

Design of a Miniature Aircraft Deployment System

Leah Crumbaker, Travis Schafhausen, Jason Farmer, James Gordon, Matthew Lenda,
Jeffrey Mullen, Jack Tatum, and Kristina Wang*

University of Colorado, Boulder, CO, 80309. Team Category

This paper describes a Miniature Aircraft Deployment System (MADS) that is integrated into an existing unmanned aircraft system (UAS) in order to carry and deploy four smaller, autonomous aircraft. The smaller, deployable sub-vehicles attach to the fuselage of the primary vehicle, beneath the wings, in a flight-ready configuration (no folding or containment) via a bracket-and-beam system. The user controls deployment via IEEE 802.15.4 modules that form a peer-to-peer network between a ground station, the primary vehicle, and the sub-vehicles. This deployment system serves as a proof-of-concept that the in-flight deployment of other air vehicles, and the overall dynamics associated with such an action is predictable, reliable and can be managed by the primary aircraft control system. The design of MADS optimizes the placement of the sub-vehicles, relative to the primary vehicle. Placement was selected to accommodate the dynamics of the sub-vehicles as they deploy from the primary vehicle, as well as minimize the impact of the entire deployment system on the controllability and handling of the primary vehicle. The behavior of the system and deployed sub-vehicles was evaluated using computational fluid dynamic methods to determine the aerodynamic and stability characteristics of both the primary vehicle and sub-vehicles, as well as all possible configurations of the integrated system. This analysis allowed for the prediction of the system both pre-deployment and post-deployment. The deployment mechanism was designed to restrict the motion of the sub-vehicle during the deployment process to ensure a reliable and safe deployment behavior.

Nomenclature

<i>CFD</i>	Computational Fluid Dynamics
<i>CG</i>	Center of Gravity
<i>DS</i>	Deployment System
<i>LBM</i>	Lattice Boltzmann Method
<i>MADS</i>	Miniature Aircraft Deployment System
<i>NS</i>	Navier-Stokes
<i>PV</i>	Primary Vehicle
<i>RECUV</i>	Research and Engineering Center for Unmanned Vehicles
<i>SV</i>	Sub-vehicle
<i>UAS</i>	Unmanned Aircraft Systems

I. Introduction

THE value of MADS is quickly becoming apparent as a developmental platform for future UAS designed for data collection. The addition of smaller, task specific, deployable aircraft to pre-existing systems is now being considered to further extend their capabilities. Deployable Sub-vehicles (SVs) are most applicable in data-collection situations that are dangerous or difficult to access for human observation, especially if SVs are designed as disposable commodities. Severe weather and wildfire observation and tracking are specific examples of such applications. Observers would be able to send an unmanned aircraft towards the event of interest which would then deploy multiple SVs to collect data. Multiple SVs allow for faster or larger search area coverage and characterization, or for several different instruments to be utilized at once for data collection. The information would then be relayed back through the primary vehicle (PV) to observers on

*Undergraduate Students, Aerospace Engineering. AIAA Student Members.

the ground. MADS was developed to meet the customer’s desire for a deployment system that was capable of carrying and deploying four SVs in flight and could be integrated into an existing RC aircraft (SIG Rascal 110). Deployment of the SVs must be on-demand via a command from the ground station. Also, the designed deployment system must not deteriorate the functionality and handling of the original PV beyond controllability. The analysis of the deployment system can be broken down into three stages: pre-deployment subsystem design, deployment electronics and software, and post-deployment system analysis. As with any design, the analysis of each section is an iterative process and may affect the other design choices throughout the project.

First, the pre-deployment subsystem design describes the physical configuration of the four SVs, their placement, configuration and attachment to the PV, as well as the deployment mechanism itself. Deployment consists of the electronics and software required to execute the deployment of the SVs. Finally, the post-deployment system analysis addresses the release of the SV and the aerodynamics and controllability of the fully assembled system (Figure 1).



Figure 1. MADS in flight with all four SVs attached.

II. Pre-deployment Subsystem Designs

Pre-deployment subsystem designs focus on the SV configuration and attachment method. This specifically addresses the configuration of the SVs themselves, their locations with respect to the PV, the reinforcement needed in the PV to carry the additional loads imparted by the deployment system, as well as the design of the deployment mechanism itself.

A. Sub-vehicle Configuration

The SV configuration consists of the vehicles externally mounted to the PV in a non-compacted, un-contained state. This means that the vehicles are mounted in a flight-ready state and are fully exposed to the freestream air-flow. Mounting the SVs in this configuration removes the complexities and possible failures associated with folding designs or the opening of containers or bays, while allowing the focus to remain on the deployment system. The SuperFly¹ (Figure 2) fits this configuration requirement, as well as the dimensional limitations of the system.

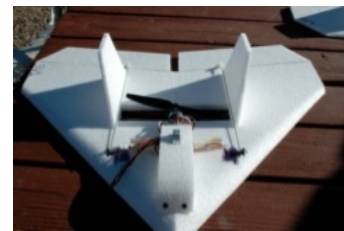


Figure 2. The chosen SV for MADS, the SuperFly.

B. Sub-vehicle Attachment Configuration

Computational Fluid Dynamics (CFD) models were used to determine the controllability of the PV with the SVs attached. This analysis was done with the worst-case loading scenario of two SVs on one side of the PV, and none on the other. In this scenario, the required control surface deflection to maintain trim was calculated, along with the remaining control surface deflection margins for maneuvering (Table 1).

Table 1. Control margins for the PV with the SVs in a pseudo-stacked attachment configuration.

Control surface	Margin
Aileron	10%
Rudder	28%
Elevator	83%

A pseudo-stacked configuration (Figure 3) with a cantilever beam was selected with the use of the results from CFD models. The chosen pseudo-stacked configuration allows adjustable placement to avoid a PV strike, as well as minimizes yawing moments while flying in asymmetric configurations (because of the proximity of the SV to the PV body). In order to secure the SVs to the PV in this configuration, a cantilevered beam attachment was chosen. This attachment method allows for the SVs to be secured to the

PV without requiring modifications to the wing or wing strut and also allows the deployment system to be removable for transportation purposes.

This SV configuration placement and attachment design has several inherent limitations. The lower SVs must be deployed first in order to avoid collisions when deploying the upper SVs in the pseudo-stacked configuration. However, given the aerodynamic considerations, space restrictions, and allowable mounting locations, the chosen design meets all necessary requirements.

C. Sub-vehicle Placement and Attachment Reinforcement

The mounting locations were selected to utilize the strongest locations on the PV fuselage as well as avoid a PV strike based on the numerical-integration drop model (discussed under Section IV: System Analysis of Post-Deployment section).

To determine the stresses in the PV structure, a Finite Element Analysis (FEA) model was created in ANSYS. This allowed for determination of the primary areas needing reinforcement, as well as the size and dimensions of the reinforcements required. The model was evaluated for five possible loading cases: 1. Four SVs - two on either side; 2. Three SVs - two on one side and one on the opposite; 3. Two SVs - one on either side connected to the upper, aft beams; 4. Two SVs both on one side; 5. One SV - attached to an upper, aft beam. Results of the modeling are summarized in Table 2. It should be noted that these stresses are the result of a loading case that was designed to determine the safety factor that could be achieved above a worst-case loading scenario of 50N of lift, 10N of drag, and a -3Nm moment imparted to the system by each SV. These results show that the empirical yield stresses² of the system structure are not reached until a factor of 1.5 times the worst case loading scenario is experienced by the system. The yield stress of the aluminum, balsa wood, and plywood were all determined empirically.

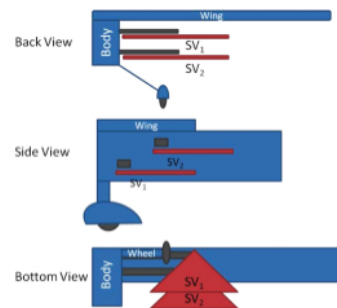


Figure 3. The chosen configuration.

Table 2. Maximum stress values for different structural materials.

Material	Empirical Yield	Case 1	Case 2	Case 3	Case 4	Case 5
Max Plywood Stress (MPa)	45	45	45	45	43	41
Max Balsa Stress (MPa)	10.8	3.3	3.3	2.8	3.3	2.8
Max Beam Stress (MPa)	280	279	279	279	278	278

The boundary conditions for the model assumed that the structure was fixed at the points where the wing attaches to the fuselage body. The worst case loading (with a safety factor) discussed earlier was applied at the deployment mechanisms located at the tip of each beam. Figure 4 shows the locations of the maximum stress incurred in the plywood and beam respectively. The plywood maximum stress is in the small gap between the wing attachment and the aluminum reinforcement. To reduce the stress on the plywood, a thin aluminum plate was added to the exterior of the PV wall in order to increase the stiffness.

The FEA model is not an entirely true representation of the physical system and has several limitations. The fixed boundary conditions do not allow for the part to move in space. This condition is not an exact approximation because the forces on the vehicle induce a small displacement in the structure. These displacements result in a decrease in the predicted stresses experienced by the PV. Second, the model assumes a perfect and rigid connection between the beams and the wall. This generates large moments at the connection point. However, in the physical model, these connection points are not truly rigid, and the small movements allowed in the physical structure would result in smaller moments and stresses in the PV wall. Lastly, the model does not contain the full fuselage structure, but rather a small section. The stresses are distributed over a larger area and more cross-members in the actual system.

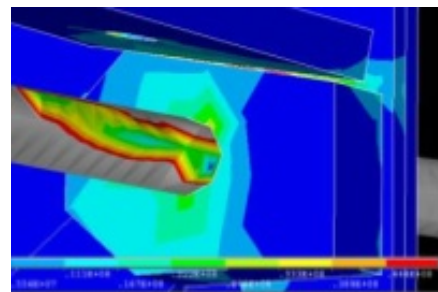


Figure 4. FEA model of the mounting point on the PV fuselage.

The mounting reinforcements were designed to meet two requirements. The first was to reduce the load carried by the wall of the PV. The second was to hold the beams during flight while still being easily

removable for transportation. The mounting reinforcements are half-inch thick aluminum blocks with a hole for the beam to slide through. The beams are held in place by two set screws. To reduce the load carried by each wall, the beams are connected across the structure with a threaded bracket in the center which that allows the beam to act as one solid piece. This distributes the load across both walls of the structure. These designs (Figure 5) were developed to minimize the stress in the wall and minimize the size (mass) of the brackets.

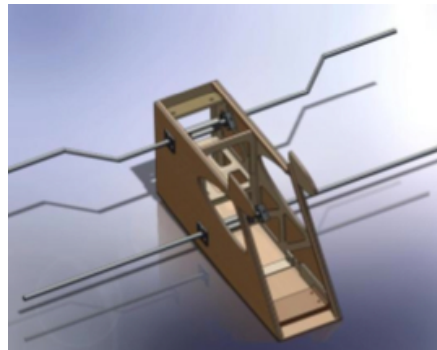
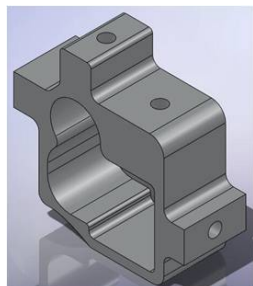


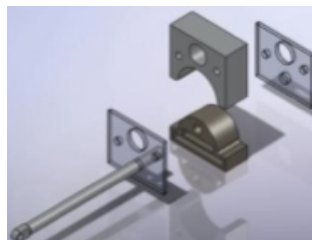
Figure 5. Deployment system beams and supports attached to the PV fuselage.

D. Deployment Mechanism

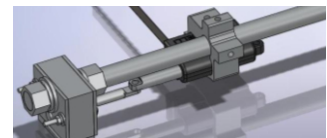
A pin-sleeve deployment mechanism was created (Figure 6) to hold the SVs to the mounting beams. The deployment mechanism was designed to restrict the motion of the SV during the deployment process to ensure reliable and safe deployment behavior. In order to integrate this with the rest of the system, the deployment mechanism design was restricted by mass, movement of the CG, interfacing requirements with the actuator, and the removability of the design. All of these restrictions limit the size and material choice.



(a) Linear actuator bracket.



(b) Exploded view of the deployment mechanism.



(c) Deployment mechanism design assembled.

Figure 6. Deployment mechanism design.

The cap SV restraint is easy to manufacture, light, and made of aluminum. The semicircle shape allows the SV to pitch freely during post-deployment, but restricts movement and rotation in all other axes. The channels of material that have been removed from the sides were designed to save weight to meet the design requirement of a total mass of 9g. The two thin aluminum sleeves bolt to either side of the cap SV restraint and corresponding beam mount. This takes torques in the roll and yaw directions off of the pin during flight and prevents jamming during deployment. These are also made of aluminum. The tight fit between the two sleeves also minimizes jamming during deployment by restricting the motion of the SV. The rod mounted restraint matches the profile of the lower restraint to within 1/1000th of an inch to ensure a secure fit during flight. This bracket is easily secured onto the end of the rod and is easily manufactured using a CNC mill.

An additional bracket was created in order to secure the linear actuator that pulls the pin during flight (Figure 6). The contour-fitting bracket fits tightly around the linear actuator and. Two set screws are used in order to further secure its position from sliding or spinning along the beam. Like the rest of the deployment mechanism, this bracket is also made of aluminum and machined with the same tolerances to ensure a precise fit.

III. Deployment Electronics and Software

The deployment phase analysis encompasses the software and electronics needed for deployment. Deployment electronics consist primarily of the autopilot on board the SV, the deployment PIC on-board the PV, and the deployment mechanism boards to control the linear actuators. Deployment software was designed to make the deployment mechanism and sequence functional from the ground station. The deployment process is defined as a three step procedure:

1. Send the command from the ground station to the deployment PIC to deploy a specified SV.
2. Monitor potentiometer data until deployment is confirmed on board the deployment PIC.
3. Upon completed pin retraction, send command to specified SV to activate predetermined thrust and elevon control sequence from the deployment PIC.

This divides the deployment software up between three components: the SV, PV, and the Ground Station.

A. Deployment Electronics

The deployment electronics consist primarily of the deployment mechanism control board, the deployment PIC on the PV, and manual override switches. A linear actuator, as opposed to a rotary servo, was selected as the mechanical element that pulls the pin attaching the SV to the deployment system due to the included potentiometer which is used to verify that the pin had been pulled. In order to use the linear actuators in flight from the ground station, custom electronic boards were needed.

The deployment mechanism required a custom PCB to control each of the actuators independently through the deployment PIC. The deployment PIC is the controller on-board the PV because of the built-in transmitter and communication link with the ground station. This deployment controller allows the ground station to activate the deployment sequence on the PV, making it one command for deployment. Once the pins have retracted, the deployment PIC also uses the pre-existing XBee radio to transmit the potentiometer data from the linear actuator back down to the ground station.

The linear actuator operates at $\pm 5V$ and is controlled by the deployment PIC from the ground during flight. Unlike servos, these linear actuators require reversal of current flow (not pulse width modulated signals) in the circuit to extend and retract the arm. Polarity reversal is achieved through the use of four MOSFETs. A fifth MOSFET is used to control the flow of current. All five MOSFETs are needed to allow complete control using two input lines. The first input line controls the direction of the current. The second line controls whether or not the current is flowing. Back-EMF (electromotive force) protection is also required to prevent unwanted voltage differentials if they are present.

With the deployment PIC and the deployment mechanism control boards, deployment is capable through commands from the ground station, however, a redundant system was created as well. Manual override switches were created and mounted onto the fuselage of the PV for every mounting location. With these switches, the SVs can be mounted onto the PV or removed without the use of the ground station.

B. Deployment Software

The deployment software is split up between the PV, SV, and the ground station. The system utilizes both wireless and wired communication between the three components as outlined by Figure 7.

1. CUPIC Software for the Primary Vehicle

The wireless interface of the PIC with the ground station indicates that there must be a wireless transmission protocol defined for the deployment mechanism electronics package. The wireless radio transmission setup of the Zigbee/XBee system already resident on the CUPIC³ gives the deployment system sufficient remote communication abilities.

The most important component of the CUPIC software is the wireless transmission of data between the deployment PIC on the PV and each of the SVs. There are two types of data transferred: data packets from each of the aircraft (PV included) and commands from the ground station. The potentiometer data from the deployment PIC on the PV and the GPS data from both the PV and the SVs are monitored in real-time by the Graphical User Interface (GUI) on the ground station. This same GUI gives the user the ability to send commands to each of the vehicles.

When the PV CUPIC is commanded to deploy a vehicle from the ground station, it immediately begins to retract the corresponding actuator. When the potentiometer on the actuator reaches the point where SV will fall away (calibrated during pre-flight), the PV CUPIC sends a command to the SV CUPIC to initiate its deployment routine. The actuator continues to retract to a final position completely clear from the housing of the mechanism. This point is also calibrated during preflight.

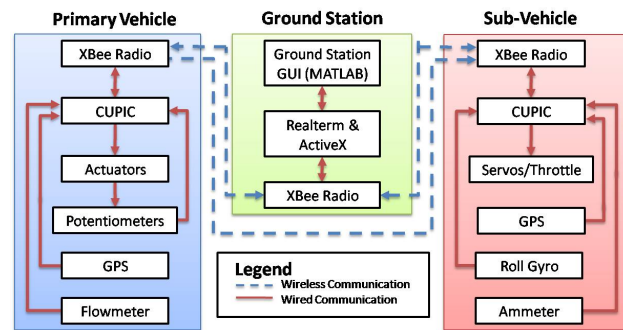


Figure 7. Communication lines between subsystems.

2. CUPIC Software for the Sub-vehicle

The autopilot control system functionality and deployment sequence are the focus of the CUPIC software for the SV. Upon receiving the deployment command, the SV performs a previously defined elevon and throttle control sequence to ensure that the SV does not collide with the PV during deployment.

The CUPIC autopilot controller uses roll to control heading rate and throttle to control altitude. The roll controller was developed based off of natural aircraft dynamics and kinematics. The rate gyro on-board the CUPIC is integrated and mixed with the GPS estimated roll angle through a high-pass filter and low-pass filter respectively to estimate the roll angle. The desired roll angle is subtracted to find the error which is mixed with the heading feedback and the roll-rate feedback to find a servo command for the elevons. The altitude is controlled by proportionally relating altitude error to throttle command while damping the vertical velocity by pitching the elevons. When the autopilots landing mode is activated, the throttle is disengaged and the vertical velocity damping is increased.

The CUPIC software (written in C) works through an interrupt based system typical to most microcontrollers. All critical aircraft routines are timed using the high and low priority interrupt system. The low priority interrupt system runs at 100Hz, while the high priority interrupt system runs asynchronously. In the main routine, a while loop runs continuously to parse commands that the CUPIC has received as well as to send data packets back to the ground station. Each time through the main while loop, it sends a data packet to the ground station. This data packet includes such parameters as node number, the longitude and latitude, the pressure altitude, roll angle, and heading. Lastly, each CUPIC has a unique node number hardcoded into the microcontroller before flight. This is how each CUPIC is identified on the ground station.

3. Ground Station Software

In order to make the ground station operable for MADS, it was designed to deploy the commanded sub-vehicle as well as receive, display, and interpret potentiometer data.

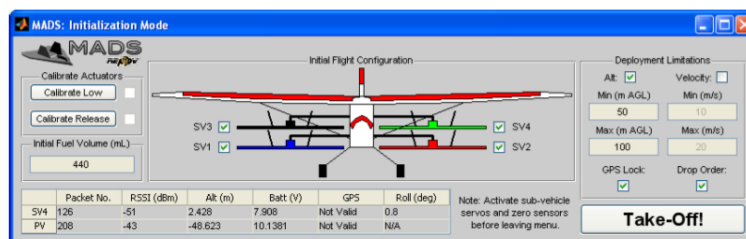


Figure 8. First screen available to the user in the ground station GUI.

The GUI interfaces with the XBee radio using Realterm and ActiveX through the use of a buffer. The GUI has several inline functions used to process the incoming packets and to send commands. Each of these functions is attached to a button in the GUI that writes the command packet to the Realterm buffer. When the command is called, it acts similarly to an interrupt. The command packet is interpreted and sent to the Realterm buffer immediately.

The GUI is separated into two major screens. The first screen (Figure 8) allows the user to calibrate the linear actuators, select deployment limitations (e.g. altitude and velocity of the system), mark what SVs are attached to the PV, and inspect the initial readings of each vehicle (e.g. battery voltage, GPS data, altitude). After the system is airborne, the next screen (Figure 9) becomes available with sensor and packet

readings for multiple vehicles, a plot for actuator positions, and a map of where each vehicle is in the air. This is also the screen that allows the user to send the release command to the PV by selecting a specific SV and clicking deploy.



Figure 9. Second screen available to the user in the ground station GUI.

IV. System Analysis of Post-Deployment

The system analysis of the post-deployment phase considers what happens to the system when the SVs are released. A numerical drop model was created to determine how the SVs would drop away from the PV taking into account several parameters: initial placement of the SV with respect to the PV, mounting angle of attack of the SV, and the speed of the system at the time of deployment. This model focused on verifying numerically that the placement and mounting orientation of the SV would be suitable to avoid collision with the PV upon release.

A. Drop Model

By numerically integrating the sum of the forces and moments acting on the sub-vehicle, the path of the SV during deployment can be modeled. In these equations, L is the lifting force, D is the drag force, W is the weight, T is the thrust, V is the freestream velocity, α is the angle of attack, and γ is the flight path angle. The sum of the moments is summed in the $-y$ direction, equal to the cross product of the x and z unit vectors. The following equations of motion⁴ depict the movement of the SV:

$$F_x = ma_x = -D\cos\gamma - L\sin\gamma + T\cos\theta \quad (1)$$

$$F_z = ma_z = -D\sin\gamma + L\cos\gamma + T\sin\theta - W \quad (2)$$

$$M = I_{yy}\dot{q} = M(\alpha, q, \delta) \quad (3)$$

The mass of the vehicle is given by m and the mass moment of inertia is given by I_{yy} . The pitch angle, θ , is the sum of the flight path angle and the angle of attack. The acceleration terms a_x , a_z , and \dot{q} can be calculated by dividing by the appropriate mass term.

To calculate the drag, lift, and moment forces,⁴ the relations shown in Equation 4, Equation 5 and Equation 6, are used, where C_D is the drag coefficient, C_L is the lift coefficient, C_M is the moment coefficient, \bar{q} is the dynamic pressure, S is the planform area, and c is the chord length.

$$D = (C_D(\alpha) + \Delta C_D(\delta))\bar{q}S \quad (4)$$

$$L = (C_L(\alpha) + \Delta C_L(\delta))\bar{q}S \quad (5)$$

$$M = (C_M(\alpha) + C_{M_q}q\frac{c}{2V} + \Delta C_D(\delta))\bar{q}S \quad (6)$$

These equations were programmed in an iterative loop using an explicit Euler method. Each time through the loop, the dynamic pressure, flight angles, and coefficients are updated to match the current flight condition. The lift, drag, and moment coefficients are a function of the angle of attack of the sub-vehicle computed from CFD results. Given a time step, Δt , the numerical integration can be calculated as

follows (i is the index number, V_x and V_z are the velocities in the x and z directions respectively, q is the pitch rate, and x, z, and θ are the positions and pitch angle):⁴

$$V_{x_{i+1}} = V_{x_i} + a_{x_i} \Delta t \quad (7)$$

$$V_{z_{i+1}} = V_{z_i} + a_{z_i} \Delta t \quad (8)$$

$$q_{i+1} = q_i + \dot{q}_i \Delta t \quad (9)$$

$$x_{i+1} = x_i + V_{x_i} \Delta t \quad (10)$$

$$z_{i+1} = z_i + V_{z_i} \Delta t \quad (11)$$

$$\theta_{i+1} = \theta_i + q_i \Delta t \quad (12)$$

Lastly, due to the design of the deployment mechanism, the motion of the SV during deployment was constrained to the z-direction. Using line-intersection method, collisions of the SV and PV could be detected. If the angle of attack exceeds the positive or negative stall angle of attack, the simulation is stopped. Figure 10 shows the path that the SV takes with respect to the PV given an initial angle of attack such that $L = W$, $V = 15\text{m/s}$, with no elevon deflection or thrust. The deployment process for varying initial angles of attack and elevon deflections can be seen using this model, allowing the team to conclude that the SV deploys without colliding with the PV.

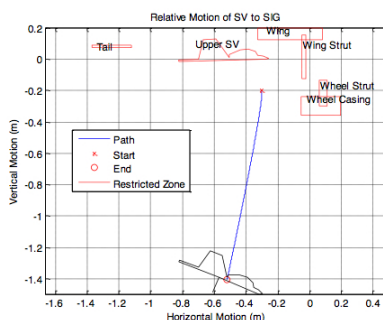


Figure 10. Example of a scenario in the numerical drop model.

B. System Stability

Conventional stability analysis tools utilize vortex-lattice methods. However, due to the complicated geometry, vortex lattice methods fail to provide any meaningful results. Therefore, a new stability-analysis approach was created to analyze the stability of MADS. CFD has been used as an analytical tool in different areas of research and applications in aerospace; the most popular research has been solving discretized Navier-Stokes (NS) equations which define single-phase fluid flow. The focus of CFD has recently turned to discretizing and simulating the Boltzmann equation (this is known as the Lattice Boltzmann Method, or LBM). The main differences between NS-CFD and LBM are that Navier-Stokes equations are simply Newton's second law applied to a fluid. The LBM is derived from the Boltzmann equation, which is a model for the statistical distribution of particles and their collisions. The resulting behavior results in Newton's second law being obeyed.

Due to the nature of LBM, it is well suited for complex geometries and boundary-layer interactions. In a flow problem when there are moving geometries (i.e. forced oscillations), there are two considerations. First, the accuracy of the results greatly depends on the ability to simulate the viscous nature of the boundary layer with high fidelity. LBM has an advantage over NS-CFD because it can resolve fine boundary layers with larger mesh sizes. Second, the solutions from moving geometries (moving pieces of the structure) will be transient solutions and the inherent numerical stability of the LBM proves beneficial in order to reduce the amount of iterations needed to ensure convergence.

Using LBM is an excellent starting point to perform forced oscillation simulations in order to estimate stability derivatives for a vehicle. To achieve this, PowerFLOW, by EXA Corp, an LBM based commercial computational dynamics solver, was used. It is designed specifically for the automobile industry for rapid aerodynamic design of vehicles with complicated geometry. PowerFLOW utilizes the immersed boundary methods and perform all the meshing behind the scene. This results in extremely quick problem set-up

in comparison to traditional NS-CFD methods. Using PowerFLOW, multiple static and forced-oscillation simulations were set-up in order to characterize the aerodynamics and stability of MADS. For each simulation, the body-frame forces in the X, Y, and Z direction and torques in the three axes were calculated. From this data, a drag-polar for the entire system could be calculated (see Figure 11).

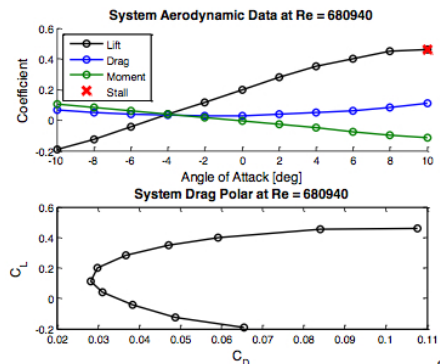


Figure 11. Drag polar for MADS.

With the flow conditions (velocity, density, pressure) and the aircraft parameter (span, wing area, mass, moments of inertia) known, all of the standard stability derivatives for a linear stability analysis were calculated. Due to time constraints, only one set of stability derivatives for a single velocity and configuration could be run. Using these stability derivatives, de-coupled, linearized state-space models of the aircraft for both the lateral and longitudinal dynamics were created with the full-system configuration (see Figure 12). The eigenvalues of these models result in the natural aircraft models (phugoid, short-period, dutch roll, roll subsidence, and spiral divergence).

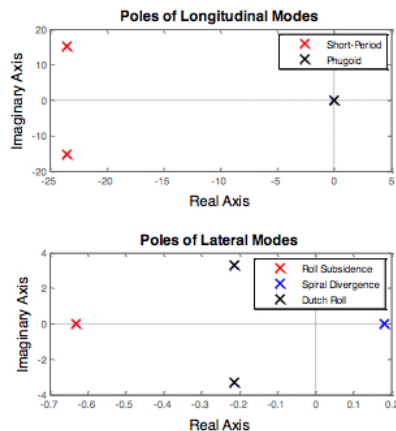


Figure 12. Longitudinal and lateral stability poles.

As seen in this plot, the system analysis of results in typical aircraft modes with the exception of the phugoid mode. This is slightly unstable, however, because these poles are extremely close to the origin, they are of no concern because the plane will never be piloted "stick-free"; a pilot will be able to control these instabilities.

V. System Development

Several months of continued development and testing of the system has been performed since the design presented in this paper was finalized at the end of the 2008 fall semester. In that time, many milestones have been achieved, and several design modifications have been implemented. The full system has been fabricated, assembled and successfully flight tested with all four SVs carried by the PV. The system has also successfully flown in all potential SV configurations, except for the worst case loading of two SVs on one side and none on the other, though plans are in place to test this configuration. Two successful in-flight deployments of non-functional SVs have been achieved as well, and continued testing should see the deployment of a functional SV under autopilot control, with the deployment routine implemented.

However, several failed deployments were experienced, which has led to a closer inspection of the linear actuators and the brackets that were designed to secure them to the beams. It was determined that pressure applied by the set screws that secured the actuators in place was causing a decrease in performance, and in some cases, internal jamming. The actuators have been disassembled and repaired, and a new mounting bracket has been designed and fabricated that will alleviate the need for set screws or large pressures to secure the actuators in place. These modifications have been tested, and the repaired actuators with the new mounting brackets successfully pull the manufacturer's specified maximum load.

Additional electronics have been implemented into the system as well. To verify the system stability models, the PV has been outfitted with a complete sensor suite that measures three-axis accelerations and angular rates of the system with a custom IMU, absolute pitch and roll angles via a horizon sensor, and vibrations at each deployment mechanism. These are recorded by two CU Aerospace DataLoggers, which are custom data logging PCBs designed and built at CU Boulder. An Eagle Tree Systems data logger is also used to measure all RC control inputs and control surface deflections, as well as engine RPM, and pitot-static pressure measurements. Finally, a flow meter has been installed in the fuel line to measure in-flight fuel consumption. The flow meter pulses are read by the PV PIC and transmitted to the ground in real time. The system has flown with this sensor suite and data analysis and comparison to the model is currently underway.

Finally, the software subsystem has achieved several milestones as well. The SVs have achieved successful flight under full autopilot control, with successful loiter circle tracking and navigation, altitude control, and unassisted landing. Additionally, the PV PIC can be implemented as a data forwarding station for the SVs such that the SVs transmit all data back to the ground station via the PV PIC. Lastly, the deployment routine for the SVs has been modified to include the roll and altitude damping functionality from the CUPIC autopilot in order to achieve a controlled deployment from the PV before switching into full autopilot control.



Figure 13. An SV deploying from the PV.

VI. Conclusion

MADS is a rapid-paced project with the main objective to design, build, and test a successful deployment system for a pre-existing RC aircraft with the ability to carry and deploy up to four sub-vehicles. Currently, the project is in system level flight tests, with two successful deployments documented. With the completion of the project in mid-April 2009, the team will be able to hand over a working deployment system that will serve as a test bed for cooperative control algorithms. Future versions of this project will lead to the use of disposable SVs with foldable wings and more advanced primary vehicles that can withstand adverse weather conditions, extending the current mission of this system.

Acknowledgments

The authors would like to thank everyone who has contributed to the MADS project over the course of the year; without any of them and their resources, this project would not have been possible. This includes, but is not limited to: The College of Engineering of the University of Colorado at Boulder, the Department of Aerospace Engineering Sciences, Dr. Eric Frew (customer), Dr. Kurt Maute (advisor), Dr. William Emery (advisor), Trudy Schwartz, Matt Rhode, Bill Pisano, Dr. Dale Lawrence, and Dr. Jean Koster. The authors would also like to thank the Arvada Associated Modelers for all of their help and guidance through choosing, building and flying our RC aircraft.

References

- ¹“Superfly RC,” <http://www.superflyrc.com>.
- ²“Matweb - Online Materials Information Resource,” <http://www.matweb.com>.
- ³“Bill Pisano,” <http://recuv.colorado.edu/pisano>.
- ⁴Schmidt, L. V., *Introduction to Aircraft Flight Dynamics*, AIAA, 1998.