

PARALLEL FINITE ELEMENT SIMULATION OF LARGE RAM-AIR PARACHUTES

V. KALRO¹, S. ALIABADI¹, W. GARRARD¹, T. TEZDUYAR^{1*}, S. MITTAL² AND K. STEIN³

¹*Aerospace Engineering and Mechanics, Army HPC Research Center, 1100 Washington Avenue South, University of Minnesota, Minneapolis, MN 55415, U.S.A.*

²*Indian Institute of Technology, Kanpur, India*

³*U.S. Army Natick RD&E Center, Natick, U.S.A.*

SUMMARY

In the near future, large ram-air parachutes are expected to provide the capability of delivering 21 ton payloads from altitudes as high as 25,000 ft. In development and test and evaluation of these parachutes the size of the parachute needed and the deployment stages involved make high-performance computing (HPC) simulations a desirable alternative to costly airdrop tests. Although computational simulations based on realistic, 3D, time-dependent models will continue to be a major computational challenge, advanced finite element simulation techniques recently developed for this purpose and the execution of these techniques on HPC platforms are significant steps in the direction to meet this challenge. In this paper, two approaches for analysis of the inflation and gliding of ram-air parachutes are presented. In one of the approaches the point mass flight mechanics equations are solved with the time-varying drag and lift areas obtained from empirical data. This approach is limited to parachutes with similar configurations to those for which data are available. The other approach is 3D finite element computations based on the Navier–Stokes equations governing the airflow around the parachute canopy and Newton's law of motion governing the 3D dynamics of the canopy, with the forces acting on the canopy calculated from the simulated flow field. At the earlier stages of canopy inflation the parachute is modelled as an expanding box, whereas at the later stages, as it expands, the box transforms to a parafoil and glides. These finite element computations are carried out on the massively parallel supercomputers CRAY T3D and Thinking Machines CM-5, typically with millions of coupled, non-linear finite element equations solved simultaneously at every time step or pseudo-time step of the simulation. © 1997 by John Wiley & Sons, Ltd.

Int. J. Numer. Meth. Fluids, **24**: 1353–1369, 1997

No. of Figures: 20. No. of Tables: 0. No. of References: 21.

KEY WORDS: parallel computations; parachutes; 3D flow simulations

1. INTRODUCTION

Small-scale gliding (ram-air) parachutes, commonly used by the sports parachute community as well as by the military for personnel drop, have reached a satisfactory level of reliability, aerodynamic efficiency and controllability. Furthermore, larger-size versions of these parachutes are being increasingly used for the recovery of large payloads.¹ Gliding parachutes coupled with automatic onboard guidance offer superior controllability and substantial wind penetration when compared with

*Correspondence to: T. Tezduyar, Aerospace Engineering and Mechanics, Army HPC Research Center, 1100 Washington Avenue South, University of Minnesota, Minneapolis, MN 55415, U.S.A.

round parachutes. Since their introduction in the early 1960s, gliding parachutes have been redesigned and refined by the sports community. These personnel-type parachutes are small but have lower wing loading than those required for large payloads.

Future military airdrops will require the deployment of high-altitude delivery systems capable of delivering up to 21 tons from 25,000 ft above ground level with increased accuracy and reduced impact velocity. Gliding parachutes which are at least an order of magnitude larger and with a wing loading three times larger than existing parachutes are currently being developed.

The deployment and control of such large parachutes pose many challenging technical problems. In the design of any parachute system it is important to predict opening forces for choice of materials. Only a limited database is available for large gliding parachutes; therefore methods for inflation analysis based on first principles may be useful in design.

In this paper, two approaches for opening force analysis are presented.^{2,3} One of the approaches is an extension of the classical Pflanz method.⁴⁻⁶ Here the lift and drag areas are assumed to vary with time and the point mass flight mechanics equations are solved as a function of time to yield the opening forces. This method requires empirical data in order to model the temporal evolution of lift and drag, so its predictive capabilities are restricted to parachutes similar to those for which data are available.

The second approach focuses on advanced finite element flow simulation techniques to use realistic, 3D computer models representing the parachute system and its deployment stages. The aerodynamics of ram-air parachutes involves a large number of complex phenomena. The deployment, extension and evolution to the gliding stage involve rapidly changing geometries, unsteady and turbulent flow behaviour and non-linear interactions between the parachute structure, aerodynamic forces and the payload. Even the gliding stage involves deformations of the canopy, changes in the orientation of the parachute and changes in relative motion between the canopy and the payload. All these behaviours require 3D simulation techniques capable of handling time-varying computational domains with formulations which are robust and accurate. Implementations on HPC platforms with sufficient computational speed and memory are necessary.

At this phase of the simulations the time-variation of the geometry of the canopy is assumed to be given, approximated using the initial and final configurations and dimensions of the canopy. However, the dynamics of the canopy, i.e. its translational and rotational motion, still needs to be determined as part of the overall solution. This motion depends on the weight and motion of the payload and also on the aerodynamic forces generated by the unsteady flow field. The airflow around the parachute canopy is governed by the 3D Navier–Stokes equations of incompressible flow with time-dependent spatial domains. The 3D dynamics of the canopy is governed by Newton's law of motion, with the forces acting on the canopy calculated from the simulated flow field. At the initial stage of canopy inflation the parachute is modelled as a falling, expanding box, whereas at the later stages the box transforms to an expanding, gliding parafoil. It is important to note that to accurately resolve the flow around such complex 3D geometries, it is essential to use very refined computational grids leading to very large systems of non-linear equations. The availability of advanced HPC platforms and efficient implementation techniques makes these computations feasible.^{7,8}

To handle the time-variant domains encountered in simulations of parachute systems, we employ the deformable-spatial-domain/stabilized-space-time (DSD/SST) finite element formulation. In this formulation the finite element interpolation polynomials are functions of both space and time and the stabilized variational formulation of the problem is written over the associated space-time domain. This method was introduced by Tezduyar *et al.*^{9,10} to solve incompressible flow problems involving free surfaces, two-liquid interfaces and fluid–structure and fluid–particle interactions. Later, similar formulations were developed for compressible gas flows¹¹ and compressible liquid flows.¹²

In this paper we discuss the steady glide aerodynamics and inflation aerodynamics of ram-air parafoils. The computations reported are performed on the Thinking Machines CM-5 and CRAY T3D massively parallel computers. On the CM-5 we use an SIMD or data-parallel implementation¹³ and on the T3D we use a message-passing paradigm based on PVM.¹⁴

In these computations, updating the finite element mesh as the spatial domain changes its shape with time becomes a major issue. There are several ways of managing this; the detailed description of these approaches together with their advantages and disadvantages can be found in Reference 15. In our studies the motion of each finite element grid point is prescribed explicitly. This is accomplished with no remeshing (i.e. without generating a new set of nodes and elements). With this approach the connectivity of the mesh remains the same throughout the simulation. As a result, both the mesh generation and parallelization set-up costs are reduced to a minimum. This is desirable for practical simulations with hundreds of time steps.

The organization of this paper is as follows. The governing equations are reviewed in Section 2. The stabilized finite element formulation is presented in Section 3. In Section 4 we present the mesh moving scheme used in our simulation of the inflation stage. Parallel implementation aspects are briefly covered in Section 5. The steady glide and inflation simulations comprise Section 6. The flight mechanics simulations are presented in Section 7.

2. GOVERNING EQUATIONS

Let $\Omega_t \subset \mathbb{R}^{n_{sd}}$ and $(0, T)$ be the spatial and temporal domains respectively, where n_{sd} is the number of space dimensions, and let Γ_t denote the boundary of Ω_t . The subscript t implies the time dependence of the spatial domain. The spatial and temporal coordinates are denoted by $\mathbf{x} = (x, y, z) \in \Omega_t$ and $t \in (0, T)$. The governing equations for the flow field computations are the Navier–Stokes equations of incompressible flows,

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \mathbf{f} \right) - \nabla \cdot \boldsymbol{\sigma} = \mathbf{0} \quad \text{on } \Omega_t, \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{on } \Omega. \quad (2)$$

where ρ is the (constant) density and \mathbf{u} is the velocity vector. Here \mathbf{f} is the external force consisting of gravity. For the Newtonian fluids under consideration the stress tensor for a fluid with dynamic viscosity μ is defined as

$$\boldsymbol{\sigma}(\mathbf{u}, p) = -p\mathbf{I} + 2\mu\boldsymbol{\epsilon}(\mathbf{u}), \quad (3)$$

where p is the mechanical pressure and $\boldsymbol{\epsilon}(\mathbf{u})$ is the strain rate tensor given by

$$\boldsymbol{\epsilon}(\mathbf{u}) = \frac{1}{2}[\nabla \mathbf{u} + (\nabla \mathbf{u})^T]. \quad (4)$$

Both Dirichlet- and Neumann-type boundary conditions are accounted for, represented as

$$\mathbf{u} = \mathbf{g} \quad \text{on } (\Gamma_t)_g \quad \mathbf{n} \cdot \boldsymbol{\sigma} = \mathbf{h} \quad \text{on } (\Gamma_t)_h, \quad (5)$$

where $(\Gamma_t)_g$ and $(\Gamma_t)_h$ are complementary subsets of the boundary. The initial condition on the velocity is specified as

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0 \quad \text{on } \Omega_0, \quad (6)$$

where \mathbf{u}_0 is divergence-free.

The parafoil is treated as a solid body with known geometric time variation. The Navier–Stokes equations are coupled together with Newton’s laws of motion for the parafoil system. Purely translational motion is considered. These equations are

$$\mathbf{F} + \mathbf{W} = (W/g)\mathbf{a}, \tag{7}$$

where \mathbf{F} is the aerodynamic force acting on the parafoil and \mathbf{W} is the gravitational force acting on the parafoil/payload system. Here \mathbf{a} is the linear acceleration of the mass centre of the parafoil/payload system.

3. DEFORMING SPATIAL DOMAIN/STABILIZED SPACE-TIME (DSD/SST) FORMULATION

In order to construct the finite element function spaces for the space–time method, we partition the time interval $(0, T)$ into subintervals $I_n = (t_n, t_{n+1})$, where t_n and t_{n+1} belong to an ordered series of time levels $0 = t_0 < t_1 < \dots < t_N = T$. Let $\Omega_n = \Omega_{t_n}$ and $\Gamma_n = \Gamma_{t_n}$. We define the space–time slab Q_n as the domain enclosed by the surfaces Ω_n, Ω_{n+1} and P_n , where P_n is the surface described by the boundary Γ_t as t traverses I_n . As is the case with Γ_t , the surface P_n is decomposed into $(P_n)_g$ and $(P_n)_h$ with respect to the type of boundary condition (Dirichlet or Neumann) being imposed. For each space–time slab we define the corresponding finite element function spaces $(\mathcal{S}_u^h)_n, (\mathcal{V}_u^h)_n, (\mathcal{S}_p^h)_n$ and $(\mathcal{V}_p^h)_n$. Over the element domain this space is formed by using first-order polynomials in space and time. Globally the interpolation functions are continuous in space but discontinuous in time.

The stabilized space–time formulation for deforming domains is then written as follows: given $(\mathbf{u}^h)_n^-$, find $\mathbf{u}^h \in (\mathcal{S}_u^h)_n$ and $p^h \in (\mathcal{S}_p^h)_n$ such that $\forall \mathbf{w}^h \in (\mathcal{V}_u^h)_n$ and $\forall q^h \in (\mathcal{V}_p^h)_n$

$$\begin{aligned} & \int_{\Omega_n} \mathbf{w}^h \cdot \rho \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h + \mathbf{f} \right) d\Omega + \int_{\Omega_n} \boldsymbol{\epsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(p^h, \mathbf{u}^h) dQ + \int_{\Omega_n} q^h \nabla \cdot \mathbf{u}^h dQ \\ & + \sum_{e=1}^{n_{el}} \int_{Q_n^e} \frac{1}{\rho} \boldsymbol{\tau} \mathbf{L}(\mathbf{w}, q) \cdot [\mathbf{L}(\mathbf{u}, p) + \rho \mathbf{f}] dQ + \sum_{e=1}^{n_{el}} \int_{Q_n^e} \delta \mathbf{V} \cdot \mathbf{w}^h \rho \nabla \cdot \mathbf{u}_h dQ \\ & + \int_{\Omega_n} (\mathbf{w}^h)_n^+ \cdot \rho [(\mathbf{u}^h)_n^+ - (\mathbf{u}^h)_n^-] d\Omega = \int_{(P_n)_h} \mathbf{w}^h \cdot \mathbf{h}^h dP. \end{aligned} \tag{8}$$

This process is applied sequentially to all the space–time slabs Q_1, Q_2, \dots, Q_{N-1} . In the variational formulations given by equation (8), the following notation is used:

$$\mathbf{L}(\mathbf{w}, q) = \rho \left(\frac{\partial \mathbf{w}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{w} \right) + \nabla q - 2\mu \nabla \cdot \boldsymbol{\epsilon}(\mathbf{w}), \tag{9}$$

$$\lim_{\epsilon \rightarrow 0} \mathbf{u}(t_n \pm \epsilon), \tag{10}$$

$$\int_{Q_n} (\cdot) dQ = \int_{I_n} \int_{\Omega_n} (\cdot) d\Omega dt, \tag{11}$$

$$\int_{P_n} (\cdot) dP = \int_{I_n} \int_{\Gamma} (\cdot) d\Gamma dt. \tag{12}$$

The computations start with

$$(\mathbf{u}^h)_0^- = \mathbf{u}_0.$$

Remarks

1. In the variational formulation given by equation (8), the first three terms and the right-hand side constitute the Galerkin formulation of the problem.
2. The first series of element-level integrals in equation (8) consists of the least squares terms based on the momentum equation. Here τ is defined as

$$\tau = \left[\left(\frac{2\|\mathbf{u}^h\|}{h} \right)^2 + \left(\frac{4\nu}{h^2} \right)^2 \right]^{-1/2}$$

where h is the element length and $\nu = \mu/\rho$.

3. The second series of element-level integrals is added to the formulation for numerical stability at high Reynolds numbers. These are the least squares terms based on the continuity equation. The coefficient δ is defined as

$$\delta = \frac{h}{2} \|\mathbf{u}^h\|_z, \quad (15)$$

where

$$z = \begin{cases} Re_u/3, & Re_u \leq 3, \\ 1, & Re_u > 3 \end{cases}$$

and Re_u is the cell Reynolds number.

4. Both stabilization terms are weighted residuals and therefore maintain the consistency of the formulation.
5. The sixth term enforces, weakly, the continuity of the velocity field across the space-time slabs.

4. MESH MOVING SCHEMES

In our finite element computations we consider two categories of mesh moving schemes: automatic schemes and special schemes.

In automatic schemes the mesh displacements are prescribed on the boundaries and the displacements in the interior are determined by solving the equations of elasticity for the domain.¹⁵ This scheme is very general and particularly suitable for unstructured meshes; however, it involves the additional cost of solving for the node displacements. Furthermore, when the mesh becomes excessively distorted, remeshing needs to be undertaken. This involves generating a new mesh and projecting the solution from the old mesh to the new mesh. Projection and mesh generation are time-consuming and pose bottlenecks in the parallel implementation.

In special mesh-moving schemes, which are normally designed for specific problems, the motion of each node is prescribed explicitly. In our computations we utilize such a scheme together with a specially designed algebraic mesh generator. In its initial state prior to inflation the parafoil is assumed to have the shape of a box. In its final state it is fully inflated. The mesh connectivity between the two states remain unchanged. The time for inflation is estimated from drop test data. The inflation process is modelled as a smooth transformation between the two states. As a result of this, the mesh generator is used to generate meshes corresponding to the two end states only, and at a

given instant during inflation the mesh is interpolated from these two states with no need for remeshing. At this time the pitching motion of the parafoil is constrained.

It is important to note that the true deformation history would come out of the solution of a complex fluid-structure interaction problem which we plan to consider in the future.

5. PARALLEL IMPLEMENTATION

We briefly describe here the parallel implementation of the finite element algorithms on the Thinking Machines CM-5 and CRAY T3D supercomputers.

The finite element formulations described in earlier sections give rise to very large systems of coupled non-linear equations which require the use of iterative strategies with update techniques such as GMRES¹⁶ for their solution. To further reduce the memory requirements, we use matrix-free iterations and thus eliminate the need to store element-level matrices.

The bulk of the computing cost is taken up by two tasks.

1. Computation of element-level quantities.
2. Communication of data across processors while forming the global equation systems. This involves the data transfer modes Gather (global/node \rightarrow local/element) and Scatter (global/node \leftarrow local/element).

Of these two, the first task is fully parallel in the sense that all operations for each element are entirely local to a processor. The second step can be a bottleneck and thus careful consideration has to be given to the use and distribution of data structures to maximize locality^{17,18} and minimize communication.

On the CM-5 we use a data-parallel programming paradigm. The task of synchronizing processing nodes is internal to the machine. The mesh is partitioned and communication traces are computed and stored through calls to routines in the Connection Machine Scientific Software Library (CMSSL). The interested reader is referred to References 7 and 19 for further details.

On the T3D we use a message-passing paradigm built upon the CRAY extension of the Parallel Virtual Machine (PVM) software. Here synchronization between processors is realized explicitly through PVM barriers placed accordingly in the code. Gather and scatter operations are performed by routines programmed to offer the same functionality available on the CM-5.

6. FINITE ELEMENT SIMULATIONS

Steady state simulations at $Re = 10^7$

All the parafoil simulations reported in this Section are carried out on the CRAY T3D massively parallel supercomputer. The aerodynamic characteristics of a parafoil are the crucial quantities which determine its performance. We have carried out steady state simulations at various angles of attack (α).

The mesh for the parafoil, with a Clark-Y cross-section with a rounded leading edge, consists of 291,437 nodes and 279,888 hexahedral elements. The aspect ratio of the parafoil is 3.0 and the ratio of line length to span 0.6. The x -axis is in the chordwise direction and the y -axis along the span. Boundary conditions consist of no-slip conditions on the parafoil surface, specification of the velocity at the inflow boundary, zero shear stress and zero normal velocities at the side boundaries and traction-free conditions at the outflow boundary. In these simulations, b refers to the parafoil span and c to the chord length.

A semidiscrete formulation coupled with matrix-free iterations was used to obtain the solutions. At every step, 1,129,248 coupled non-linear equations are solved. Turbulence is incorporated into the simulations using a simplified version of the algebraic Baldwin–Lomax²⁰ model,

$$\mu_t = \rho l^2 |\boldsymbol{\omega}|, \quad (16)$$

$$l = \kappa n [1 - \exp(-n^+/A^+)], \quad (17)$$

where μ_t is the eddy viscosity, $\boldsymbol{\omega}$ is the vorticity vector, $\kappa = 0.41$ and $A^+ = 25.0$ are constants, n is the normal distance to the wall and n^+ is the same distance expressed in wall units. In our computations this distance is measured from the closest node on the surface.

Figure 1 shows the chordwise pressure distribution on the parafoil surface at $y/b = 0.0$ (midspan) for various angles of attack. The pressure profile is similar to that of a 2D aerofoil with suction present on the upper surface. Figure 2 shows the chordwise pressure distribution on the parafoil surface close to the parafoil tip for various angles of attack. 3D effects are distinctly present, with the suction decreased owing to leakage of fluid around the tips from the lower to the upper surface. Figures 3–5 show the spanwise pressure distribution on the parafoil surface at different chord lengths for various angles of attack. The pressure distribution is close to elliptical, with the effect of parafoil bumps clearly prevalent at the leading edge; the distribution flattens out progressively towards the trailing edge.

Figure 6 shows the lift (C_l) and drag (C_d) coefficients (which are based on the freestream velocity and the projected area of the parafoil) as functions of the angles of attack. To account for the inlet, a factor²¹ of $0.5h/c$ is added to the drag coefficient. Here h is the inlet height. A typical value of $h/c = 0.1$ is used. We compared these coefficients with available experimental data on parafoils and the calculated values are in the expected range. We could not carry out a closer comparison because of differences in the shape of our parafoil model and those for which data are available. Moreover, while experimental data reveal that most parafoils stall at around 8° – 12° , our model does not indicate this. We think that the reasons for this are as follows: first, our model has a rounded leading edge as opposed to an inlet; second, the predictive capability of the turbulence model we use diminishes at high angles of attack in the regions of flow separation; third, in our computations we assume that the parachutes are not deformable, but this is not the case. Figure 7 shows the lift-to-drag ratio (L/D) as a function of the angle of attack. The typical maximum L/D for the ram-air parafoil alone is in the

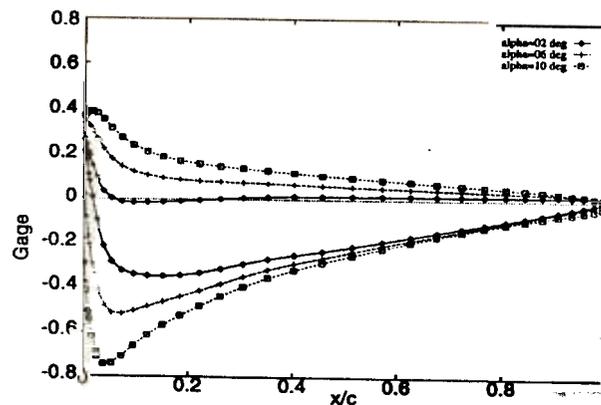


Figure 1. Steady state simulations at $Re = 10^7$ for various angles of attack: chordwise pressure distribution on parafoil surface at $y/b = 0.00$ (midspan)

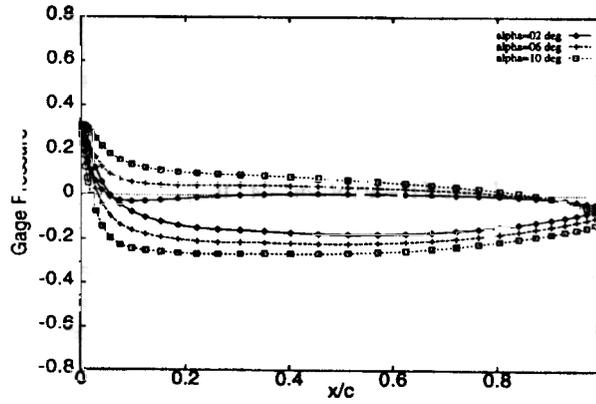


Figure 2. Steady state simulations at $Re = 10^7$ for various angles of attack: chordwise pressure distribution on parafoil surface at $y/b = 0.45$

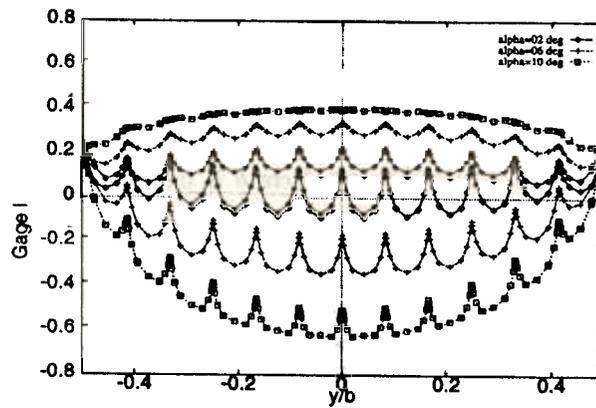


Figure 3. Steady state simulations at $Re = 10^7$ for various angles of attack: spanwise pressure distribution on parafoil surface at $x/c = 0.02$

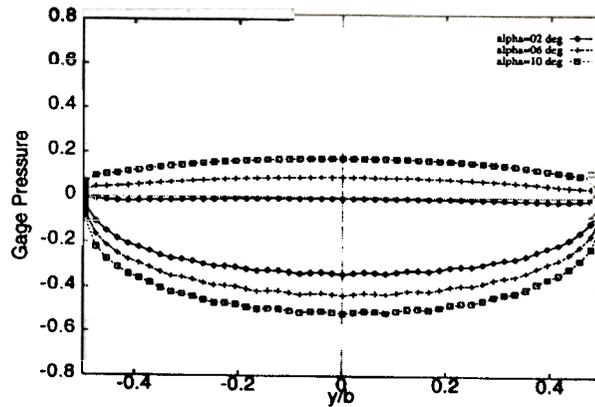


Figure 4. Steady state simulations at $Re = 10^7$ for various angles of attack: spanwise pressure distribution on parafoil surface at $x/c = 0.25$

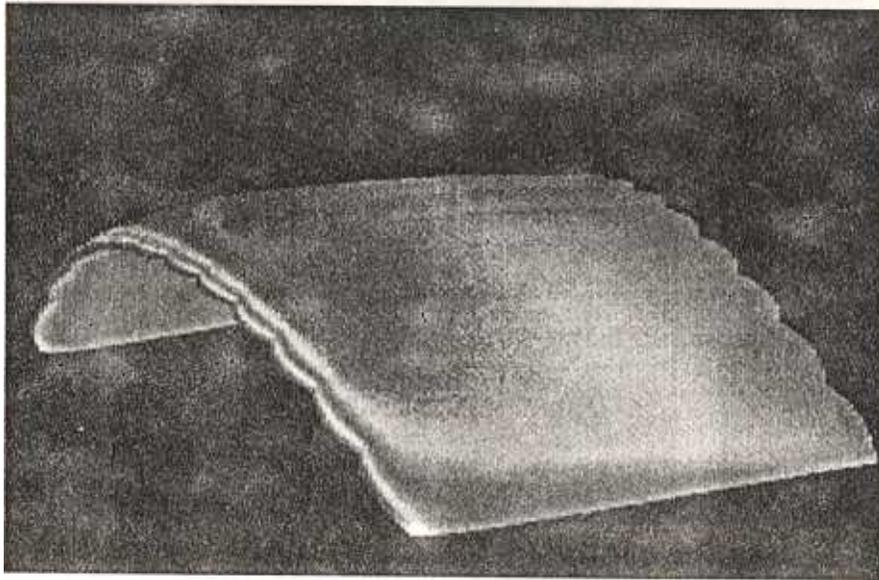


Plate 1. Steady-state simulations at $Re=10^7$, at $\alpha = 2^\circ$: pressure distribution on the parafoil surface

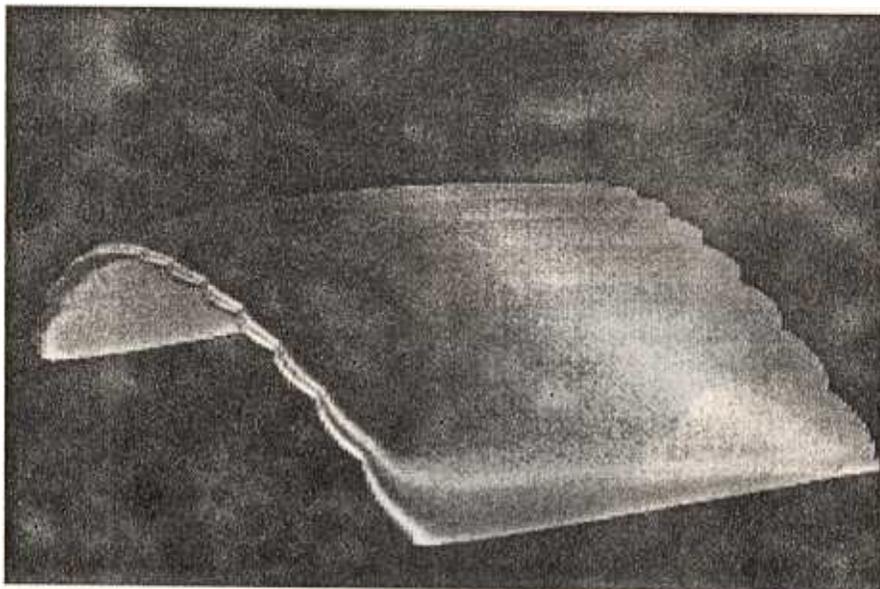


Plate 2. Steady-state simulations at $Re=10^7$, at $\alpha = 6^\circ$: pressure distribution on the parafoil surface

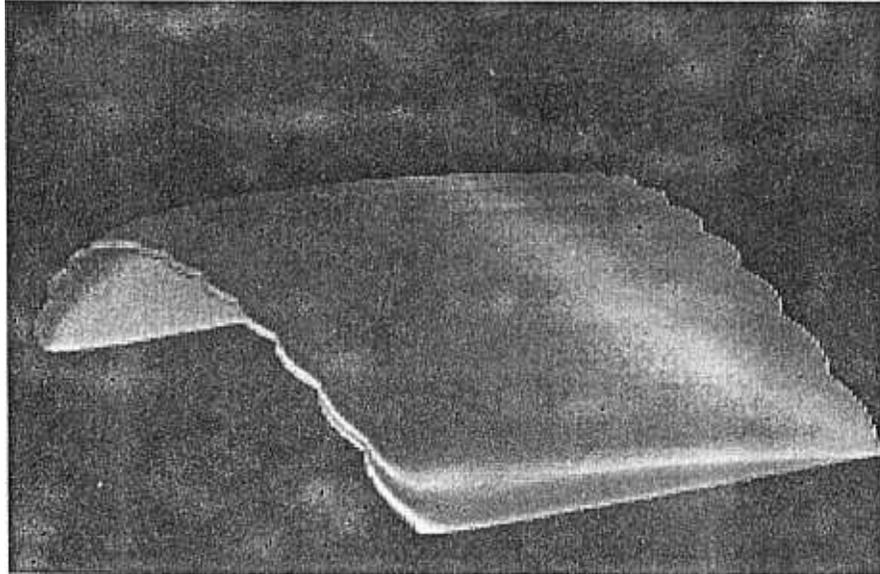


Plate 3. Steady-state simulations at $Re=10^7$, at $\alpha=10^\circ$: pressure distribution on the parafoil surface

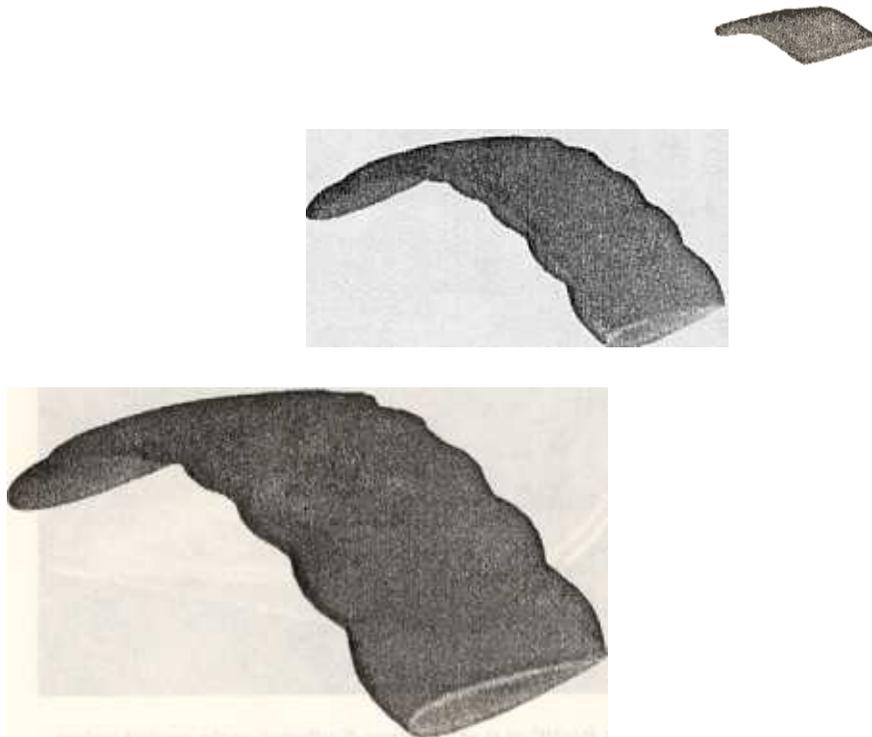


Plate 4. Transformation of the box to a parafoil: pressure distribution on the parafoil surface during the transformation

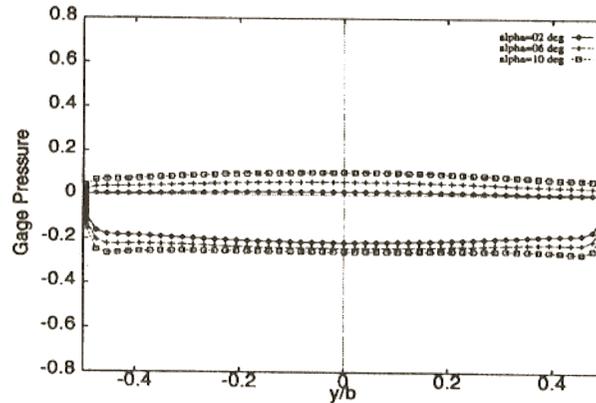


Figure 5. Steady state simulations at $Re = 10^7$ for various angles of attack: spanwise pressure distribution on parafoil surface at $x/c = 0.50$

range 3.0–5.0. This is somewhat higher than obtainable for typical parafoil systems; however, our model does not include line and payload drag.

To establish the convergence of our results, at an angle of attack of 6° , the pressure distribution on the parafoil surface is compared with that obtained on a mesh with 594,587 nodes and 575,968 elements. Figures 8 and 9 indicate good agreement between the solutions obtained on the two meshes. Plates 1–3 show the pressure distribution on the parafoil surface for various angles of attack.

Transformation of the Box to a Parafoil

The inflation of a ram-air inflated gliding parachute takes place in three stages.⁴ During the first stage the canopy expands with little cell inflation. The canopy then pitches forwards and air rushes into the separate cells of the gliding wing, causing them to inflate. As the cells inflate, the parachute begins to take on the shape of an aerofoil, causing lift to be produced while drag decreases. The parachute then transits into equilibrium glide. Since we do not take into account the pitching motion in the computations reported here, only part of the entire process is simulated.

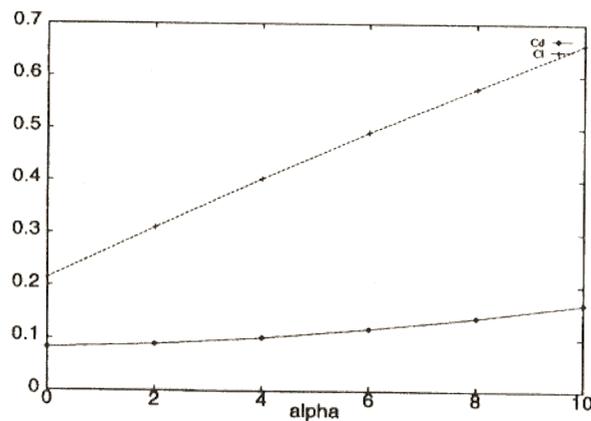


Figure 6. Steady state simulations at $Re = 10^7$: drag and lift coefficients as functions of angle of attack

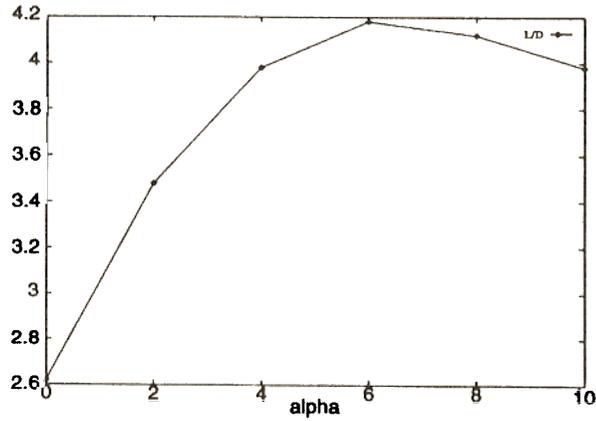


Figure 7. Steady state simulations at $Re = 10^7$: lift/drag (L/D) ratio as function of angle of attack

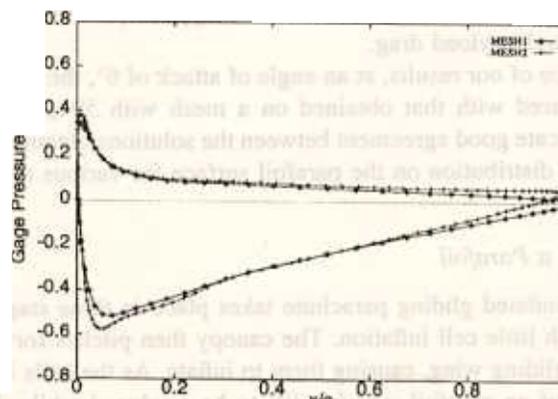


Figure 8. Steady state simulations at $Re = 10^7$ for $\alpha = 6^\circ$: chordwise pressure distribution on parafoil surface at $y/b = 0.00$. Comparison of solutions on mesh 1 (291,437 nodes) and mesh 2 (594,587 nodes)

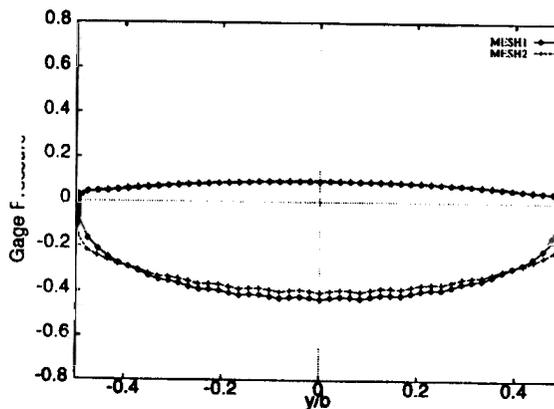


Figure 9. Steady state simulations at $Re = 10^7$ for $\alpha = 6^\circ$: spanwise pressure distribution on parafoil surface at $x/c = 0.25$. Comparison solutions on mesh 1 (291,437 nodes) and mesh 2 (594,587 nodes)

All the parafoil simulations reported in this case are carried out on the Thinking Machines CM-5 massively parallel supercomputer. In these computations the parafoil is allowed to fall under the influence of gravity, inflate and tend towards a steady gliding state. Here a partially inflated box with dimensions of chord \times span \times thickness = 48.0 \times 33.4 \times 12.0 ft³ transforms to a gliding parafoil at a prescribed rate. The box expands by factors of 1.5 and 6.5 along the chord and span respectively and the cross-section becomes the same as that of an NACA 0025 aerofoil. Figure 10 shows the surface mesh for the transformation of the box to a parafoil. The mesh in the middle is interpolated from the two others using the mesh moving scheme described earlier.

The finite element mesh used in this simulation consists of 170,950 nodes and 161,856 hexahedral elements. The boundary conditions consist of specifying the freestream velocity (which is zero) at the inflow boundary, zero normal velocity and zero shear stress at the side boundaries, traction-free conditions at the outflow boundary and no-slip conditions on the parafoil surface. At every time step, 1,304,606 coupled non-linear equations are solved.

The computation starts at $t = 0.0$ s, which corresponds to the steady state solution at the initial configuration of the box at 10° angle of attack and with a velocity of 112 ft s⁻¹. At $t = 2.0$ s the

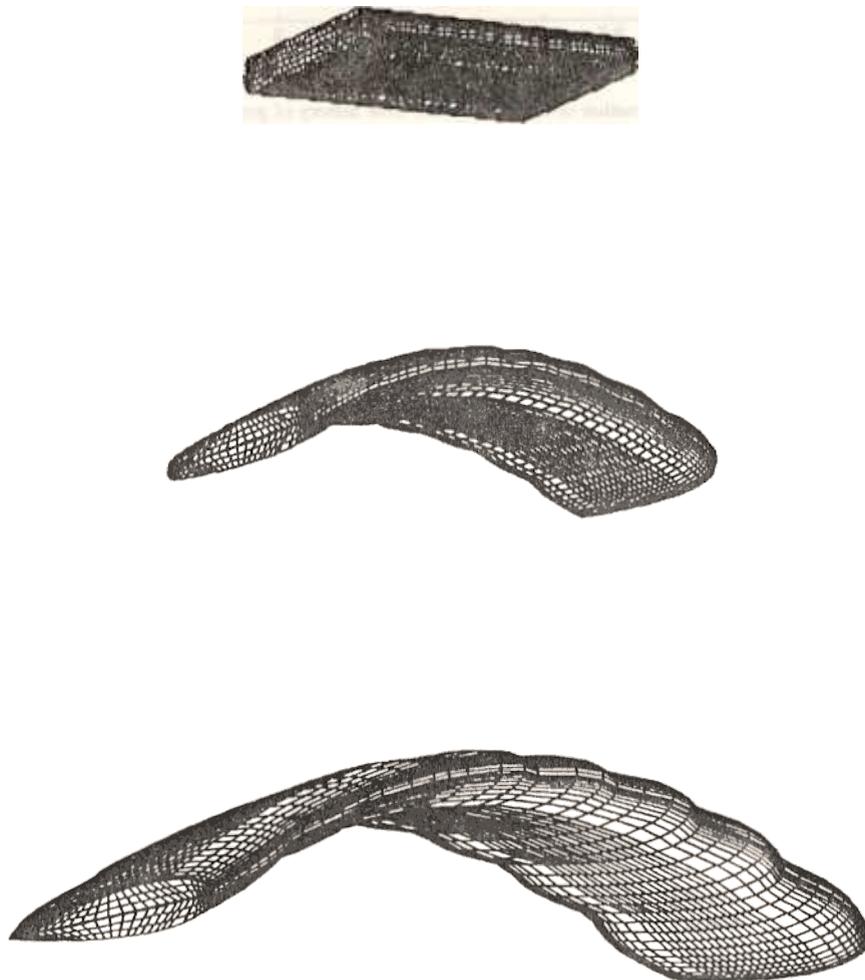


Figure 10. Surface mesh for transformation of box to parafoil

transformation ends, but the computation continues until $t = 3.5$ s. The parafoil/payload system mass for this case is 22,000 lb.

Figures 11 and 12 show the time histories of the projection area and velocity respectively. Initially the box has very little lift and gains speed owing to gravity. As it expands rapidly at high drop velocities, very large aerodynamic forces are generated, leading to the peaks in Figure 13. We

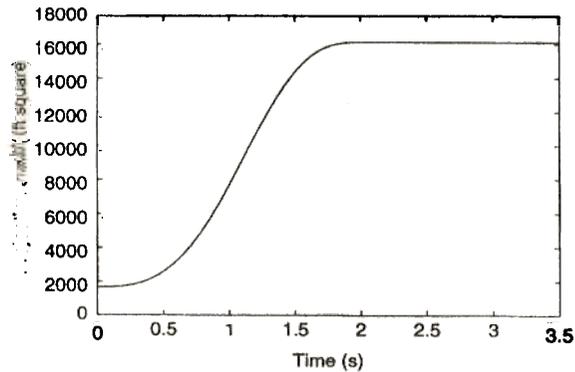


Figure 11. Transformation of box to parafoil: time history of projection area

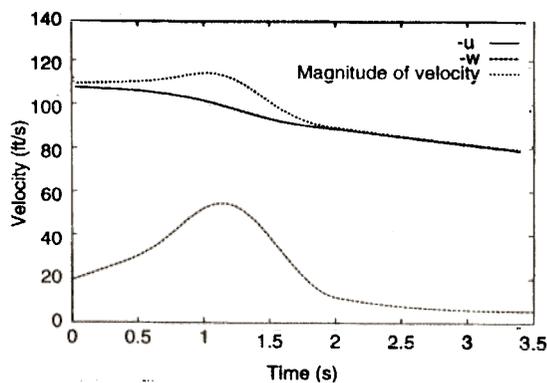


Figure 12. Transformation of box to parafoil: time history of parafoil velocity

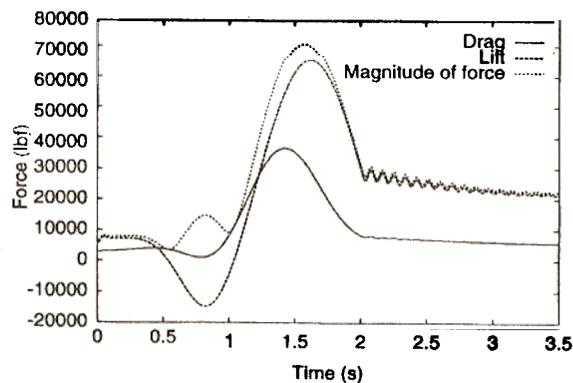


Figure 13. Transformation of box to parafoil: time history of forces acting on parafoil

SIMULATION OF LARGE RAM-AIR PARACHUTES

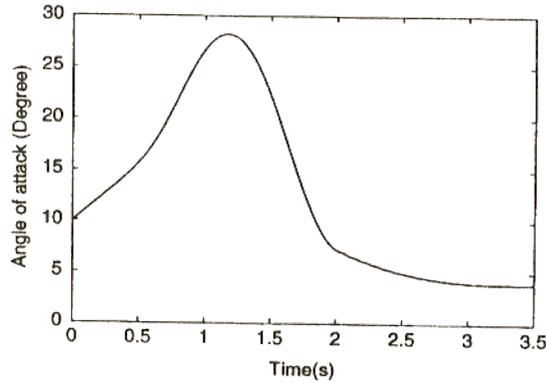


Figure 14. Transformation of box to parafoil: time history of angle of attack

observe that the parafoil attains an almost steady 4° angle of attack as indicated by Figure 14. Plate 4 shows the pressure distribution on the parafoil surface during the transformation.

7. FLIGHT MECHANICS ANALYSIS

In the flight mechanics simulation the dynamics of the parachute/payload system is modelled as a point mass described by the differential equations

$$\frac{dV}{dt} = \frac{g\rho C_d S(t)}{2W} V^2 + g \sin \gamma, \quad (18)$$

$$\frac{d\gamma}{dt} = \frac{g\rho C}{2}. \quad (19)$$

The trajectory and opening force can be determined by solving these equations on a computer. The primary task is to model correctly the evolution of the lift and drag areas as functions of time. The lift and drag areas were extracted from data on drop tests 9-309 and 9-310 of ram-air inflated personnel parachutes and are shown in Figure 15. During the initial inflation of the canopy the airflow is nearly perpendicular to the chord of the canopy and the aerofoil is at a 90° angle of attack. This causes the drag coefficient of the parachute to be much higher than during gliding flight. In addition, the airflow is perpendicular to the inlet openings of the cells. Thus these cells do not inflate and little lift is produced.

During canopy inflation the simulation uses a parabolic growth curve for the drag area as a function of time. By matching the simulation output to measured data, the drag area was found to have a maximum value of 0.80 S_o during the period of canopy inflation, where S_o is the canopy area. The lift area was assumed to be zero during this time.

The time of inflation for the canopy was determined using the relation

$$K_1 = t_{f1} \frac{V_s}{D_o}, \quad (21)$$

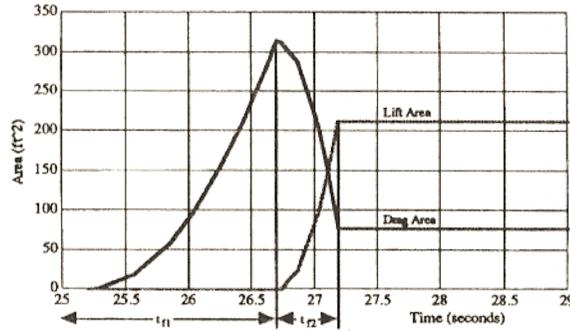


Figure 15. Flight mechanics analysis of drop 9-309: lift and drag areas as functions of time

where $D_o = (S_o/4/\pi)^{0.5}$, t_{f1} is the time for the drag area to grow from zero to its full value ($0.8S_o$) and V_s is the parachute velocity at line stretch. By matching the simulation output to the measured data for personnel-type gliding parachutes with slider reefing, K_1 was determined to have a value of 18.0.

Near the end of inflation the canopy begins to pitch forwards, causing the cells to inflate. As they inflate, the drag coefficient drops and lift begins to be produced. The time between the beginning of cell inflation and the beginning of the glide stage, t_{f2} , is determined similarly to t_{f1} , except that $K_1 = 2.5$ and V_s is the velocity at the beginning of cell inflation. Once again the simulation uses a parabolic curve for the lift area growth and drag area reduction as a function of time. During the final stage of inflation the drag and lift areas are assumed to be constant, with $Cd_o = 0.21$ and $Cl_o = 0.58$.

The simulation was applied to an MC-4 personnel-type parachute with the following characteristics:

$$S_o = 370.0 \text{ ft}^2, \quad b = 28.5 \text{ ft}, \quad c = 13.0 \text{ ft}.$$

The rigged mass was 360 lb. A great deal of detailed information was available on a number of tests for this parachute. The simulation is shown to give good results when compared with measured data. Figures 16 and 17 show comparisons between calculated force and measured force on the parachute for two separate drop tests. The simulation predicts a peak force almost equal to the measured peak force in both cases, but it shows the peak force occurring before the actual measured force.

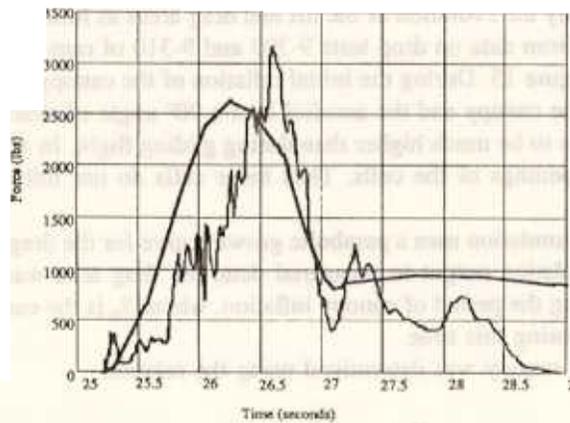


Figure 16. Flight mechanics analysis of drop 9-309: comparison between computed and measured forces

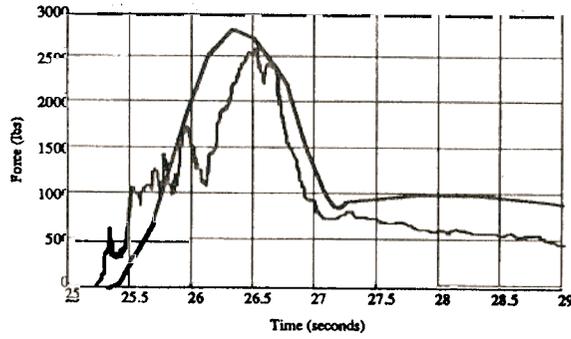


Figure 17. Flight mechanics analysis of drop 9-310: comparison between computed and measured forces

Figure 18 shows the simulated parachute velocity as a function of time. It can be seen that the velocity drops quickly to its steady state gliding value. The flight path angle as a function of time is shown in Figure 19. Initially the payload is falling vertically, but as the parachute deploys and starts to glide, the initial path angle decreases. Figure 20 shows the altitude as a function of range; this clearly shows that the parachute is entering into its steady state gliding mode. This method was

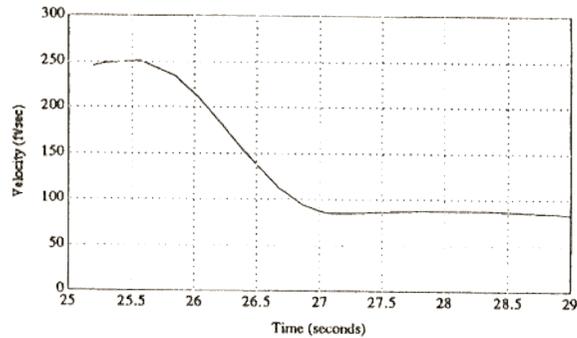


Figure 18. Flight mechanics analysis of drop 9-309: parachute velocity as function of time

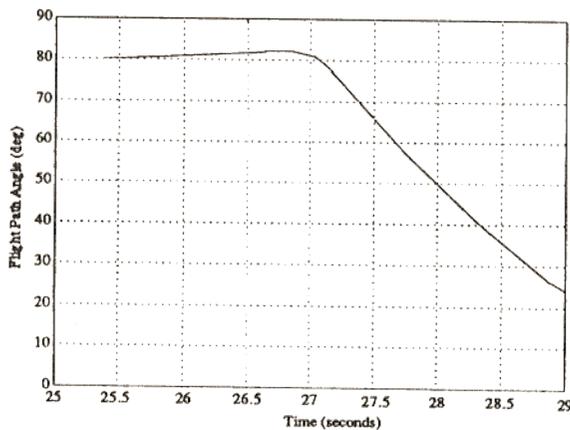


Figure 19. Flight mechanics analysis of drop 9-309: parachute flight path angle as function of time

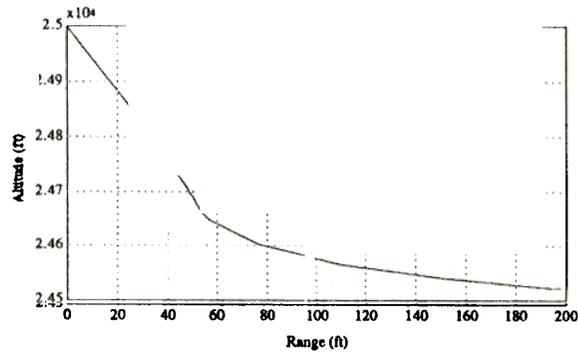


Figure 20. Flight mechanics analysis of drop 9-309: parachute altitude as function of range

applied to inflation of a large gliding parachute with sequential reefing. After considerable manipulation of the lift and drag functions it was possible to obtain opening forces which were similar to experimental data. However, this approach requires access to flight test data and would not be useful for analysis of a design for which no data are available. We are currently working on improving our finite element model to use it to predict accurate drag and lift areas histories for such situations.

8. CONCLUDING REMARKS

We demonstrated the use of two approaches for the simulation of parachute dynamics. One approach employed state-of-the-art finite element formulations and massively parallel computing technology for numerical simulation of flow around a parafoil. Computed lift and drag coefficients and L/D ratios for steady glide were in reasonable agreement with experimental values. Preliminary simulations of the inflation stages were made, with the assumption that the time history of the parafoil surface through the transformation is known. The actual evolution of the surface would involve a complex fluid–structure interaction problem and this will be the direction of our future research. The second approach incorporated lift and drag time histories from flight data. These histories were input to the flight mechanics equations. The use of this method is limited to parachutes similar to those for which data are available. As our computational model evolves, we hope to use a hybrid technique where flight data could be obtained from numerical simulations, thus reducing expensive drop tests.

ACKNOWLEDGEMENTS

This research was sponsored by ARO under grant DAAH04-93-G-0514, by ARPA under NIST contract 60NANB2D1272, by NASA-HSC under grant NAG 9-449 and by the Army High Performance Computing Research Center under the auspices of the Department of the Army, Army Research Laboratory co-operative agreement number DAAH04-95-2-0003/contract number DAAH04-95-C-0008, the content of which does not necessarily reflect the position or the policy of the government, and no official endorsement should be inferred. CRAY C90 time was provided in part by the University of Minnesota Supercomputer Institute.

REFERENCES

1. W. Wailes, 'Development testing of large ram air inflated wings', *AIAA Paper 93-1204*, 1993.
2. W. L. Garrard, T. E. Tezduyar, S. K. Aliabadi, V. Kalro, J. Luker and S. Mittal, 'Inflation analysis of ram air inflated gliding parachutes', *AIAA Paper 95-1565*, 1995.

3. S. K. Aliabadi, W. L. Garrard, V. Kalro, S. Mittal, T. E. Tezduyar and K. R. Stein, 'Parallel finite element computations of the dynamics of large ram air parachutes', *AIAA Paper 95-1581*, 1995.
4. J. S. Lingard, 'A semi-empirical theory to predict the load-time history of an inflating parachute', *AIAA Paper 84-0814*, 1984.
5. J. S. Lingard, 'Unsteady aerodynamics', in *University of Minnesota Parachute Systems Technology Short Courses*, League City, TX, 1994.
6. W. Garrard, 'Application of inflation theories to preliminary force and stress analysis', *AIAA Paper 91-0862*, 1991.
7. T. Tezduyar, S. Aliabadi, M. Behr, A. Johnson and S. Mittal, 'Parallel finite-element computation of 3D flows', *IEEE Comput.*, **26**(10), 27-36 (1993).
8. T. E. Tezduyar, S. K. Aliabadi, M. Behr and S. Mittal, 'Massively parallel finite element simulation of compressible and incompressible flows', *Comput. Methods Appl. Mech. Eng.*, **119**, 157-177 (1994).
9. T. E. Tezduyar, M. Behr and J. Liou, 'A new strategy for finite element computations involving moving boundaries and interfaces—the deforming-spatial-domain/space-time procedure: I. The concepts and the preliminary tests', *Comput. Methods Appl. Mech. Eng.*, **94**, 339-351 (1992).
10. T. E. Tezduyar, M. Behr, S. Mittal and J. Liou, 'A new strategy for finite element computations involving moving boundaries and interfaces—the deforming-spatial-domain/space-time procedure: II. Computation of free-surface flows, two-liquid flows, and flows with drifting cylinders', *Comput. Methods Appl. Mech. Eng.*, **94**, 353-371 (1992).
11. S. K. Aliabadi and T. E. Tezduyar, 'Space-time finite element computation of compressible flows involving moving boundaries and interfaces', *Comput. Methods Appl. Mech. Eng.*, **107**, 209-224 (1993).
12. G. P. Wren, S. E. Ray, S. K. Aliabadi and T. E. Tezduyar, 'Space-time finite element computation of compressible flows between moving components', *Int. j. numer. methods fluids*, **21**, 981-991 (1995).
13. M. Behr, A. Johnson, J. Kennedy, S. Mittal and T. E. Tezduyar, 'Computation of incompressible flows with implicit finite element implementations on the Connection Machine', *Comput. Methods Appl. Mech. Eng.*, **108**, 99-118 (1993).
14. A. Geist, *et al.*, *PVM: Parallel Virtual Machine, A User's Guide for Networked Parallel Computing*, MIT Press, Cambridge, MA, 1994.
15. A. A. Johnson and T. E. Tezduyar, 'Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces', *Comput. Methods Appl. Mech. Eng.*, **119**, 73-94 (1994).
16. Y. Saad and M. Schultz, 'GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems', *SIAM J. Sci. Stat. Comput.*, **7**, 856-869 (1986).
17. A. Pothen, H. D. Simon and L. Wang, 'Spectral nested dissection', *Tech. Rep. RNR-92-003*, NASA Ames Research Center, Moffett Field, CA, 1992.
18. Z. Johan, 'Data parallel finite element techniques for large-scale computational fluid dynamics', *Ph.D. Thesis*, Department of Mechanical Engineering, Stanford University, 1992.
19. V. Kalro and T. Tezduyar, 'Parallel finite element computation of 3D incompressible flows on MPPs', in W. G. Habashi (ed.), *Solution Techniques for Large-Scale CFD Problems*, Wiley, New York, 1995, pp. 59-81.
20. B. Baldwin and H. Lomax, 'Thin layer approximation and algebraic turbulence model for separated turbulent flows', *AIAA Paper 78-257*, 1978.
21. J. S. Lingard, 'Ram-air parachute design', *AIAA Paper*, 1995.