

# Strategies of Path-Planning for a UAV to Track a Ground Vehicle

Jusuk Lee, Rosemary Huang, Andrew Vaughn, Xiao Xiao, and J. Karl Hedrick\*

Marco Zennaro and Raja Sengupta‡

**Abstract**—In this paper, we present a strategy of path-planning for an unmanned aerial vehicle (UAV) to follow a ground vehicle. The ground vehicle may change its heading and vary its speed from a standstill up to the velocity of the UAV, while the UAV will maintain a fixed airspeed and will maneuver itself to track the ground vehicle. The algorithm also allows the UAV to track the ground vehicle with an offset vector (i.e. the user may wish the UAV to stay ahead of the ground vehicle or to its sides). Since the ground vehicle may operate in a range of velocities, the algorithm must plan the UAV's path with the appropriate schemes for the various ground vehicle speeds. The natural effect of wind injects a disturbance into the system, and so wind compensation techniques had to be developed. In order to maintain the focus of this project on path-planning strategies, the path-planning algorithm was implemented on top of a system that already controls the dynamics of the UAV. Simulation of aircraft and ground vehicles was performed with a hardware-in-the-loop simulation environment to test for mission feasibility. After attaining satisfactory simulation results, an experiment was conducted to confirm the path-planning strategy.

**Index Terms**—Aircraft navigation, Mobile robot motion-planning, Surveillance, Tracking, Unmanned aerial vehicles.

## I. INTRODUCTION

In order for a UAV to function in a useful manner in any application, care must be taken to construct a path to ensure mission success. Current techniques for path planning of UAVs are often done on a “per-application” basis, and some even require manual computation of navigation information in real-time, which severely hinders UAVs from achieving a more autonomous role [3]. Certainly, different applications of the UAV call for different path-planning strategies. For, example the path of a drone conducting border patrol missions will vary from that of one performing terrain mapping or planetary exploration [1, 2, 3]. Nevertheless, the high-level

navigation needs of many types of applications may be answered through waypoint navigation [4].

The purpose of this effort in UAV research is to provide local as well as “over-the-horizon” visual coverage for a ground vehicle from a UAV that is equipped with a camera. The constant aerial coverage from the UAV is achieved by flying the UAV autonomously over a region of interest. This region of interest may be directly on top of the ground vehicle, or be as far as a mile ahead of the ground vehicle's velocity vector. An additional requirement of maintaining a constant airspeed for the UAV is also imposed for fuel efficiency purposes. Not forgetting the human factor, a user-friendly interface must be generated to ensure maximum functionality of such UAV system [5].

To fulfill these demands, an algorithm based on waypoint strategy was created. Under the guidance of this navigation scheme, the UAV will fly in a sinusoidal manner and change the amplitude of the sinusoid, all the while maintaining a constant velocity and tracking the ground vehicle that has varying speed. Additionally, cases were also taken into account where the ground vehicle is at a stand still. Finally, simplicity was maintained when designing the user interface.

This paper focuses on the implementation strategies of tracking a ground vehicle using a UAV. Special emphasis is placed on the details of generating the sinuous path. Results from simulation as well as a real flight test are presented to demonstrate the effectiveness of this autonomous navigation scheme.

## II. PATH-PLANNING ALGORITHM

The central goal of the path-planning algorithm is to maneuver the UAV to track the movement of a ground vehicle. The tracking procedure may be at some offset distance with respect to the ground. In other words, the UAV could be half a kilometer to the east of the ground vehicle, and if the ground vehicle were to move north, the UAV must also fly north but maintaining the offset to the east of the ground vehicle. The tracking procedure must also change its strategy when the ratio of the UAV velocity,  $v_p$ , to that of the ground vehicle velocity,  $v_B$ , goes above a threshold ratio (in this application, approximately 3:1). This preset ratio is determined by the limitation of the UAV's autopilot avionics.

As a result, the behavior of the UAV is separated into two

Manuscript received May 5, 2003. (submitted for review on Feb. 24, 2003) This work was supported in part by the Office of Naval Research (ONR) under STTR Phase I Grant N00014-02-M-0236

\* Authors are with the Department of Mechanical Engineering at the University of California, Berkeley. (e-mail: {jlee, rosemary, xiao, acvaughn, khedrick}@vehicle.me.berkeley.edu).

‡ Authors are with the Department of Civil and Environmental Engineering at the University of California, Berkeley. (e-mail: {zennaro, raja}@path.berkeley.edu).

modes (*loitering* or *sinusoidal*) depending on the velocity of the ground vehicle. If the velocity of the ground vehicle is much slower than that of the UAV (i.e. velocity ratio is above the threshold), the UAV will be in the *loitering* mode; otherwise, the UAV will go into the *sinusoidal* mode. The sinusoidal curve in Fig. 1 illustrates the desired path the UAV will follow using the *sinusoidal* algorithm. The amplitude of the sinusoid,  $A$ , varies according to the ratio of  $v_p$  and  $v_B$ . In this figure, the UAV has no angular offset but only a distance offset, which leads the ground vehicle by a distance,  $d$ . This will be the assumption for the rest of the derivations in this section, as the concept may be extended to the case when a different angular and/or distance offsets are desired. In the figure the dashed line is the projected path the ground vehicle is to follow. It will be referred to as the ground vehicle's "travel path" from now on.

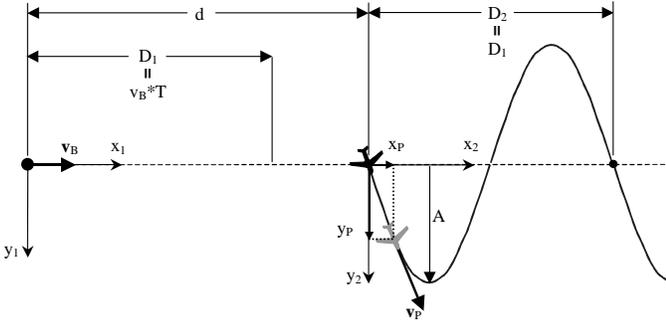


Fig. 1. Top view of path-planner algorithm in *sinusoidal* mode

In Fig. 1, two coordinate systems are shown, one fixed to the ground vehicle,  $x_1$ - $y_1$ , and the other the beginning of the sinusoid,  $x_2$ - $y_2$ . Both are oriented in the same direction with the  $x$ -axes pointing the direction of the ground vehicle's travel.  $D_2$  is the distance in the  $x$  direction the UAV will travel in one time period,  $T$ , of the sinusoid. It will equal the distance,  $D_1$ , that the ground vehicle will travel in the same direction. Thus,  $D_1 = v_B T$ , where the period  $T$  is arbitrarily chosen. Using this fact, the equation of a sinusoid can be transformed into an equation that describes the sinusoidal path in terms of the  $x_2$ - $y_2$  coordinate system,

$$y_p = A \sin\left(\frac{2\pi x_p}{D_1}\right). \quad (1)$$

Here,  $x_p$  and  $y_p$  are the desired position of the UAV relative to the stationary  $x_2$ - $y_2$  coordinate system. Taking the time derivative yields

$$\dot{y}_p = A' \cos\left(\frac{2\pi x_p}{D_2}\right) \cdot \dot{x}_p \quad (2)$$

where  $A' = 2\pi A/D_2$ . The magnitude of the UAV velocity,  $v_p$ , is related to its  $x$  and  $y$  components via

$$\dot{x}_p^2 + \dot{y}_p^2 = v_p^2. \quad (3)$$

Substituting equation 2 into 3 results in

$$\dot{x}_p^2 + A'^2 \cos^2\left(\frac{2\pi x_p}{D_2}\right) \cdot \dot{x}_p^2 = v_p^2, \quad (4)$$

which after some algebraic manipulation becomes

$$\dot{x}_p = \frac{v_p}{\sqrt{1 + A'^2 \cos^2\left(\frac{2\pi x_p}{D_2}\right)}} \quad (5)$$

Equation 2 and 5 are used in the implementation of this algorithm to calculate the desired path of the UAV. Proceeding further, an equation is now derived that will relate the ratio of the UAV velocity and the ground vehicle velocity with the amplitude. First, note that  $\dot{x}_p = dx_p/dt$ , which allows us to express equation 5 as the following integral,

$$\int_0^T dt = \frac{1}{v_p} \int_0^{D_2} \sqrt{1 + A'^2 \cos^2\left(\frac{2\pi x_p}{D_2}\right)} dx_p \quad (6)$$

Now, since

$$\int_0^T dt = T, \quad T = D_1/v_B, \quad \text{and} \quad D_2 = D_1, \quad (7)$$

Equation 6 becomes

$$\frac{D_1}{v_B} = \frac{1}{v_p} \int_0^{D_1} \sqrt{1 + A'^2 \cos^2\left(\frac{2\pi x_p}{D_1}\right)} dx_p, \quad (8)$$

Let the velocity ratio be  $\sigma = v_p/v_B$ , then

$$\sigma = \frac{1}{D_1} \int_0^{D_1} \sqrt{1 + A'^2 \cos^2\left(\frac{2\pi x_p}{D_1}\right)} dx_p, \quad (9)$$

This equation is used in the implementation to determine the  $A$ , amplitude of the sinusoid, based on  $\sigma$ . Equation 9 is a variation of a complete elliptic integral of the second kind, which means it can be expressed as

$$\frac{1}{\pi} E((-A^2)^{1/2}) \quad (10)$$

where  $E(\dots)$  is the aforementioned elliptic integral expressed in function form. [6].

Fig. 2 is a plot of  $\sigma$  versus the ratio  $A/D_1$  based on Equation 9. The plot shows that using this equation, the amplitude of the sinusoid will increase as the velocity ratio gets larger. This makes sense because a larger  $\sigma$  corresponds to an increasing  $v_p$  or a decreasing  $v_B$ . In both cases the amplitude needs to be enlarged to slow the rate at which the plane follows the travel path.

If the value of  $\sigma$  is above a certain threshold value,  $h_\sigma$ , the UAV will exit the *sinusoidal* mode that generates the trajectory as discussed above and enter into the *loitering* mode. The UAV will then loiter about a specified position, at the offset angle and distance, relative to the ground vehicle. The value of  $h_\sigma$  is set to avoid the large amplitude that would be caused by the high value of  $\sigma$ . If the value of  $\sigma$  decreases below  $h_\sigma$ , then the sinusoidal algorithm will continue.

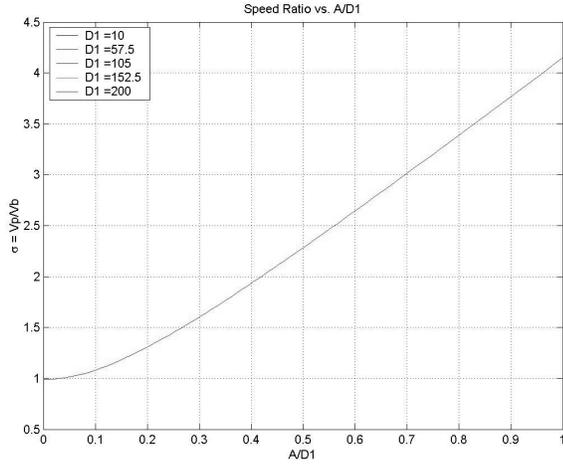


Fig. 2. Speed ratio,  $\sigma$ , vs amplitude/distance,  $A/D_1$ , ratio of path-planning algorithm (all  $D_1$  values overlap)

In the *loitering* mode, the UAV enters into a circle or rose curve trajectory (this is a user-defined option). In the circle trajectory, the plane circles about a set point and essentially maintains a constant bank angle. The rose curve is beneficial because it will allow a camera on the bottom of the plane to face the ground for a greater amount of time than when circling. The rose curve is created by giving the plane waypoints in a line and then after the plane has gone through those points the line is rotated about a fixed center at the desired offset from the ground vehicle. Once the line is rotated, waypoints are given along the new line. This pattern continues, until  $\sigma$  decreases below  $h_\sigma$ . In either of the path-planning modes, the UAV's offset is centered at the ground vehicle and defined by a cardinal direction,  $\theta$ , and a distance,  $L$ , as seen in Fig. 3.

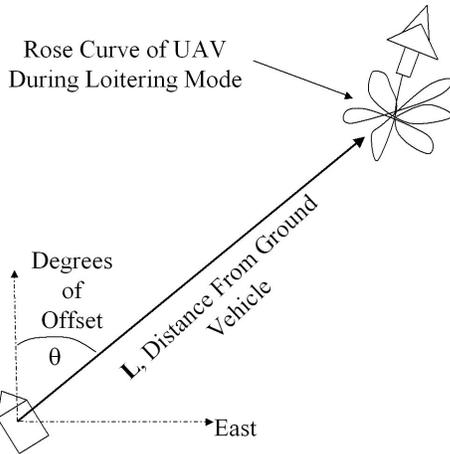


Fig. 3. Top view of path-planning algorithm in *loitering* mode

### III. WIND COMPENSATION

The ground velocity of the UAV is used for the path-planning algorithm. However, when wind is present the UAV's ground velocity changes, while the true air speed of the

UAV is kept constant. Therefore, not only will the UAV have difficulty following the sinusoidal path, but also the path-generation algorithm will also generate paths with undesirable features. Fortunately, the UAV has the capability to estimate the wind velocity, which can be used by the path-generation algorithm. This new path is offset at a ratio of the wind velocity vector; therefore adding or subtracting to the distance of the next waypoint for the UAV to go to. Additionally, a hysteresis was added to eliminate frequent switching between the *loitering* and *sinusoidal* modes that is caused by the fluctuation of the UAV's ground speed.

### IV. SOFTWARE PLATFORM

In order to most efficiently develop, test, and debug the system, a “controller development platform” (CDP) was developed on top of the low-level UAV controller software, which is provided by CloudCap Technology Incorporated [7]. The CDP facilitates the development process by taking care of the tasks of data collection, unit conversion, and communication. It simplifies the testing and debugging of any controller algorithm by offering a hardware-in-the-loop simulation environment and a real-time feedback of the controller behaviors through the CDP GUI. The hardware-in-the-loop strategy reduced development time, especially because the simulation software differs from the actual software merely by a few compiler flags.

The software architecture of the CDP is illustrated in Fig. 4. The control developer is only required to implement one of the modules, the *controller module*, while everything else is ready for test flight.

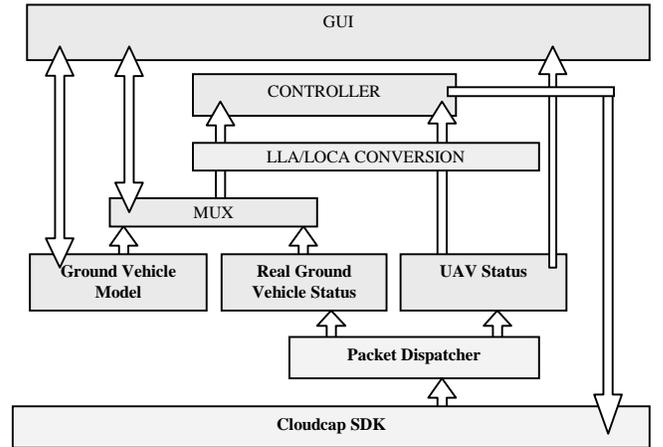


Fig. 4. CDP software architecture

The bottom layer of the CDP is the CloudCap Communication SDK, which is a library that provides communication primitives between the controller and the ground station through a serial port. The library comes with a simple packet dispatcher. Inside this module we nested our routines to forward the packets that we receive from the ground station to one of the two modules: the *real ground vehicle status module* or the *UAV status module*. In turn, these modules keep track of the status of their respective vehicles.

The MUX module is a set of switches that route the ground vehicle information appropriately.

The LLA / LOCA conversion module is used for coordinate transformations between the ground vehicle's local coordinate system (LOCA) and GPS Longitude Latitude Altitude format used by the UAV autopilot.

The interaction between the user and the software is managed through a *GUI module*. The user can observe the state of the UAV, the ground station, and the controller algorithm, and also command and alter the behavior of the path-planning controller. In addition, the GUI is used to "drive" the simulated ground vehicle during the development phase.

## V. SIMULATION IMPLEMENTATION AND RESULTS

We have successfully simulated the path-planning algorithm using the CDP and simulation hardware and software provided by CloudCap [7]. A ground vehicle simulation was also developed in order to aid in the simulation and send its status to the *ground vehicle model module* of the CDP. The discrete time equations for the position of the ground vehicle are given by Equations 11 and 12:

$$x_{Lat}(k+1) = x_{Lat}(k) + v_N \Delta T \quad (11)$$

$$x_{Long}(k+1) = x_{Long}(k) + v_E \Delta T \quad (12)$$

where the position of the ground vehicle model is reported in degrees of latitude,  $x_{Lat}$ , and longitude,  $x_{Long}$ . The velocity magnitude in the north direction is  $v_N$  and the velocity in the east direction is  $v_E$ . Since the algorithm is run once every second, an update of the ground vehicle model's position occurs at 1Hz, and so the  $\Delta T$  in Equations 11 and 12 is set to one.

The velocity vector of the ground vehicle is determined through the heading and the speed. Equations 13 and 14 calculate the velocity in terms of radius of the earth and in units of degrees.

$$v_N = \frac{\|v_B\| \cos(\psi)}{r_{Lat}} \cdot \frac{180}{\pi} \quad (13)$$

$$v_E = \frac{\|v_B\| \sin(\psi)}{r_{Long}} \cdot \frac{180}{\pi} \quad (14)$$

where  $r_{Lat}$  and  $r_{Lon}$  are the radius of the earth in latitude and longitude direction, respectively;  $v_B$  is the magnitude of the ground vehicle's velocity vector whereas  $\psi$  and heading.

The ground vehicle's position and velocity are used with the path-planning algorithm in order for a simulated UAV to follow a simulated ground vehicle before implementation with the actual plane and ground vehicle. The simulated plane was previously developed by CloudCap Technology.

Simulations were conducted to assist in the development of the path-planning strategies and to confirm that the software would work with an actual plane. The speed of the UAV is held constant at approximately 20-23 m/s throughout all of the simulations and experiments. The results of a simulation test with the ground vehicle heading south at a constant 10 m/s are

shown in Fig. 5. The value of  $\sigma$  is approximately 2, which is lower than the threshold value of  $h_\sigma = 3$ ; thus, the path is a sine wave. There is no wind in this simulation.

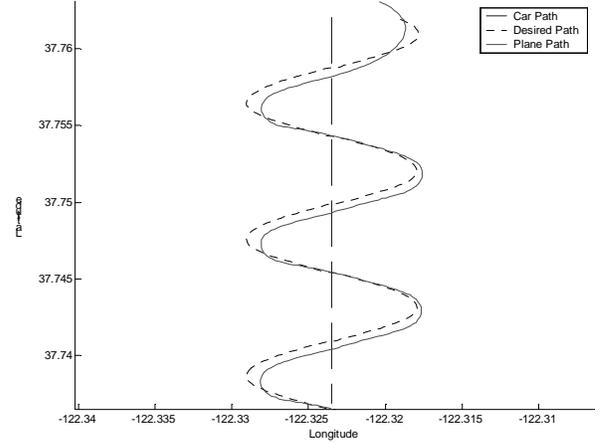


Fig. 5. Simulation of UAV and ground vehicle with no wind and the ground vehicle traveling south at a constant velocity

The next step in the simulation process is to test the wind compensation algorithm. The simulated plane estimates the simulated wind velocity, which is used in the wind compensation algorithm. The wind is simulated at 10 m/s coming from the south. First, a simulation was conducted without any wind compensation in the path-planning algorithm. The resulting path is shown in Fig. 6. The ground vehicle is traveling at 10 m/s to the north and then turns and heads at 10 m/s to the east. The UAV has difficulty following the sine wave with tail wind; the UAV goes too far and then has to cut back. When the UAV has a crosswind it stays too far away from the ground vehicle (to the side).

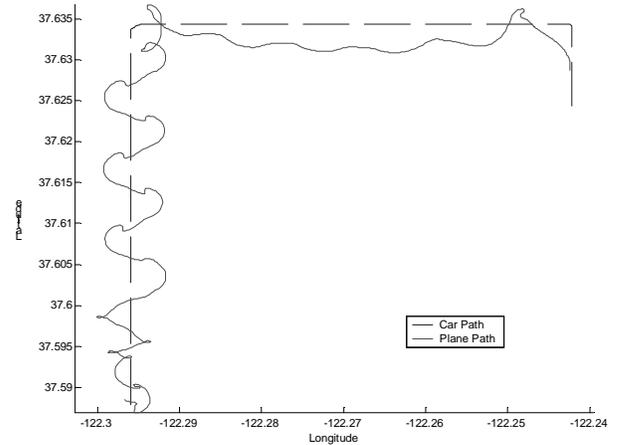


Fig. 6. Simulation with 10 m/s south wind, constant speed in the ground vehicle, and no wind compensation

Next, a simulation is shown with the wind compensation added (Fig. 7). The wind and vehicle conditions are the same as the previous experiment. Notice that the sine wave paths are much better and the path is centered over the ground vehicle. In the simulations and algorithms that we derived, we only worked with constant wind; gusts have not yet been considered.

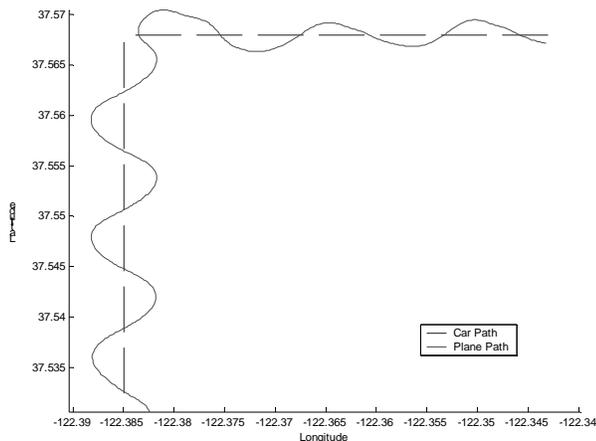


Fig. 7. Simulation with 10 m/s south wind, constant speed in the ground vehicle, and wind compensation added

Fig. 8 demonstrates the viability of the *loitering* and *sinusoidal* modes and the switch between the two modes. The ground vehicle is heading approximately north at 8 m/s and then comes to a halt. At that point the UAV enters into a *loitering* mode and starts circling the ground vehicle. After a couple of seconds, the offset distance is slowly increased so that the plane will loiter over a region ahead of the ground vehicle. (This can be seen by the circles that continue after the car path ends.)

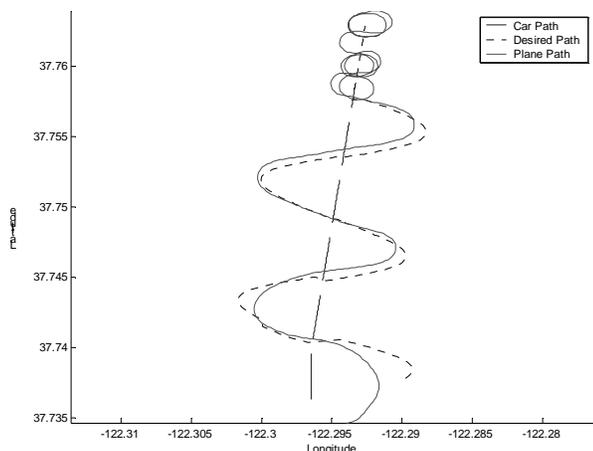


Fig. 8. Simulation with no wind and changing speed in the ground vehicle (switch from *sinusoidal* to *loitering* mode)

As demonstrated above, reassuring simulation results were attained, which increased the confidence for an experiment to verify the path-planning strategy.

## VI. EXPERIMENTAL RESULTS

A UAV furnished by Advanced Ceramics Research (ACR) was outfitted with CloudCap Technology's Piccolo® system for low-level control. The path-planning algorithm was incorporated into the CloudCap's ground station software, and the ground station was loaded in the bed of a truck.

The truck was driven at speeds varying from 0 to 45 mph throughout the test. At all times the UAV followed the motions of the truck by traveling either in a sine wave trajectory or loitering. There was no angular offset for the test. The UAV was set to be 40m in front of the truck at all times and then loiter directly above the truck. There were low wind conditions for the day of the test.

Fig. 9 shows the entire data from the experiment. The truck mainly made 90° turns, per constraint of the desolate desert highways in Tucson, Arizona. The truck first began to travel towards the west with a high value of  $\sigma$  (i.e. slow moving truck); therefore the UAV was in *loitering* mode. The truck then returned to the starting point and then began to travel south. At this point the  $\sigma$ -value is at around 2:1. Following, the truck headed toward the east with the same sigma value. Two miles later, the truck made a U-turn and reversed its route to return to the starting point. Throughout the return trip the  $\sigma$ -value was roughly 2:3. The long stretch throughout most of the plot had a low enough  $\sigma$ -value for the UAV to stay in *sinusoidal* mode.

Fig. 10 exhibits a close-up of the *loitering* mode. Notice the path is circular instead of a rose curve. This was the loitering pattern chosen for the day of the experiment. Fig. 11 separates the long east to west stretch of the experiment from the rest of the experimental data. The experimental results shown here verify the simulation results.

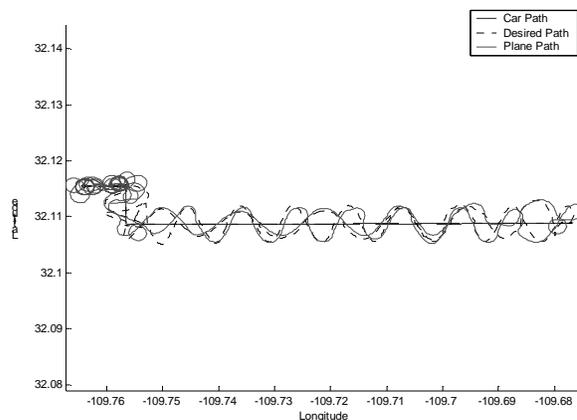


Fig. 9. Data from the entire experiment, with the UAV following a truck. The truck started on the west side of the plot and drove east at varying speeds, then turned back and retraced its path.

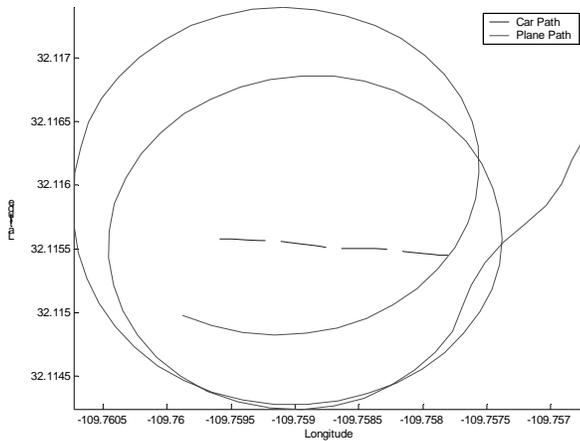


Fig. 10. Close up of experimental data highlighting *loitering* mode

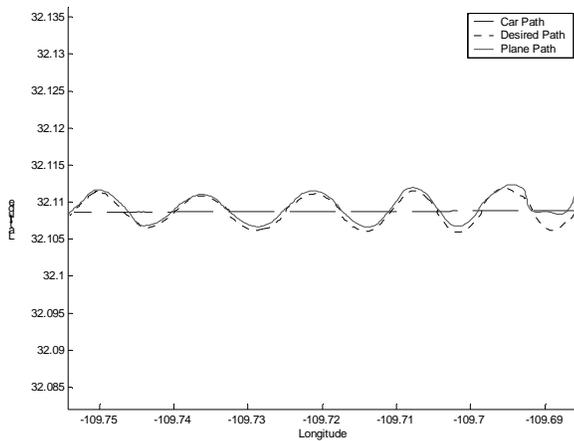


Fig. 11. Close up of experimental data highlighting *loitering* mode. The truck was traveling from the east to the west.

The real time video feed provided situational awareness coverage at nearly all times during the test. The picture in Fig. 12 was captured from the video footage provided from onboard the plane. The picture displays the plane's view as it passes over the truck. The points at which the plane path and truck path cross are essentially the zero-point of the plane's sine wave trajectory.

## VII. CONCLUSIONS AND FUTURE WORK

We have presented a path-planning strategy for an unmanned aerial vehicle (UAV) to follow a ground vehicle, which can change its headings and velocity. If the ground vehicle is not moving, or its speed is under a selected threshold, the UAV starts to loiter, following a circular or rose curve trajectory. When the vehicle is moving above the threshold, the UAV follows it along a sinusoidal trajectory with dynamically adjusting amplitude to compensate for vehicle speed changes.

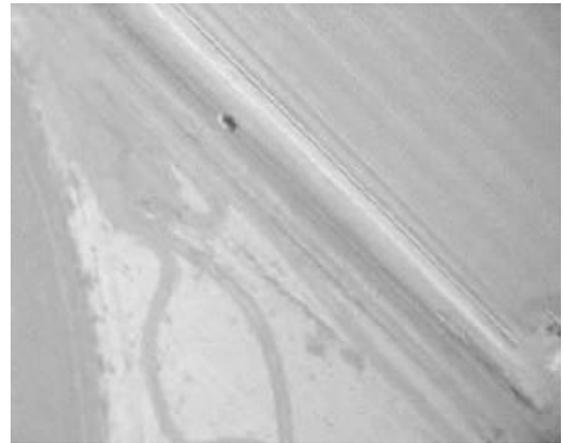


Fig. 12. Picture of video taken from a camera on the bottom of the UAV.

The wind introduces a disturbance in the system that has been addressed by using the calculated wind velocity and offsetting the planned UAV trajectory accordingly.

The path-planning algorithm has been developed, tested and debugged using the "controller development platform" we have implemented on top of the simulation software and hardware provided by CloudCap Technology.

The system has been successfully tested on a real UAV. The experimental results reflect the results obtained in the simulation phase. The ground vehicle was successfully tracked even under ground vehicle speed and heading changes.

The authors are currently working on extending the strategies presented in this paper to incorporate multiple vehicles. In order to do so, the CDP has to be extended to be able to support multiple vehicles simulations, and the control has to be expanded to avoid collisions and coordinate the motion of the UAVs.

## REFERENCES

- [1] D.A. Schoenwald, "AUVs: In space, air, water, and on the ground", *IEEE Control Systems Magazine*, 20(6):15-18, December 2000.
- [2] K. C. Wong, C. Bil, G. Gordon, P.W. Gibbens, "Study of the Unmanned Aerial Vehicle (UAV) market in Australia", *Aerospace Technology Forum Report*, August 1997.
- [3] *Unmanned Aerial Vehicles Roadmap: 2002-2027*, Office of the Secretary of Defense, December 2002.
- [4] M. Niculescu. Lateral track control law for Aerosonde UAV. *39th AIAA Aerospace Sciences Meeting and Exhibit*, Paper 2001-0016, January 2001.
- [5] S. Banda, J. Doyle, R. Murray, J. Paduano, J. Speyer, and G. Stein, "Research needs in dynamics and control for uninhabited aerial vehicles," <http://www.cds.caltech.edu/murray/notes/uav-nov97.html>, November 1997, Panel Report.
- [6] M. Abramowitz and I.A. Stegun, *Handbook of Mathematical Functions*, Dover Publications, 1965, 17.6.
- [7] CloudCap Technology – Piccolo avionics system for small Unmanned Aerial Vehicles. <http://www.cloudcaptech.com/piccolo.htm>, 2001-2003.