Verifying DART Systems (DART)

Presentation to CERDEC
Sagar Chaki
January 15, 2015

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
16 JAN 2015	**N/A**	

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Verifying DART Systems (DART)	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
Chaki /Sagar	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited.

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**SAR**	**14**	

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM-0002080

Driving Vision

DARTs coordinate physical agents in an uncertain and changing physical world.

- Coordination – physical agents
- Timeliness – safety critical
- Resource constrained - UAVs
- Sensor rich – sensing physical world
- Intimate cyber physical interactions
- Automated adaptation to physical context and rational adversaries
- Computationally complex decisions

Coordination, adaptation, and uncertainty pose key challenges for assuring safety and mission critical behavior of distributed cyber-physical systems.

The DART project uses develops and packages sound techniques and tools for engineering high-assurance distributed CPS.

DART Assurance Today

Currently validated via testing

- Low coverage, late in development

Rigorous & exhaustive analysis provides higher assurance

- Non-compositional V&V does not scale
- Probabilistic & deterministic requirements

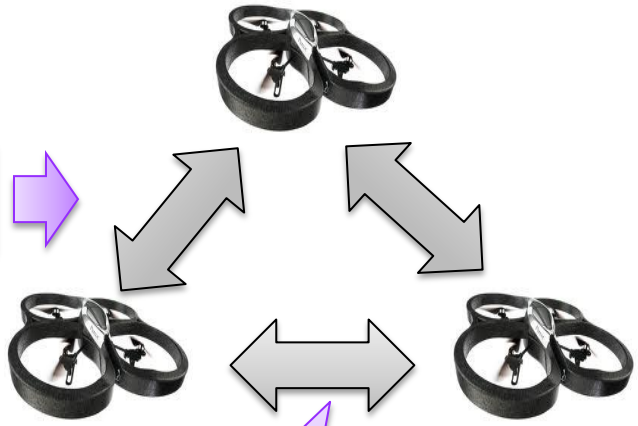
Goal: Develop new theories, analyses and tools to engineer high-assurance DARTs with evidence of correctness

DART in a Nutshell

1. Enables compositional and requirement specific verification
2. Use proactive self-adaptation and mixed criticality to cope with uncertainty and changing context

1. ZSRM Schedulability (Timing)
2. Software Model Checking (Functional)
3. Statistical Model Checking (Probabilistic)

System + Requirements (AADL + DSL) → **Verification** → **Code Generation** →

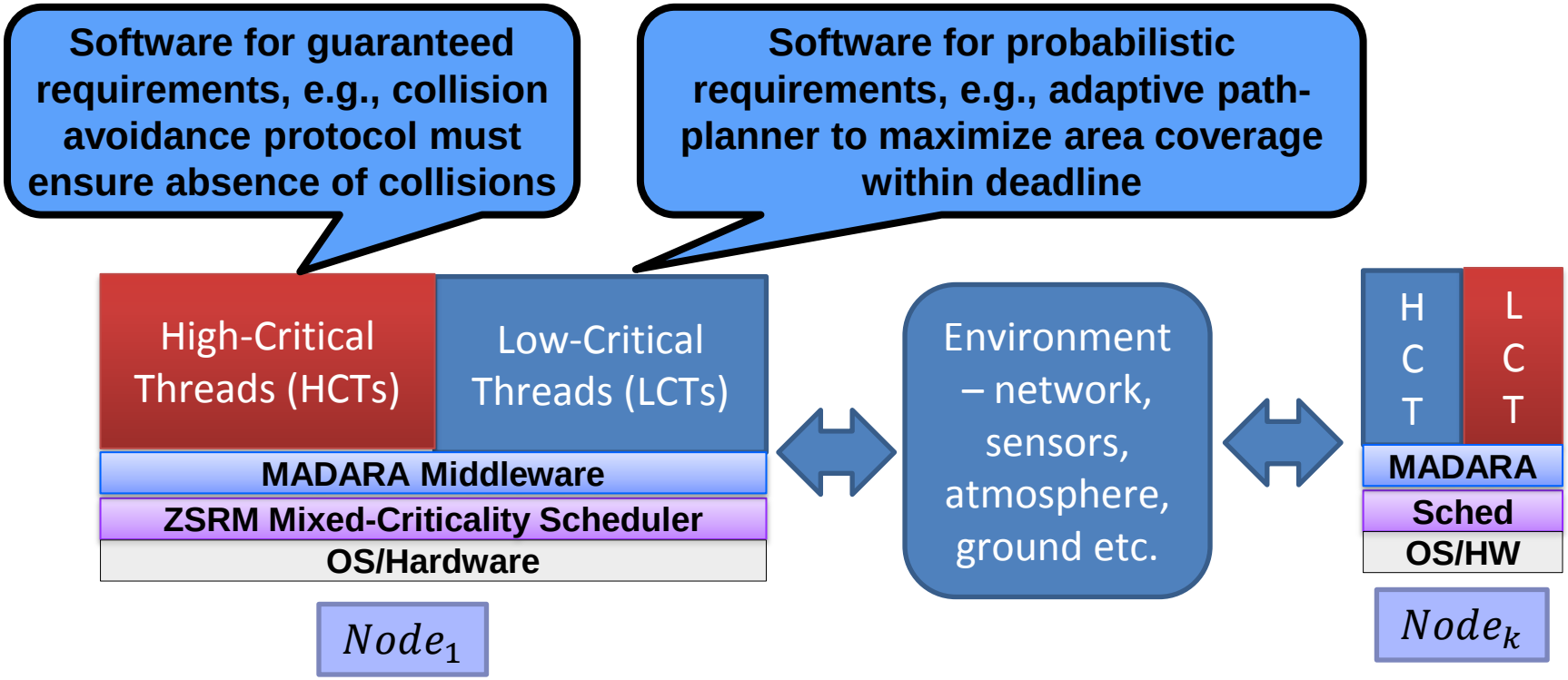


1. Middleware for communication
2. Scheduler for timing contracts
3. Monitor for functional contracts

Demonstrate on DoD-relevant model problem (DART prototype)

- Engaged stakeholders
- Technical and operational validity

DART High-Level Architecture



Research Thrusts

- Proactive Self-Adaptation
- Statistical Model Checking
- Real-Time Schedulability
- Functional Verification

Validation Thrusts

- Model Problem
- Workbench

Roadmap & Foundations

Thrust Area	Jan	Apr	Jul	Oct
Proactive Self-Adaptation	Latency-aware Self-Adaptation	CMU/SCS FY14		Disaggregation, Machine-learning
Verification				
Real-Time Schedulability	ZSRM scheduler integrated with DART workbench	HCCPS FY12-FY14		Mixed-criticality among multi-agents & end-to-end OR with Input/Output
Functional Verification	Bounded Model Checking of Synchronous Software	HCCPS FY12-FY14		Unbounded Model Checking of Asynchronous Software
Statistical Model Checking	Crude Monte-Carlo based SMC, applied to simple examples	AFOSR FY14		Heterogeneous Fault Regions and Systems with Non-determinism, HPC Simulation
Workbench	Preliminary version of DSL, Code generation, ZSRM, CBMC, V-REP simulation, simple examples	MCDA FY14		Completed DSL, model problem, ODroid Code Generation, AADL/OSATE, Verification Tools
Coordination (ELASTIC)	Synchronous, multi-agent	GAMS FY14		Asynchronous, multi-agent

Simple Model Problem: Coordinated Protection

Guaranteed Properties

No collision

Best Effort

Defensive perimeter

Resource conservation (e.g., fewest moves)

Adaptation w/ Uncertainty (next step)

Lose of a Protector

Lose of a Leader (new election)

Directional threats (shield formation vs. perimeter formation)

Fleet's Initial State

Assumptions

2D Universe (X by Y matrix)

Perfect communications between agents

Perfect localization for each agent

11 nodes

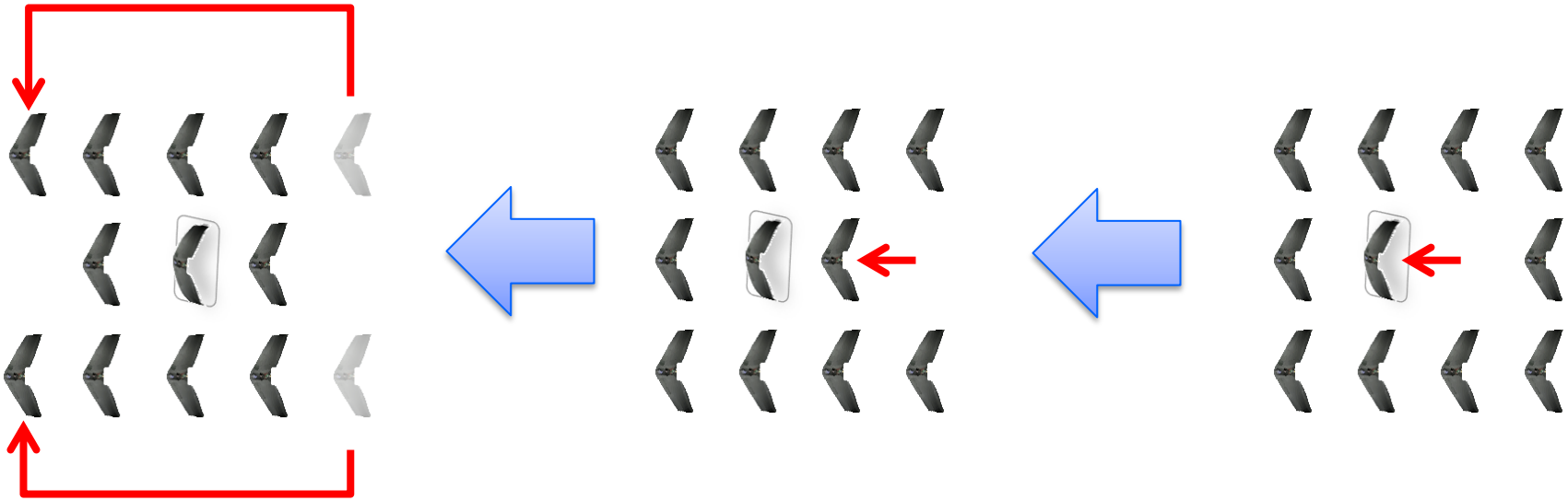
- N_0 is the leader
- N_1 – N_{10} are the protectors

Operation

N_0 moves from $(x, y) \rightarrow (x', y')$

N_1 – N_{10} move to maintain defensive perimeter

Fleet Operation: Defensive Posture



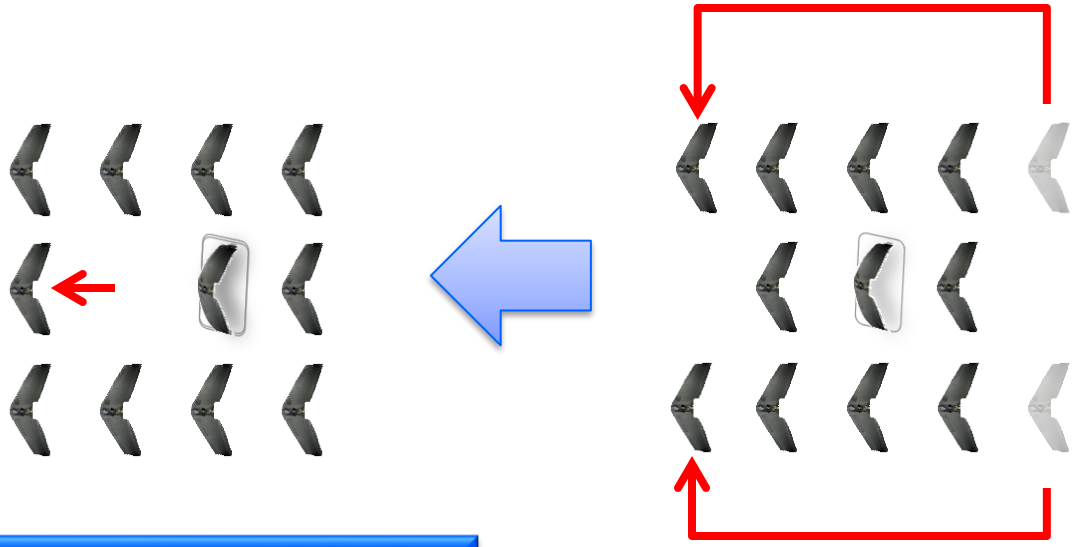
Free guard UAVs move around to front, simultaneously

Rear guard closes gap, leaving two free guard UAVs

N_0 moves from $(x, y) \rightarrow (x', y')$

Coordination needed at each step to avoid collision

Fleet Operation: Defensive Posture

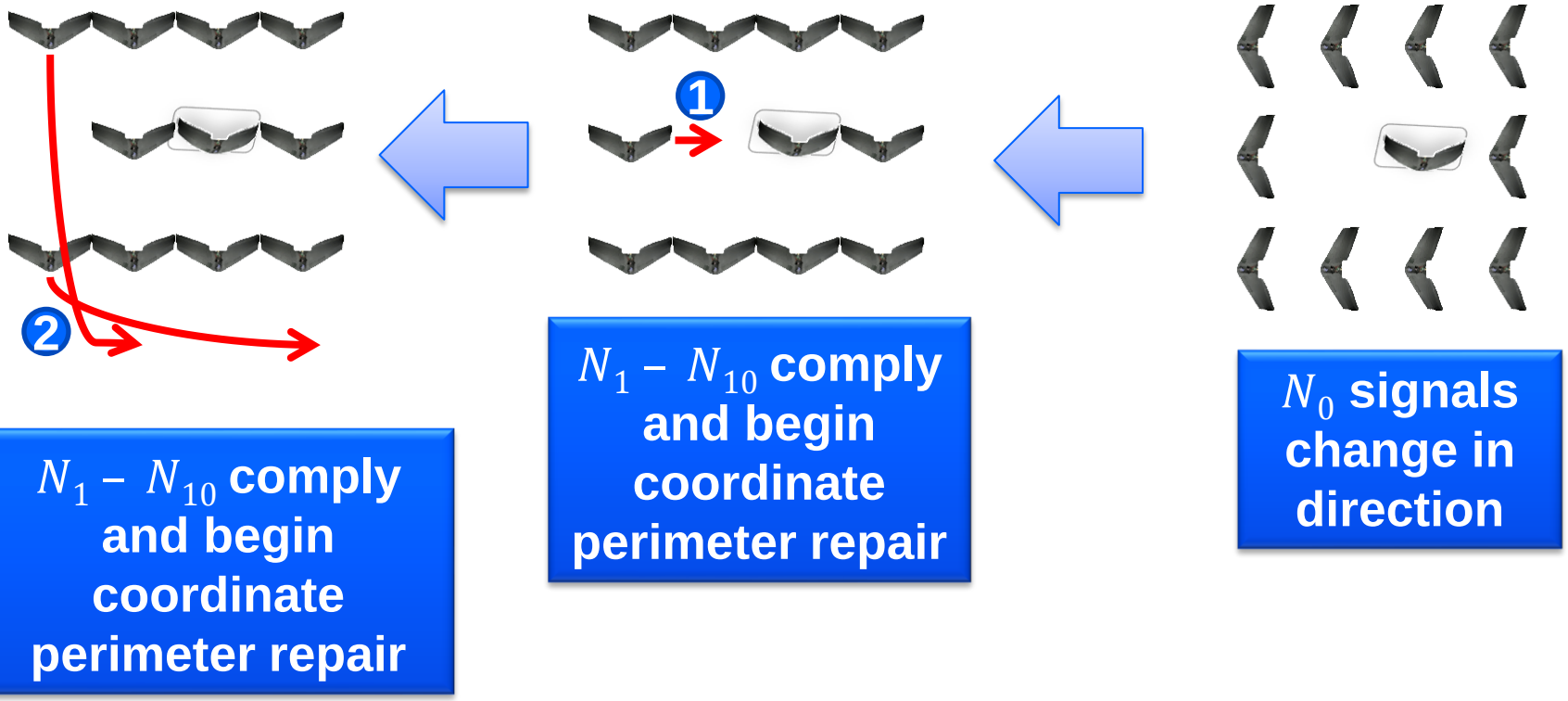


Front guard UAV makes space for N_0 to move forward

Free guard UAVs move around to front, simultaneously

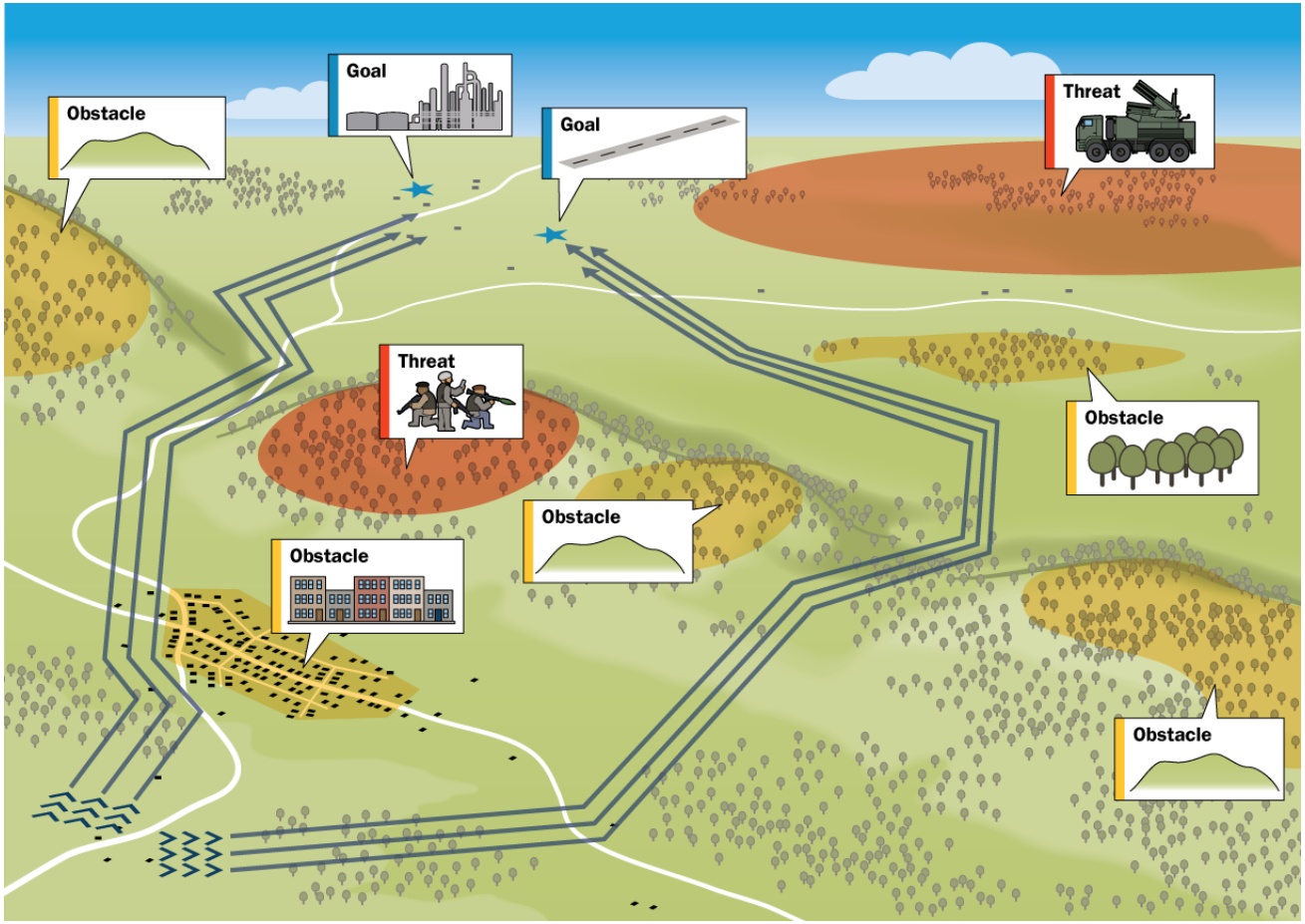
Coordination needed at each step to avoid collision

Fleet Operation: Defensive Posture



Coordination needed at each step to avoid collision

Broader Model Problem



Mission assurance
- Goals
- Objectives

Resiliency
- Design time Verification
  - Guaranteed behavior
  - Best-effort behavior
- Runtime Assurance
  - Critical Timing behavior
  - Coordination
  - Adaptation

QUESTIONS?

Contact Information Slide Format

Sagar Chaki
Senior MTS
SSD/CSC
Telephone: +1 412-268-1436
Email: chaki@sei.cmu.edu

U.S. Mail
Software Engineering Institute
Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612
USA

Web
www.sei.cmu.edu
www.sei.cmu.edu/contact.cfm

Customer Relations
Email: info@sei.cmu.edu
Telephone: +1 412-268-5800
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257