2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

FrA04.5

Forest Fire Monitoring With Multiple Small UAVs

David W. Casbeer
Randal W. Beard
Timothy W. McLain
Brigham Young University

Sai-Ming Li
Raman K. Mehra
Scientific Systems Company, Inc.

Abstract— **Frequent updates concerning the progress of a forest fire are essential for effective and safe fire fighting. Since a forest fire is typically inaccessible by ground vehicles due to mountainous terrain, small Unmanned Air Vehicles (UAVs) are emerging as a promising means of monitoring large forest fires. We present an effective UAV path planning algorithm utilizing infrared images that are collected on-board in real-time. To demonstrate the effectiveness of our path planning algorithm in realistic scenarios, we simulated the propagation of a forest fire with the EMBYR model. A new cooperative control mission concept is indroduced where multiple Low-Altitude, Short-Endurance (LASE) UAVs are used for fire monitoring. By emplying multiple UAVs, the effectiveness of the mission in terms of information update rate can be improved dramatically.**

I. Introduction

Forest fires cause billions of dollars in damage to property and the environment every year. To combat forest fires effectively, early detection and continuous tracking is vital. Many methods have been developed to detect remote forest fires using satellite images [1], [2]. Such images are taken by low earth orbiting satellites with an orbital period of about ten hours and whose resolution is only sufficient for fire detection. However, fire fighters need frequent and high-quality information of a fire's development to conduct an effective and safe fire fighting mission. Because forest fire monitoring is difficult, fighters often enter the scene with little knowledge of how and where the fire is propagating, placing their lives at risk. For these reasons, there is an urgent need to develop more effective fire monitoring technologies.

Low-altitude, short-endurance (LASE) UAVs are expected to be a key technology for closing the fire monitoring gap. These UAVs can capture high resolution imagery and broadcast frequent updates to fire crews. NASA is actively pursuing this possibility with research aimed at tracking the growth of fires using LASE UAVs [3]. However, a number of challenges must be solved before LASE UAVs can be used. First, since the fire is growing and changing directions, UAVs need path planning ability using limited real-time information. Second, LASE UAVs cannot carry enough fuel to endure a long mission, which means they need the intelligence to return for refueling.

In this paper we present an effective path planning algorithm for UAVs tasked to monitor a forest fire. To demonstrate the effectiveness of our path planning algorithm in realistic scenarios, we implemented the fire model EMBYR in Simulink, which generates the time-evolution of a typical forest fire. We also introduce a new cooperative control mission concept utilizing multiple LASE UAVs to monitor the perimeter of the fire. By using multiple UAVs, the effectiveness of the mission in terms of the information update rate can improve dramatically. On the other hand, new problems such as coordination of UAV paths to cover the most critical areas, when and which UAV should be taken down for refueling, and how to measure the performance of the entire fleet of UAVs, must be addressed.

The paper is organized as follows: The fire monitoring problem is described in Section II, followed by a description the fire simulation model in Section III. Section IV describes the fire perimeter tracking algorithm used, with simulation results. Section V describes our cooperative control approach using a simplified scenario where the fire is assumed to be of circular shape. Finally, Section VI concludes the paper with comments on future work.

II. Problem Statement

Figure 1 shows the forest fire monitoring scenario considered in this paper. The orange pixels represent the areas where fire is burning, while the area enclosed by them represents the burnt area. A base station, represented by the red truck in Figure 1, sends out a UAV to monitor the propagation of the fire. The UAV's objective is to image the perimeter of the fire and upload the location of the fire perimeter (with associated imagery) to the base station as frequently as possible.

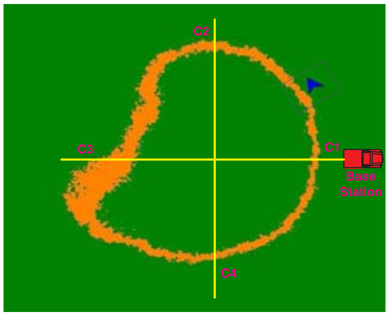


Fig. 1. Fire monitoring scenario.

The fire will spread in an unknown fashion, which implies the UAV cannot follow a pre-planned path. The UAV is assumed equipped with an infrared camera that captures images of a small region beneath it at regular sampling instants, indicated by the rectangle in Figure 1. The UAV can use the infrared images collected for navigational purposes. The UAVs are also assumed to have limited communication range, which means they cannot upload data to the base station unless they are within range of the station. Finally, the UAV is assumed to have limited flight range and must return to the base station for refueling after after a certain period of time.

To compare various fire monitoring algorithms, two simple performance metrics are used. Two perpendicular lines crossing at the geometric center of the fire define four points (C1, C2, C3, C4) around the periphery of the fire. The first metric is the update frequency from each checkpoint to the base station. The second metric is the time delay required to relay information from each checkpoint to the ground station.

III. The EMBYR Model

The EMBYR model [4]–[6] was chosen for this study because it is straightforward to implement and it produces simulations that closely resemble the propagation of actual forest fires.

EMBYR divides the landscape into a lattice of cells, in which single or multiple fires may be ignited. The fire propagates according to independent stochastic events at every time step. The fire can spread in a time step by two methods: diffusion from one cell to an adjacent cell or by firebrands being thrown to distant cells. The probability of spread as well as the rate of occurrence for firebrands are a function of the strength and direction of the wind and the combustibility of fuel types due to fuel moisture content.

1) Diffusive spread: The fire will spread from one cell to any of eight adjacent unburned cells with probability I_0 [5], [6], $(0 \leq I_0 \leq 1)$ in a given time step.

2) Wind and Slope Effects: The effects of wind and slope on the fire's propagation is modeled using the binomial distribution [5]. First, the probability of spread to an adjacent cell is calculated as $I_w = 1-(1-I_0)^{b_w}$, where b_w is a bias term proportional to wind in a given direction. Secondly, the probability of spread I is calculated using a bias term for the slope, b_s, as $I = 1-(1-I_w)^{b_s}$ [7].

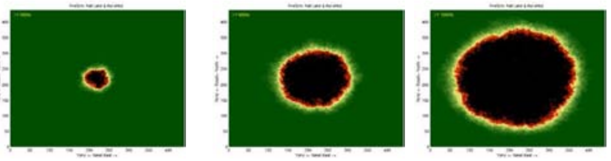
3) Firebrands: Firebrands are carried by updrafts due to the heat of the fire. The number of firebrands produced by a cell depends on its fuel moisture class and fuel type. The distance traveled by the firebrands increases with wind, and the likelihood of the targeted cell igniting depends on its fuel type and moisture class.

A. Simulated Fire Snapshots

The original EMBYR code is written in FORTRAN and C and is available online [8]. The essential parts of the model have been rewritten in MATLAB for this study. This code was run 100 times and averaged to achieve the simulations shown below. The terrains are 400 cells wide each with a constant moisture class and each cell's fuel type was generated randomly. In Figure 2, the black regions represent areas that have already burned, the yellow and orange colored areas are currently on fire, and the outside green colored regions are unburned.

We define the maximum speed of the UAVs to be $V_{\text{max}} \triangleq$ 30 mph = 13.4 m/sec. Given V_{max}, it will take 8.7 min for an UAV to travel 7 km. We also define a circle with a radius of 100 cells to have a circumference of 7 km, which makes one cell 11.1 m wide. To define the length of a time step (T_{step}), we let the maximum speed of the fire be V_{fmax} = 10 mph = 4.47 m/sec. Because the fastest moving fire advances at about 80 cells per time step each time step is found to be $T_{\text{step}} \approx 200$ sec.

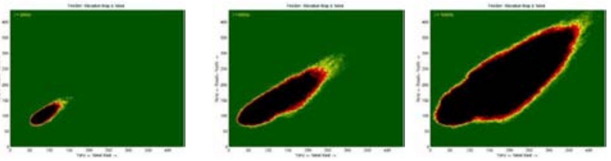
Figure 2 represents a fire on flat ground with no wind with the fire starting in the center of the image. In theory it would burn as a pure circle if there were uniform fuel type. It is burning outward from the original ignition point at a rate of about 0.12 m/sec or 0.56 km/hr.



(a) t = 3800 sec (b) t = 9800 sec (c) t = 15800 sec

Fig. 2. Fire simulation on flat land with no wind

In high wind situations the fire spread takes an elliptical pattern as shown in Figure 3 with the fastest front moving at about 10 m/sec. Here the fire was ignited from the bottom left.



(a) t = 2800sec (b) t = 6800sec (c) t = 10800sec

Fig. 3. Fire simulation of high wind conditions with an elevation gradient

These simulations are realistic enough to allow the study of fire perimeter tracking schemes using single and multiple UAVs.

IV. Path Planning Algorithm

We now consider tracking of the fire's edge by one UAV. We assume each UAV is equipped with an infrared camera and the processing ability to detect a fire edge in the image using either gradient operators or a thresholding method [9]. Further analysis of actual infrared fire images is needed to identify robust methods of detecting a forest fire's perimeter, and is beyond the scope of this paper. The following control algorithm relies on the UAV's current position, heading, and pixel coordinates belonging to the fire edge and it is assumed that the image processing algorithm identifies which side of the edge is burning.

We denote the coordinate frame of a vector by the subscripts i,v,b, and c, which represent the inertial, vehicle, body, and camera frames respectively. When the vector represents the position of the UAV in the given coordinate frame an p is added to the subscript. In the camera frame we define the x-axis as vertical and forward, with the y-axis being horizontal to the right.

The camera has a view angle of 30° and is forward looking in the xz-plane with respect to the z-axis by θ_c degrees and produces a picture of size $i_p \times i_p$ pixels. Here ψ is the heading of the UAV, which is flying at a constant altitude of h in meters by means of an autopilot [10]. The camera is kept at this angle with the use of pan and tilt gimbal control [11], [12].

Given these definitions the UAV will be be able to view the ground (assuming flat land) $D_f = h\tan(15^\circ - \theta_c)$ meters forward and $D_r = h\tan(15^\circ + \theta_c)$ meters in reverse. Letting the center of the image be the origin, the projected location of the UAV in the image is determined to be

$$\mathbf{x}_{pc} = \begin{bmatrix} x_{pc} \\ y_{pc} \end{bmatrix} = \begin{bmatrix} -\frac{i_p}{2} + \frac{D_r}{k} \\ 0 \end{bmatrix} , \tag{1}$$

where $k = \frac{D_r + D_r}{i_p}$ is a scaling factor with units of m/pixel.

We can transform a point $\mathbf{x}_c$ in the camera frame to $\mathbf{x}_i$ in the inertial frame by,

$$\mathbf{x}_i = \mathbf{R}_{b\to v} k(\mathbf{x}_c - \mathbf{x}_{pc}) + \mathbf{x}_{pi} , \tag{2}$$

where $\mathbf{R}_{b\to v}$ is a rotation matrix given by,

$$\mathbf{R}_{b\to v} = \mathbf{R}_{v\to b}^{-1} = \begin{bmatrix} \cos(\psi) & \sin(\psi) \\ -\sin(\psi) & \cos(\psi) \end{bmatrix}^{-1} \tag{3}$$

and $\mathbf{x}_{pi}$ is the UAV's position in the inertial coordinate frame.

We next describe the path planning algorithm. We assume the image processing stage passes an image of the fire's edge points along with a label indicating which side of the perimeter is burning. It is important that we have this second piece of information since folds in the perimeter and sharp edges would cause problems.

The first step is to approximate the fire edge by a straight line. This will help smooth the path of the UAV if there are lakes, rivers, or boulder fields that cause gaps in the fire edge as well as help with noise in the image. We note as a reminder that in the camera frame the y axis is aligned horizontally and the x axis is vertical. The parameters a and b for the straight line $x_c = ay_c + b$ in the camera frame can be estimated using a Least Squares criteria.

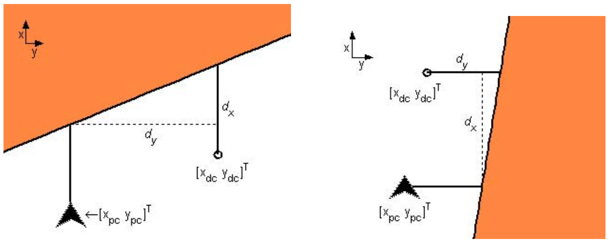
The path planning algorithm places waypoints along the approximated fire perimeter and the autopilot determine the control required to reach the waypoints. At every time step, a waypoint is placed before the UAV and set out from the straight line approximation. More precisely, let $d_p = -1$ if the UAV is circling the fire counterclockwise, otherwise $d_p = 1$. $\theta = \tan^{-1}(|a|)$, which is the absolute angle of inclination of the approximated fire perimeter. If $\theta \leq 45^\circ$ we will refer to the fire as being above or below the line. Similarly, for $\theta > 45^\circ$, the fire will be referred to as being to the left or to the right (in the camera frame) of the line. If the fire is below or to the right of the line we will assign direction parameter $d_f = 1$, otherwise we will assign $d_f = -1$. The variables d_f and d_p are used to determine the direction to move the next waypoint.

The waypoint is determined according to θ. This waypoint is then converted from the camera frame to the inertial frame so the autopilot can determine a good path to the waypoint. When $\theta \leq 45^\circ$ the next waypoint $[x_{dc}, y_{dc}]^\top$ is:

$$\begin{bmatrix} x_{dc} \\ y_{dc} \end{bmatrix} = \begin{bmatrix} ay_{dc} + b + d_f d_x \\ y_{pc} + d_f d_p d_y \end{bmatrix} \tag{4}$$

where d_y is measured along the y axis from the plane's position and the next waypoint (*i.e.,* $d_y = |y_{pc} - y_{dc}|$). d_x is the distance from the approximated fire perimeter to the next waypoint measured along the x axis. Visually the effects of d_f, d_p, d_y, and d_x can be seen in Figure 4a. Here the plane is flying counterclockwise thus $d_p = -1$ and since the fire is above the perimeter approximation $d_f = -1$. From equation 4 we see the next waypoint will move in the positive y direction from the plane's location since $d_f d_p = 1$ and the waypoint will be located below the fire perimeter approximation because $d_f = -1$. In simulations, we have used $d_y = 9$ pixels and $d_x = 13.5$ pixels. When $\theta > 45^\circ$ the next waypoint is:

$$\begin{bmatrix} x_{dc} \\ y_{dc} \end{bmatrix} = \begin{bmatrix} x_{pc} + d_f d_p d_x \\ (x_{dc} - b)/a - d_f d_y \end{bmatrix} . \tag{5}$$



(a) $\theta \leq 45^\circ$ (b) $\theta > 45^\circ$

Fig. 4. Next waypoint is shown as $[x_{dc}, y_{dc}]^\top$.

In Figure 4b the plane is flying clockwise yeilding $d_p = 1$. The fire is found on the right of the fire edge approximation making $d_f = 1$. The next waypoint moves a positive distance of d_x along the x axis since $d_f = d_p = 1$ in equation 5 and because $d_f = 1$ is subtracted, the waypoint is moved to the left of the fire edge by d_y. For simulations, $d_x = 9$ pixels and $d_y = 13.5$ pixels.

V. Fire Monitoring Using Multiple-UAVs

A centralized scheme will be developed where multiple UAV's cooperate to pass information concerning a fire to a base station. A brief discussion of a decentralized solution will then be given. The communication range for these simulations is assumed to be much smaller than the circumference of a forest fire, which is typically greater than 50 km.

Before proceeding, a few definitions are needed. To simplify the problem for study, the fire is assumed to be a spreading circle with radius R. We will be using the fire center as the origin of an inertial coordinate frame (i.e., X is north, Y is east, and Z points into the ground). The data will be delivered to a base station located at $p_b = [x_b \, y_b]^\top$. There are $2N$ planes with constant velocity V_m, each with the ability to track the fire's perimeter as described in IV.

Because the UAVs are able to track the fire autonomously, the explicit path along the fire perimeter can be eliminated from the coordination effort and the UAVs can simply coordinate rendezvous based on their relative angular position from the orgin. The position of the i^{th} UAV along the fire will be written as θ_i. We will measure θ_i counterclockwise with respect to east.

Because the fire's circumference is much larger than the UAVs communication range it is evident that the UAVs must either individually deliver the data collected to the base station or pass the data to another UAV that subsequently transmits this information to the base station. The data is delivered when $\theta_i = \theta_b$ where θ_b is the angle of the base station.

To measure the performance of the monitoring scheme, we assign four checkpoints along the fire's perimeter (c1 (East) to c4 (South) in Figure 5). The base station is c1 (*i.e* $\theta_b = 0$). Information is acquired as a UAV passes directly over a checkpoint and is exchanged when two UAVs are colocated on the fire perimeter. Two metrics will be monitored: the update frequency for each checkpoint and the maximum delay to relay information from each checkpoint to the base station. Three cases will be analyzed: single UAV, two UAVs, and greater than two UAVs.

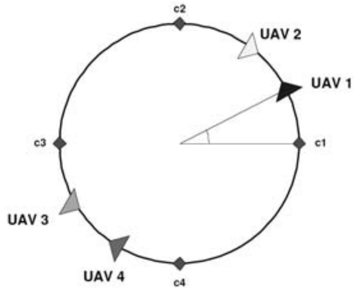


Fig. 5. c1: East, c2:North, c3:West, c4:South

A. Case 1: Single UAV

This is the trivial case where the optimal monitoring scheme is to follow the perimeter in a one direction. It will take

$$T = \frac{2\pi R}{V_m} \tag{6}$$

seconds to cover the entire fire perimeter once. Hence the update frequency for all checkpoints at the base station is $1/T$ Hz.

For all checkpoints, the maximum delay of information for the UAV is equal to T, as the UAV reaches each of them only once each cycle. The maximum delay of information for the base station, which is located at checkpoint 1, is also equal to $7T/4$ for c2 (assuming the UAV travels in the counter-clockwise direction).

B. Case 2: Two UAVs

In this case, the optimal strategy is to have the two UAVs start from checkpoints 1 and 3, and travel towards each other. When they meet, they can either continue in the same direction, or synchronize their data and reverse direction. Both strategies yield the same performance in this case. However, in the next case where $N > 1$, the latter strategy proves more efficient. Hence we will consider this strategy here as well.

Fig. 6 shows the angular positions of the two UAVs versus time. In this simulation we choose V_m such that $T = 12$ sec. Since the UAVs turn around whenever they meet, each UAV covers half of the circle (UAV 1 covers $(-\frac{\pi}{2}, \frac{\pi}{2})$, while UAV 2 covers $(\frac{\pi}{2}, \frac{3\pi}{2})$. We will label the two rendezvous angles by $p_{1rv} = \frac{\pi}{2}$ and $p_{2rv} = \frac{3\pi}{2}$. The rendezvous times at these two points are,

$$\begin{aligned} t_{1rv} &= \frac{2k+1}{2N} T + t_o \\ t_{2rv} &= \frac{2k}{2N} T + t_o, \qquad k = 0, 1, 2, \ldots \end{aligned}$$

where t_o, the initial rendezvous time was 3 sec.

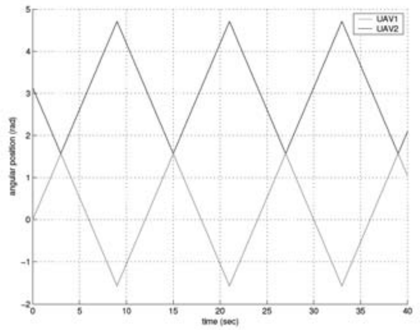


Fig. 6. Two UAV angular position

Fig. 7 shows the amount of time passed since the base station and UAV 1 were last updated with the data from the corresponding checkpoint. Notice that except for checkpoint 1, which coincides with the base station, the data's delay at the base station for all the checkpoints is always greater than or equal to the delay for UAV 1. This is because UAV 1 obtains updates of the other checkpoints through UAV 2 from time to time. However, until UAV 1 returns to c1, the base station will not have any updates.

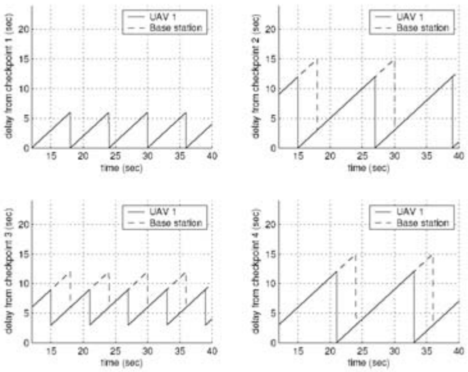


Fig. 7. Two UAV data delay from checkpoints: top left (c1), top right (c2), bottom left(c3), bottom right (c4). The solid line indicates time passed since UAV 1 has been updated while the dashed line shows the time delay since the base station has been updated with data from the specific waypoint.

As can be seen, the maximum delay (15 sec) of data for the base station occurs from checkpoints 2 and 4. Ironically, the maximum delay for checkpoint 3, which is the farthest from the base station, is smaller (12 sec). The update frequencies for data on checkpoints 1 and 3 has doubled to $2/T$ Hz compared to the single UAV case, while those for checkpoints 2 and 4 remain at $1/T$ Hz.

C. Case 3: $2N$ UAVs, $N > 1$

Here the optimal strategy starts at equidistant points around the perimeter of the fire. Alternate UAVs are assigned to two groups (odd or even). The two groups travel in opposite directions until meeting at rendezvous angles $p_{irv} = \frac{(2i+1)\pi}{2N}$, $i = 0, \ldots, 2N-1$. When UAVs meet, they turn around and travel in the opposite direction. Notice that odd subscripted rendezvous ideally occur simultaneously, as well as the even:

$$\begin{aligned} t_{i_o rv} &= \frac{2k+1}{2N}T + t_o \\ t_{i_e rv} &= \frac{2k}{2N}T + t_o, \qquad k = 0, 1, 2, \ldots \end{aligned}$$

where i_o indicates odd values in the range $i = [0, \ldots, 2N-1]$ and i_e indicates even values in the same range. In this example $t_o = 1.5$ sec.

Fig. 8 shows the angular position of the UAVs against time when four UAVs are used to track the fire perimeter. The trajectories are similar to those in the two UAV case except the portion of fire covered by each UAV is halved. Figure 9 shows the delay of data from each checkpoint on both UAV 1 and the base station. The frequencies of update for checkpoints 1 and 3 are both $4/T$ Hz, while those for checkpoints 2 and 4 are $2/T$ Hz. The maximum delay for checkpoint 3 is 9 sec, or $3T/4$ sec at the base station, which is a significant improvement from the two-UAV case. Maximum delays for other checkpoints are also shorter in this case.

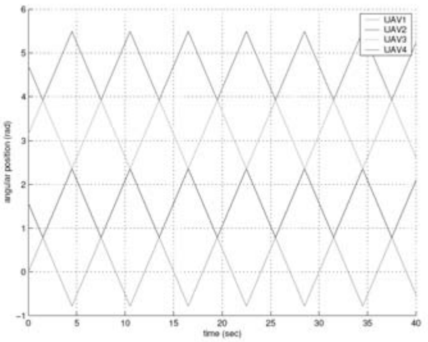


Fig. 8. Four UAV angular position

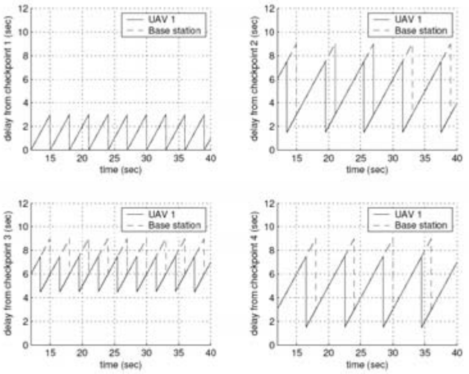


Fig. 9. Four UAV data delay from checkpoints: top left (c1), top right (c2), bottom left(c3), bottom right (c4). The solid line is the amount of time since UAV 1 has been updated with data from that checkpoint, while the dashed line is the amount of time delay since the base station has been updated.

Figure 10 shows the delay of data from each checkpoint for both UAV 1 and the base station when eight UAVs are deployed. The frequencies of update for checkpoints 1 and 3 are both $8/T$ Hz, while those for checkpoints 2 and 4 are $4/T$ Hz. The maximum delay for checkpoint 3 is 7.5 sec, or $5T/8$ sec at the base station, which is a significant improvement over the two-UAV case.

In general, when $2N$ UAVs are deployed, each UAV should to cover an arc of $2\pi/(2N)$ along the perimeter. Checkpoint 1 (the base station) lies in the middle of the path for UAV 1, hence the update frequency of checkpoint 1 is $2N/T$ Hz. We expect the update frequencies or checkpoints 2 and 4 to be half that for checkpoint 1, since UAV 1 gets updates from those checkpoints only once per cycle. Finally, since checkpoint 3 lies opposite checkpoint 1, the UAVs from both sides of the fire pass its updates to UAV 1, its update frequency is twice that of checkpoints 2 and 4, at $2N/T$ Hz.

Using this scheme, the data at each checkpoint is relayed to the base station in minimum time, *i.e.,* the time for a UAV to travel from the checkpoint to the base station. The maximum delay for each checkpoint consists of two components: (i) the time for the data to travel from the checkpoint to the base station; (ii) the time between updates for that checkpoint at the base station. The first component for checkpoints 2 and 4 are $T/4$, while that for checkpoint 1 is $T/2$. The second component for checkpoints 2 and 4 are $2 \times T/(2N) = T/N$, as it takes UAV 1 two cycles to update those checkpoints at the base station. For checkpoint 3, the corresponding time is $T/(2N)$. In summary, the maximum delays for checkpoints 2 through 4 are

$$\delta_{2,4} = \frac{T}{4} + \frac{T}{N} \tag{7}$$

$$\delta_3 = \frac{T}{2} + \frac{T}{2N} . \tag{8}$$

This can be verified from Figures 7 through 10.

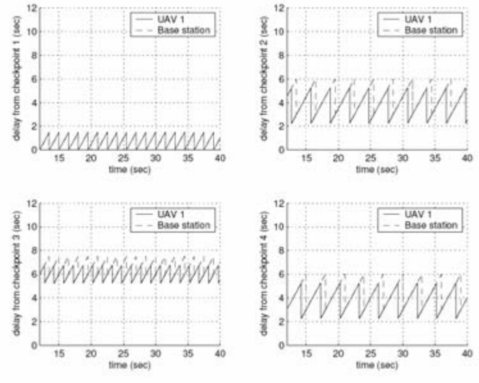


Fig. 10. Eight UAV data delay from checkpoints: top left (c1), top right (c2), bottom left(c3), bottom right (c4). The solid line is the amount of time since UAV 1 has been updated with data from that checkpoint, while the dashed line is the amount of time delay since the base station has been updated.

A problem arising with this scenario is fires tend not to be circular due to wind, elevation changes, and fuel availablilty. Thus mapping the fire's perimeter to θ_i is not one-to-one. However this is not important, since the explicit path can be decoupled from the coordination effort given the autonumous tracking in section IV. If the UAVs also have the ability to integrate their paths traveled, it is shown in [13] that the distance ($L = 0, \ldots, P$ where P is the perimeter length) from the base station along the perimeter can be used as a position measurement instead of angle (θ_i). In this case the variable P must be known, which can be estimated using cooperative timing [14] or consensus based approach [15] as the UAVs meet to pass information.

VI. Conclusions and Future Work

A vision based fire perimeter tracking algorithm has been shown to work effectively on simulated forest fires. This allows a UAV to follow the edge of a forest fire autonomously to obtain images of the spreading fire. It can be seen that by incorporating multiple UAVs, updates to fire fighters concerning the location and growth of a forest fire can be made more frequently and with greater detail.

While the approach presented has promise, numerous technical issues remain to be resolved including determination of the initial rendezvous time, dealing with fuel contingencies and refueling, implementation with irregular and growing fire shapes, and determining factors that allow the perimeter length to be updated frequently enough.

Acknowledgement

We would like to thank Chad Frost, Scott Poll, and Francis Enomoto at NASA Ames Research Center for initiating this research problem and pointing out related work in this area. This research was supported by NASA under STTR contract No. NNA04AA19C to Scientific Systems Company, Inc (SSCI) and Brigham Young University (BYU) and by the National Science Foundation under Information Technology Research Grant CCR-0313056.

References

[1] J.-I. Kudoh and K. Hosoi. Two dimensional forest fire detection method by using noaa avhrr images. In *2003 IEEE International Geoscience and Remote Sensing Symposium, IGARSS '03*, volume 4, pages 2494 – 2495, July 2003.

[2] J. Fujiwara, K.; Kudoh. Forest fire detection in 2001 using three-dimensional histogram. In *2002 IEEE International Geoscience and Remote Sensing Symposium, IGARSS '02*, volume 4, pages 2057 – 2059, June 2002.

[3] White paper on uav over-the-horizon disaster management demonstration projects, Feb 2000. http://geo.arc.nasa.gov/sge/UAVFiRE/whitepaper.html.

[4] D. Albright and B.N. Meisner. Classification of fire simulation systems. *Fire Managment Notes*, 59(2):5–12, Spring 1999.

[5] W.W. Hargrove, R.H. Gardner, M.G. Turner, W.H. Romme, and D.G. Despain. Simulating fire patterns in heterogeneous landscapes. *Ecological Modelling*, 135:243–263, 2000.

[6] R.H. Gardner, W.W. Hargrove, M.G. Turner, and W.H. Romme. *Climate change, disturbances, and landscape dynamics*, pages 149–172. Global Change and Terrestrial Ecosystems. International Geosphere-Biosphere Programme Book Series - Book #2. Cambridge University Press, Great Britain, 1996.

[7] W.W. Hargrove. Personal communication, 2004.

[8] http://research.esd.ornl.gov/hnw/embyr/.

[9] Rafael C. Gonzales and Richard E. Woods. *Digital Image Processing*. Prentice-Hall, New Jersey, 2nd edition, 2002.

[10] Derek Kingston, Randal Beard, Timothy McLain, Michael Larsen, and Wei Ren. Autonomous vehicle technologies for small fixed wing UAVs. In *AIAA 2nd Unmanned Unlimited Systems, Technologies, and Operations–Aerospace, Land, and Sea Conference and Workshop & Exhibit*, San Diego, CA, September 2003. Paper no. AIAA-2003-6559.

[11] Randal W. Beard, D.J. Lee, Sarita Thakoor, and Steve Zornetzer. A new approach to observation of descent and landing of future mars mission using bioinspired technology innovations. *AIAA Journal of Aerospace Computing, Information, and Communication*, 2004. (in review).

[12] Morgan Quigley, Michael A. Goodrich, Steve Griffiths, Andrew Eldredge, and Randal W. Beard. Target acquisition, localization, and surveillance using a fixed-wing mini-uav and gimbaled camera. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2005. (in review).

[13] David W. Casbeer, Derek B. Kingston, Randal W. Beard, Timothy W Mclain, Sai-Ming Li, and Raman Mehra. Cooperative forest fire surveillance using a team of small unmanned air vehicles. *International Journal of Systems Sciences*, 2005, To appear. Technical Report available at, https://dspace.byu.edu/handle/1877/55.

[14] Timothy W. McLain and Randal W. Beard. Coordination variables, coordination functions, and cooperative timing missions. *AIAA Journal of Guidance, Control, and Dynamics*, 2004. (to appear).

[15] Derek B. Kingston, Wei Ren, and Randal W. Beard. Cooperative timing using Kalman consensus. In *Proc. of the American Control Conference*, 2005. (Submitted for review.).