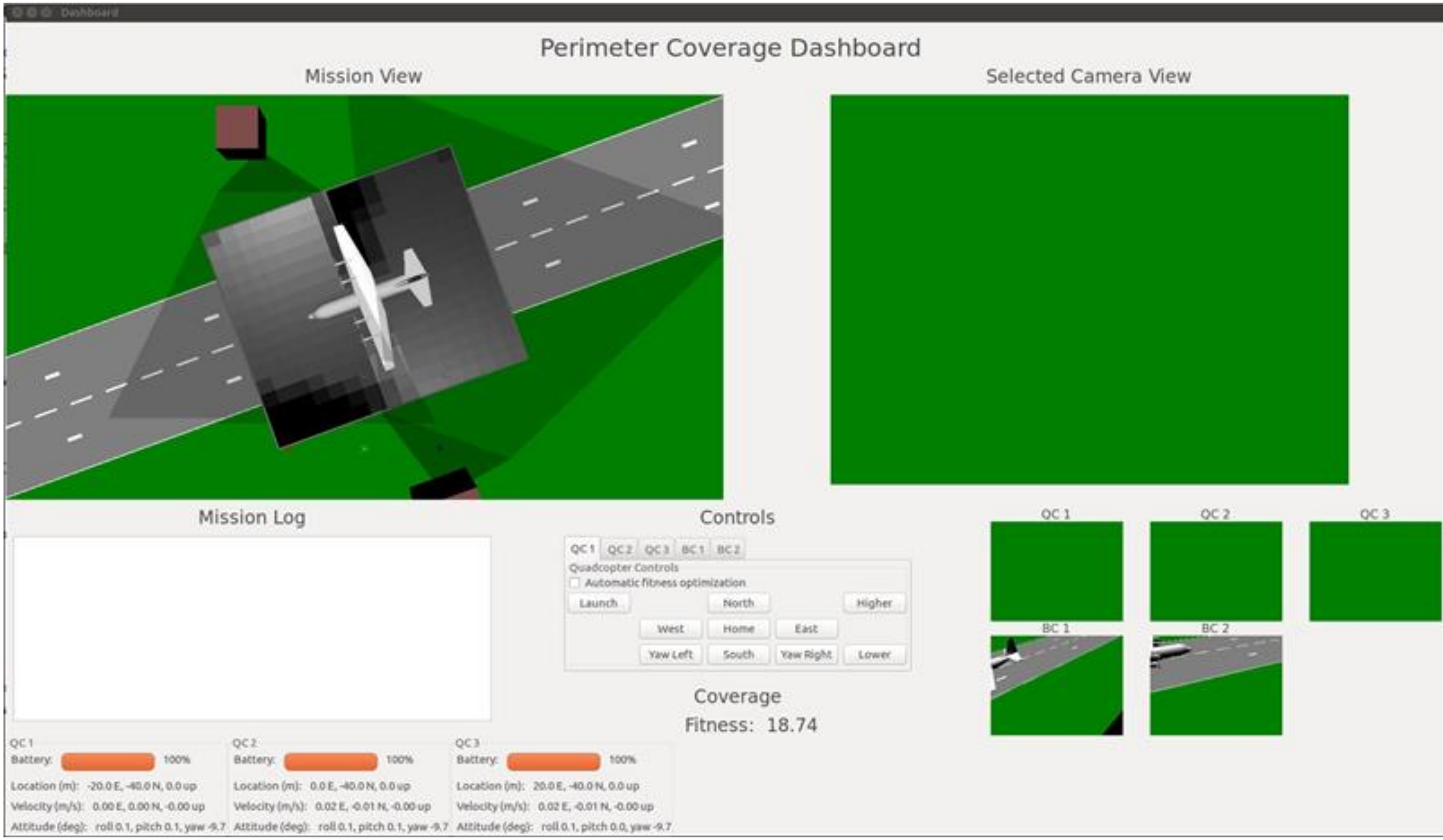
Smart Dashboard Prototype

- Purpose
  - monitoring and control of multiple simulated and physical vehicles
- Underlying technologies
  - ontology-driven customizable interface
  - dronekit platform with visualization facilities
  - quadcopters (hardware) and quadcopter simulation models
  - quadcopter planning and decision-making model
  - quadcopter controller
  - decision tree for motion classification
- Key capabilities
  - simulated vehicles exhibit behavior of physical vehicle
  - same commands used to control vehicle models and the physical vehicles (quadcopters)
  - can switch from simulated to physical vehicles, and vice versa

Perimeter Coverage Scenario: Simulator Dashboard

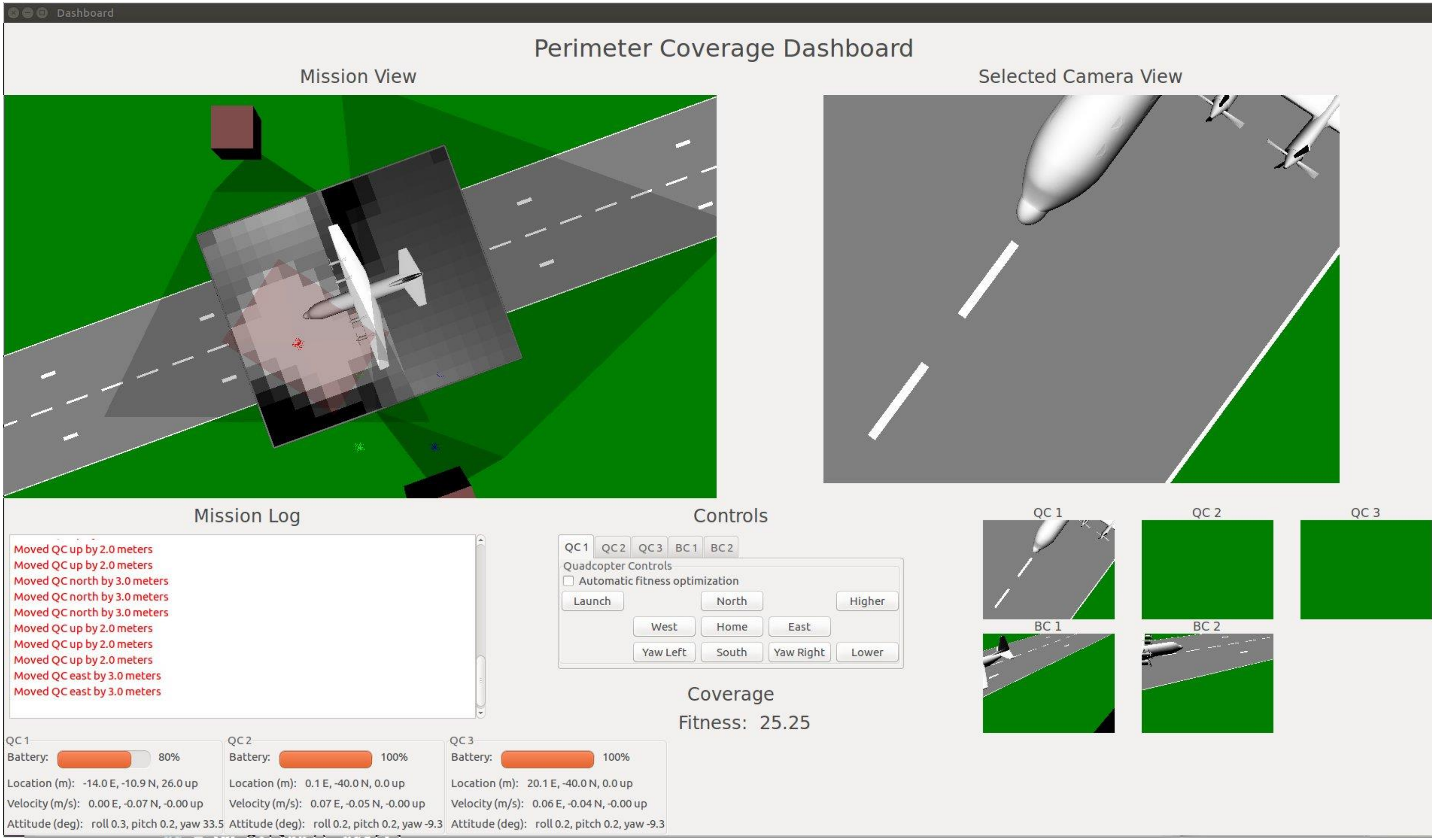


Dashboard Showing Coverage Area

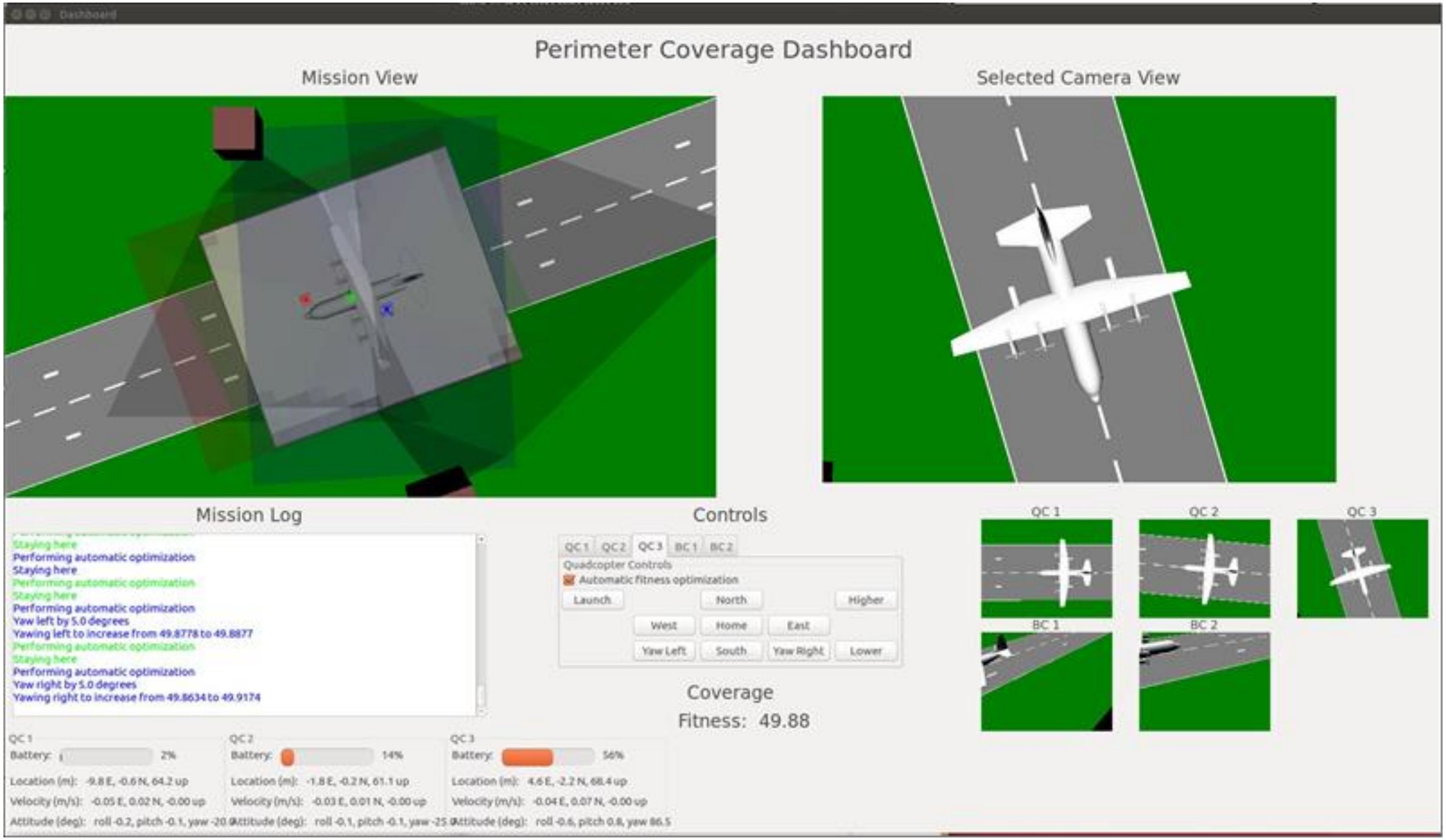


Dashboard with One QC During Optimization of Fitness Function

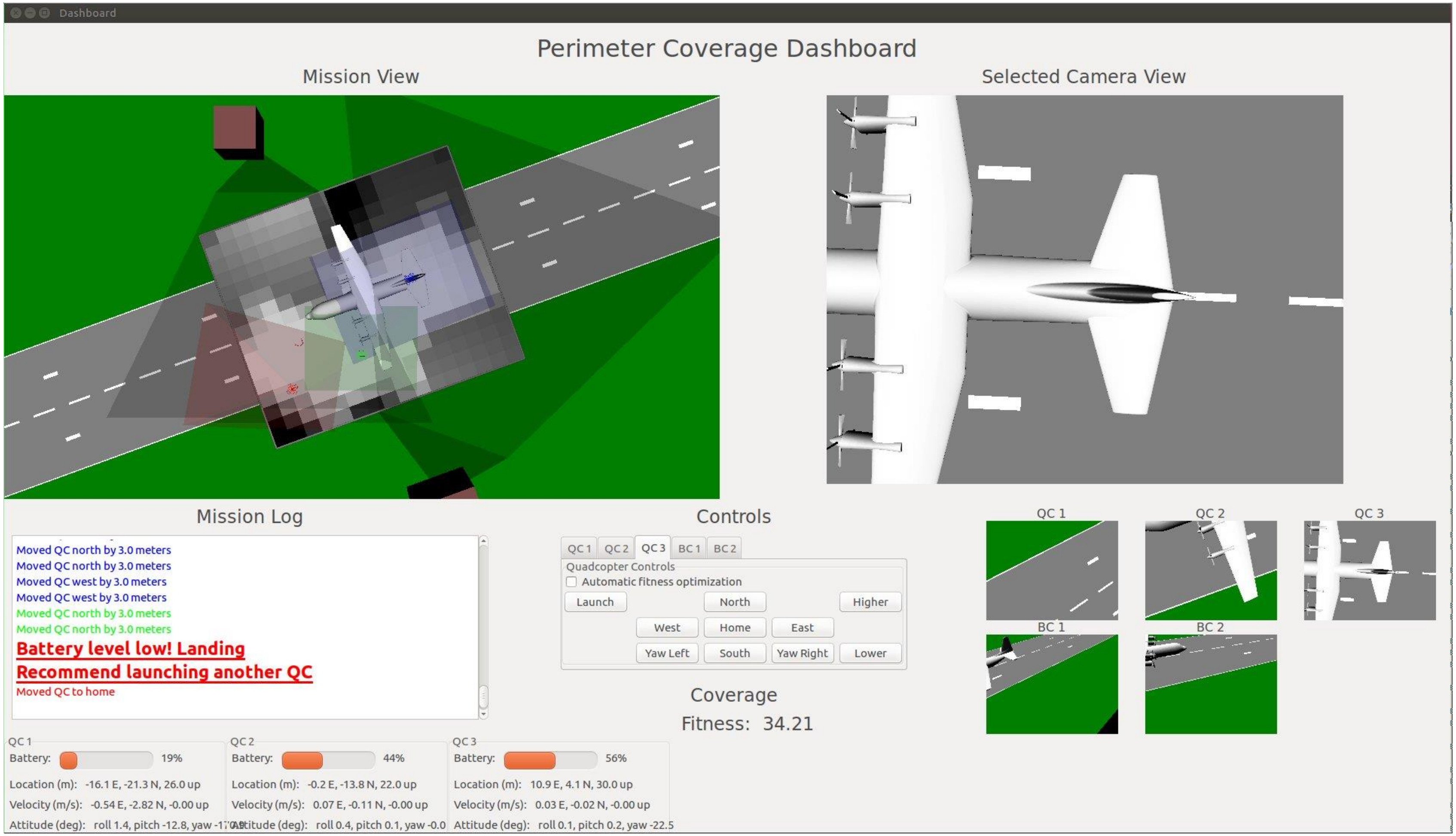
Dashboard Showing Optimal Location for a Single Quadcopter



Dashboard Showing Optimal Location for Three Quadcopters



Dashboard Showing 3 Flying QCs with One Low on Battery and Ready to Land



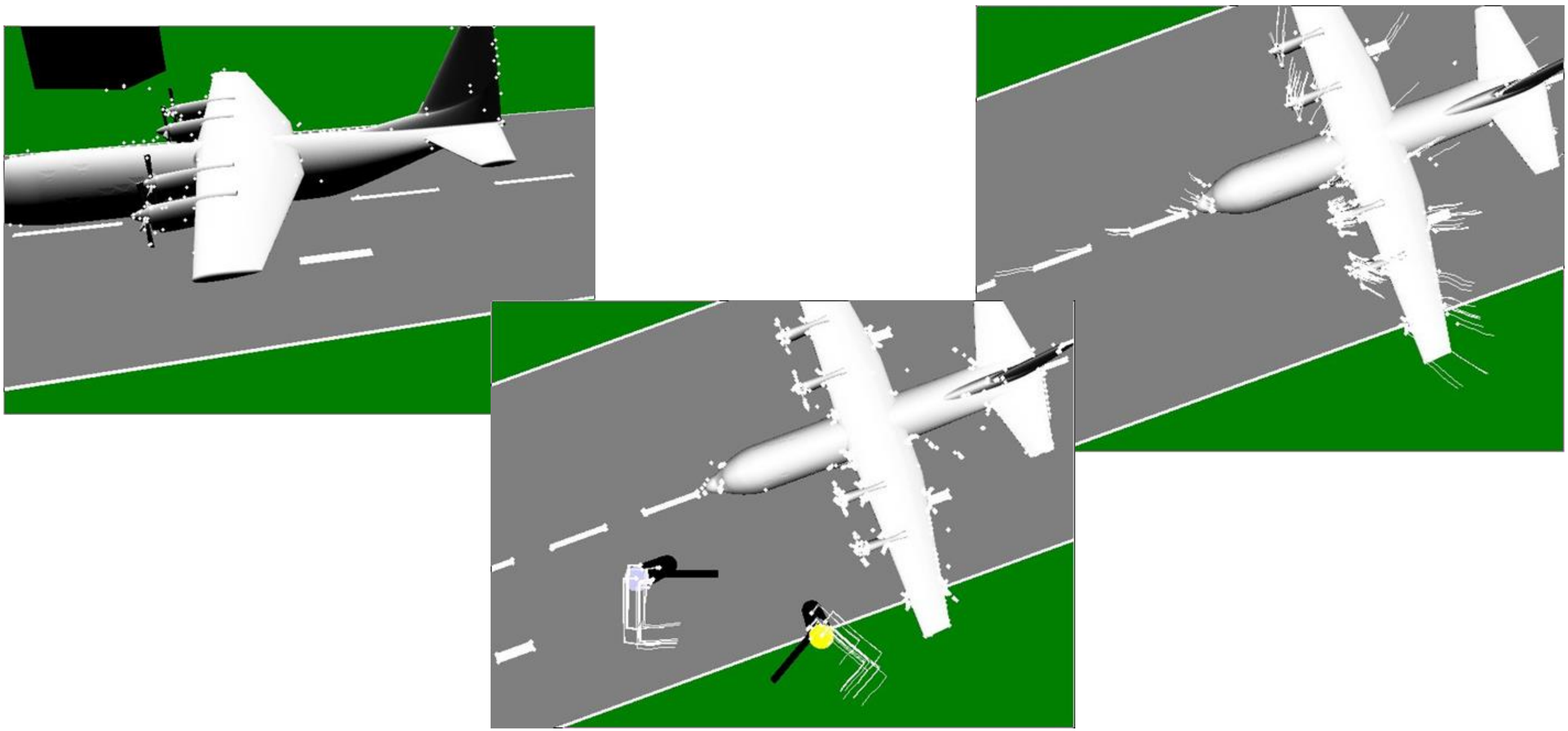
Multi-Asset Control Approach

- **Problem:** control the collection assets (UAVS and fixed cameras) to optimize multi-sensor coverage of the aircraft perimeter

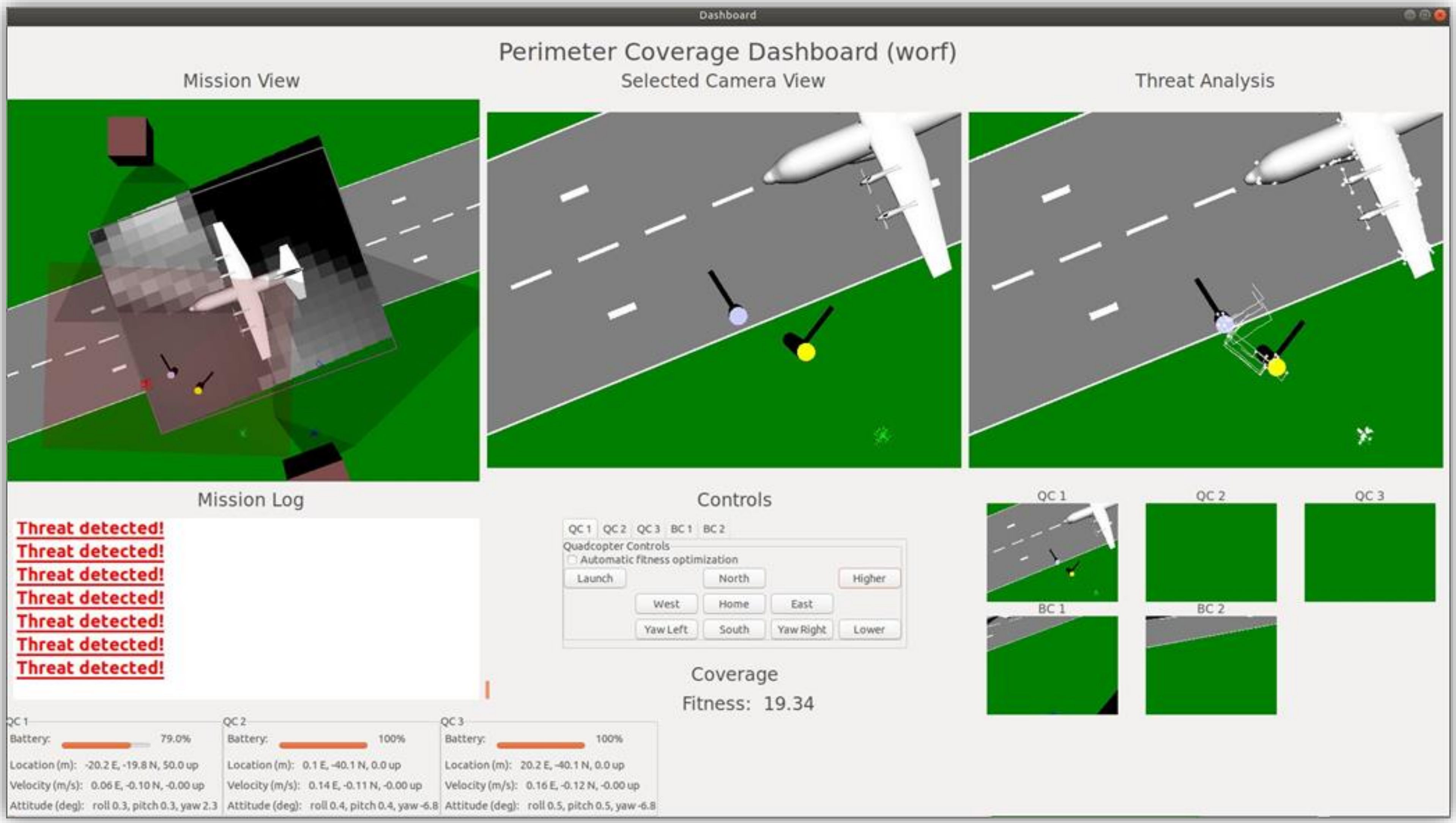
- **Fitness function** to characterize perimeter coverage
  - Employs multiple levels to flexibly allocate and move assets to optimize coverage:
    - Multi-agent control
    - Adaptivity
    - Human-in-the-loop

Multi-Asset Control Approach

- **Motion detection:** image analysis using open-source OpenCV computer vision software library
  - **Feature identification:** SIFT (Scale-Invariant Feature Transform)
  - **Optical flow:** Lucas-Kanade (LK) pyramid method



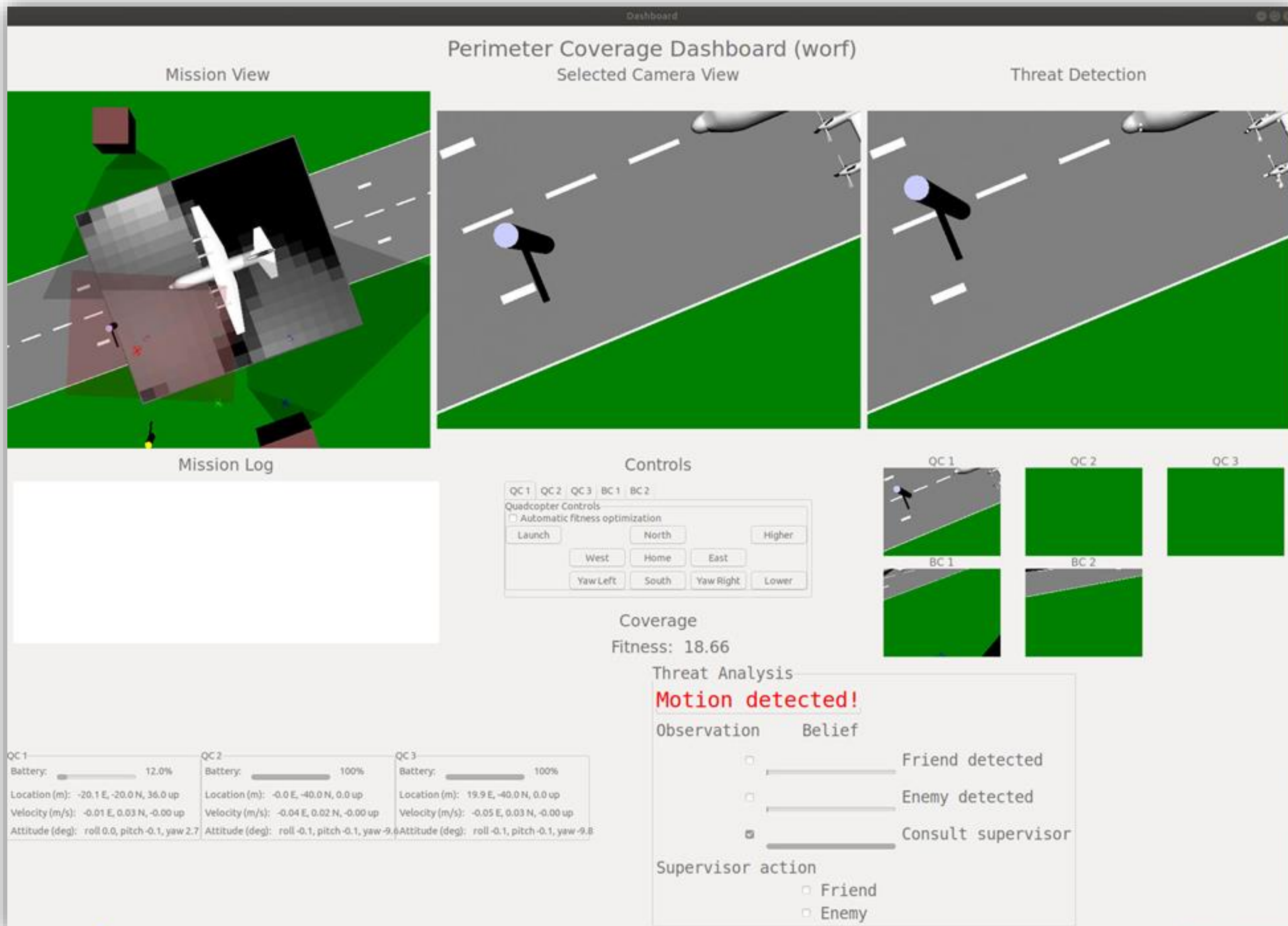
Threat Analysis View



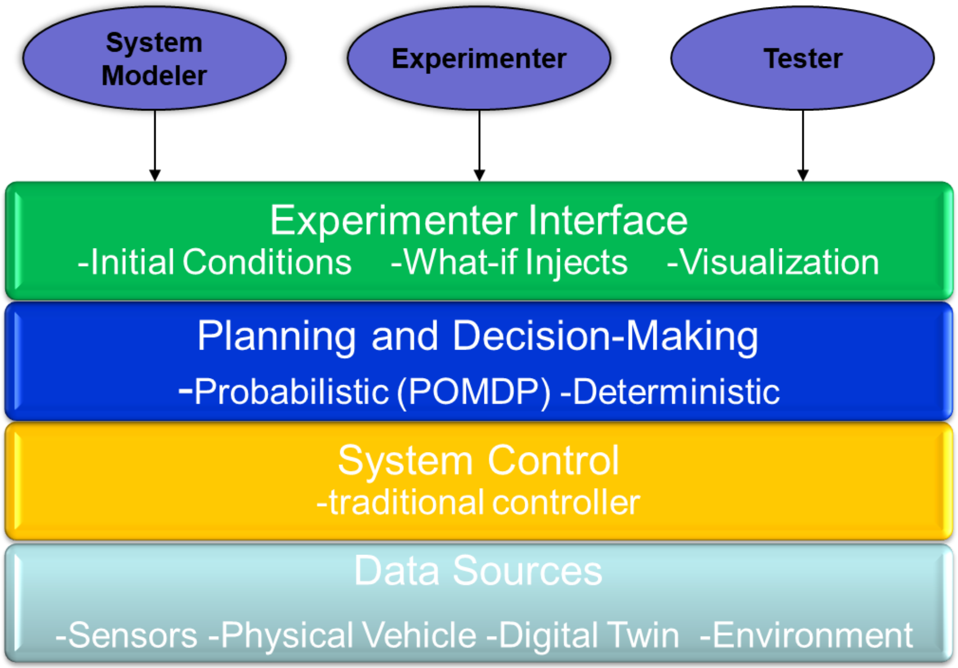
Agent-In-The-Loop

- **Agent-in-the-loop** learning from a **human supervisor**
  - Integration of **decision tree** within simulation dashboard
    - Invoked whenever a moving object is seen within the field of view of the active camera
    - Decision tree analysis of moving object produces three possible outcomes: friendly, enemy, or consult human supervisor
    - Data collected from human supervisor is used to tune the decision tree in a batch off-line learning mode
      - E.g. Classification accuracy was 0.567 initially that increased to 0.98 after tuning the parameters based on collected data.

Agent-In-The-Loop



Testbed Architecture



- Developed concurrently with prototype system
- Currently supports system modeling, model verification, system behavior simulation, threat simulation
- Simulations runs on separate machines within a distributed, networked architecture

Implementation:
Distributed Simulation Architecture

- C-130 perimeter defense sim: distributed on 3 computers:
  - World server, Perimeter defense computer, Enemy computer
- World server
  - maintains state of all entities in the world
  - runs a continuous dynamic simulation of all QCs
- Perimeter defense computer
  - runs dashboard which controls the QCs
- Enemy computer
  - runs the enemy dashboard which controls enemy soldiers
- 2 dashboard computers communicate with world server to
  - obtain entity state (x,y,z,) to display all entities on screen
  - send motion commands to move entities they control
  - sensors modeled by defense dashboard

SYSTEMS ENGINEERING RESEARCH CENTER

Technical Findings

- Key problem in implementing hybrid models
  - resolving mismatch between planning & decision-making layer and vehicle control layer
- Mismatch resolution
  - ensure that propagated commands from PDM layer to controller do not violate physical and regulatory constraints
  - propagate execution constraints from control layer to PDM layer for PDM layer to account for when issuing commands
  - incorporate heuristics (e.g., priorities, region of influence) to resolve conflicts and simplify computation
- POMDP and vehicle controller work on different time scales
  - dynamics model runs every 0.01 seconds (accuracy)
  - POMDP runs slower (high level decisions/commands)
  - waypoint navigation problem - minimize response time to action
  - ideal sampling period for POMDP determined experimentally

Overall Findings and Summary

- Concurrent creation of dashboard and testbed was a plus
  - enabled rapid iterations on dashboard design
  - dashboard is an essential component of adaptive CPHS and debugging aid
  - enabled early demos of evolving system to DoD, SERC, and transition partner
  - gained valuable experience to create MBSE testbed for SERC community
- Model type and complexity are a function of problem context
  - size of system state-space
  - knowledge of system states
  - environment observability and uncertainty
- Even a relatively simple fitness function yielded promising results
  - can develop more sophisticated fitness functions in the future
- What next?
  - incorporate physical system data into virtual model to create Digital Twin
  - enhance verification and testing - expand MBSE coverage of system life cycle

Takeaways

- DoD systems in 21st century need to be resilient and operate safely in uncertain, partially observable, potentially hostile environments
- Adaptive CPHS, an example of a 21st century system, poses unique modeling, analysis and realization challenges
- Adaptation implies not only changes in model parameters but also modeling construct ("principle of proportional complexity")
- Distributed simulation well-suited to implementing adaptive CPHS
- Approach successfully applied to perimeter security of military aircraft
- Successfully demonstrated supervisory and autonomous control of QCs
- Demonstrated value of a context-aware dashboard in maximizing situation awareness and exploring what-ifs
- **Successful Transition:** Research product along with product of RT-166/210 transitioned to The Aerospace Corporation's MBSE team

Relevant Journal Publications

- Madni, A.M., Sievers, M. and Madni, C.C. Adaptive Cyber-Physical-Human Systems: Exploiting Cognitive Modeling and Machine Learning in the Control Loop, *INSIGHT*, 21,3, (87-93), 2018.

- Madni, A.M. and Sievers, M. "Model Based Systems Engineering: Motivation, Current Status, and Research Opportunities," *Systems Engineering*, 20th Anniversary Issue, vol. 21, issue 3, pp. 172-190, 2018.

- Madni, A.M., and Madni, C.C. Architectural Framework for Exploring Adaptive Human-Machine Teaming Options in Simulated Dynamic Environments. *Systems*. 2018; 6(4):44.

- Madni, A.M., and Madni, C.C. Lucero, S.D. Leveraging Digital Twin Technology in Model-Based Systems Engineering. *Systems*. 2019; 7(1):7.

- Madni A.M. and Purohit S. Economic Analysis of Model-Based Systems Engineering. *Systems*. 2019; 7(1):12.

Azad M. Madni

- Professor, Astronautical Engineering, University of Southern California
- Executive Director, Systems Architecting and Engineering Program
- Director, Distributed Autonomy and Intelligent Systems Laboratory
- Founder and CEO, Intelligent Systems Technology Inc.
- INCOSE Fellow, Pioneer and Founder
- Life Fellow, IEEE; Fellow, AAAS; Fellow, AIAA; Life Fellow, SDPS; Life Fellow, IETE
- Ph.D., M.S., B.S. in Engineering, UCLA; Graduate of Stanford's Executive Program
- **Research Interests**: Formal and Probabilistic System Modeling; Resilient Cyber-Physical-Human Systems; Interactive Storytelling in Virtual Worlds, Intelligent Systems Engineering
- **2019 Awards and Honors**
  - *2019 Presidential Award from Society of Modeling and Simulation International*
  - *2019 AIAA/ASEE Leland Atwood Award* for excellence in aerospace engineering
  - *2019 ASME CIE Leadership Award* for advancing use of computers in engineering
  - *2019 INCOSE Founders Award* for increasing global awareness of INCOSE
  - *2019 EC William B. Johnson International Inter-Professional Founders Award*
  - *2019 OCEC Prestigious Pioneering Educator Award*
- **Recent Books**
  - Madni, A.M., Boehm, B. et al. (eds.) *Disciplinary Convergence: Implications for Systems Engineering Research*, Springer, 2018.
  - Transdisciplinary Systems Engineering: Exploiting Convergence in a Hyper-Connected World (foreword by Norm Augustine) Springer, 2017
  - Tradeoff Decisions in System Design (foreword by John Slaughter), Springer, 2016
  - Madni, A.M. and Boehm, B. (eds), *"Engineered Resilient Systems: Challenges and Opportunities in the 21st Century,"* Procedia Computer Science 28 (2014), ISSN 1877-0509, Elsevier, 2014

Thank You