



A Distributed Constraint-Based Algorithm for Dynamic Task Allocation Among UAVs

Bin Yu

**Quantum Leap Innovations
Newark, DE**



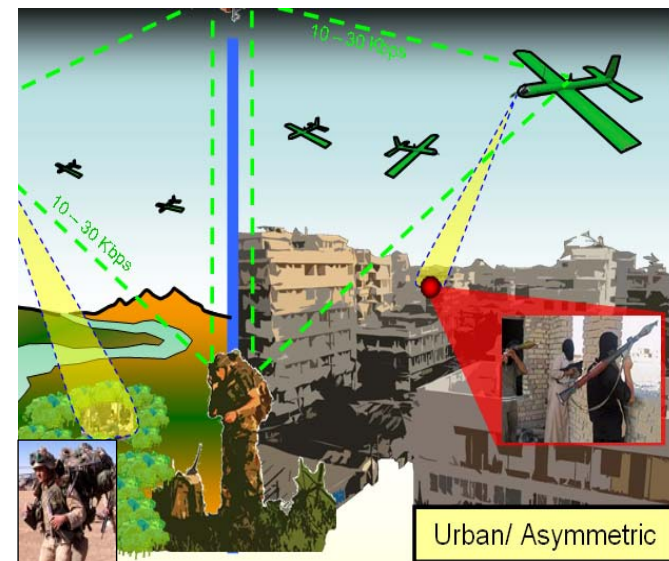
Background of our Company

- Founded in 1999 with ~30 scientist and developers
 - Offices in Newark, DE and Arlington, VA
 - Focus on AI, multiagent systems and distributed modeling & simulation
- Contracts with the governments
 - Office of Naval Research (Integrated Warfighter Biodefense Program)
 - Department of Homeland Security (Unified Incident Command and Decision Support – UICDS) with SAIC
 - ONR Code 30 (Sense & Respond Logistics Program) with BBN
- Research Technology Transfer
 - Condition based maintenance (CBM) for NAVSEA
 - Disease modeling and control for PACOM (Cobra Gold'08)

The Problem: Dynamic Task Allocation

Robust, high-level task allocation among UAVs based on small unit intelligence needs

- Heterogeneous platforms/tasks
- Multiple sensing modalities
- Limited communications
- Limited resources
- Performance guarantees



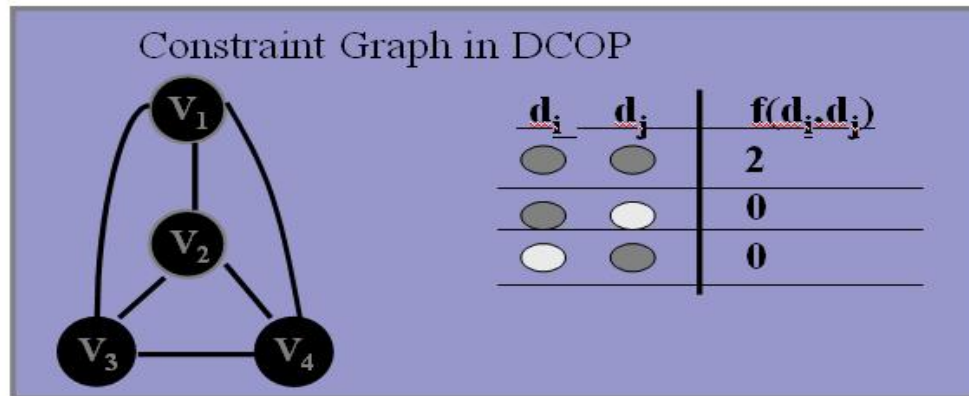
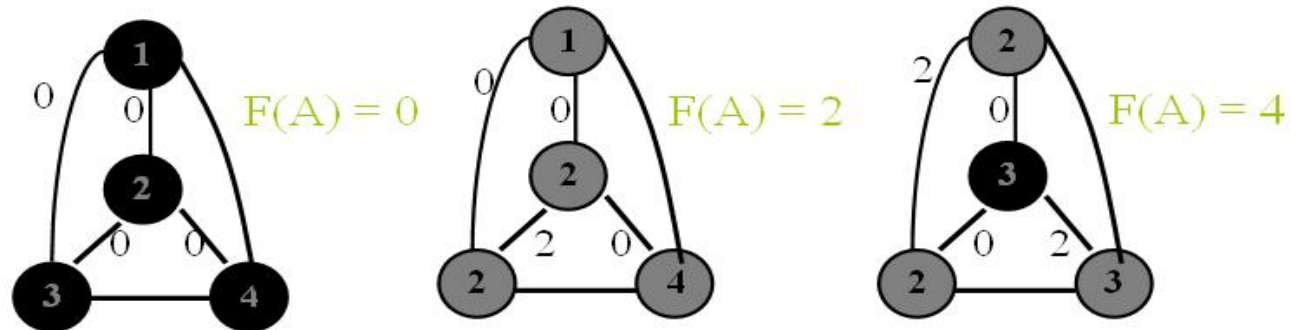
Continuously evolving missions of expeditionary forces



Existing Approaches to UAV Coordination

- Auction-based algorithms
 - Without performance guarantee
 - Severe bottlenecks for dynamic task reallocation in auction
- Decision theoretic approaches such as MDP/POMDP
 - Intractable with limited and intermittent communication
- Distributed Constraint Optimization (DCOP)
 - With strong global quality guarantees
 - Only feasible for static and small-scale problems

An Example of DCOP



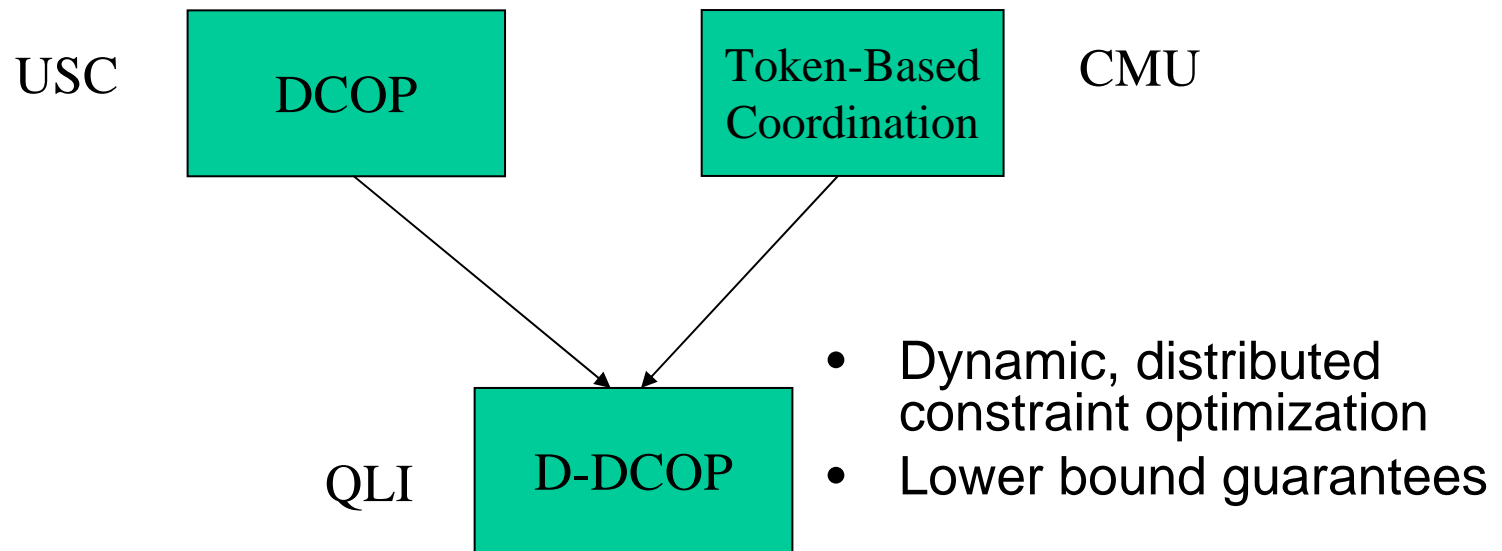


Drawbacks of DCOP

1. The dense constraint graph for task allocation dramatically increase communication with DCOP algorithms.
2. DCOP has to solve a new constraint graph when either the number of UAVs or the number of tasks changes.
3. DCOP is vulnerable to communication failures and only works for a static topology of UAV networks.

Explore Synergies between DCOP and Token-Based Approach

- Based on constraint graphs
- Only feasible for static and small-scale problems
- A heuristics-based algorithm
- No performance guarantee

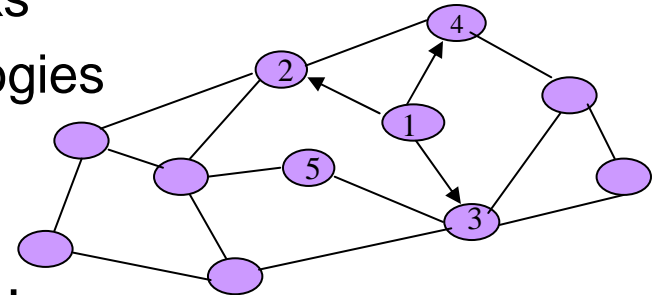


[1] Pragnesh Jay Modi, Wei-Min Shen, Milind Tambe, Mokoto Yahoo, Adopt: Asynchronous distributed constraint optimization with quality guarantees, Artificial Intelligence, 161(1-2):149-180, 2005

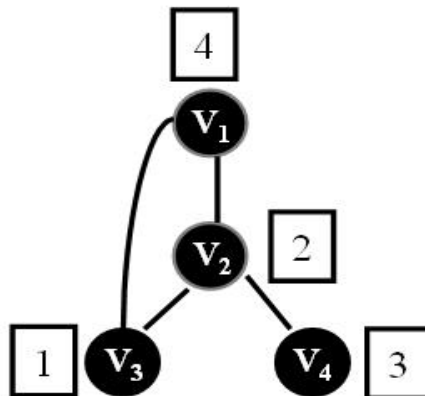
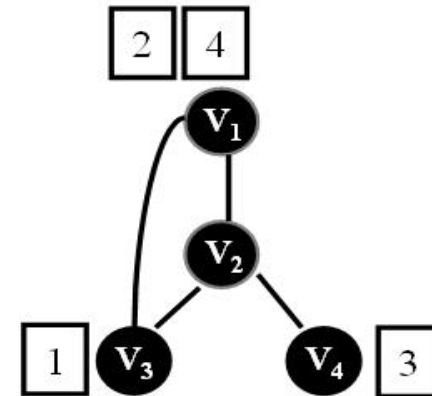
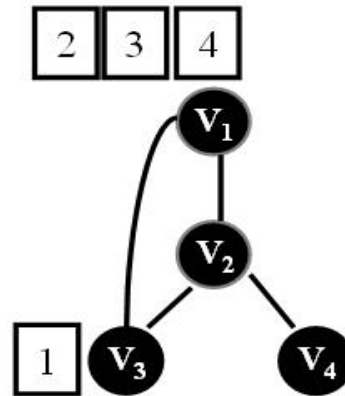
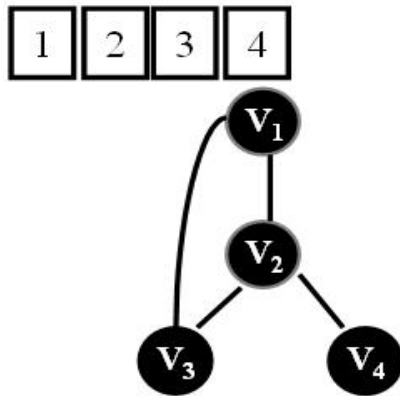
[2] Yang Xu, Paul Scerri, Bin Yu, Katia Sycara, Michael Lewis, An integrate token-based algorithm for scalable coordination, AAMAS, 2005

D-DCOP: **D**ynamic, **D**istributed **C**Onstraint Optimization

- Modeling
 - UAV platforms, costs and rewards of tasks
 - A connected network with dynamic topologies
 - Mission goals are decomposed into tasks
- Task allocation via local utility estimation
 - An **anytime** algorithm for dynamic task allocation
- Task execution via robust motion planning
 - Use **rendezvous points** to synchronize tightly-coupled tasks, e.g., coverage



An Example of D-DCOP



No need to represent task exclusiveness constraints



Key Ideas of D-DCOP

- **Representation** (from token-based coordination)
 - Task information is encapsulated into tokens
 - No need for mutually exclusive constraints
- **Local search:** (from k-optimal DCOP algorithms)
 - Each vehicle exchanges the task information with its neighbors
 - The vehicle transfers the task to a neighbor with a higher **Utility**
- Similar algorithms exist in distributed constraint satisfaction problems
 - DBA: distributed breakout algorithm
 - DSA: distributed stochastic search algorithm

[3] Makoto Yokoo and Katsutoshi Hirayama, Distributed breakout algorithm for solving distributed constraint satisfaction problems, ICMAS, 1996

[4] Weixiong Zhang and Lars Wittenburg. Distributed stochastic search for constraint satisfaction and optimization: Parallelism, phase transitions and performance. In Workshop on Probabilistic Approaches in Search AAAI-2002,



A Comparative Study

		Dynamic Tasking	Dynamic Topologies	Limited/intermittent Communication	Performance Guarantees
Centralized Solutions	MILP	No	N/A	No	Yes
	Auction	Yes	N/A	No	Yes
Distributed Solutions	DCOP	No	No	No	Yes
	D-DCOP	Yes	Yes	Yes	Yes (suboptimal)



Tasks and Vehicles

- Tasks
 - IDs: list of vehicles which accepted or held the task before
 - Status: accepted/not accepted
 - Category: ID, coverage, tracking
 - (t_s, t_d) : initial and finishing time
- Vehicles
 - ID: the vehicle's ID
 - (x, y, z) : existing location

Utility Estimation for any UAV V_i and Task T_j

$$U(V_i, T_j) = \frac{[c * \text{cap}(V_i, T_j) * L_j - C(V_i, T_j)]}{(t_{\text{delay}}/T_{\text{avg}} + 1)}$$

Utility
Reward
Cost

- Goals of task allocation
 - Task priority control
 - Maximize utility
 - Resource management
 - Performance guarantees

Parameters

c : maximal travel time

$\text{cap}(V_i, T_j)$: capability of V_i for task T_j

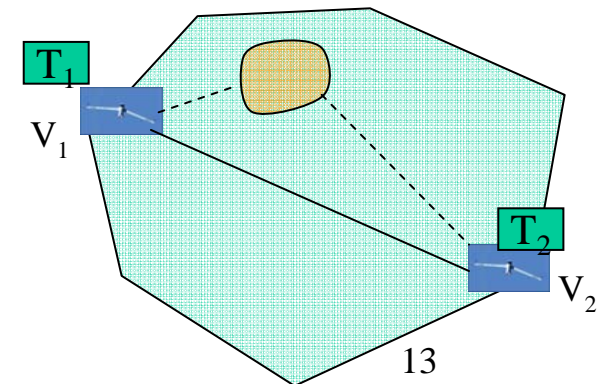
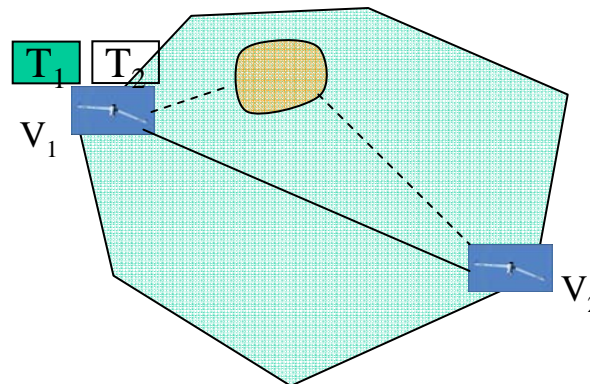
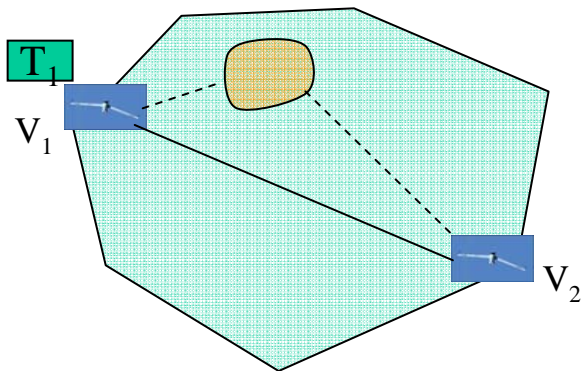
L_j : priority level of task T_j

$C(V_i, T_j)$: travel time to task T_j

t_{delay} : waiting time for task T_j

T_{avg} : average task execution time

Goals

$$U(V_i, T_j) \geq \max_{i \in \text{neighbor}(V_i)} U(V_i, T_j)$$




Simulation Testbed

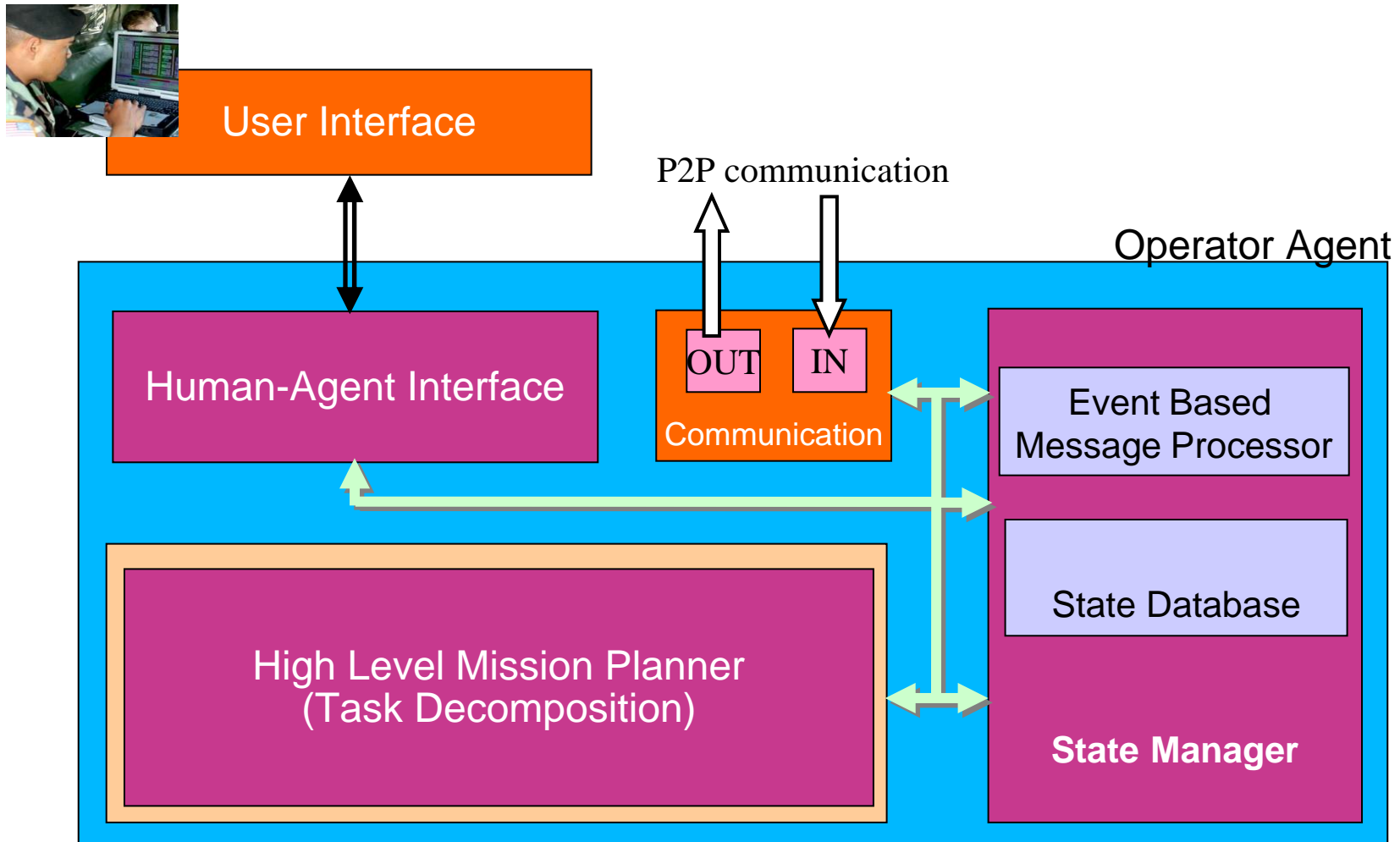
- Representative scenario
 - 1 operator
 - ~10 UAVS
 - Heterogeneity
 - Search/coverage, identification
 - May require multiple sensing modalities
- Communication constraints
 - Short range, low bandwidth communication capabilities on small UAVs
- Solutions:
 - Mandatory check-in at rendezvous points
 - Constrained by time and/or location



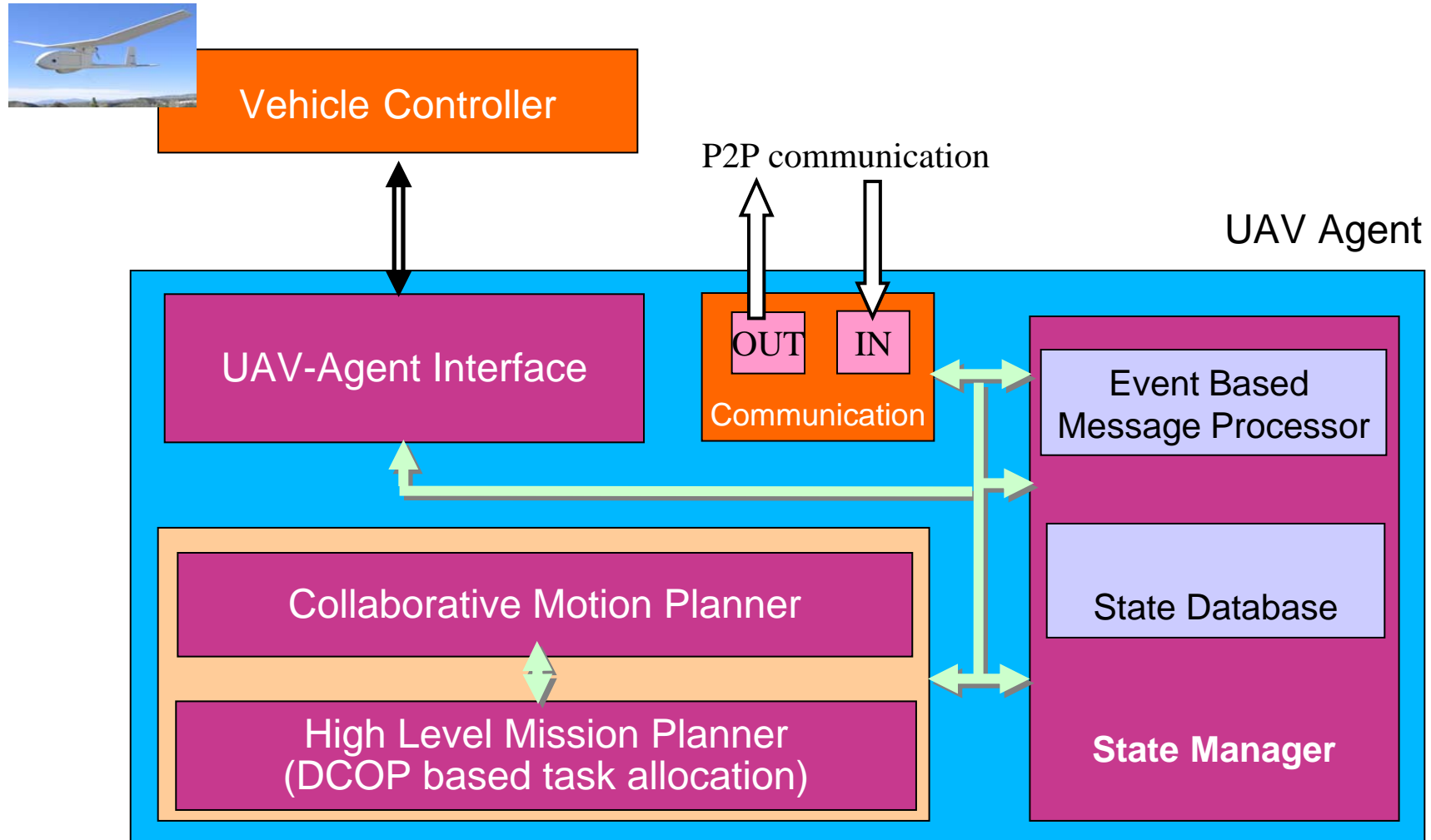
Highlights of System Development

- Algorithm development
 - Motion planning in Matlab
 - Java wrapper for Matlab code
 - D-DCOP in Java
 - Tasks can be any spatio-temporal specification with constraints
- Inter-module communications
 - A simple MAS in Java (an agent/per JVM)
 - Publish-Subscribe network model
 - Common XML schema
- Distributed and decentralized architecture
- A simple GUI

Architecture of an Operator Agent



Architecture of a UAV Agent





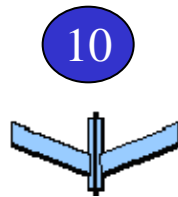
Scenarios

- Task templates:
 - (A) High altitude (1000m) coverage for surveillance
 - (B) Medium altitude (200m) high resolution imaging and identification
- A team of 10 UAVs
 - 5 DragonEye like UAVs: EO or IR for task B
 - Speed: 18 m/s; minimal turning radius: 50 m; max. task duration: 2~4 hours; communication: 10 km + 125 Kbps
 - 2 Scan Eagle like UAVs: EO + IR for task A
 - Speed: 25.7 m/s; minimal turning radius: 117 m; max. task duration: 15 hours; communication: 200 km + 4 mbps
 - 3 Fire Scout like UAVs: EO + IR + SAR for tasks A and B
 - Speed: 0~28 m/s, cruise at 28 m/s; minimal turning radius: 0 ~ 140 m for cruise speed; max. task duration: 4.2 hours ; communication: 200 km + 10.7Mbps down / 200Kbps up

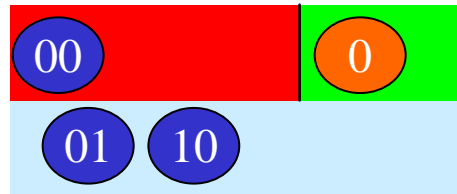
Symbols in Demos

- Vehicles

Querying message



Tasks being held



Tasks being executed

Tasks in the waiting list

- Tasks

- Coverage tasks



1: zone 1 | 0: first task for zone 1

- ID tasks





Demo



Conclusion

- Successfully demonstrated the integration of *distributed, dynamic* task allocation with *continuous* motion planning
 - Continuously evolving missions
 - Limited and intermittent communications
 - Real-time, dynamic task allocation
- Future work
 - Synergies between task allocation and resource allocation
 - Dynamic team formation for tightly-coupled tasks



Acknowledgements

- Penn: Vijay Kumar, Peng Cheng, Jim Keller,
- ONR: Marc Steinberg