Multi-UAV Network Control through Dynamic Task Allocation: Ensuring Data-Rate and Bit-Error-Rate Support

Andrew Kopeikin, Sameera S. Ponda, Luke B. Johnson, and Jonathan P. How

Abstract— **A multi-UAV system relies on communications to operate. Failure to communicate remotely sensed mission data to the base may render the system ineffective, and the inability to exchange command and control messages can lead to system failures. This paper describes a unique method to control communications through distributed task allocation to engage under-utilized UAVs to serve as communication relays and to ensure that the network supports mission tasks. The distributed algorithm uses task assignment information, including task location and proposed execution time, to predict the network topology and plan support using relays. By explicitly coupling task assignment and relay creation processes the team is able to optimize the use of agents to address the needs of dynamic complex missions. The framework is designed to consider realistic network communication dynamics including path loss, stochastic fading, and information routing. The planning strategy is shown to ensure that agents support both data-rate and interconnectivity bit-error-rate requirements during task execution. System performance is characterized through experiments both in simulation and in outdoor flight testing with a team of three UAVs.**

I. Introduction

As unmanned systems technology progresses, teams of multiple unmanned air vehicles (UAV) will play an increasingly important role in a broader range of applications. These "real-world" scenarios involve executing complex missions where the number, status, and types of tasks, as well as the environment vary dynamically. As such, the multi-UAV system must continuously assess its capabilities and properly allocate resources to overcome changes and maximize performance. The objective of dynamic task allocation is to ensure that tasks are performed effectively by the right UAV, at the right place, at the right time.

A significant challenge in this process involves maintaining proper communications to execute the mission. For multi-UAV teams to cooperatively execute a set of objectives, the communication network must exchange command and control messages, and when necessary, remotely sensed data. The inability to communicate sensed data, such as live video, to the designated processing centers may render the multi-UAV system ineffective for its mission [1]. Less intuitively, failure to properly exchange command and control messages can lead to dangerous system failures or unintended consequences. This is particularly true in decentralized systems, where these messages enable vehicle control and team decision making. Inadequate team control can lead to formation instability where one or multiple agents increasingly diverge from the desired behavior, and risk colliding [2]. Poor decision making may prevent the team from reaching consensus on a plan and remain idle without accomplishing anything, or worse, take action on an undesirable plan leading to unwanted consequences [3].

Because UAVs are highly mobile vehicles, information is most commonly exchanged across the network using wireless communication. Signals containing encoded messages travel between transmitting and receiving radio modules over wireless channels. The quality of the channel is fundamentally based on the strength of the signal at the receiver compared to noise and interference in the environment, or Signal to Noise Ratio (SNR) [4]. The quality of the link drives the probability that information transmitted will successfully be received, and affects the rate at which information can be exchanged over that channel.

The network itself can be controlled through its topology, which describes the set of interconnections between nodes, and how strong those connections are. Since wireless channels generally degrade with increasing distance and obstacles in the line of sight, a relevant control method in multi-vehicle systems involves properly positioning agents to support the network. Common ways to do this include planning the motion of agents using gradient ascent [5], potential fields [6], reactive control [7], and adaptive strategies [8] to steer vehicles to stay within communication range of each other. Another method is to deploy agents designated as mobile communication relays to support a network of sensor nodes, using both graph theoretic [9] and network optimization schemes [10].

This paper is unique in that the network is controlled through task allocation. The team cooperates to (1) ensure tasks undertaken are supported by the network, and (2) dynamically assign under-utilized agents to serve as communication relays. By explicitly coupling the task assignment and relay planning processes, the team is able to better optimize the use of agent resources to address current mission needs. This leads to improved performance and added flexibility in real-time dynamic mission scenarios. The framework is formulated to consider realistic networking dynamics such as path loss, fading, and information routing.

The paper proceeds as follows: the communication models and task assignment formulation employed in the paper are provided in Sect. II. Section III outlines the CBBA with Relays algorithm designed to control the network considering realistic communication constraints. In Sect. IV experiments are conducted in both simulation and outdoor flight testing

A. Kopeikin, S.M student from the Aerospace Controls Lab, MIT
S. Ponda, Ph.D. candidate in the Aerospace Controls Lab, MIT
L. Johnson, Ph.D. candidate in the Aerospace Controls Lab, MIT
J. How, R.C. Maclaurin Professor of Aeronautics & Astronautics, MIT
{kopeikin,sponda,lbj16,jhow}@mit.edu

to characterize the performance of the framework. Finally, Sect. V describes ongoing and future work.

II. Problem Statement

A. Network Model

The objective in this paper is to enable a multi-UAV team to cooperatively execute a mission and to ensure that its wireless network supports the transmission of sensor data and command and control (C2) messages. This section describes the communication model employed in this process.

A fundamental factor in network wireless channel performance is the Signal to Noise Ratio (SNR). In general, channels degrade with increasing inter-node distance due to path loss. Signal shadowing occurs due to obstacles in the environment such as buildings, mountains, or even the platform vehicle itself. Finally, multipath replicas of the signal reflecting and scattering off these obstacles constructively or destructively interfere with each other at the receiver [11]. A practical way to model SNR in uncertain environments is to vary SNR (γ) according to a lognormal distribution with variance σ^2 which depends on the density of obstacles in the environment to capture both shadowing and multipath fading effects [4]. The SNR is conveniently expressed in decibels by Eq. (1):

$$\gamma_{dB} = 10 \log \left(\frac{P}{N_o W} \right) + K_{dB} - 10\alpha \log \left(\frac{d}{d_o} \right) - \mathcal{N}(0, \sigma_{dB}^2) \tag{1}$$

where P is the transmission power, $\frac{N_0}{2}$ is the power spectral density of the environment noise, W is the bandwidth of the signal, K_{dB} is a gain based on equipment characteristics, d is the Euclidean distance between the transmitter and receiver, and α is the path loss exponent which equals 2 in free space, and up to 6 in environments congested with obstacles [4,12].

A multi-UAV system network typically needs to support two types of data. First, sensor data may need to be transmitted to designated analysis centers [1]. Communication requirements depend on the data type, but live-video, for instance, requires minimal delay, high data-rates, and has some fault tolerance [13]. As such, the *Shannon Capacity* $u = W \log_2(1+\gamma)$ is used as a representative performance indicator to determine whether a channel satisfies data-rate requirements. The second data type consists of C2 messages including state information (telemetry), observations of the world (e.g. estimated target location), and control data (e.g. waypoints or task allocation). These messages have low bandwidth requirements, but must be exchanged with minimal delay and error for effective team coordination. This study therefore employs the *bit-error-rate* (BER) as a metric for C2 messaging success. BER is the probability an information bit will be dropped, and is defined generally as $\texttt{BER} = Q(\sqrt{2\gamma})$ [4]. To evaluate network performance, the locations of UAVs are used to determine the expected SNR of wireless channels between agents and predict the available data-rate and BER for information transfer between them.

Information must be routed along links with sufficient capacities, and arrive at its destination with acceptable error-rates (BER), as dictated by data requirements. Information routing protocols in ad-hoc wireless networks consistent with a distributed multi-UAV architecture often use shortest path routing mechanisms, such as AODV [14]. A node which needs to transmit broadcasts a "request to send" message which identifies the destination node. Neighbors receive the message, compute a cost metric (see Sect. III), and rebroadcast the message. The process continues, where cost is computed at every hop, until the destination is reached. Each node stores its lowest cost originator. Once a designated amount of time has elapsed, the destination node sends a unicast response to its lowest cost originator, which similarly relays the response to backtrack the lowest cost route [14]. These protocols are implemented in practice because of their rapid response to changes in wireless channel performance and topology. A significant drawback to them, however, is that information from different nodes is routed mostly independently of other nodes, and tends to use the same high performance routes leading to congestion and unfairness in node usage [15].

B. Task Allocation

The complex missions envisioned to be executed by autonomous teams of robotic agents will require different agents to have different roles and responsibilities during execution, much like modern day sports teams. The task allocation problem involves planning the resource allocation for a team of N_a unmanned systems, to accomplish a set of N_t tasks. In this work, it is formulated as the following optimization problem:

$$\begin{aligned} \underset{\mathbf{x},\boldsymbol{\tau}}{\text{argmax}} \quad & \sum_{i=1}^{N_a} \left(\sum_{j=1}^{N_t} c_{ij}(\mathbf{p}_i(\mathbf{x}_i, \boldsymbol{\tau}_i)) x_{ij} \right) \\ \text{s. t.} \quad & \sum_{j=1}^{N_t} x_{ij} \leq L_i, \quad \forall i \in \mathcal{I} \\ & \sum_{i=1}^{N_a} x_{ij} \leq 1, \quad \forall j \in \mathcal{J} \\ & x_{ij} \in \{0,1\}, \tau_{ij} \in \{\mathbb{R}^+ \cup \emptyset\}, \ \forall (i,j) \in \mathcal{I} \times \mathcal{J} \end{aligned} \tag{2}$$

where $x_{ij} = 1$ if agent i is assigned to task j, and $\mathbf{x}_i$ is its vector of assignments. Vector $\mathbf{p}_i$ represents the *path* for agent i, a sequence of tasks listed in the order to be executed at times in $\boldsymbol{\tau}_i$. The length of $\mathbf{p}_i$ is limited by L_i, the planning horizon. The objective in Eq. (2) is to maximize the sum of the reward for each agent. The score c_{ij} that agent i obtains by performing task j is a function of the value of the task, the cost of execution (e.g. fuel consumed), and the execution time τ_{ij} compared to the optimal task execution time [16].

Because of all inter-dependencies in number of possible assignments, Eq. (2) is very difficult to solve if planning for more than 1 task for each agent. A popular method to solve the task assignment problem involves using auction algorithms which are efficient but sub-optimal [17]. The Consensus-Based Bundle Algorithm (CBBA) is one such auction protocol that provides provably good solutions for heterogeneous decentralized agent multi-task assignment functions. Each CBBA iteration is composed of two phases. First, in the bundle building phase each agent greedily bids on an ordered bundle of tasks based on its capabilities and its location with respect to the task. Second, in the consensus phase, agent bid information is shared, and conflicts in task assignment are identified and resolved. The algorithm iterates until it converges on a conflict free plan [17]. The CBBA algorithm runs in polynomial time, and has demonstrated good scalability with increasing numbers of agents and tasks. It has been demonstrated in numerous applications with teams of heterogeneous unmanned systems [18].

III. Task Allocation for Network Control

A complicating factor not considered in the CBBA algorithm is that the multi-UAV system needs to meet communication requirements described in Sect. II to operate effectively. Task assignments not supported by the network should be avoided since they provide no reward but waste energy in execution and can degrade the ability to cooperate as a team. To prevent this, communication constraints can be explicitly considered in the planning process. Task allocation information, such as task locations and planned execution times, can be leveraged by the agents to predict the network topology at execution and identify communication constraint violations. The CBBA with Relays algorithm described in the previous installment of this work [19], was therefore developed as an outer-loop to the CBBA algorithm to prevent network disconnects. When potential disconnects are identified in a CBBA task allocation, relay tasks are created to repair the network. The CBBA algorithm then iterates, with the relay tasks. If agents can accommodate relay tasks into their plan they place appropriate bids. Following each iteration of the algorithm, unassigned relays are deleted. The tasks dependent on the deleted relays are then placed in the bidding agent's forbidden task list to ensure that the CBBA with Relays algorithm eventually converges on a task assignment solution. Interested readers are referred to [16] for more information.

In this paper, the algorithm is further developed to handle realistic networking dynamics including data-rate, bit-error-rate, and information routing. The models listed in Sect. II are used with agent positions to compute the network topology of inter-agent channel data-rate capacities and BER.

A. Supporting Data-Rate

In order to transmit live sensor data back to the base, the multi-UAV network must ensure information routes have sufficient end-to-end channel capacity at the time of execution. A model representative of shortest path routing protocols (similar to AODV discussed in Sect. II) is implemented. The cost metric to determine the shortest transmission route is set to $C_{A_{ij}} = \frac{1}{u_{ij}}$ between each node i and j, as a generalized version of the airtime metric popular in ad-hoc networking applications [20]. This identifies routes with high data-rates, but also minimizes the number of hops to reach the destination, which is a desirable behavior for live video transfers [4].

The protocol is used in planning to predict how information will be routed from each active agent to the base. Route discovery is performed in simulation using the well known Dijkstra's shortest path algorithm [21]. Then, the data flow from each task at rate f_{task} is summed over each generated path to determine the channel capacity requirement between each node. Predicted data-rate violations are identified when $u_{req_{ij}} > u_{ij}$. This is then used by the planner to decide: (1) which tasks are allowed to be executed given the present configuration, and (2) how best to create relays to support disconnected tasks.

Since the mission objective is to maximize team reward, a selection process is required to find the combination of proposed tasks which provides the highest reward without violating data-rate requirements. To avoid an exhaustive search over the set of all tasks and to reduce the number of network recomputations when discarding tasks, the algorithm leverages the *Minimum Spanning Tree* (MST) network graph structure of the routing paths. The tree is rooted at the base and each node sends its data to a single other node. The algorithm greedily prunes lowest value leaves (or end node tasks) that are up-link of edges with data-rate violations until all remaining tasks are supported by the network.

Once the set of disconnected tasks is identified, the planning framework sequentially reconnects them in a greedy value optimization scheme. In each step, the next most valuable reconnection is determined as the sum of the reconnected task values divided by the number of relays required. The algorithm uses the known required link capacity $u_{req_{ij}}$ to satisfy flow requirements on the lowest cost route to solve for the maximum inter-node distance d_{max} to meet the requirement. The number of required relays and equally spaced positions is computed ($N_r = \lceil \frac{d_{ij}}{d_{max}} \rceil$). Once relays are created, information routing, which may now have changed for other nodes, is recomputed to identify possible new link violations. This process repeats until all tasks in the configuration are connected or designated infeasible. Tasks are infeasible if they require more relays than the number of available agents, and are placed in the bidding agent's forbidden task list to ensure convergence of the algorithm.

B. Supporting Error-Rate Requirements

In addition to satisfying data-rate requirements, the multi-UAV network must also ensure that agents meet BER requirements to adequately exchange C2 messages. Routing for these messages is assumed to be independent of routing sensor data back to the base. As data travels over multiple hops to its destination, each hop has a probability of error. Over n hops, the probability of error is $p_e = 1 - ((1-p_{e1})(1-p_{e2})...(1-p_{en}))$ [13]. Since error probabilities are multiplicative over multiple hops, the additive cost metric of each hop is set to $C_{ber} = -\ln(1 - \mathtt{BER}_{ij})$ [8]. Dijkstra's Algorithm is used again to generate the minimum cost routes between all agents. BER violations occur if any pair of agents i and j have $\mathtt{BER}_{ij} > \mathtt{BER}_{thresh}$ which is the threshold maximum allowable error rate for the system to be effective.

Once BER violations are identified, the framework greedily prunes disconnected tasks. First, all tasks which violate BER with the base are set as disconnected. Next, the algorithm considers the data-rate routing Minimum Spanning Tree, and prunes its low value end-nodes with BER violations in an effort to avoid changes in satisfactory data-routing. Finally, if violations are still present, the algorithm greedily selects low value nodes to remove. Disconnected tasks are then greedily repaired by sequentially adding relays along the computed shortest BER path between the two neighboring nodes with weakest link until the threshold is met. This process is run in series with the one described in the previous subsection to ensure data-rate and BER requirements are met for all agents executing tasks at the time of execution. Relay tasks are then inserted into the CBBA with Relays algorithm to allow under-utilized agents to support the network.

This process does not ensure, however, that other agents not busy executing tasks at the time of network prediction are connected. In order to drive these other agents, which are either idle or traveling to their next task, to an interconnected state, a decentralized reactive motion control policy is proposed. Its objectives are to minimize additional fuel costs incurred from extra travel, but also ensure that agents can satisfy their proposed plans. Each agent i uses its current knowledge of other agent locations and communication environment parameters to locally estimate the network topology and BER values with other agents. It determines if it is at risk of violating BER with any other agent m if $\texttt{BER}_{\texttt{im}} > \texttt{BER}_{\texttt{plan}}$, where $\texttt{BER}_{\texttt{plan}}$ is a tunable planning threshold less than $\texttt{BER}_{\texttt{thresh}}$. It then plans to reconnect with the agent most at risk of being disconnected with, above this threshold, through the estimated minimum BER route. It computes its minimum distance $d_{min_{ij}}$ to the next agent j in the route needed for $\texttt{BER}_{\texttt{im}} = \texttt{BER}_{\texttt{plan}}$ and starts traveling to meet that requirement. If that distance $d_{min_{ij}}$ is less than a safety threshold for collision avoidance, it skips on toward the next agent k in the BER route. As agents move about the environment, $d_{min_{ij}}$ is dynamically updated to steer i. Because agents plan for discrete task execution times instead of for continuous time, it is not possible to guarantee that all agents will be interconnected at all times using this framework. However, results in Sect. IV show good performance of this strategy both in simulation and outdoor flight tests.

IV. Experiments & Results

A. Simulation

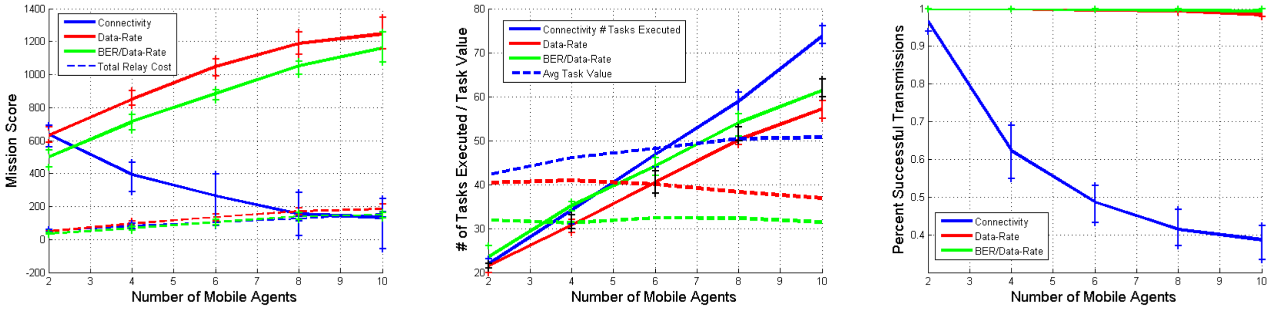
A simulation experiment was conducted to evaluate the proposed framework planning for different communication requirements. The mission scenario consisted of gathering information in the environment using small UAVs. Tasks were created at uniformly distributed random locations and times. The task value was scaled linearly as a function of its distance from the base (value = 0 at base vs. 100 at furthest location). This relates well to real-world remote sensing objectives which often consist of surveying out-of-reach areas. Tasks further away offer more reward, but require agents to cooperate to execute them successfully. The expected SNR at a distance of $R_{comm} = 30\%$ the environment size was used to establish required task data-rate and the single hop BER threshold. Reward was only generated when data-rate requirements to the base were supported at each time step. In other words, if a specific task was only supported 50% of the time during its execution, it received 50% of its reward.

Three planning strategies were evaluated in the mission. First, the team planned to meet over-simplified *connectivity* requirements only, where the network is supported so long as all agents are within R_{comm} of one another without regard for more complex networking dynamics (as was done in the original version of this work). Next, planning was performed to meet data-rate requirements, while considering information routing. Finally, planning was conducted to simultaneously meet data-rate and BER requirements through relay placement and using the reactive motion control policy. To compare the planning effectiveness in supporting the network, these experiments assumed a deterministic communication environment with no fading ($\sigma_{dB} = 0$). The experiment varied the team size from 2 to 10 UAVs, and 30 scenario trials were executed per data point.

Figure 1 shows the performance results of the experiments. All strategies performed similarly for small counts of agents. However as the team size grows, teams considering connectivity only, and not the more complex networking dynamics, quickly overload the network with their plans. As can be seen in Fig. 1 (b), agents planning for connectivity only without stricter requirements attempt more and higher value tasks. However Fig. 1 (c) shows these plans are too ambitious and often lead to data-rates not supported by the network (as low as 38.6% average supportability for teams of 10 UAVs) which results in significant mission performance degradations compared to other strategies considering data-rate.

Although data-rates are better supported when the planning algorithm specifically takes it into account, Fig. 1 (c) reveals a limitation of this control framework. Since network prediction is performed at discrete task execution times, as opposed to continuous time, the network is vulnerable to changes in routing caused by agents not active during this prediction (such as agents traveling to tasks). While network supportability and plans are still good, with 98.4% average network support in the **worst** case, perfect performance guarantees are not possible with this framework and real-world protocols.

Finally, while not plotted here, the experiment showed the performance of the system maintaining inter-agent connectivity. Teams not considering inter-agent BER requirements spend significant amounts of time disconnected from one another during the mission. On the other hand, if BER is considered as a requirement in addition to connectivity or data-rate alone, and inactive or traveling agents plan their motion properly, the team can maintain good inter-connectivity performance (98.2% in the worst case for small teams, and 99.95% with larger teams).



(a) Mission scores

(b) Number and value of tasks

(c) Data-rate support success rate

Fig. 1. Simulation experiment results

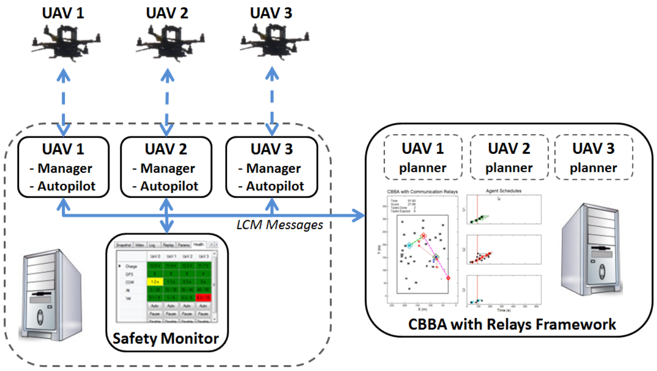


Fig. 2. CBBA with Relays flight test architecture

B. Outdoor Flight Testing

In order to demonstrate the algorithm in a more operationally realistic environment a field experiment was designed to show the capability of CBBA with Relays in outdoor flight testing. There are several challenges and ultimately benefits to implementing a distributed multi-agent system in an outdoor uncontrolled environment: (1) it demonstrates real-time functionality of the system, (2) the system is subjected to non-linear dynamics not considered in simulation due to imperfect modeling and environmental effects, and (3) it evaluates the robustness of the framework to system degradations, uncertainties, and imperfects which can significantly hinder the system from behaving as intended.

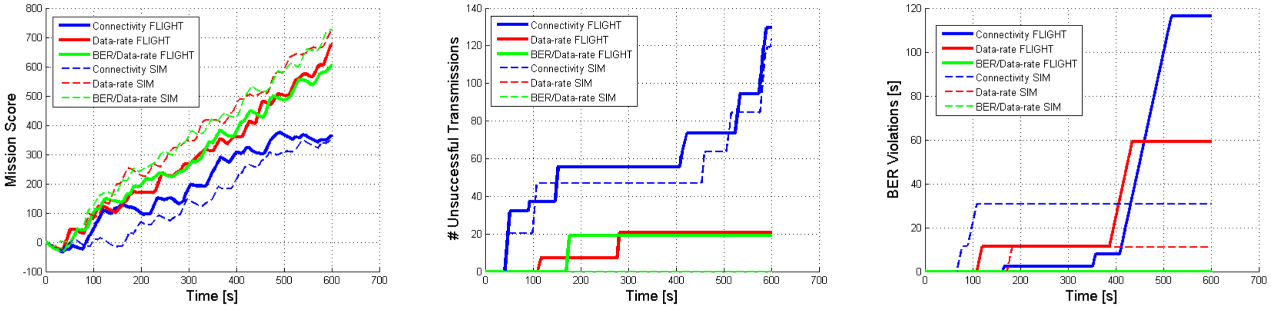
A similar scenario as used in the simulation experiments was employed in this experiment with a team of three Ascending Technologies Quadrotor UAVs. Three missions were executed to evaluate the different planning strategies described above: considering (1) connectivity only, (2) data-rate requirements, and (3) joint data-rate and BER requirements. Flight testing was conducted at a military base to take advantage of restricted airspace. Interested readers are referred to the previous installment of this work for a relevant description of the hardware implementation, which is summarized in Fig. 2. The communication network in this flight test is simulated using the model presented in Sect. II.

Results in Fig. 3 compare the planning strategies for the three flights. All three strategies are also compared to their respective performance predicted in simulation. Results first show that the two strategies planning to meet data-rate requirements outperform the original CBBA with Relays algorithm considering connectivity only which results in frequent link capacity overloads and missed reward. In addition, the green line in Fig. 3(c) indicates that the framework successfully maintains interconnectivity between nodes through relay placement and using motion control, as opposed to the two other strategies.

Interestingly though, results in Fig. 3(b) reveal three instances of overloaded capacity using the two strategies which specifically plan for this requirement (each step up in the graph represents a separate event) which were not predicted in simulation. In all three instances, an agent traveled slower than predicted due to head wind and arrived at its task late. In each case the task was still in its window of validity and agents started late anyways. Shortly after, other agents start other tasks which had not considered the late task still being active during network prediction in planning. This led to the three unpredicted overloads which were directly responsible for reducing the mission score in data-rate and joint BER/data-rate planning strategies compared to their simulated runs. Different solutions are proposed to fix this issue, but need further evaluation: (1) the planner could use more conservative velocities in planning, (2) network predictions could include time window buffers to account for possible task delays, and (3) agents could be forced to stop executing or not execute a task and stay put in the event of a late arrival. Overall though, the results show the algorithm generally works well and as intended, and improves the performance of the multi-UAV team subject to more realistic communication requirements.

V. Conclusion

The framework in this paper enables teams of multiple UAVs to cooperatively execute complex scenarios using dynamic mission planning to support communication requirements. The distributed algorithm uses task assignment information, including task location and proposed execution time, to predict the network topology and plan support using relays. By explicitly coupling task assignment and relay creation processes the team is able to optimize the use of agents to address the needs of dynamic complex missions. This also ensures the resulting network topology



(a) Comparison of mission scores

(b) Comparison of data-rate violations

(c) Comparison of BER violations

Fig. 3. Outdoor flight network control performance comparison using three different planning strategies

and information routing support data-rate and bit-error-rates required for the system to be effective.

On-going work is underway to enable the team to dynamically estimate unknown or changing communication environment parameters, such as the path loss exponent α, the fading variance σ^2, and the ambient noise density N_0 using network measurements. These values can then be dynamically updated in Eq. (1) with a tunable level of conservatism to formulate plans which account for uncertainty in network performance due to shadowing and fading. Furthermore, on-going work is investigating theoretical optimal routing protocols based on centralized linear programing to study the performance of the planning strategy without the limitations specific to individual real world routing mechanisms.

Future work will include incorporating robust planning strategies being investigated in [22] to plan for the uncertainty in vehicle dynamics which led to network violations in flight testing. In addition, a next step needed in this work is to validate the performance of the system in flight test with actual networking modules instead of a simulated network.

Acknowledgments

This research acknowledges support from Aurora Flight Sciences and Mike Klinker during field testing, and was financially supported by the Aurora Fellowship program.

References

[1] N. Jodeh and M. Mears, “An overview of the cooperative operations in urban terrain (counter) program,” in *AIAA Guidance, Navigation, and Control Conference (GNC)*, August 2008.

[2] M. Schwager, N. Michael, V. Kumar, and D. Rus, “Time scales and stability in networked multi-robot systems,” in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2011.

[3] S. Ponda, J. Redding, H.-L. Choi, J. P. How, M. Vavrina, and J. Vian, “Decentralized planning for complex missions with dynamic communication constraints,” in *American Control Conference (ACC)*, June 2010.

[4] A. Goldsmith, *Wireless Communications*. New York, NY: Cambridge University Press, 2005.

[5] C. Dixon and E. Frew, “Maintaining optimal communication chains in robotic sensor networks using mobility control,” *Mobile Networks and Applications*, vol. 14, no. 3, pp. 281–291, 2009.

[6] M. M. Zavlanos, M. B. Egerstedt, and G. J. Pappas, “Graph theoretic connectivity control of mobile robot networks,” *IEEE Journal*, vol. 99(9), pp. 1525–1540, 2011.

[7] M. A. Hsieh, A. Cowley, R. V. Kumar, and C. J. Taylor, “Maintaining network connectivity and performance in robot teams,” *IEEE Journal*, vol. 25(1-2), pp. 111–131, 2008.

[8] Y. Yan and Y. Mostofi, “Robotic router formation in realistic communicatin environments- a bit error rate approach,” in *IEEE Military Communications Conference*, 2010.

[9] R. Olfati-Saber, J. A. Fax, and R. M. Murray, “Consensus and cooperation in networked multi-agent systems,” *IEEE Proceedings*, vol. 95, no. 1, pp. 215–233, January 2007.

[10] E. Craparo, J. P. How, and E. Modiano, “Throughput optimization in mobile backbone networks,” in *SIAM Conference on Optimization*, May 2008.

[11] Y. Mostofi, A. G. Ruiz, A. Ghaffarkhah, , and D. Li, “Characterization and modeling of wireless channels for networked robotic and control systems - a comprehensive overview,” in *IEEE International Conference on Intelligent Robots and Systems*, 2009.

[12] P. Santi, *Topology Control in Wireless Ad Hoc and Sensor Networks*. Chichester, U.K.: John Wiley and Sons, 2005.

[13] E. Modiano, “16.36: Communication systems and networks,” Lecture Notes, Massachusetts Institute of Technology, Cambridge MA, 2012.

[14] C. Perkins and E. Belding-Royer, “Ad-hoc on-demand distance vector (aodv) routing,” in *IEEE Workshop on Mobile Computing Systems*, February 1999.

[15] T. Salonidis, M. Garetto, A. Saha, and E. Knightly, “Identifying high throughput paths in 802.11 mesh networks: a model-based approach,” in *IEEE International Conference on Network Protocols*, Octoboer 2007.

[16] 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*, 2012.

[17] H. Choi, L. Brunet, and J. P. How, “Consensus-based decentralized auctions for robust task allocation,” *IEEE Transaction on Robotics*, vol. 25, pp. 443–455, 2009.

[18] A. N. Kopeikin, A. Clare, O. Toupet, J. P. How, and M. L. Cummings, “Flight testing a heterogeneous multi-uav system with human supervision,” in *AIAA Guidance, Navigation, and Control Conference (GNC)*, August 2012.

[19] A. N. Kopeikin, S. S. Ponda, H.-L. Choi, and J. P. How, “Real-Time Dynamic Planning to Maintain Network Connectivity in a Team of Heterogeneous Unmanned Systems,” in *Wi-UAV 2011, 2nd International Workshop on Wireless Networking for Unmanned Autonomous Vehicles at the IEEE GlobeComm Conference*, Dec 2011.

[20] A. Ribeiro, “Notes on implementing a ieee 802.11s mesh point,” in *Euro NGI Workshop, Barcelona, Spain*, January 2008.

[21] D. Bertsimas and J. N. Tsitsiklis, *Introduction to Linear Optimization*. Belmont, MA: Athena Scientific, 1997.

[22] S. S. Ponda, L. B. Johnson, and J. P. How, “Distributed chance-constrained task allocation for autonomous multi-agent teams,” in *American Control Conference (ACC)*, June 2012 (to appear). [Online]. Available: http://acl.mit.edu/papers/ACC2012_ChanceConstrainedCBBA_final_submitted.pdf