Using Direct Methods for Terminal Guidance of Autonomous Aerial Delivery Systems

Oleg A. Yakimenko and Nathan J. Slegers

Abstract—**This paper discusses the usage of direct methods of calculus of variations for real-time optimization of a final turn-into-the-wind maneuver for autonomous guided parafoil-based delivery systems. It reviews several approaches that are currently being pursued by different developers of such systems and discusses the necessity of usage and applicability of a direct method to optimize the final turn. The proposed approach seeks for an optimal solution within a certain class of parameterized inertial trajectories and further employs inverse dynamics to find the corresponding control. This approach has been successfully used in a variety of real-time applications already and proved to work well in a developed miniature aerial delivery system, Snowflake, as well. The paper presents the results of the most recent real drops and ends with a discussion of the developed algorithm as compared to another one, pursued by Draper Laboratory, where instead of parameterizing a reference trajectory the control itself is being parameterized. The paper ends with conclusions.**

I. Introduction

During the last decade there were many efforts to replace old-fashioned uncontrolled circular parachutes with maneuverable autonomously guided ram-air parafoils that can assure more accurate delivery of a variety of payloads from high altitudes and large standoff distances [1].

Autonomous parafoil capability implies bringing the aerial delivery system (ADS) to a desired landing point from an arbitrary release point using onboard computer, sensors and actuators. The navigation subsystem of the guidance, navigation and control (GNC) unit onboard the ADS manages data acquisition, processes sensor data and provides guidance and control subsystems with information about parafoil states. Using this information along with local wind profiles, the guidance subsystem plans the mission and generates a feasible trajectory to the desired landing point. Finally, it is the responsibility of the control system to track this trajectory using the information provided by the navigation subsystem and onboard actuators.

A variety of different-weight ADSs were developed and demonstrated during Precision Airdrop Technology Conference and Demonstration (PATCAD) in 2001, 2003, 2005 and 2007 [2,3] in the U.S. and Precision Airdrop Demonstration Capability (PCAD) 2006 and 2008 [4] near Bordeaux, France. Analyzing the results of dozens of airdrops however it can be stated that practically all systems have difficulties meeting their circular error probable (CEP) accuracy requirement (established at 75m for the lightest systems, below 15kg, all way up to 300m for the 19-ton systems). The major reason for that are unknown and variable winds near the ground. It is relatively easy to bring the system close to the target (usually upwind), while an altitude reserve still exists, but it is quite difficult to decide when to exit an energy management (EMGMT) pattern and actually fly towards the target, especially aiming at landing into the winds (for soft landing).

This paper specifically concentrates on the terminal phase of a guided descent and is organized as follows. Section II presents some of the known strategies and advocates the necessity of real-time optimization during the last portion of a descent to mitigate the effect of changing winds. Section III presents general ideas of the direct method of calculus of variations as the only tool to handle real-time terminal guidance optimization. Section IV proceeds with a detailed derivation of optimization algorithm based on the inverse dynamics in the virtual domain (IDVD) method. Section V describes the miniature prototype of a generic ADS that was built and extensively tested with the developed algorithms followed by the results of the most recent drops. Section VI introduces and discusses a slightly different approach, still based on the direct method, recently proposed by the Draper Laboratory, followed by Section VII, which compares two approaches. The paper ends with conclusions.

II. Terminal Guidance Strategies

Usually, industry does not disclose control algorithms employed on their systems. However, some of the ideas can be easily picked up by analyzing the test results (if available). To this end, Figs.1 and 2 present a bird-eye view of typical trajectories of different ADSs (demonstrated at PATCAD and PACD) that are trying to steer towards a target (the authors tried to pick the best ones among those that were published).

To start with, some of the systems employ either way-point navigation or some kind of heuristic algorithms that mimic behavior of a human jumper. As seen from Fig.1, it may lead to not only missing the target, but also to a failure to land into the wind. Once again, the major reason for this is always unknown winds below the altitude the ADS is currently at.

Figure 2 presents another approach pursued by three other companies. They bring their ADSs to the target and further spiral down trying to keep it in the middle. At a certain altitude (based on an estimate of a current velocity) the algorithm commands to exit a spiral towards the target. Theoretically, in

Manuscript received June 15, 2009.

O. A. Yakimenko is with the Department of Mechanical and Astronautical Engineering of Naval Postgraduate School, Monterey, CA 93943-5107 USA (phone: 831-656-2826; fax: 831-656-3125; e-mail: oayakime@nps.edu).

N. J. Slegers is with the Mechanical and Aerospace Engineering of the University of Alabama, Huntsville, AL 35899 USA.

ISBN 978-963-311-369-1

this case a miss distance is roughly limited by the radius of a spiral. However in practice, the ground winds take their toll and may steer a system further away. It is especially true for the two-stage Screamer ADS, when at a certain altitude a main round chute, bringing the system down to the ground uncontrollably, deploys. Of course, neither of systems attempt to land into the wind. (It should be noted though that this moth-pattern algorithm could be easily modified to allow landing into the winds [5].)

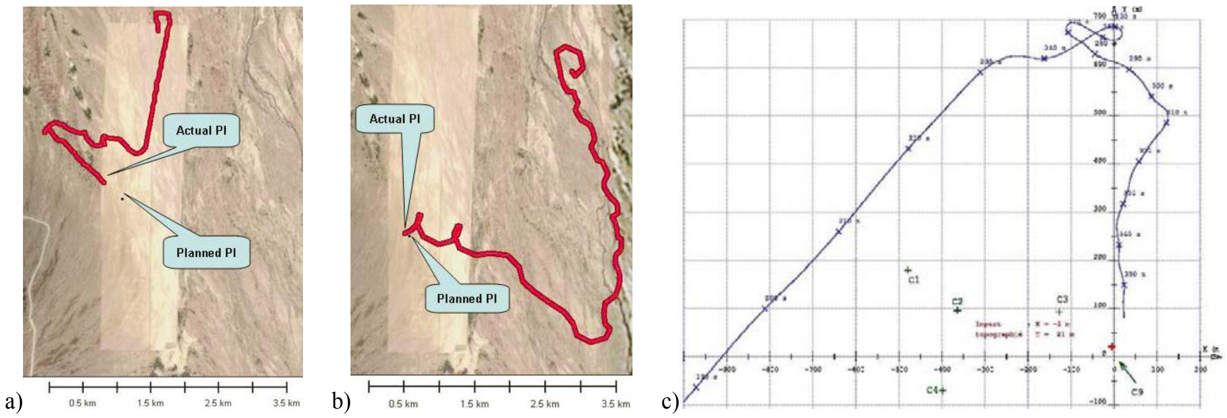

Fig. 1. Examples of a flight path of: a) Panther ADS (by Pioneer Aerospace Corp. / Aerazur) [2], b) Sherpa ADS (by MMIST, Inc.) [2], and c) SPADES ADS (by Dutch Space) [4].

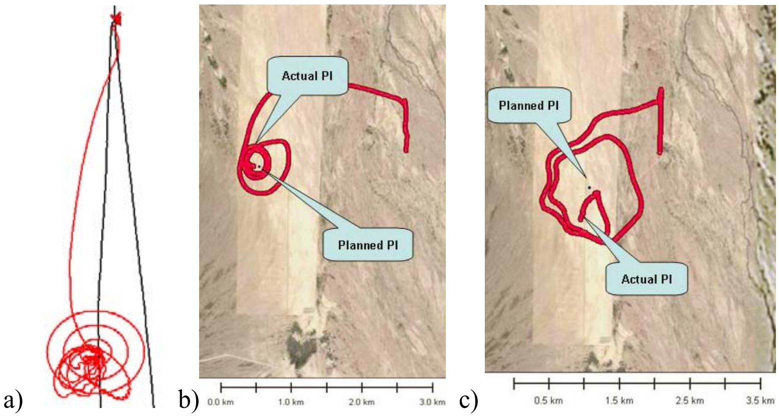

Fig. 2. Examples of a flight path of: a) Mosquito ADS (by Stara technologies, Inc.) [1], b) Onyx ADS (by Atair Aerospace, Inc.) [2], and c) Screamer ADS (by Strong Enterprises) [2].

Since it is well understood that terminal guidance determines touchdown accuracy it is crucial to adapt the guidance algorithm (computing a reference trajectory) to changing winds, i.e. to allow constant real-time re-computation of the final turn maneuver to accommodate wind disturbances as much as possible.

One such attempt realized on large systems was a GNC unit developed by Draper Laboratory [6, 2]. Being unable to optimize a maneuver in real-time, an optimized table-look-up terminal flight path was chosen based on the current conditions while entering a terminal area. This approach was not robust, and lead to the latest development using a direct method of calculus of variations to be able to produce a quasi-optimal trajectory in real time [7].

It turns out that direct methods have been used in flight mechanics since the 60s [8]. Specifically, the IDVD method [9] was recently used for a variety of real-time applications for unmanned underwater and air vehicles, rotorcraft, spacecraft and missiles. Also, it was suggested to be used for coordinated payload delivery [10]. Recently, it was applied to terminal guidance of ADS [11].

In what follows, we first consider the general ideas behind using direct methods in flight mechanics, followed by the approach and results of using IDVD for ADS final turn optimization. Then we will come back and review a Band-Limited Guidance (BLG), put forward by Draper Laboratory, from the standpoint of direct methods.

III. Basics of Direct Methods

To approximate the Cartesian coordinates of a vehicle and its speed (four states), Prof. Taranenko suggested using the following continuous, unequivocal and differentiable functions [12]

$$x_i(\tau) = x_{i0} + \frac{x_{if} - x_{i0}}{\tau_f - \tau_0}(\tau - \tau_0) + \Phi_i(\tau)\,,\ i = 1,...,4\,. \qquad (1)$$

Obviously these functions automatically satisfy initial conditions (IC) $x_i(\tau_0) = x_{i0}$ and terminal conditions (TC) $x_i(\tau_f) = x_{if}$ as long as $\Phi_i(\tau_0) = \Phi_i(\tau_f) = 0$. As an argument τ any monotonically changing parameter, e.g., time, path, full mechanical energy, etc. can be used. However, to decouple time and space, Taranenko suggested using virtual arc as oppose to time as an argument. The so-called speed factor

$$\lambda = \frac{d\tau}{dt}\,, \qquad (2)$$

allows to map the virtual domain to the physical domain.

Functions $\Phi_i(\tau)$, depending on several varied parameters a_{ik} , $k = 0,1,...$ define a variety of candidate trajectories, and their choice depends on a specific problem. In general, the more terms (varied parameters) the functions $\Phi_i(\tau)$ have, the more accurate (closer to the really optimal) solution can be found. Some unknown coefficients a_{ik} can be found using the boundary conditions (BC) imposed on the first and higher-order derivatives $x_i^{(l)} = \dfrac{d^l x_i}{d\tau^l}$, $l = 1, 2, ...$, while other (free) coefficients become varied parameters in addition to τ_f .

Once the Cartesian coordinates and speed are defined using reference functions (1), the remaining states and controls are determined using inverse dynamics of the original non-linear equations driving the system's dynamics. All the states and control history are then mapped to the time domain using (2).

Taranenko's direct method has a huge advantage over indirect methods. As known, in classical optimal control the problem is reduced to determining the trajectory with the given initial states and control time-histories by solving the Cauchy problem, i.e. integrating differential equations. Taranenko's direct method in general does not require solving the Cauchy problem. Instead, having the desired trajectory from the very beginning the inverse dynamics is applied to retrieve time-histories for all the controls.

Another approach could be to start from parameterizing controls time histories. However in this case the Cauchy problem will need to be solved. As a result: *i*) the TC will not be satisfied automatically as in (1), *ii*) integrating equations of motion will significantly slow the optimization routine, and finally, *iii*) the convergence is not guaranteed. These are the pitfalls of the BLG approach as will be discussed in Section VI.

Prof. Taranenko's approach was further advanced and perfected in IDVD method, specifically designed for real-time applications [9]. The details of its application to terminal guidance of ADS can be found in [11]. For consistency, the following section briefly repeats the main results.

IV. Inverse Dynamics in Virtual Domain

In an attempt to mitigate the effect of unknown variable winds we run into the following two-point boundary-value problem (TPBVP) (Fig.3). Staring at some initial point at $t = 0$ with the state vector defined as

$$\mathbf{x}_0 = \left[x_0, y_0, \psi_0\right]^T \tag{3}$$

(x - downrange, y - crossrange and ψ - heading) we need to bring our ADS influenced by the last known (estimated onboard) constant wind $\mathbf{w} = \left[W, 0, 0\right]^T$ to another point

$$\mathbf{x}_f = \left[(\hat{V}_h^* + W)t_{app}^{des}, 0, -\pi\right]^T \tag{4}$$

at $t = t_f$ ($\hat{V}_h^*$ is the estimate of a steady-state horizontal speed and t_{app}^{des} is the desired final approach time t_{app}^{des}). Figure 3 shows the portion of a guided descent we would like to optimize (shown with two vertical lines) that occurs between some turn initiation (TI) point at altitude $\hat{z}_0$ (defined by the estimates of W, and the components of the ADS velocity vector) and final approach capture (FAC) point.

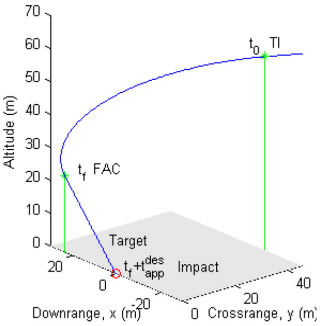

Fig. 3. Final turn and approach.

Hence, we need to find the trajectory that satisfies these BC along with the constraint imposed on the control (heading rate), $|\dot{\psi}| \le \dot{\psi}_{\max}$, and allows completion of the maneuver in exactly

$$\Delta t_{turn} = \frac{\hat{z}_0}{\hat{V}_v^*} - t_{app}^{des} \tag{5}$$

($\hat{V}_v^*$ is the estimate of a steady-state descent rate). The optimal control $\dot{\psi}_{opt}(t)$ that does the job is then to be tracked by the ADS' GNC unit. Obviously, the unaccounted winds $\mathbf{w}^{dist}(h) = \left[w_x, w_y, w_z\right]^T$ will not allow exact tracking of the calculated optimal trajectory. Therefore, the optimal trajectory needs to be constantly updated during the final turn, each time starting from the current (off the original trajectory) IC and still forcing the ADS to be at (4) within an updated Δt_{turn} (5).

Let us start from the simplest kinematic equations:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} -W + V_h^* \cos\psi \\ V_h^* \sin\psi \end{bmatrix}. \tag{6}$$

From these two equations it follows that if the final-turn trajectory is given, the yaw angle along this trajectory is related to the change of inertial coordinates as follows:

$$\psi = \tan^{-1} \frac{\dot{y}}{\dot{x} + W}. \tag{7}$$

Differentiating this equation provides with the yaw rate control required to follow the reference final-turn trajectory in presence of a constant wind W:

$$\dot{\psi} = \frac{\ddot{y}(\dot{x} + W) - \ddot{x}\dot{y}}{(\dot{x} + W)^2 + \dot{y}^2}. \tag{8}$$

Now, following the general idea of IDVD we will assume the solution of the TPBVP to be represented analytically as the functions of some scaled abstract argument $\bar{\tau} = \tau\tau_f^{-1} \in [0;1]$ as $x(\bar{\tau}) = P_1(\bar{\tau})$ and $y(\bar{\tau}) = P_2(\bar{\tau})$, with $P_\eta(\bar{\tau})$, $\eta = 1, 2$ being:

$$P_\eta(\bar{\tau}) = a_0^\eta + a_1^\eta \bar{\tau} + a_2^\eta \bar{\tau}^2 + a_3^\eta \bar{\tau}^3 + b_1^\eta \sin(\pi\bar{\tau}) + b_2^\eta \sin(2\pi\bar{\tau}) \tag{9}$$

The coefficients a_i^η and b_i^η in this formula are defined by the BC up to the second-order derivative at $\bar{\tau} = 0$ and $\bar{\tau} = 1$. According to the problem formulation and Eqs.(3),(4),(6) these BC are as follows:

$$\begin{bmatrix} x \\ y \end{bmatrix}_{\bar{\tau}=0} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}, \begin{bmatrix} x \\ y \end{bmatrix}_{\bar{\tau}=1} = \begin{bmatrix} (V_h^* + W)t_{app}^{des} \\ 0 \end{bmatrix};$$
$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix}_{\bar{\tau}=0} = \begin{bmatrix} V_h^* \cos\psi_0 - W \\ V_h^* \sin\psi_0 \end{bmatrix}, \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix}_{\bar{\tau}=1} = \begin{bmatrix} -V_h^* - W \\ 0 \end{bmatrix}; \tag{10}$$
$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix}_{\bar{\tau}=0} = \begin{bmatrix} -\dot{\psi}_0 V_h^* \sin\psi_0 \\ \dot{\psi}_0 V_h^* \cos\psi_0 \end{bmatrix}, \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix}_{\bar{\tau}=1} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Note, while TC (10) will be permanent (the second-order derivatives are zeroed for a smooth arrival), the IC will reflect the current state of the system at each cycle of optimization. The mapping between the virtual domain $[0;\tau_f]$ and physical domain $[0;t_f]$ is carried using the speed factor (2) as described in [11].

Differentiating Eqs.(9) two times with respect to τ

$$\begin{aligned}\tau_f P'_\eta(\bar\tau) &= a_1^\eta + 2a_2^\eta\bar\tau + 3a_3^\eta\bar\tau^2 + \pi b_1^\eta\cos(\pi\bar\tau) + 2\pi b_2^\eta\cos(2\pi\bar\tau)\\ \tau_f^2 P''_\eta(\bar\tau) &= 2a_2^\eta + 6a_3^\eta\bar\tau - \pi^2 b_1^\eta\sin(\pi\bar\tau) - (2\pi)^2 b_2^\eta\sin(2\pi\bar\tau)\end{aligned} \quad (11)$$

and equating these derivatives at the terminal points to the known BC (10) yields a system of linear algebraic equations to solve for coefficients a_i^η and b_i^η (for each chosen value of a single varied parameter τ_f) [11]. Applying Eqs. (7) and (8) yields the time history of a single control $\dot\psi(h)$.

When solving this problem numerically over a fixed set of N points ($j = 1,\ldots,N$), spaced evenly along the virtual arc $[0;\tau_f]$ with the interval

$$\Delta\tau = \tau_f(N-1)^{-1}, \quad (12)$$

all states and control are found at these nodes, so that the TPBVP is reduced to

$$\min_{\tau_f} J \text{ subject to } \Delta \le \varepsilon\,, \quad (13)$$

where the performance index is

$$J = \left(\sum_{j=1}^{N-1}\Delta t_j - \Delta t_{turn}\right)^2 \quad (14)$$

(time intervals Δt_j corresponding to $\Delta\tau$ are found via mapping between two domains [11]), the penalty function

$$\Delta = \max_j\left(0;\left|\dot\psi_j\right| - \dot\psi_{\max}\right)^2, \quad (15)$$

and ε is some tolerance.

ADS Prototype and Real Drops Results

This section presents the latest results of employing the aforementioned IDVD-based algorithm on a miniature 1.95kg ADS Snowflake (Fig.4), featuring 0.85m^2 (1.37m span) two-skin canopy and capable of carrying 1.8-2.2kg payload [13].

Deployed from an aircraft or helicopter (so far from as high as 3km) this ADS exhibits the following performance: the descend rate of ~3.6m/s, forward speed of ~7.2m/s (glide ratio of 2:1), and minimum turning radius of about 15m.

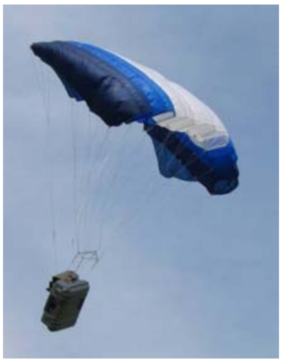

Fig. 4. Snowflake ADS.

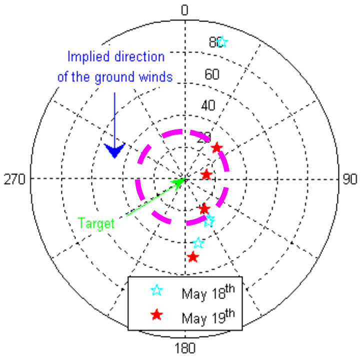

Fig. 5. Overall performance of Snowflake ADS with optimal terminal guidance.

To date over 40 drops of the developed ADS were successfully fulfilled and Fig.5 presents the overall results of the most recent drops that occur in May of 2009. On May 18th the drops were performed right after thunderstorm with a quite turbulent atmosphere and on May 19th the weather was more cooperative. The CEP for all eight drops is 30.6m while for four drops performed on May 19th is slightly less (26m), with the closest ADS landing within 14.4m from the target.

For more detailed analysis bird-eye-view trajectories for the best and the worst drops (from the standpoint of touchdown accuracy) are presented in Fig. 6 and 7, respectively. The first observation is that compared to trajectories presented in Figs.1 and 2 all Snowflake ADS trajectories look alike. The reason for that is that the system establishes and accurately tracks the inertial trajectory developed based on the target location and prevailing ground winds [11].

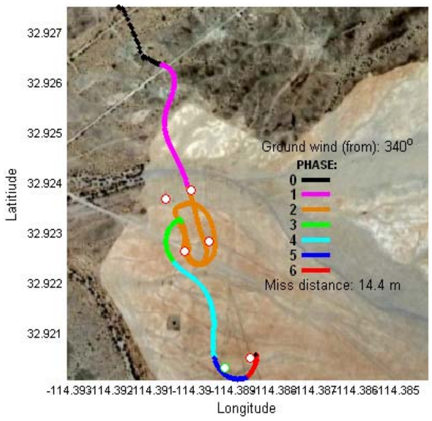

Fig. 6. The most successful drop (14.4m accuracy).

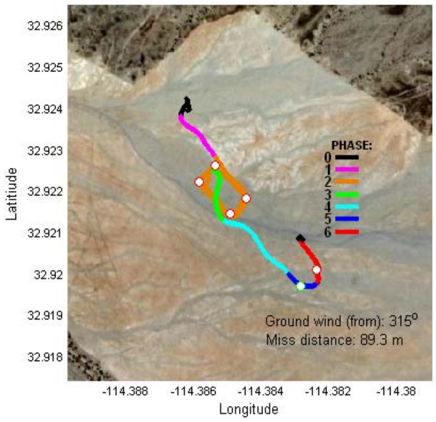

Fig. 7. The least successful drop (89m accuracy).

Upon exiting a deployment platform (phase 0) the Snowflake ADS steers towards a rectangular loiter area (phase 1), loiters to manage energy and estimate winds (phase 2), exits loiter area (phase 3) to enter a downwind leg as a manned aircraft would do (phase 4), and eventually performs a final turn / final approach (phase 5 and 6, respectively).

During the last portion of the descent the optimal guidance algorithm described in the previous section continuously re-compute the reference inertial trajectory. To be more specific, problem (13) is recast as:

$$\min_{\tau_f}\left(w_1 J + w_2 \Delta\right) \qquad (16)$$

(where w_1 and w_2 are some weighting coefficients) with no constraints, and solved numerically using the straight forward golden section search and inverse parabolic interpolation algorithm (based on the MATLAB's *fminbnd* function). A 16bit 80MHz processor, the core of Snowflake ADS' GNC unit, allows computation of an about 20-second turn maneuver with N=20 in only 10 iterations, which takes as little as 0.07s all together. With the control update rate of 0.25s that means that the trajectory can be updated as often as every control cycle! In practice however we allow tracking errors to build up, and perform re-computation of the reference trajectory every 10 control cycles.

The question is what made two drops (presented in Figs. 6 and 7) to be so different regardless the fact that there were no changes in GNC algorithm. The answer is – winds. To this end, Figs. 8 and 9 provide with the winds measured within a few minutes before each drop (shown in Figs. 6 and 7, respectively), and projected into the direction of the intended landing. As seen from Fig.8, for the most accurate drop the last wind estimate W, provided by Snowflake ADS' GNC at 68.3m altitude above ground (AGL) was about 3.6m/s while the actual downrange component of the wind gradually decreased from about 5.8m/s down to 2.7m/s at the ground. On the average the winds were about 0.8m/s stronger than it was estimated, but the capability to re-optimize the maneuver while descending allowed compensation for this change, so that there is almost 0m downrange error. Taking into account 22.5s that took ADS to land, without re-optimization the ADS would land about 18m short of the target.)

For the least accurate drop (Figs. 7 and 9) the last wind estimate W of 2m/s came at 51m AGL but the actual winds were more than -3m/s (tailwind) all the way down. These ~5.4m/s unaccounted winds (for a 20.5s descent) resulted in 89.3m overshoot (again, without re-optimization we would have 111m overshoot).

VI. Band-Limited Guidance

For the sake of comparison let us also consider a somewhat similar algorithm for the terminal guidance, BLG, developed by Draper Laboratory [7]. As opposed to parameterizing the coordinates as in Eq.(9), the BLG is reduced to parameterizing a control, heading rate, as

$$\psi'(h)=\sum_{k=0}^{M}\psi'_k\frac{\sin(\xi_k)}{\xi_k}, \qquad (17)$$

where

$$\xi_k=\frac{\pi(h-k\Delta h)}{\Delta h} \qquad (18)$$

(to obtain the heading rate command $\dot{\psi}_c$ Eq.(17) needs to be multiplied by V_v^*). Specifically, for an Airborne Systems' Megafly ADS (with $V_v^*=5m/s$) a final turn and approach maneuver begins at exactly 800m AGL, followed by the application of full brakes (flare) at 60m AGL, so Δh=200m and M=4 were chosen.

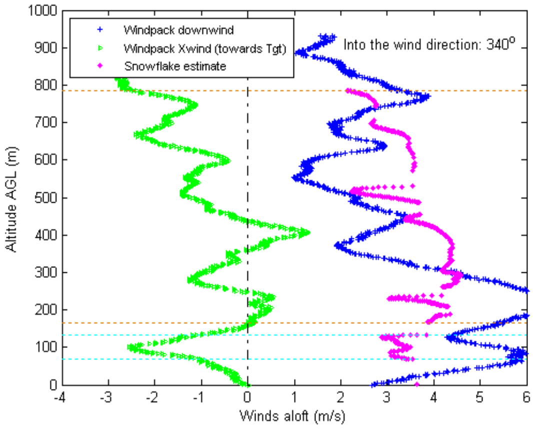

Fig. 8. Wind data for the drop depicted in Fig.6.

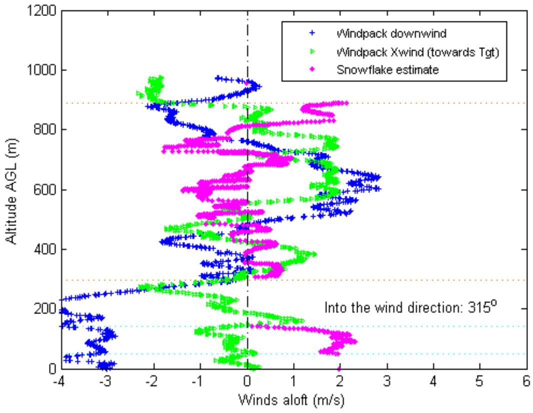

Fig. 9. Wind data for the drop depicted in Fig.7.

Five varied parameters ψ'_k, which represent turning rates at consecutive multiples of Δh, are defined by solving the following unconstrained optimal terminal control problem:

$$\min_{\psi'_k,k=0,\ldots,4} J, \qquad (19)$$

where J is comprised of a weighted sum of a squared miss distance and squared heading error:

$$J=w_1((x_f-x_T)^2+(y_f-y_T)^2)+w_2\sin^2\tfrac{1}{2}(\psi_f-\psi^{des}) \qquad (20)$$

(ψ^{des} is the desired final heading). Of course all the states are to be computed via integration of Eqs.(6). As stated in [7], integration of the kinematic equations is done in fixed point arithmetic for efficiency. Using the simplex search Nelder-Mead algorithm (MATLAB's *fminsearch* function), for the optimization allows re-solving Megafly final approach guidance at 1Hz rate (using each previous solution as the new initial guess). With the imposed 100 limit on the number of iterations per 1Hz cycle this means that convergence to an optimal solution is

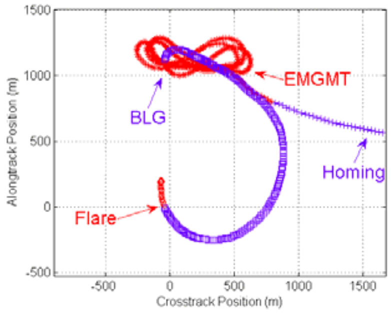

Fig. 10. Example of a flight path of Megafly with Draper's GNC [7].

obtained only gradually. An example of the last portion of a guided descent is presented in Fig.10.

VII. CONCLUSIONS

Let us now compare the two different direct method approaches. Table 1 presents the results of a side-by-side comparison.

TABLE I
COMPARISON OF IDVD AND BLG

Characteristic	IDVD	BLG
Computing the coordinates	parameterized analytical function	via integrating equations of motion
Computing the heading rate	via inverse dynamics	parameterized analytical function
Satisfying TC on coordinates	automatically	via optimization (Eq.(20))
Satisfying TC on heading	automatically	via optimization (Eq.(20))
Satisfying TC on heading rate	automatically	not capable
Satisfying IC on heading rate	automatically	not capable
Satisfying the control bandwidth constraint	via optimization (Eq.(15))	automatically
Number of varied parameters	1	5
Minimization engine	fminbnd	fminsearch
Convergence	guaranteed	not guaranteed
Number of iterations to converge	10	>100*
Onboard CPU	16bit 80MHz	unknown
CPU time to converge	0.07s	1s*

* - means if converges to the solution.

As seen, the BLG method can exactly limit the control bandwidth, while IDVD can only do it indirectly by penalizing $\dot{\psi}$. On the other hand, IDVD allows to automatically satisfy all BC and guarantees fast convergence (apparently running on a much less powerful computer compared to that of Megafly's GNC unit). Moreover, IDVD approach has even more potential by adjusting the TC according to the current accuracy of reference trajectory tracking as outlined in [14]. Both approaches allow and will definitely benefit from uplinking the ground winds so that they can be accordingly accommodated in (6) (see [14]).

VIII. CONCLUSIONS

This paper reviewed different concepts used to guide autonomous ADS towards a predetermined target, and concentrated on the two most prominent ones (IDVD and BLG) using inertial terminal-guidance trajectory optimized in real time. Both algorithms utilize the major property of direct methods, reducing the optimization space to a certain class of predetermined solutions, but pursue different computational paradigms. Based on the analyses presented in this paper it seems that if limiting control bandwidth is a priority then using BLG may be justified. Otherwise, the IDVD-based guidance is more robust and guarantees a better accuracy than that of BLG.

ACKNOWLEDGMENT

The authors would like to thank Mr. Bourakov, CDR Hewgley, and the personal of the U.S. Army Yuma Proving Ground for helping with the experiments. Also, they would like to thank the U.S. SOCOM for supporting the development and testing of Snowflake ADS financially.

REFERENCES

[1] Benney, R., Henry, M., Lafond, K., Meloni, A., and Patel, S., "DOD New JPADS Programs & NATO Activities," *Proceedings of the 20th AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar (ADSTC&S)*, WA, May 4-7, 2009.

[2] *Precision Airdrop Technology Conference and Demonstration 2007*, Final Report, U.S. Army RDECOM, Natick, MA, February 2008.

[3] Benney, R., Meloni, A, Cronk, A, and Tiaden, R., "Precision Airdrop Technology Conference and Demonstration 2007," *Proceedings of the 20th ADSTC&S,* Seattle, WA, May 4-7, 2009.

[4] de Lassat de Pressigny, Y., Benney, R., Henry, M., Bechet, R., and Wintgens, J.H., "PACD2008: Operational Requirements Fulfilled," *ibid.*

[5] Kaminer, I., and Yakimenko, O., "Development of Control Algorithm for the Autonomous Gliding Delivery System", *Proceedings of the 17th ADSTC&S,* Monterey, CA, May 19-22, 2003.

[6] Hattis, P., "Autonomous Large Parafoil Guidance, Navigation, and Control System Design Status," *Proceedings of the 19th ADSTC&S*, Williamsburg, VA, May 21-24, 2007.

[7] Carter, D., Singh, L., Wholey, L., Rasmussen, S., Barrows, T., George, S., McConley, M., Gibson, C., Tavan, S., and Bagdonovich, B., "Band-Limited Guidance and Control of Large Parafoils," *Proceedings of the 20th ADSTC&S*, WA, May 4-7, 2009.

[8] Taranenko, V.T., *Experience of Employing Ritz's, Poincaré's, and Lyapunov's Methods for Solving Flight Dynamics Problems*, Air Force Engineering Academy, Moscow, 1968 (in Russian).

[9] Yakimenko, O., "Direct Method for Rapid Prototyping of Near-Optimal Aircraft Trajectories," *AIAA Journal of Guidance, Control, and Dynamics,* vol.23, no.5, 2000, pp.865-875.

[10] Kaminer, I., Yakimenko, O., and Pascoal, A., "Coordinated Payload Delivery using High-Glide Ratio Parafoil Systems," *Proceedings of the 18th ADSTC&S*, Munich, Germany, May 23-26, 2005.

[11] Slegers, N., and Yakimenko, O., "Optimal Control for Terminal Guidance of Autonomous Parafoils," *Proceedings of the 20th ADSTC&S*, Seattle, WA, May 4-7, 2009.

[12] Taranenko, V.T., and Momdzhi, V.G., *Direct Method of Calculus of Variations in Boundary Problems of Flight Dynamics*, Maschinostroenie, Moscow, 1986 (in Russian).

[13] Yakimenko, O., Slegers, N., and Tiaden, R., "Development and Testing of the Miniature Aerial Delivery System Snowflake," *Proceedings of the 20th ADSTC&S*, Seattle, WA, May 4-7, 2009.

[14] Bourakov, E.A., Yakimenko, O.A., and Slegers, N.J., "Exploiting a GSM Network for Precise Payload Delivery," *ibid.*