Multimedia Communications and Systems Lab

Performance of the Asynchronous Consensus Based Bundle Algorithm in Lossy Network Environments

Matt Rantanen*, Jalil Modares*, **Nick Mastronarde***,
Farshad Ghanei** and Karthik Dantu**

*Dept. of Electrical Engineering **Dept. of Computer Science and Engineering
University at Buffalo, The State University of New York, Buffalo, NY

nmastron@buffalo.edu
http://www.eng.buffalo.edu/~nmastron/

July 9, 2018

IEEE SAM Special Session
Signal Processing and Communications for Resilient Autonomous Swarms

The Intersection of Planning and Communications

- Without reliable wireless communications, drones cannot coordinate
- Consequences of network losses
  - Failed delivery of sensed data to processing nodes
  - Insufficient situational awareness for effective in-field planning
  - Delayed/lost command and control messages (focus of this work)
  - Worst case: mission failure!
- Despite these adverse effects, most planning literature assumes perfect communication among nearby agents
- This has led to various techniques to maintain network connectivity

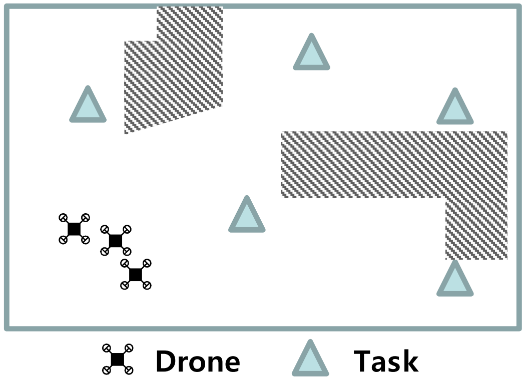
Related Work

- "Binary" connectivity
  - Connectivity-as-a-service [Cornejo, '09]
    - Refine arbitrary motion plan to preserve network connectivity and meet goals
  - Control-theoretic connectivity [Zavlanos, '11]
    - Convex optimization and subgradient descent algorithms to maximize network's algebraic connectivity
    - Potential fields to control network topology
  - Connectivity-aware task allocation [Ponda, '12]
    - Extend the well-known Consensus Based Bundle Algorithm (CBBA [Choi, '09]) to include planning for relays
- Connectivity with variable reliability
  - BER- and throughput-aware task allocation [Kopeikan, '12]
    - Extend CBBA with relays to meet BER and throughput constraints

Prior work investigates how planning affects communication, but not how *unreliable communication affects planning*

The Task Allocation Problem

- Given
  - A set of drones
  - A set of tasks
- Goal
  - Allocate tasks to drones (at most one drone per task)
  - Maximize sum utility



Example environment (with obstacles)

We investigate the effect of realistic network environments on the Asynchronous CBBA (ACBBA [Johnson, ‘10], [Johnson, ‘11])

Problem Formulation: Notation

- $A = \{1,2,\ldots,N_a\}$: Set of agents (drones)
  - $i \in A$: Specific agent
- $T = \{1,2,\ldots,N_t\}$: Set of tasks
  - $j \in T$: Specific task
- $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iN_t})$: agent i's assignment vector
  - $x_{ij} = 1$ if agent $i \in A$ is assigned task $j \in T$
  - $x_{ij} = 0$, otherwise
- $\boldsymbol{p}_i$: ordered sequence of tasks assigned to agent i
- $u_{ij}(\tau_{ij}(\boldsymbol{p_i}))$: agent i's utility for completing task j at time $\tau_{ij}(\boldsymbol{p_i})$
  - $u_{ij}(\tau_{ij}(\boldsymbol{p_i})) = r_j \lambda^{\tau_{ij}(\boldsymbol{p_i})}$, where r_j is reward for task j and $\lambda \in [0,1)$

Problem Formulation: Optimization

$$\max \sum_{i \in A} \left(\sum_{j \in T} u_{ij}(\tau_{ij}(\boldsymbol{p}_i))\, x_{ij} \right)$$

$$\text{subject to} \quad \sum_{j \in T} x_{ij} \leq L_t, \qquad \forall i \in A$$

$$\sum_{i \in A} x_{ij} \leq 1, \qquad \forall j \in T$$

$$x_{ij} \in \{0,1\}, \qquad \forall (i,j) \in A \times T$$

If agents form a *connected network* and there are *no transmission errors*, then the CBBA guarantees a non-conflicting task assignment. This assignment achieves within 50% of the optimal utility.

Internal State Information in the CBBA

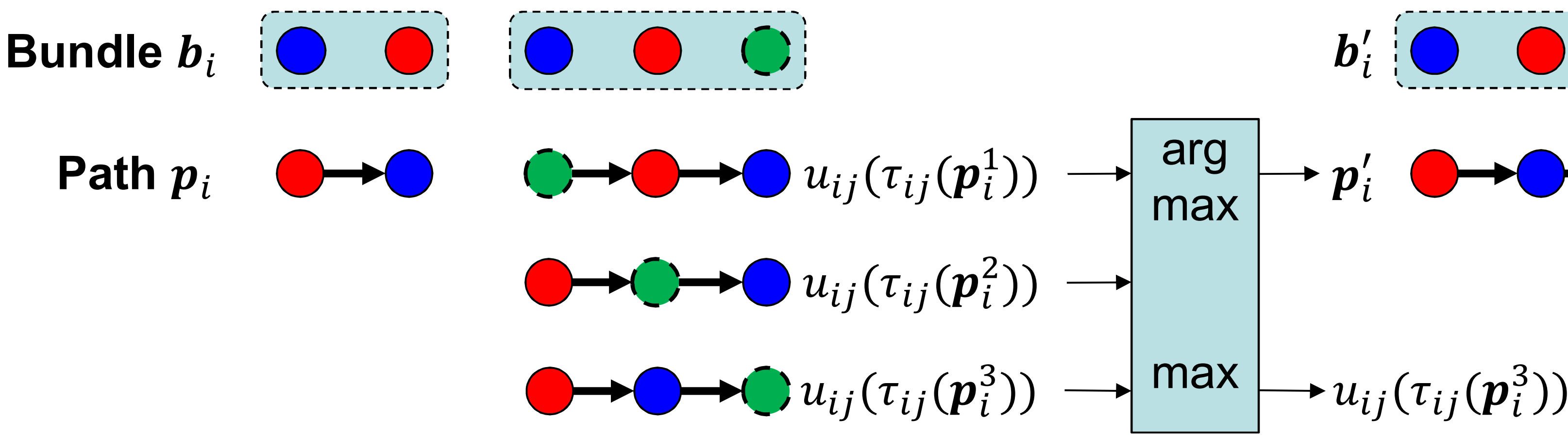
Each agent $i \in A$ maintains the following five internal state vectors

- Bundle vector $\boldsymbol{b}_i$
  - Element $b_{in} \in T$ corresponds to the nth task assigned to agent i
  - Tasks are ordered based on when they are "won"
- Path vector $\boldsymbol{p}_i$
  - Contains same tasks as bundle, but ordered based on when they will be completed
- Winning agent vector $\boldsymbol{z}_i$
  - Element $z_{ij} \in A$ indicates who agent i believes has highest bid for task j
- Winning bid vector $\boldsymbol{y}_i$
  - Element $y_{ij} \in R_+$ corresponds to agent z_{ij}'s winning bid for task j
- Timestamp vector $\boldsymbol{t}_i$
  - Element $t_{ij} \in R_+$ indicates when agent z_{ij} placed bid y_{ij} on task j

CBBA Iterations

The CBBA iterates among three phases

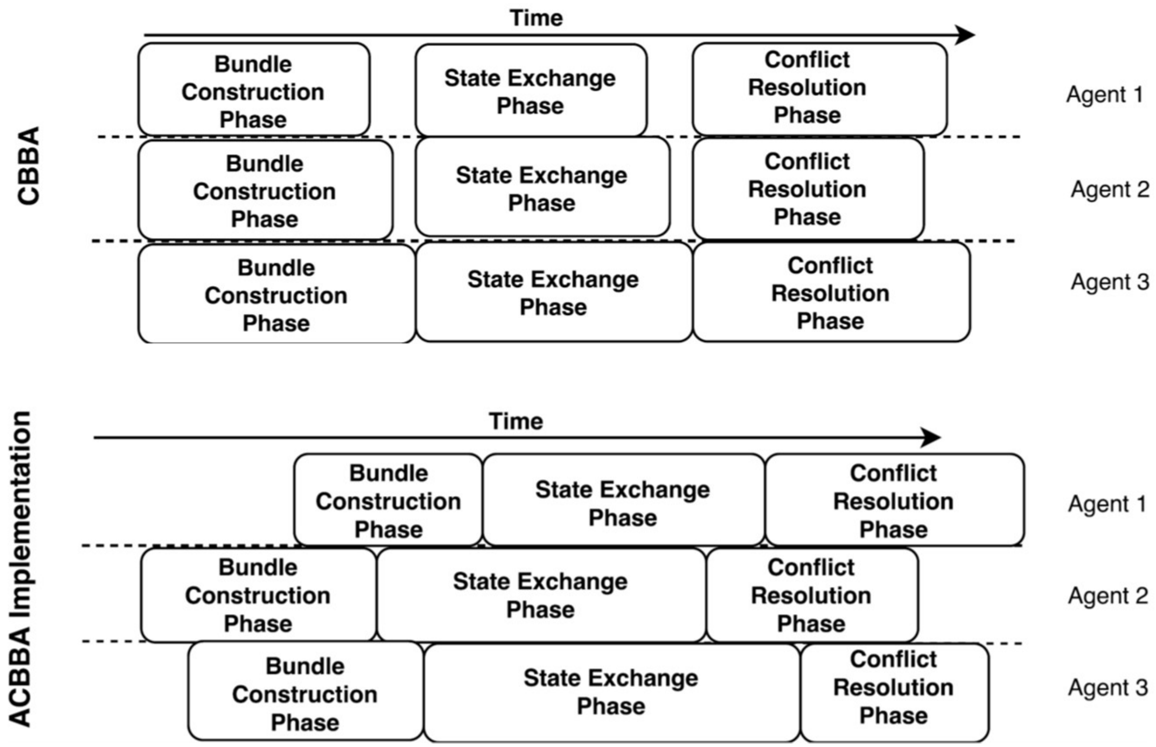
- Bundle construction phase
  - Each agent adds tasks to its bundle in a *sequential greedy* fashion



- State exchange phase
  - Each agent communicates its winning agent vector $\boldsymbol{z}_i$, winner bid vector $\boldsymbol{y}_i$, and timestamp vector $\boldsymbol{t}_i$
- Conflict resolution phase
  - Each agent releases tasks it was outbid on and tasks added thereafter

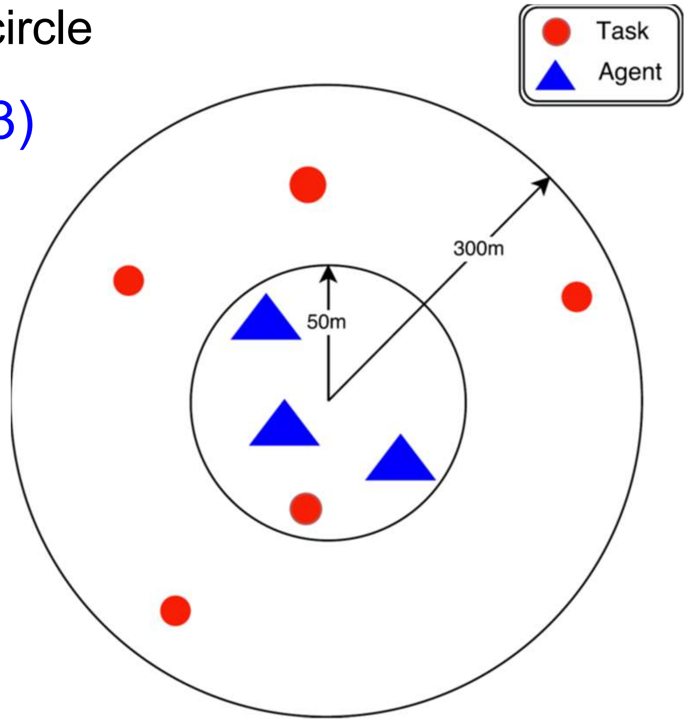
Asynchronous CBBA (ACBBA) vs. CBBA

- ACBBA is conceptually similar to the CBBA, but
  - Each agent builds its bundle and performs consensus asynchronously
  - Each agent only transmits the winning agent, winning bid, and timestamp for a *single task* at a time (less bandwidth required)



ACBBA Simulation Setup

- 100 ACBBA simulation scenarios
  - $N_t = 1,2, \dots 10$ tasks and $N_a = 1,2, \dots 10$ drones
- Each scenario executed 100 times
  - Drones randomly dropped in 50 m radius circle (ensures connectivity)
  - Tasks randomly dropped in 300 m radius circle
- IEEE 802.11b Wi-Fi broadcast mode (ns-3)
  - No ACKs
  - No retransmissions
  - No exponential backoff
- UDP (ns-3)
  - Connectionless transport protocol



Evaluation Metrics

- Redundant task assignments
  - If n_j agents are assigned the same task $j \in T$, then there are $n_j - 1$ redundant assignments of task j
  - Total number of redundant task assignments $n_r \coloneqq \sum_{j \in T} \max(n_j - 1, 0)$
- Total number of transmission/reception events
  - $n_{TX} \coloneqq \sum_{i \in A} n_{TX,i}$, where $n_{TX,i}$ is the number of times agent i broadcasts its state information
  - $n_{RX} \coloneqq \sum_{i \in A} n_{RX,i}$, where $n_{RX,i}$ is the number of times agent i receives state information
- Fraction of received packets
  - $f_{RX} \coloneqq n_{RX} / [n_{TX} \cdot (N_a - 1)]$
- Negotiation time
  - Elapsed time from the first bundle construction phase to the last conflict resolution phase

Simulation Results (1/3)

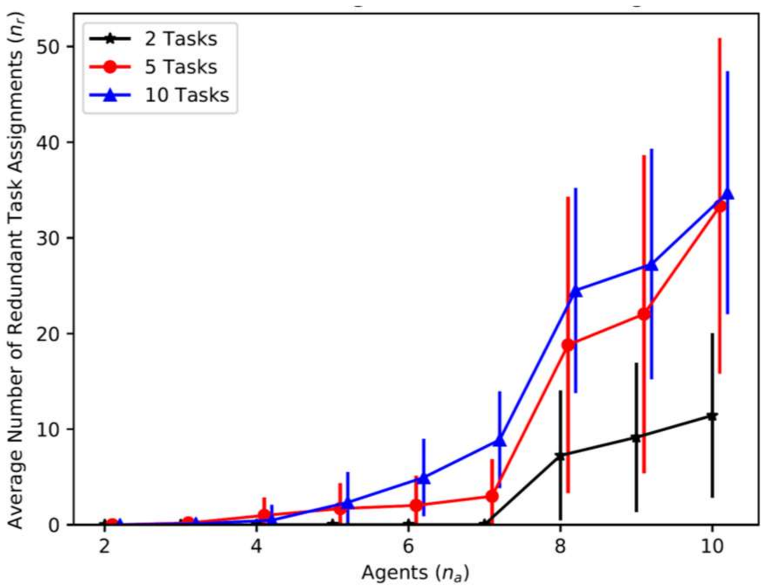


Fig A. Redundant task assignments n_r vs. number of agents N_a.

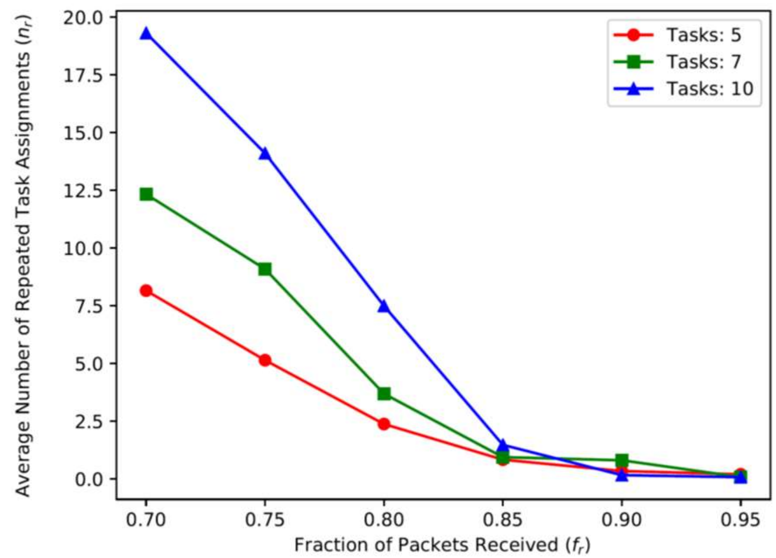


Fig B. Redundant task assignments n_r vs. fraction of packets received f_{RX}.

- Number of redundant task assignments increases with number of agents
- Number of redundant task assignments is negatively correlated with the fraction of packets received
- *What causes this? Channel errors and/or collisions?*

Simulation Results (2/3)

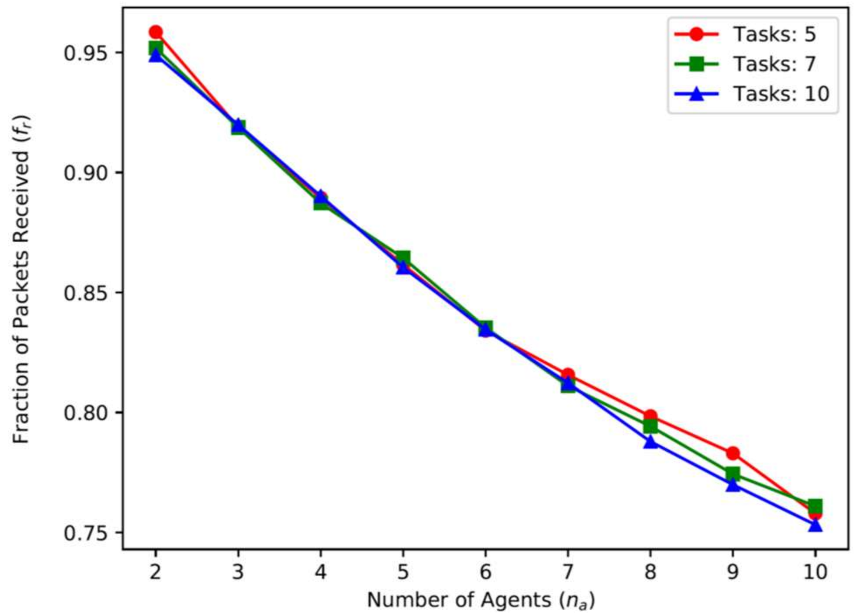


Fig C. Fraction of packets received f_{RX} vs. number of agents N_a

- The fraction of received packets decreases with the number of agents and is approximately invariant in the number of tasks
- Simulation channel errors are independent of number of agents ➔ Performance degradation is primarily due to collisions
- *Why are collisions so problematic?*

Simulation Results (3/3)

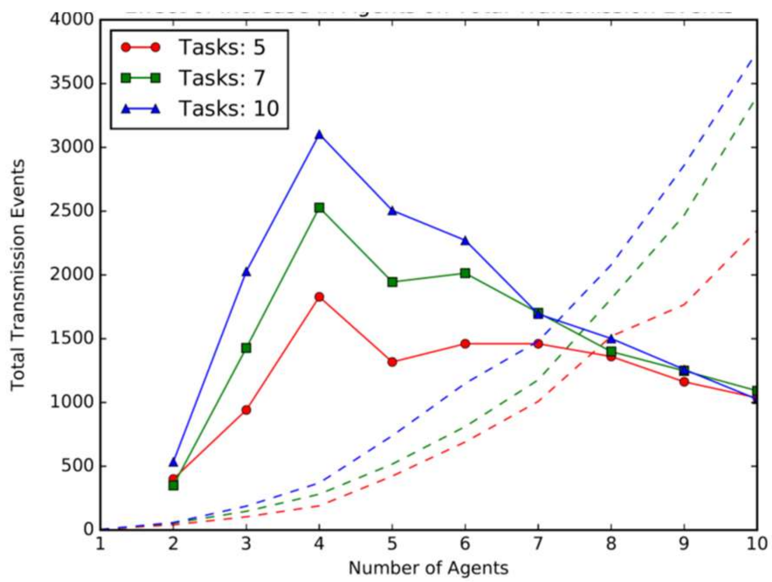


Fig D. Total transmissions n_{TX} vs. number of agents N_a. Solid and dashed lines show lossy and lossless results, respectively.

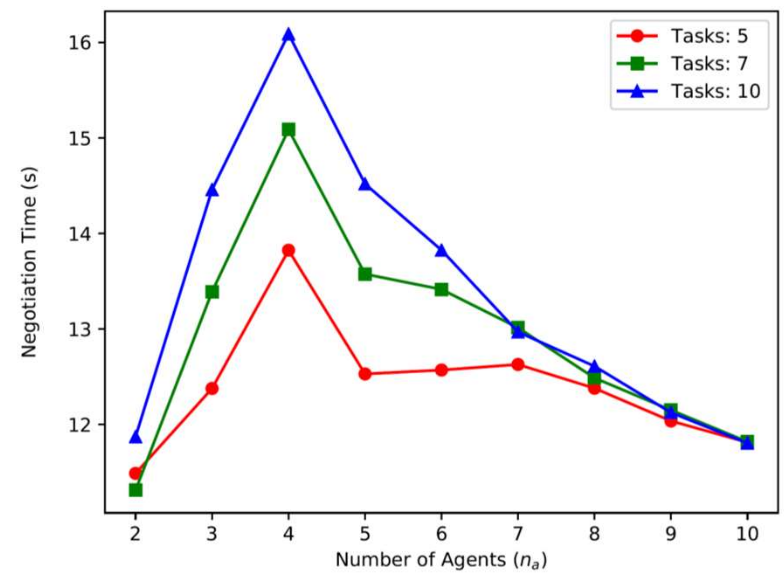


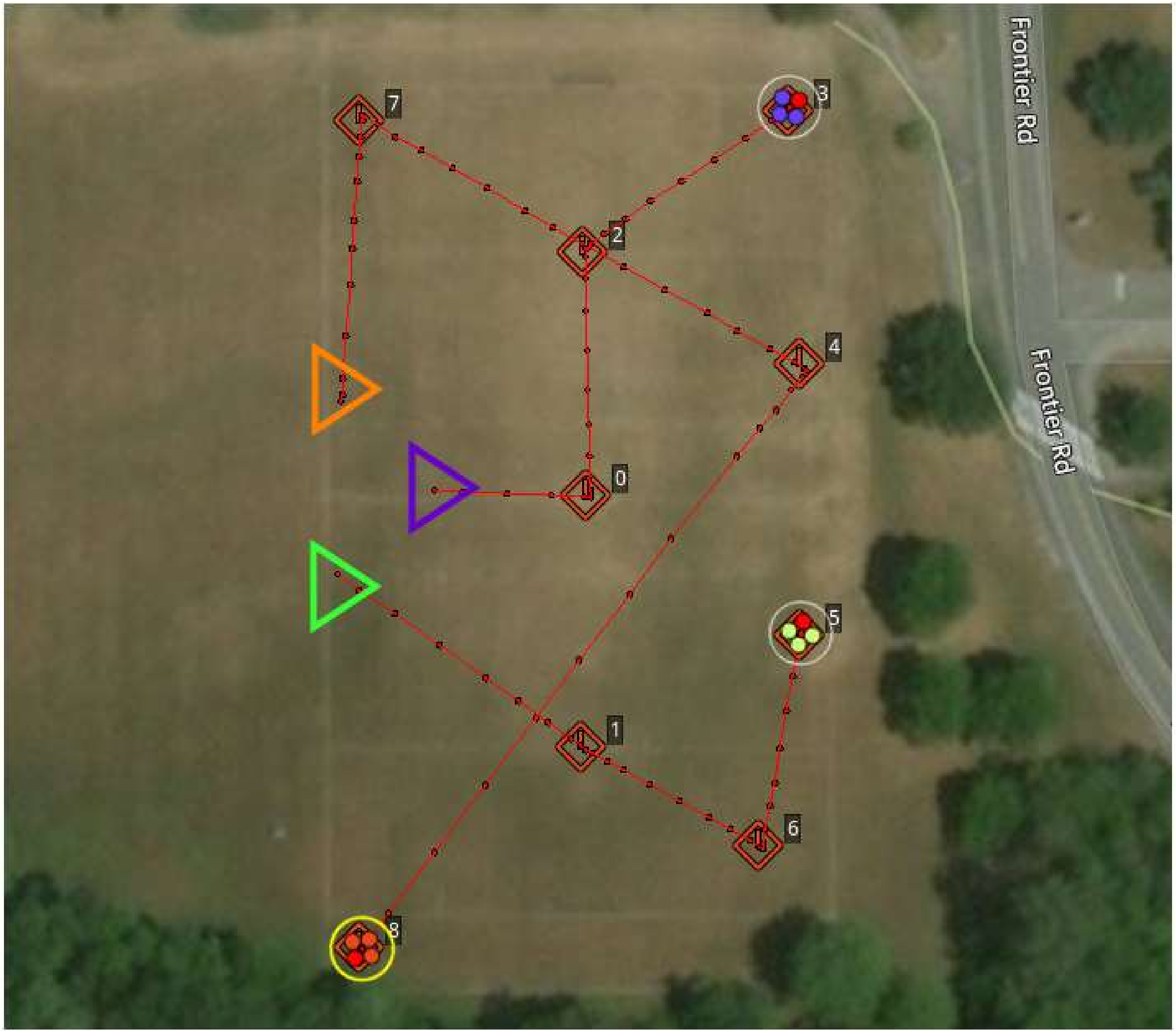
Fig E. Negotiation time (s) vs. number of agents N_a.

- *Lossless communication:* TX events increase with number of agents
- *Lossy communication:* TX events initially increase with number of agents, but eventually decline due to effect of collisions
- Decline in transmission events is related to shortened negotiation time

Thank You!

- ACBBA yields inefficient task assignments in lossy networks
- Agents mistakenly attribute absence of new messages in network to reaching consensus, when actually due to lost packets
  - Collisions have more significant impact than channel errors
- UDP + IEEE 802.11 broadcast mode provides insufficient QoS
- Ongoing work:
  - Study performance of ACBBA under other network configurations
    - UDP + IEEE 802.11 unicast mode
    - TCP + IEEE 802.11 unicast mode
- Future work:
  - Make ACBBA more robust to network disruptions
  - Study interaction of planning and communications for other applications
    - Swarming, formation control, etc.

Example Execution of ACBBA in the UB-ANC Emulator



References
Task Allocation and Planning for Connectivity

- **[Cornejo, '09]** A. Cornejo, F. Kuhn, R. Ley-Wild, and N. A. Lynch, "Keeping mobile robot swarms connected.," in DISC, pp. 496–511, Springer, 2009.
- **[Zavlanos, '11]** M. M. Zavlanos, M. B. Egerstedt, and G. J. Pappas, "Graph-theoretic connectivity control of mobile robot networks," Proceedings of the IEEE, vol. 99, no. 9, pp. 1525–1540, 2011.
- **[Ponda, '12]** S. S. Ponda, L. B. Johnson, A. N. Kopeikin, H.-L. Choi, and J. P. How, "Distributed planning strategies to ensure network connectivity for dynamic heterogeneous teams," IEEE Journal on Selected Areas in Communications, vol. 30, no. 5, pp. 861–869, 2012.
- **[Choi, '09]** H.-L. Choi, L. Brunet, and J. P. How, "Consensus-based decentralized auctions for robust task allocation," IEEE transactions on robotics, vol. 25, no. 4, pp. 912–926, 2009.
- **[Johnson, '10]** L. B. Johnson, S. Ponda, H.-L. Choi, and J. P. How, "Improving the efficiency of a decentralized tasking algorithm for uav teams with asynchronous communications," in AIAA Guidance, Navigation, and Control Conference (GNC), vol. 5, pp. 5406–5411, 2010.
- **[Johnson, '11]** L. B. Johnson, S. S. Ponda, H.-L. Choi, and J. P. How, "Asynchronous decentralized task allocation for dynamic environments," in Proceedings of the AIAA Infotech@ Aerospace Conference, vol. 2, pp. 2–2, 2011.
- **[Kopeikin, '12]** A. Kopeikin, S. S. Ponda, L. B. Johnson, and J. P. How, "Multi-uav network control through dynamic task allocation: Ensuring data-rate and bit-error-rate support," in Globecom Workshops (GC Wkshps), 2012 IEEE, pp. 1579–1584, IEEE, 2012.