

2-D Parachute Simulation

by the Immersed Boundary Method

Yongsam Kim^{*} and Charles S. Peskin[†]

Abstract

Parachute aerodynamics involves an interaction between the flexible, elastic, porous parachute canopy and the high speed airflow (relative to the parachute) through which the parachute falls. Computer simulation of parachute dynamics typically simplify the problem in various ways, e.g. by considering the parachute as a rigid bluff body. Here, we avoid such simplification by using the immersed boundary (IB) method to study the full fluid-structure interaction. The IB method is generalized to handle porous immersed boundaries, and the generalized method is used to study the influence of porosity on parachute stability.

Keywords: parachute, canopy, payload, porosity, oncoming velocity, control mechanism, stability, immersed boundary method

AMS subject classifications: 65-04, 65M06, 76D05, 76M20

1 Introduction

The purpose of this paper is to show that the immersed boundary (IB) method can be properly applied to the parachute problem and to demonstrate how the porosity of the parachute canopy affects the stability of parachute motion. The problem of parachute

^{*}ICES, University of Texas at Austin, Austin, TX 78712, USA. kimy@ices.utexas.edu

[†]Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY 10012 USA. peskin@cims.nyu.edu

aerodynamics encompasses several of the most complex phenomena in classical fluid dynamics, such as porous bluff-body aerodynamics and highly deformable structures. The cloth part of the parachute, which is called the canopy, acts as a bluff-body resisting a moving flow and changes its shape rapidly in response to the surrounding flow field, but the airflow that generates the aerodynamic forces depends on the shape of the parachute canopy. Thus, parachute aerodynamics is inherently a fluid-structure interaction phenomenon, and to express this requires the time-dependent position of the parachute as well as the usual variables that appear in the Navier-Stokes equation [4, 12, 14, 16].

The IB method was developed to study flow patterns around heart valves, and is a generally useful method for problems in which elastic materials interact with a viscous incompressible fluid. In the IB formulation, the action of the elastic canopy immersed in the air flow appears as a localized body force acting on the fluid. This body force arises from the elastic stresses in the parachute canopy. Moreover, the parachute canopy is required to move at the local fluid velocity as a consequence of the no-slip condition. This condition is modified, however, in the case of canopy porosity, as described below. The central idea of the IB method is that the Navier-Stokes solver does not need to know anything about the complicated time-dependent geometry of the elastic boundary, and that therefore we can escape from the difficulties caused by the interaction between the elastic boundary and the fluid flow. This whole approach has been applied successfully to problems of blood flow in the heart [17, 18, 19, 20, 21, 23], wave propagation in the cochlea [3, 6], platelet aggregation during blood clotting [7], and several other problems [1, 8, 9, 11, 27].

An interesting and important fact about the parachute problem is that it is a porous bluff-body motion. Because the purpose of a parachute is to provide drag, parachute fluid dynamics is irrevocably associated with the airflow around and through a bluff body. This bluff body naturally causes air flowing around the parachute to separate and induces vortex shedding [2]. But since the vorticity behind the bluff canopy shape usually is not symmetric and is too complicated to predict and control, it may affect canopy motion in an asymmetric way and induce an instability of the parachute motion [14]. The parachute canopy is generally equipped with a vent at its apex, which helps to control the inflation process and to stabilize the near-wake

region. Even with an extremely small opening through the apex, the reduction of drag as well as the suppression of asymmetric wake oscillations are observed [10]. Besides its central vent, a typical parachute also has a lot of gaps between ribbons or ringslots on its canopy that make it effectively a porous body. Air flow allowed to go through this porous body collides with and then suppresses large-scale vortex motion behind the canopy and helps to stabilize parachute motion. For more details concerning parachute design and the porosity effect, see [4, 12, 16].

Besides demonstrating the application of the IB method to the parachute problem, it is the main purpose of this paper to give porosity to the parachute and investigate its effect on parachute movement. The real parachute has many discrete holes in the canopy that make it porous such as a vent at the apex of the canopy, and gaps between ribbons and ringslots. In the computation, however, it is impractical to resolve these holes individually. To do so would require that each hole be at least a few fluid meshwidths in diameter. Otherwise the hole would be effectively closed. Thus a very fine mesh would be required. Instead of modeling each hole individually, we give the canopy as a whole a porosity which depends neither on the meshwidth of the boundary nor on that of the fluid.

The idea of how to do this comes from the fact that porosity reduces the drag force of fluid. The drag force is the most significant aerodynamic characteristic of the parachute, but a drag force produced by a porous body is smaller than when a body has no porosity. In the usual application of the IB method, the immersed boundary moves at the local fluid velocity. This is the familiar no-slip condition of a viscous fluid. In the case of a porous immersed boundary, however, we have to allow relative slip between the boundary and the surrounding fluid. This slip is only in the normal direction; the tangential no-slip condition still holds. We assume that the normal relative velocity is determined by Darcy's law [15], i.e., that is proportional to the pressure difference across the boundary. Fortunately, there is no need to evaluate this pressure difference directly, since the IB method provides the force that the immersed boundary applies to the fluid, and the normal component of this force is proportional to the pressure difference across the boundary [23]. Thus we allow a relative slip between boundary and fluid, in the normal direction only, by an amount proportional to the normal component of the boundary force. The constant of proportionality is

called the porosity.

2 Equations of Motion

We begin by stating the mathematical formulation of the equations of motion for a system comprised of a two-dimensional viscous incompressible fluid containing an immersed, elastic, massless, porous boundary.

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u}\right) = -\nabla p + \mu \nabla^2 \mathbf{u} + \mathbf{f}, \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (2)$$

$$\mathbf{f}(\mathbf{x}, t) = \int \mathbf{F}(s, t) \delta(\mathbf{x} - \mathbf{X}(s, t)) ds, \quad (3)$$

$$\begin{aligned} \frac{\partial \mathbf{X}}{\partial t}(s, t) &= \mathbf{u}(\mathbf{X}(s, t), t) + \lambda(\mathbf{F}(s, t) \cdot \mathbf{n}(s, t))\mathbf{n}(s, t), \\ &= \int \mathbf{u}(\mathbf{x}, t) \delta(\mathbf{x} - \mathbf{X}(s, t)) d\mathbf{x} + \lambda(\mathbf{F}(s, t) \cdot \mathbf{n}(s, t))\mathbf{n}(s, t), \end{aligned} \quad (4)$$

$$\mathbf{F}(s, t) = \frac{\partial}{\partial s}(T(s, t)\tau(\mathbf{s}, \mathbf{t})), \quad (5)$$

$$T(s, t) = c\left(\left|\frac{\partial \mathbf{X}(s, t)}{\partial s}\right| - 1\right), \quad (6)$$

$$\tau(s, t) = \frac{\partial \mathbf{X}(s, t)/\partial s}{|\partial \mathbf{X}(s, t)/\partial s|}, \quad (7)$$

$$\mathbf{n}(s, t) = \tau(s, t) \times \mathbf{e}_3. \quad (8)$$

Eqs (1) and (2) are the familiar Navier-Stokes equations for a viscous incompressible fluid. The constant parameters ρ and μ are the fluid density and viscosity, respectively. The unknown functions in the fluid equations are the fluid velocity, $\mathbf{u}(\mathbf{x}, t)$; the fluid pressure, $p(\mathbf{x}, t)$; and the force per unit area applied by the immersed boundary to the (2-D) fluid, $\mathbf{f}(\mathbf{x}, t)$, where $\mathbf{x} = (x, y)$ are fixed Cartesian coordinates, and t is the time.

Eqs (5)-(8) are the immersed boundary equations which are written in Lagrangian form. The unknown $\mathbf{X}(s, t)$ completely describes the motion of the immersed boundary, and also its spatial configuration at any given time. Other unknown functions

of (s, t) that appear in the boundary equations are the unit tangent vector to the boundary, $\tau(s, t)$; the tension, $T(s, t)$; force density at the boundary, $\mathbf{F}(s, t)$; and the unit normal to the boundary $\mathbf{n}(s, t) = \tau(s, t) \times \mathbf{e}_3$, where $\mathbf{e}_3$ is a constant unit vector normal to the plane of the (2-D) flow. The equation for the elastic tension T follows the Hooke's Law for a simple spring which resists both stretching and contracting with a linear constitutive relation but does not resist bending. But it could be easily generalized to include bending resistance or a more complicated nonlinear constitutive law.

Finally, we come to the interaction equations (Eqs (3) and (4)). These both involve the two-dimensional Dirac delta function $\delta(\mathbf{x}) = \delta(x)\delta(y)$, which expresses the local character of the interaction. Eq (3) simply expresses the relation between the two corresponding force densities $\mathbf{f}(\mathbf{x}, t)d\mathbf{x}$ and $\mathbf{F}(s, t)ds$. We can see this fact by integrating each side of Eq (3) over an arbitrary region Ω . It should be noted, however, that, in Eq (3), since $\delta(\mathbf{x})$ is the two dimensional Dirac delta function but integration is only over one dimensional boundary contour, $\mathbf{f}(\mathbf{x}, t)$ is a singular function like a one dimensional delta function.

Eq (4) is the equation of motion of the immersed elastic boundary. It is explained as follows. First consider the special case $\lambda = 0$. Then Eq (4) is the familiar no-slip condition. In the first form of Eq (4), the expression $\mathbf{u}(\mathbf{X}(s, t), t)$ is the fluid velocity evaluated at the boundary. This is rewritten in terms of the Dirac delta function in the second form of Eq (4). We do so in order to expose a certain symmetry with Eq (3), in which the force generated by the immersed boundary is re-expressed as a body force acting on the fluid. This symmetry is important in the construction of our numerical scheme. In the following discussion of porosity, however, we shall simply use the notation $\mathbf{U}(s, t)$ for the fluid velocity evaluated at the boundary point $\mathbf{X}(s, t)$, i.e.,

$$\mathbf{U}(s, t) = \mathbf{u}(\mathbf{X}(s, t), t) = \int \mathbf{u}(\mathbf{x}, t)\delta(\mathbf{x} - \mathbf{X}(s, t))d\mathbf{x}. \quad (9)$$

Now consider the porosity of the immersed elastic boundary. (This discussion will be phrased in terms of the 2-D case, but of course a similar discussion could be made in 3-D, with a few changes of units.) Consider the interval $(s, s + ds)$ of the immersed elastic boundary. Let β be the number density of pores, in the sense that βds is the number of pores in the interval $(s, s + ds)$. Let each pore have an aerodynamic

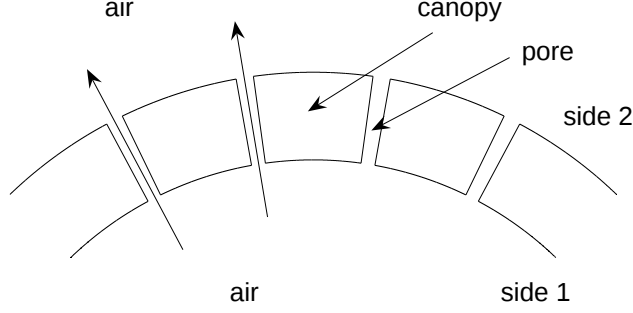


Figure 1: Porous boundary. The pores have small diameter in comparison to their length (the thickness of the boundary).

conductance (reciprocal of resistance) equal to γ . This means that the flux through the pore is equal to $\gamma(p_1 - p_2)$ where p_1 and p_2 are the pressures on the two sides of the boundary, see Figure 1. Then the flux through the interval $(s, s + ds)$ of the boundary is given by $\beta\gamma(p_1 - p_2)ds$. This flux can be evaluated in another way by considering the difference between the fluid velocity at the boundary and the velocity of the boundary itself. The resulting expression for the flux is

$$(\mathbf{U}(s, t) - \frac{\partial \mathbf{X}}{\partial t}(s, t)) \cdot \mathbf{n} \left| \frac{\partial \mathbf{X}(s, t)}{\partial s} \right| ds, \quad (10)$$

where $\mathbf{n}$ is the unit normal to the boundary pointing from side 1 towards side 2. The factor $\left| \frac{\partial \mathbf{X}(s, t)}{\partial s} \right|$ appears because $\left| \frac{\partial \mathbf{X}(s, t)}{\partial s} \right| ds$ is the arc length of the segment $(s, s + ds)$. Setting these two expressions for the flux equal to each other, we get

$$\beta\gamma(p_1 - p_2)ds = (\mathbf{U}(s, t) - \frac{\partial \mathbf{X}}{\partial t}(s, t)) \cdot \mathbf{n} \left| \frac{\partial \mathbf{X}(s, t)}{\partial s} \right| ds. \quad (11)$$

But $(p_1 - p_2)$ can be related to the normal component of the boundary force $\mathbf{F}(s, t)$. The normal equilibrium of our massless boundary requires that

$$(p_1 - p_2) \left| \frac{\partial \mathbf{X}}{\partial s}(s, t) \right| ds + \mathbf{F}(s, t) \cdot \mathbf{n} ds = 0. \quad (12)$$

Combining these equations, we find

$$\left(\frac{\partial \mathbf{X}}{\partial t}(s, t) - \mathbf{U}(s, t) \right) \cdot \mathbf{n} = \frac{\beta\gamma}{\left| \frac{\partial \mathbf{X}(s, t)}{\partial s} \right|^2} \mathbf{F}(s, t) \cdot \mathbf{n}. \quad (13)$$

We also need a tangential component for the porous boundary condition. This is a complicated issue, see for example [3], where a slip boundary condition is derived

at the interface of a fluid and a porous solid. In that paper, the solid has isotropic porosity. Here we assume that the parachute canopy is a thin shell of porous material *with pores oriented normal to the surface of the shell*. Moreover, we assume that the pores have small diameter in comparison to their length (the thickness of the shell). Under these conditions, the flow in each pore is normal to the surface of the canopy, so the tangential velocity of the flow in each pore is zero. Between the pores, we also have zero tangential velocity by the no-slip condition at a solid-fluid interface (see Figure 1). Combining these observations, it seems clear that the appropriate tangential boundary condition is the familiar no-slip condition, despite the porosity.

$$\left(\frac{\partial \mathbf{X}(s, t)}{\partial t} - \mathbf{U}(s, t)\right) \cdot \boldsymbol{\tau} = 0. \quad (14)$$

Then

$$\frac{\partial \mathbf{X}}{\partial t}(s, t) = \mathbf{U}(s, t) + \frac{\beta\gamma}{|\partial \mathbf{X}(s, t)/\partial s|^2} (\mathbf{F}(s, t) \cdot \mathbf{n}) \mathbf{n}, \quad (15)$$

which is equivalent to Eq (4), provided we set

$$\lambda = \frac{\beta\gamma}{|\partial \mathbf{X}(s, t)/\partial s|^2}. \quad (16)$$

A question that still remains is whether β and γ depend on $|\partial \mathbf{X}(s, t)/\partial s|$. Recall that $|\partial \mathbf{X}(s, t)/\partial s|$ is the ratio of arc length to unstressed arc length, so it measures how stretched the material is. Intuitively, one would think that stretch would tend to increase either the number of pores or their conductance or both. Thus, one would expect $\beta\gamma$ to increase with $|\partial \mathbf{X}(s, t)/\partial s|$ but in a manner that would be hard to determine a priori. Here we make the simple assumption that λ is independent of $|\partial \mathbf{X}(s, t)/\partial s|$, i.e, that $\beta\gamma$ is proportional to $|\partial \mathbf{X}(s, t)/\partial s|^2$. More information about the material would be needed to refine this assumption.

3 Numerical Method

We now describe a formally second-order IB method to solve the equations of motion [13, 22]. The word ‘formally’ is used as a reminder that this scheme is only second-order accurate for problems with smooth solutions. Even though our solutions are not smooth (the velocity has jumps in derivative across the immersed boundary), the use of the formally second-order method results in improved accuracy, see [13].

The specific formally second-order method that we use is the one described in [22]. In this method, each time step proceeds in two substeps, which are called the preliminary and final substeps. In the preliminary substep, we get data at time level $n + \frac{1}{2}$ from data at n by a first-order accurate method. Then the final step starts again at time level n and proceeds to time level $n + 1$ by a second-order accurate method. This Runge-Kutta framework allows the second-order accuracy of the final substep to be the overall accuracy of the scheme.

We use a superscript to denote the time level. Thus $\mathbf{X}^n(s)$ is shorthand for $\mathbf{X}(s, n\Delta t)$, where Δt is the duration of the time step, and similarly for all other variables. Our goal is to compute updated $\mathbf{u}^{n+1}$ and $\mathbf{X}^{n+1}$ from given data $\mathbf{u}^n$ and $\mathbf{X}^n$.

Before describing how this is done, we have to say a few words about the spatial discretization. There are two such discretizations: one for the fluid and one for the elastic boundary. The grid on which the fluid variables are defined is a fixed uniform lattice of meshwidth $h = \Delta x_1 = \Delta x_2$. Now we define the central difference operator D_i , defined for $i = 1, 2$ as follows:

$$(D_i\phi)(\mathbf{x}) = \frac{\phi(\mathbf{x} + h\mathbf{e}^i) - \phi(\mathbf{x} - h\mathbf{e}^i)}{2h}, \quad (17)$$

where $\mathbf{e}^i$ is the unit vector in the i -th coordinate direction. As the notation suggests, the difference operator in i -th direction D_i corresponds to the i -th component of the differential operator ∇ . Thus $\mathbf{D}p$ will be the discrete gradient of p , and $\mathbf{D} \cdot \mathbf{u}$ will be the discrete divergence of $\mathbf{u}$.

We shall also make use the central difference Laplacian L .

$$(L\phi)(\mathbf{x}) = \sum_{i=1}^2 \frac{\phi(\mathbf{x} + h\mathbf{e}^i) + \phi(\mathbf{x} - h\mathbf{e}^i) - 2\phi(\mathbf{x})}{h^2}. \quad (18)$$

The immersed boundary variables are defined as functions of s with meshwidths Δs . For any function $\phi(s)$, define D_s :

$$(D_s\phi)(s) = \frac{\phi(s + \Delta s/2) - \phi(s - \Delta s/2)}{\Delta s}. \quad (19)$$

The fluid mesh and the elastic boundary mesh defined above are connected by a smoothed approximation to the Dirac delta function. It is denoted δ_h and is of the

following form:

$$\delta_h(\mathbf{x}) = h^{-2} \phi\left(\frac{x_1}{h}\right) \phi\left(\frac{x_2}{h}\right), \quad (20)$$

where $\mathbf{x} = (x_1, x_2)$, and the function ϕ is given by

$$\phi(r) = \begin{cases} \frac{3-2|r|+\sqrt{1+4|r|-4r^2}}{8} & , \quad \text{if } |r| \leq 1 \\ \frac{5-2|r|-\sqrt{-7+12|r|-4r^2}}{8} & , \quad \text{if } 1 \leq |r| \leq 2 \\ 0 & , \quad \text{if } 2 \leq |r|. \end{cases} \quad (21)$$

The motivation and derivation for this particular choice is discussed in [19, 21].

We are now ready to describe a typical timestep of the numerical scheme. The preliminary substep which goes from time level n to $n + \frac{1}{2}$ proceeds as follows:

First, update the position of the massless boundary $\mathbf{X}^{n+\frac{1}{2}}(s)$.

$$\frac{\mathbf{X}^{n+\frac{1}{2}} - \mathbf{X}^n}{\Delta t/2} = \sum_{\mathbf{x}} \mathbf{u}^n(\mathbf{x}) \delta_h(\mathbf{x} - \mathbf{X}^n(s)) h^2 + \lambda(\mathbf{F}^n \cdot \mathbf{n}^n) \mathbf{n}^n, \quad (22)$$

where $\mathbf{n}^n$ is the unit normal to the boundary $\mathbf{X}^n$, and $\mathbf{F}^n$ can be calculated from $\mathbf{X}^n$ (see below). In general $\mathbf{n} = \tau \times \mathbf{e}_3$ in Eq (22), but τ is not defined at each boundary point. To overcome this, we use

$$\tau^n(s) = \frac{\mathbf{X}^n(s + \Delta s) - \mathbf{X}^n(s - \Delta s)}{|\mathbf{X}^n(s + \Delta s) - \mathbf{X}^n(s - \Delta s)|} \quad (23)$$

and then get $\mathbf{n}^n(s) = \mathbf{e}_3 \times \tau^n(s)$.

Next, we calculate the force density $\mathbf{F}^{n+\frac{1}{2}}$ from the deformation of elastic boundary $\mathbf{X}^{n+\frac{1}{2}}$. The force density $\mathbf{F}^{n+\frac{1}{2}}$ can be obtained by the discretization of Eqs (5)-(7):

$$\tau^{n+\frac{1}{2}} = \frac{D_s \mathbf{X}^{n+\frac{1}{2}}}{|(D_s \mathbf{X}^{n+\frac{1}{2}})|}, \quad T^{n+\frac{1}{2}} = c(|D_s \mathbf{X}^{n+\frac{1}{2}}| - 1), \quad \mathbf{F}^{n+\frac{1}{2}} = D_s(T^{n+\frac{1}{2}} \tau^{n+\frac{1}{2}}). \quad (24)$$

Note that, since $\tau^{n+\frac{1}{2}}$ and $T^{n+\frac{1}{2}}$ are defined at values of s halfway between those at which $\mathbf{X}^{n+\frac{1}{2}}$ is defined, and so $\mathbf{F}^{n+\frac{1}{2}}$ are defined at the same values of s as $\mathbf{X}^{n+\frac{1}{2}}$.

Now we have to change this elastic force defined on Lagrangian grid points into the force at Eulerian spatial grid points to be applied in the Navier-Stokes equations.

$$\mathbf{f}^{n+\frac{1}{2}} = \sum_s \mathbf{F}^{n+\frac{1}{2}}(s) \delta_h(\mathbf{x} - \mathbf{X}^{n+\frac{1}{2}}(s)) \Delta s \quad (25)$$

With $\mathbf{f}^{n+\frac{1}{2}}$ in hand, we can turn to solving the Navier-Stokes equations.

$$\rho\left(\frac{u_i^{n+\frac{1}{2}} - u_i^n}{\Delta t/2} + \frac{1}{2}(\mathbf{u} \cdot \mathbf{D}u_i + \mathbf{D} \cdot (\mathbf{u}u_i))^n\right) + D_i p^{n+\frac{1}{2}} = \mu L u_i^{n+\frac{1}{2}} + f_i^{n+\frac{1}{2}}, \quad (26)$$

for $i = 1, 2$, and

$$\mathbf{D} \cdot \mathbf{u}^{n+\frac{1}{2}} = 0 \quad (27)$$

Note that the unknowns in Eqs (26) and (27) are $u_i^{n+\frac{1}{2}}$ and $p^{n+\frac{1}{2}}$ and that they enter into these equations linearly. Since all the coefficients of these equations are constants, the system of Eqs (26)-(27) can be solved by Fast Fourier Transform with the periodic boundary condition [19, 21].

The final step is the correction of $\mathbf{u}^{n+\frac{1}{2}}$ and $\mathbf{X}^{n+\frac{1}{2}}$ obtained in the preliminary step.

First, using the fluid velocity $\mathbf{u}^{n+\frac{1}{2}}$, we can find the boundary configuration $\mathbf{X}^{n+1}$.

$$\frac{\mathbf{X}^{n+1} - \mathbf{X}^n}{\Delta t} = \sum_{\mathbf{x}} \mathbf{u}^{n+\frac{1}{2}}(\mathbf{x}) \delta_h(\mathbf{x} - \mathbf{X}^{n+\frac{1}{2}}(s)) h^2 + \lambda(\mathbf{F}^{n+\frac{1}{2}} \cdot \mathbf{n}^{n+\frac{1}{2}}) \mathbf{n}^{n+\frac{1}{2}}, \quad (28)$$

where $\mathbf{n}^{n+\frac{1}{2}}$ is the unit normal to the boundary $\mathbf{X}^{n+\frac{1}{2}}$ and can be obtained in the same manner as $\mathbf{n}^n$ in the preliminary step.

The last thing that we have to do is to update the fluid velocity data.

$$\rho\left(\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{1}{2}(\mathbf{u} \cdot \mathbf{D}u_i + \mathbf{D} \cdot (\mathbf{u}u_i))^{n+\frac{1}{2}}\right) + D_i p^{n+\frac{1}{2}} = \frac{1}{2}\mu L(u_i^{n+1} + u_i^n) + f_i^{n+\frac{1}{2}}, \quad (29)$$

for $i = 1, 2$, and

$$\mathbf{D} \cdot \mathbf{u}^{n+1} = 0 \quad (30)$$

Since we have now computed $\mathbf{u}^{n+1}$ and $\mathbf{X}^{n+1}$, the timestep is complete.

4 Two-dimensional Parachute Model

In this section we introduce a 2-dimensional computational model of a parachute. We present the initial configuration of our model and display the physical and computational parameters which are used in the numerical experiments.

Consider the incompressible viscous fluid in a square box $(0, 4\text{m}) \times (0, 4\text{m})$ with periodic boundary conditions (but see below) which contains an immersed elastic

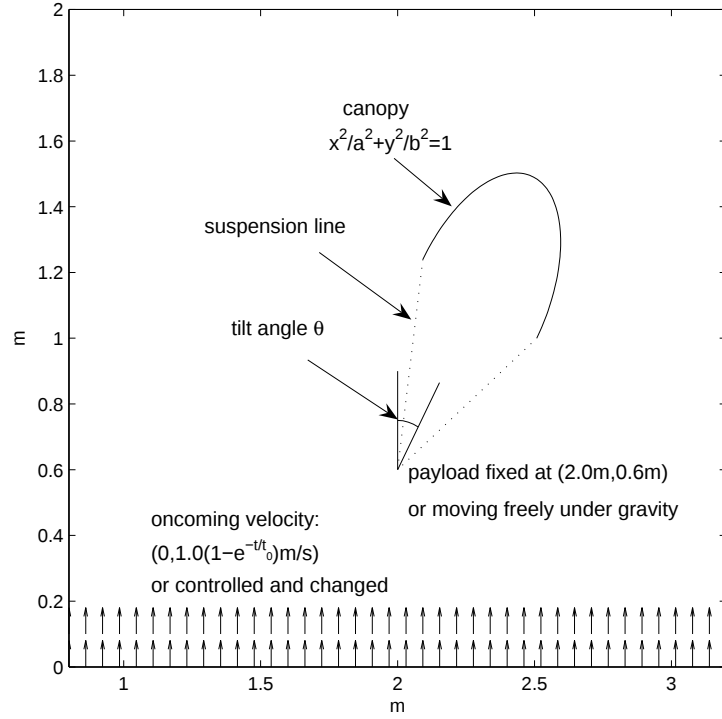


Figure 2: The initial configuration of the parachute. The tilt angle from the y-axis is the initial attack angle θ with respect to the oncoming velocity $\mathbf{u}_0 = (0, V_0(t))$. The payload is either fixed or moving freely. While the oncoming velocity is prescribed in the fixed-payload case, it changes determined by a control mechanism in the free-payload case.

parachute. Figure 2 shows the initial constructed configuration of a 2-D parachute and a part of the computational domain. The canopy has as its initial shape an upper half ellipse centered at $(0, 0)$ which is expressed $\{(a\cos(s), b\sin(s)) : 0 \leq s \leq \pi\}$ as a parameterized curve. Linking the two end points of the canopy to the payload position completes the basic symmetric parachute which is then rotated about the payload point by an arbitrary angle θ . The first simulation has its payload position fixed at the point $(2.0\text{m}, 0.6\text{m})$ which does not move and, in each time step, is just attached to two end points of the updated canopy position with two straight lines. These two lines, which are called suspension lines, are pure force generators. They do not interact with the fluid along their length, but simply act as linear springs which apply force to the two end points of the canopy.

The most natural way to model a parachute would be to let it fall, under the influence of gravity acting on its payload, through air which would be at rest at a large distance from the parachute. Here, however, we either fix the payload at the point (2.0m,0.6m) or let it move freely with gravity and drive the air upwards. To keep the air flowing upwards, at each time step, we apply to the Navier-Stokes equation an external force

$$\mathbf{f}_0(\mathbf{x}, t) = \begin{cases} \alpha_0(\mathbf{u}_0(t) - \mathbf{u}(\mathbf{x}, t)) & , \quad \mathbf{x} \in \Omega_0(h) \\ 0 & , \quad \text{otherwise,} \end{cases} \quad (31)$$

where h is the meshwidth and $\Omega_0(h)$ is the set of grid points containing more than two grid lines on which we want to control the oncoming velocity. In the coarsest meshwidth, we choose two grid lines for $\Omega_0(h)$ and, as the meshwidth becomes finer, $\Omega_0(h)$ becomes bigger reversely. $\mathbf{u}_0(t)$, which is called the oncoming velocity, is the desired velocity on those lines, and α_0 is a constant. When α_0 is large, the grid velocity is driven rapidly towards $\mathbf{u}_0(t)$ within $\Omega_0(h)$.

For the fixed payload case, for example, the oncoming velocity $\mathbf{u}_0(t)$ is given by $(0, 1.0(1 - \exp(-\frac{t}{t_0}))\text{m/s})$. Here, in order to avoid the abrupt change of velocity field and then parachute configuration, we set the oncoming velocity as a function of time which is initially (0,0) but increases gradually up to a constant value (0,1.0m/s).

Although this method of specifying the oncoming velocity is crude, it is not quite as crude as one might think. we address here two concerns that may occur to the reader. First, Does the velocity field on $\Omega_0(h)$ really match the oncoming velocity $\mathbf{u}_0(t)$? When we look closely at the velocity field in this region, we can see that the real velocity in this region quickly catch up with the oncoming velocity and becomes very close to it. Second, since the pressure is computed by solving a periodic problem, doesn't this result in unwanted interaction between the top and bottom of the domain? To investigate this, we have compared the pressures above and below the lines on which the oncoming velocity is specified. After an initial transient, these pressures are actually *anti*-correlated, thus showing that the specified oncoming velocity has effectively broken the periodicity, as one would expect by considering, say, the effect of thin porous plug in a circular pipe.

The second simulation is done with the same parachute as before but we remove the tethered point and allow the payload to move. The point payload of our model has

no direct effect on the fluid such as vortex shedding, and its movement is independent of the local fluid velocity around the payload. But it has a point mass M and therefore feels a gravitational force $M\mathbf{g}$, where $\mathbf{g}$ is the downward acceleration $(0, -g)$. Since the payload is also loaded by the stresses of two suspension lines, let $T_i(t)$, $(i = 1, 2)$ be the tension of each suspension line and $\tau_i(t)$, $(i = 1, 2)$ be the unit vectors pointing from the payload to the two end points of the canopy, respectively, we then have the total force $\mathbf{F}_p(t)$ acting on the payload,

$$\mathbf{F}_p(t) = M\mathbf{g} + \sum_{i=1}^2 T_i(t)\tau_i(t). \quad (32)$$

Let the velocity of the payload be $\mathbf{U}_p(t)$ and the position be $\mathbf{X}_p(t)$, then we have the equations of motion:

$$\frac{d\mathbf{U}_p(t)}{dt} = \frac{\mathbf{F}_p(t)}{M}, \quad (33)$$

$$\frac{d\mathbf{X}_p(t)}{dt} = \mathbf{U}_p(t). \quad (34)$$

In the fixed-payload case, the oncoming velocity was arbitrary, but with a free payload if we specify the oncoming velocity arbitrarily we shall find that it is either too small, in which case the parachute will fall out the bottom of the domain, or too large, in which case the parachute will rise out the top of the computational domain. To keep the parachute within the domain, and away from the meshlines on which the oncoming velocity is specified, we use a control mechanism to adjust the oncoming velocity in such a manner that the y-coordinate of the payload settles to a predetermined value. The equation governing this control mechanism is as follows:

$$\frac{dV_0(t)}{dt} = k(y_{target} - Y_p(t)) - \sigma V_p(t). \quad (35)$$

In Eq (35), $Y_p(t)$ and $V_p(t)$ are obtained by taking the y-components of $\mathbf{X}_p(t)$ and $\mathbf{U}_p(t)$ from Eqs (33) and (34) respectively. The velocity $(0, V_0(t)) = \mathbf{u}_0$ is the oncoming velocity at time t , y_{target} is the fixed value at which the y-coordinate of the payload is supposed to have its equilibrium, and k and σ are constant coefficients. The equation says that if, at some time, the height of the payload $Y_p(t)$ is lower than the target position of the payload y_{target} , the oncoming velocity increases, and if $Y_p(t)$ is greater than y_{target} , the oncoming velocity decreases. But the change of the

oncoming velocity is damped according to $V_p(t)$ in order to avoid large oscillations of the oncoming velocity. The coefficients k and σ are chosen so that the y-coordinate of the payload is stable around the target position y_{target} . Note, however, that we allow the parachute to move out the side of the domain, in which case we should handle the data outside the domain by duplicating them into the domain in a periodic way.

The readers may wonder in the free-payload case, why we need an oncoming velocity at all. Why not use the periodicity to let the parachute fall out the bottom of the domain and reappear as it does so at the top? Aside from the interaction of the parachute with its own wake that would then occur, there is a more fundamental problem. Since the periodic domain contains only a finite mass of fluid to which a constant force (the weight of the payload) is applied, the total downward momentum of the system will increase linearly with time, and no terminal velocity will exist. We avoid this difficulty through the use of the oncoming velocity.

The overall performance of a parachute can be summarized by the relationship between the drag force it generates and the speed at which it is falling (relative to the air at a large distance from the parachute). The two types of computer experiments introduced above assess this relationship in different ways. When the payload is fixed in place and the oncoming velocity is arbitrary, then the speed of the parachute (relative to the distant air) is the independent variable, and the drag force is computed. Indeed, the drag force can be determined simply by examining the tensions and angles of the suspension lines. When the payload is free and the oncoming velocity is adjusted to keep the parachute from falling or rising, then the drag force is the independent variable, since it has to be equal to the specified weight of the payload. In this case, the speed corresponding to the given drag force is just the equilibrium value of the oncoming velocity, as set by the control mechanism. Although we would expect to get the same steady-state relationship between speed and drag force from either type of computational experiment, the dynamics of the two cases could certainly be different. This is because the free-payload case has two additional degrees of freedom and one additional parameter (the payload mass), which might well be expected to influence the dynamics.

Two important parameters of our computational experiments are the initial tilt angle θ (defined in Figure 2) and the porosity λ (defined in Eq (16) and discussed

Table 1: Physical parameters.

Physical parameters	symbol	magnitude	unit
Density	ρ	1.2	kg/m ³
Viscosity	μ	0.002	kg/(m · s)
Gravitational Acceleration	g	9.8	m/s ²
Mass of Payload	M	0.08	kg
Computational Domain		4×4	m × m
Canopy Length		1.0561	m
Suspension Line Length		0.6462	m
Opening Length	$2a$	0.1 – 0.48	m
Porosity	λ	0.0 – 0.2	m ² /(N · s)
Initial Tilt Angle	θ	0.0 – $\pi/6$	

above). An initial tilt angle is needed to break the left-right symmetry of the problem in order to explore the possibility that the symmetric configuration of the parachute may be unstable, and that the parachute may oscillate from side to side. If the symmetric configuration is linearly unstable, then any nonzero initial tilt angle will lead to such oscillations. But if it is only nonlinearly unstable, then there may be a threshold value of the initial tilt angle below which the parachute settles into a steady symmetrical configuration and above which it settles into a sustained side-to-side oscillation. The porosity λ is introduced in order to study its influence on the stability of the steady, symmetrical parachute configuration. To avoid numerical difficulties at the ends of the parachute canopy, we make λ a function of s that is constant at its maximum value near the center of the parachute canopy and then tapers smoothly to zero at the ends of the canopy. When we report a numerical value of λ , that refers to the maximum value.

Table 1 shows other physical parameters as well as θ and λ . Our model of parachute is chosen to have a small dimension compared to real parachutes. The density is the same of air, but the dynamic viscosity is 0.002 which is 100 times bigger than that of air. The reason for the small size of parachute and the big viscosity

is to reduce the Reynolds number. We believe that, with computational resolution affordable so far, we can properly compute up to a Reynolds number of several hundreds. With those values, we have the Reynolds number as large as 400 based on the oncoming velocity and the diameter of the canopy. The initial canopy configuration is a half-ellipse with semi-axes a and b . In our numerical experiments, the data that we vary are the porosity λ , the initial tilt angle θ , and the initial opening length $2a$. The ranges of these parameters are given in Table 1.

5 Results and Discussion

We first verify that the computation of IB method for a porous boundary is robust and consistent. To do that, we choose a parachute with porosity coefficient $\lambda = 0.09 \text{ m}^2/(\text{N}\cdot\text{s})$, tilt angle $\theta = \pi/12$, and the initial opening length $2a = 0.48\text{m}$. The parachute is fixed at $(2.0\text{m}, 0.6\text{m})$ with two suspension lines. For $N=256, 512$, and 1024 , we change the timestep $\Delta t=0.0256/N$, the space meshwidth $h=4/N$ and the boundary meshwidth $\Delta s=(128L_0)/(70N)$, where L_0 is the canopy length. That is, when we refine the meshwidths h and Δs by a factor of 2, the timestep Δt is also reduced by the same factor.

The top of Figure 3 shows the x-coordinate of the midpoint of the parachute canopy about that of the fixed point as a function of time. This indicates how much a parachute deviates and oscillates from its symmetry. While the oscillations of parachute in three different meshwidths are very close to each other, the magnitude of the oscillations gets smaller in high resolution than in low resolution. This might be because the denominator $|\partial \mathbf{X}(s, t)/\partial s|^2$ of porosity λ in Eq (16) is usually smaller and then λ is bigger in high resolution than in low resolution. Though the difference of parachutes in three different meshwidths looks clear to exist, the convergence ratio is roughly equal to 2 in L_1 norm. Let (u_N, v_N) be the velocity field, and let $\|\cdot\|_1$ be L_1 norm. Then the bottom of Figure 3 shows the convergence ratios $\|u_{256} - u_{512}\|_1/\|u_{512} - u_{1024}\|_1$ and $\|v_{256} - v_{512}\|_1/\|v_{512} - v_{1024}\|_1$ as a function of time. The convergence ratio 2 implies that the scheme has first order accuracy which the general IB method satisfies. (We choose the middle meshgrid $N=512$ for the whole simulation in this paper).

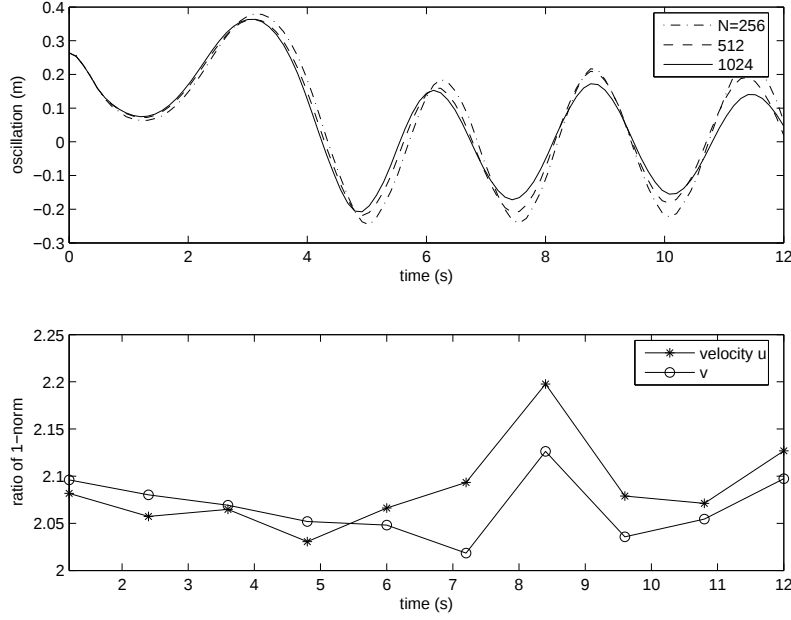


Figure 3: The top graph plots the difference between the x-coordinate of the midpoint of the canopy and the payload of the parachute (in meters) as a function of time (in seconds), which represents the side-by-side oscillations. With three different mesh-widths, the oscillations are similar. The bottom represents the convergence ratios of the velocity field (u, v) in 1-norm. The convergence ratio 2 implies that the scheme is first order accurate.

The first simulation that we consider involves the process of parachute inflation, starting from a nearly closed configuration and studying the changes in shape of the parachute at early times, see Figure 4. Of all the results that we consider, it is this one that shows most clearly the need for a method that can handle the unknown changes in shape of the parachute canopy. In the immersed boundary method, this is done without any re-gridding, since the canopy is represented in the fluid dynamics computation by a force field defined on a uniform grid. For alternative approaches, see [5, 25, 26].

Figure 4 shows the inflation of a parachute which has the payload tethered at $(2.0\text{m}, 0.6\text{m})$ and has no initial tilt angle, and for which the porosity coefficient $\lambda = 0 \text{ m}^2/(\text{N}\cdot\text{s})$ and the initial opening length $2a = 0.1 \text{ m}$. At the beginning of parachute inflation, the upper part of canopy is expanding more than the lower part. But

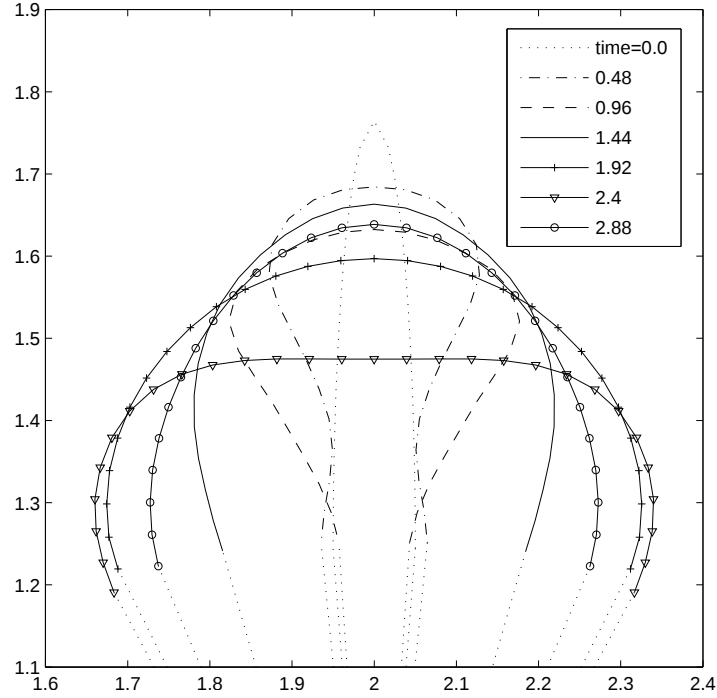


Figure 4: Inflation of parachute at early times. Note that inflation occurs first in the upper part of the canopy. In fact the suspension lines actually move slightly closer together at first but then move apart as inflation spreads to the entire canopy.

air captured by the canopy finally applies enough pressure to inflate the parachute completely. After time proceeds beyond the last time shown in Figure 4, the parachute moves stably without further change in its configuration.

An important theme of this paper is the relationship between porosity and stability. The effect of porosity on stability is investigated in Figures 5 and 6. Each panel of Figure 5 is a graph showing the oscillation of the parachute which is defined as the difference between the x-coordinates of midpoint of parachute canopy and fixed point as a function of time. In the left-hand column, the initial tilt angle is $\theta = \pi/60$, which we regard as a small perturbation from symmetry about a line $x=2$. In the right-hand column, a larger initial tilt angle ($\theta = \pi/6$) is used. In each row of Figure 5, the parachute canopy has a different porosity, beginning with $\lambda = 0.0$ (no porosity) at the top and increasing to 0.1 in the middle and 0.2 $\text{m}^2/(\text{N}\cdot\text{s})$ at the bottom.

The top row shows that the oscillating steady state (limit cycle) is unstable for a

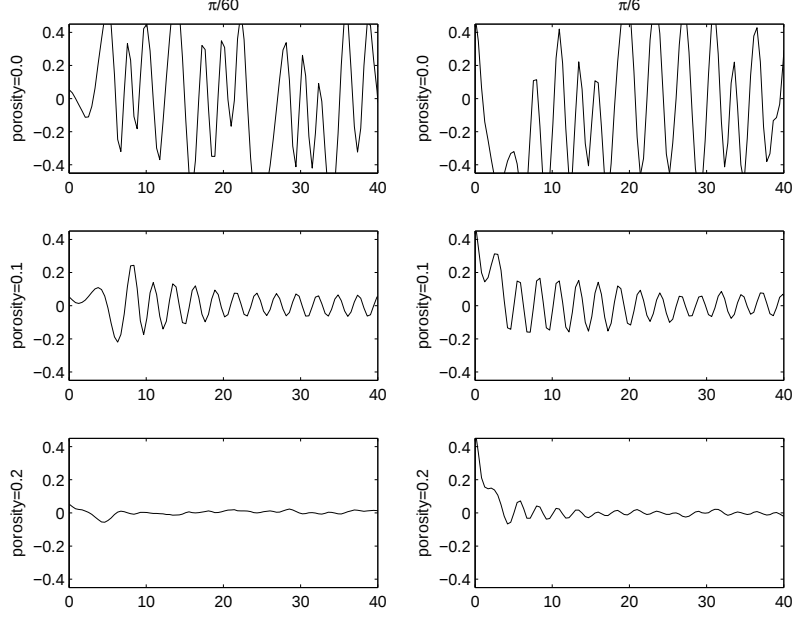


Figure 5: These graphs compare the stability of parachutes which have $\pi/60$ (left column) and $\pi/6$ (right column) as the initial tilt angle, and different coefficients λ of porosity equal to 0 (top row), $0.1 \text{ m}^2/(\text{N}\cdot\text{s})$ (middle row), and $0.2 \text{ m}^2/(\text{N}\cdot\text{s})$ (bottom row). Each graph plots the difference between the x-coordinates of the midpoint of the parachute canopy and the fixed payload (in meters) as a function of time (in seconds). Three qualitatively distinct behaviors are seen: unstable oscillations with large variable amplitude (top), oscillating steady state with damped amplitude (middle), or a rapid approach to the symmetrical steady state (upon which only small movement, possibly related to asymmetric vortex shedding, are superimposed).

parachute with no porosity, at least under the conditions of our simulation. For both initial conditions tried, the parachute does not settle into an oscillating steady state in which it rocks back and forth in an arbitrary way. The parachute of the middle row has the intermediate porosity $\lambda = 0.1 \text{ m}^2/(\text{N}\cdot\text{s})$. With this porosity, we can see the oscillation damped, and the parachute seems to be settling into the oscillating steady state (limit cycle). Finally, the bottom row suggests that the symmetrical steady state is globally stable for sufficiently high porosity, as almost no oscillations are seen in this case for either of the two initial conditions tried.

We have found that the transition point from the unstable motion to the oscil-

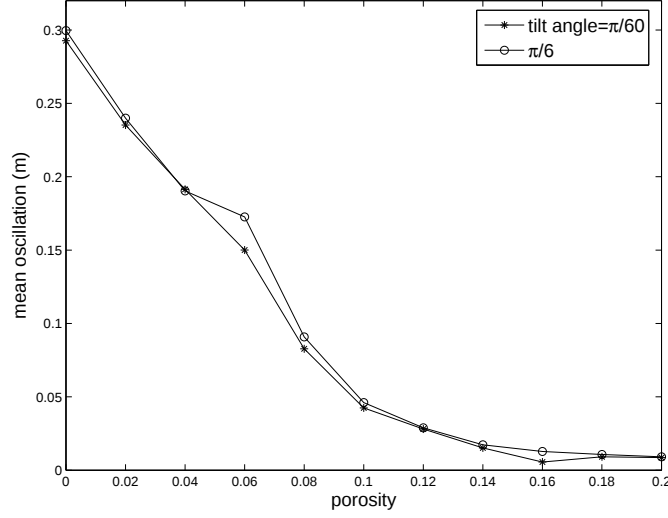


Figure 6: The mean oscillation in terms of porosity. The mean oscillation of parachute decreases almost linearly with the increasing porosity independent of the initial perturbation.

lating steady state is when the porosity is around $0.025 \text{ m}^2/(\text{N}\cdot\text{s})$. As the porosity increases beyond that point, the amplitude of the oscillation reduces almost linearly. The changing point of porosity from the oscillating steady state to the symmetrical steady state is about $0.15 \text{ m}^2/(\text{N}\cdot\text{s})$. The amplitude of parachute motion with the same porosity is independent of the initial perturbation. Only difference is that the parachute of a large initial tilt angle takes more time to settle down to its generic state than that of a small angle. We can see these facts from Figure 6 which shows the mean absolute oscillation for both cases. When we change the porosity (x-axis), the absolute value of oscillation is averaged from time 25 s to 42 s, and the averaged value is drawn. We choose 25 s as a transient to remove the initial perturbation effect. The mean oscillation of both cases are very close and both decreasing almost linearly with the increasing porosity. Note that the decrease of the mean oscillation stops above porosity $0.15 \text{ m}^2/(\text{N}\cdot\text{s})$ in which case the parachute has the symmetrical steady state.

The next result concerns the parachute with a moving payload instead of the fixed payload. As we discussed above, to prevent the parachute from leaving the computational domain, we use the oncoming velocity controlled by Eq (35). From

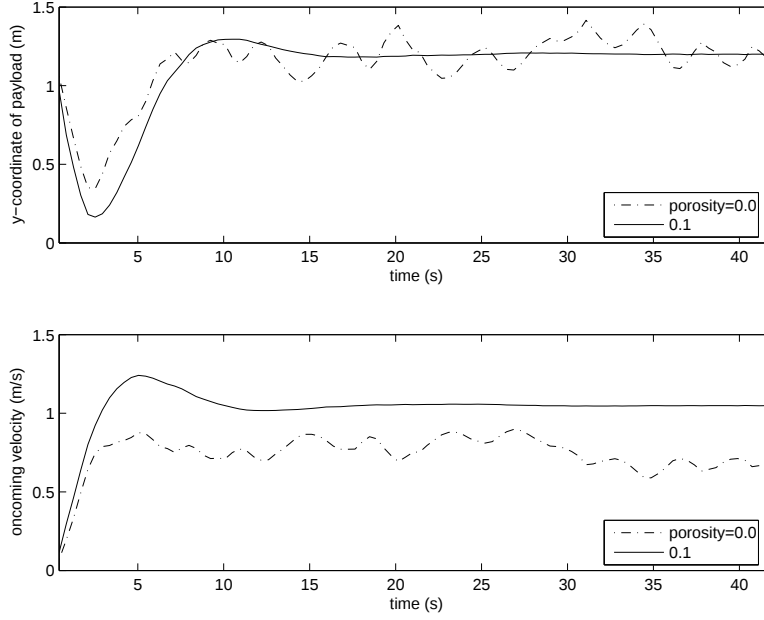


Figure 7: Behavior of the control mechanism that regulates the oncoming velocity in the free-payload case. Upper graph shows the y-coordinate of the payload $Y_p(t)$ as a function of time. $Y_p(t) \rightarrow 1.2 \text{ m} = y_{target}$ as $t \rightarrow \infty$ (with porosity) or $Y_p(t)$ oscillates around y_{target} (without porosity). Lower graph shows the oncoming velocity $V_0(t)$ as a function of time. $V_0(0) = 0 \text{ m/s}$, and $V_0(t) \rightarrow 1.05 \text{ m/s}$ as $t \rightarrow \infty$ for the parachute with porosity $\lambda = 0.1 \text{ m}^2/(\text{N}\cdot\text{s})$, but $V_0(t)$ changes around 0.76 m/s for the parachute with no porosity.

Eqs (34) and (35), since $\frac{dY_p(t)}{dt} = V_p(t)$ and $\frac{dV_0(t)}{dt} = k(y_{target} - Y_p(t)) - \sigma V_p(t)$, we get $V_p(t) = 0$ and $Y_p(t) = y_{target}$ as the equilibrium state for y-component of the velocity and vertical position of the moving payload. So we can predict analytically which values of k and σ induce the stable state. We choose $k = 0.25 \text{ s}^{-2}$ and $\sigma = 0.5 \text{ s}^{-1}$ through the analytical prediction and numerical experiments.

Figure 7 shows that, with these values of the parameters, the control mechanism of Eq (35) behaves in a stable manner. In the upper graph, as time goes on, while the y-coordinate of the payload $Y_p(t)$ is converging to the target position $y_{target} = 1.2 \text{ m}$ for the parachute with porosity coefficient $\lambda = 0.1 \text{ m}^2/(\text{N}\cdot\text{s})$, $Y_p(t)$ oscillates around the target position for the parachute without porosity. Similarly the oncoming velocity $V_0(t)$ stays near one value after a short time with porosity 0.1 (lower graph), but no

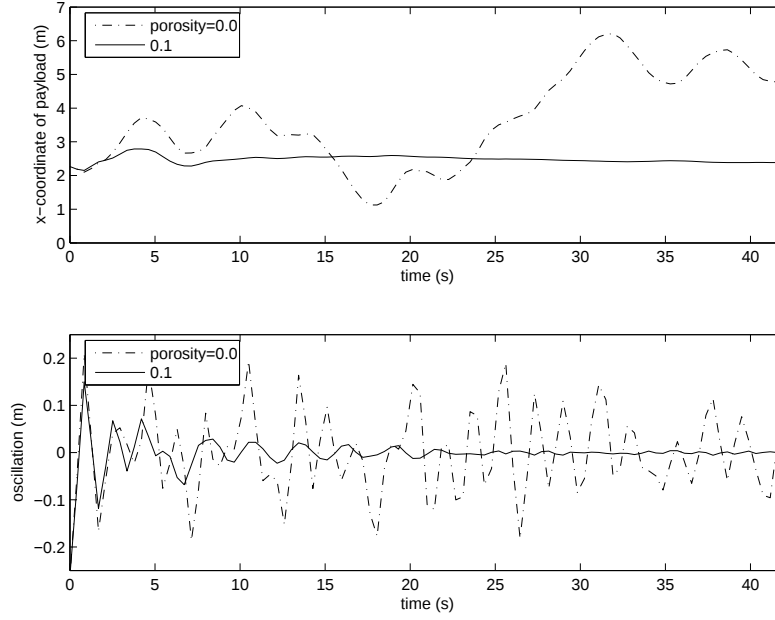


Figure 8: The graph compares the lateral stability of parachutes which have the same initial tilt angle $\pi/10$ but different coefficient λ of porosity. For each parachute, both the x-coordinate of the payload and the x-coordinate of the midpoint of the canopy are plotted

porosity case goes through an unstable oscillation around another value. These little oscillations around the expected equilibrium values for the parachute with no porosity are related to its unstable oscillatory motion, see below. Note from the lower graph that the parachute with porosity needs a larger oncoming velocity than the parachute without porosity to support the payload. Because porosity reduces the drag force of the fluid and causes the parachute with porosity to fall faster, a larger oncoming velocity is needed to generate a drag equal to the weight of the payload.

In this relatively stable situation, which we can regard as y-directional stability, we now investigate the relation between the porosity and the x-directional stability of the parachute. Figure 8 compares the x-directional stabilities of two parachutes which have the same initial tilt angle $\pi/12$, the initial opening length 0.48 m and two different porosities $\lambda = 0$ and $0.1 \text{ m}^2/(\text{N}\cdot\text{s})$. From the top panel which compares the x-coordinates of payload of the two parachute, we can see that, while the parachute with porosity settles at a point after a short time, the parachute without porosity

moves greatly going in and out the side of the periodic domain. In this case, the data outside the domain should be duplicated into the domain in the periodic way. The bottom panel shows the oscillation (difference between the x-coordinates of midpoint of canopy and the payload) of the two parachutes. The parachute with porosity $\lambda = 0.1 \text{ m}^2/(\text{N}\cdot\text{s})$ moves more stably than the parachute without porosity.

From Figures 5-8, we can see that porosity helps to stabilize the parachute against both side-to-side motion about the payload and x-directional motion as a whole (free payload only). As stated in the Introduction, asymmetry of vortex shedding is the probable cause of the parachute instability, but porosity can reduce this asymmetry of the vortex wake in the neighborhood of the canopy. To see that this idea is plausible, Figure 9 compares the vorticity contours of three cases of parachutes which have the same initial tilt angle $\theta = \pi/60$ but different porosity. The parachutes have fixed payloads. The parachutes in the first column have no porosity, those in the second column have porosity $\lambda = 0.1 \text{ m}^2/(\text{N}\cdot\text{s})$, and those in the third column has $\lambda = 0.2 \text{ m}^2/(\text{N}\cdot\text{s})$. Each row represents a certain fixed time. At time 0, all parachutes have the same configuration as our initial position in Figure 2 and there is neither wind velocity nor vorticity. Figure 9 shows that the porous parachute settles into the oscillating steady state (second column) or symmetrical steady state (third column), but the parachute without porosity continues to oscillate in an unstable way. We can also observe that the no-porosity case has very large and asymmetric vorticity. However, the parachute with porosity 0.1 has an oscillating vortex shedding, and the parachute with porosity 0.2 has a relatively symmetric vortex wake.

6 Summary and Conclusions

We have presented numerical experiments concerning the parachute problem in the two-dimensional case. Two basic configurations have been studied: one with a fixed payload in a prescribed updraft, and the other with a free payload in a controlled updraft, the controller being designed to adjust the updraft so that the parachute stays within the computational domain. The coupled equations of motion of the air and the flexible parachute canopy have been solved by the immersed boundary (IB) method. We have used this methodology to simulate the details of parachute inflation,

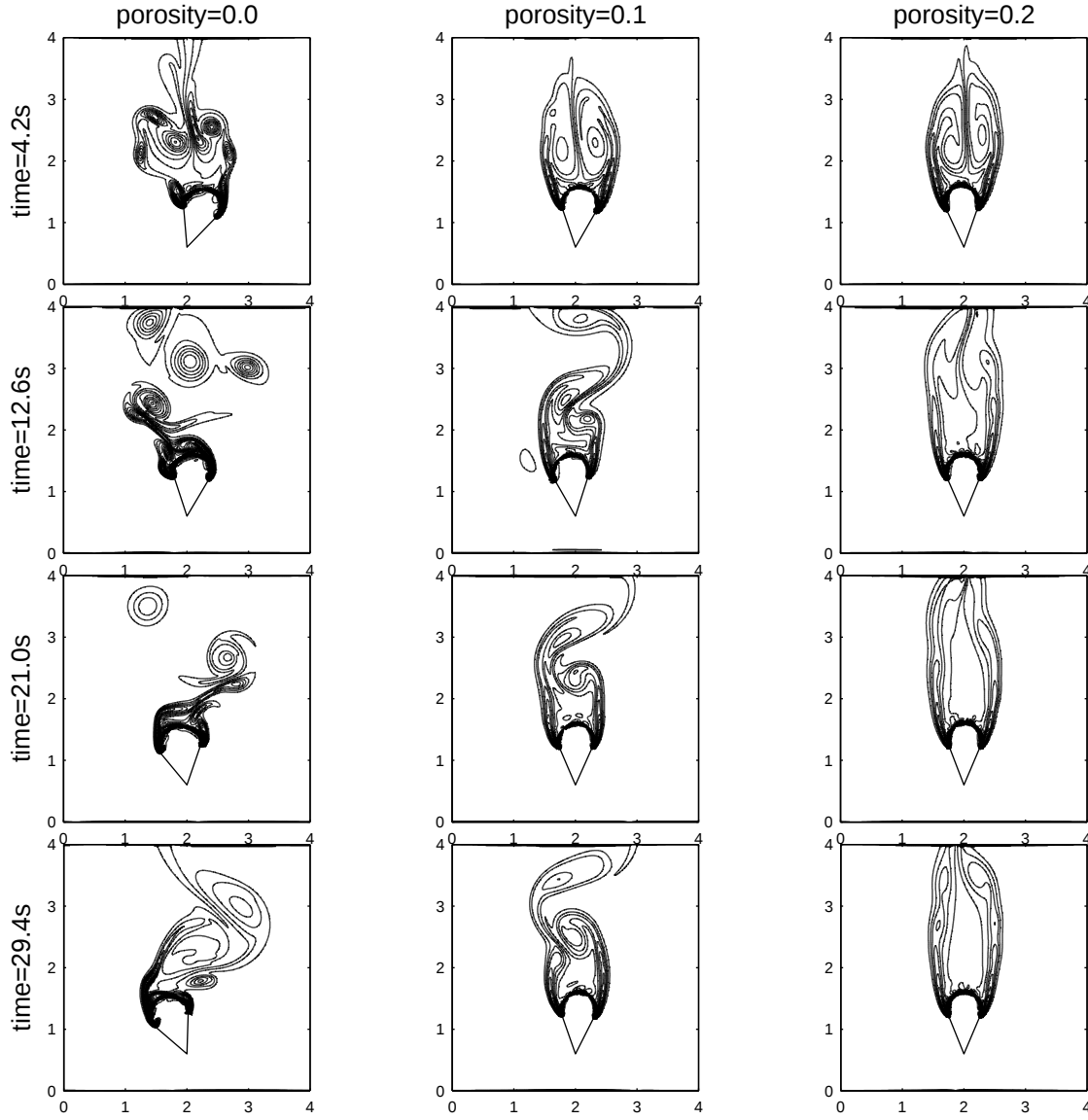


Figure 9: Vorticity contours in the wake of parachutes with fixed payloads. The parachute in the first column has zero porosity, the parachute in the second column porosity $\lambda=0.1$, and the parachute in the third column $\lambda=0.2 \text{ m}^2/(\text{N}\cdot\text{s})$. Each row shows a different time. A side-to-side oscillation and a large asymmetric vortex wake are observed in the case of the parachute with zero porosity. The parachute with porosity 0.1 has an oscillating vortex shedding and the parachute with porosity 0.2 has a relatively symmetric vortex wake.

and to study the influence of canopy porosity on the lateral stability of the parachute.

Future work will include the generalization to the three-dimensional case, and studies concerning the influence of wind shear on parachute dynamics.

Acknowledgement

This work was supported by the National Science Foundation under KDI research grant DMS-9980069.

References

- [1] K.M.Arthurs, L.C.Moore, C.S.Peskin, E.B.Pitman, and H.E.Layton. *Modeling arteriolar flow and mass transport using the immersed boundary method*. J. Comput. Phys. 147:402-440, 1998
- [2] P.W.Bearman and M.Takamoto. *Vortex shedding behind rings and disc*. Fluid Dynamics Research 3:214-218, 1988
- [3] R.P.Beyer. *A computational model of the cochlea using the immersed boundary method*. J. Comput. Phys. 98:145-162, 1992
- [4] D.J.Cockrell. *The Aerodynamics of Parachutes*. AGARDograph No. 295, July 1987.
- [5] I.V.Dneprov. *Computation of aero-elastic characteristics and stress-strained state of parachutes*. AIAA Pap. 93-1237, 1993
- [6] E.Givelberg. *Modeling elastic shells immersed in fluid*. Ph.D.Thesis, Mathematics, New York University, 1997.
- [7] A.L.Folgelson. *A mathematical model and numerical method for studying platelet adhesion and aggregation during blood clotting*. J. Comput. Phys. 56:111-134, 1984
- [8] A.L.Folgelson and C.S.Peskin. *A fast numerical method for solving the three-dimensional Stoke's equations in the presence of suspended particles*. J. Comput. Phys. 79:50-69, 1988

- [9] L.J.Fauci and C.S.Peskin. *A computational model of aquatic animal locomotion*. J. Comput. Phys. 77:85-108, 1988
- [10] H.Higuchi and F.Takahashi. *Flow past two-dimensional ribbon parachute models*. J. Aircr. 26(7):641-649, 1989
- [11] E.Jung and C.S.Peskin. *Two-dimensional simulations of valveless pumping using the immersed boundary method*. SIAM J.Sci.Comput. 23:19-45, 2001
- [12] T.W.Knacke. *Parachute Recovery Systems Design Manual*. NWC TP 6575, Naval Weapons Center, China Lake, California, June 1987
- [13] M.C.Lai and C.S.Peskin. *An Immersed Boundary Method with Formal Second-Order Accuracy and Reduced Numerical Viscosity*. J. Comput. Phys. 160:705-719, 2000
- [14] R.C.Maydew and C.W.Peterson. *Design and Testing of High-Performance Parachutes*. AGARD-AG-319. 1991
- [15] D.A.Nield and A.Bejan. *Convection in Porous Media*. Springer-Verlag 1991
- [16] C.W.Peterson, J.H.Strickland, and H.Higuchi. *The fluid dynamics of parachute inflation*. Annu.Rev.Fluid.Mech. 28:361-387. 1996
- [17] C.S.Peskin. *Flow patterns around heart valves:A numerical method*. J. Comput. Phys. 10:252-271,1972
- [18] C.S.Peskin. *Numerical analysis of blood flow in the heart*. J. Comput. Phys. 25:220-252, 1977
- [19] C.S.Peskin and D.M.McQueen. *Three dimensional computational method for flow in the heart :Immersed elastic fibers in a viscous incompressible fluid*. J. Comput. Phys. 81:372-405, 1989
- [20] C.S.Peskin and D.M.McQueen. *A general method for the computer simulation of biological systems interacting with fluids*. Symposia of the society for Experimental Biology. 49:265-276, 1995

- [21] C.S.Peskin and D.M.McQueen. *Fluid dynamics of the heart and its valves*. In: *Case studies in Mathematical Modeling: Ecology, Physiology, and Cell Biology*. Prentice Hall, Englewood Cliffs NJ, 1996, pp. 309-337
- [22] C.S.Peskin and D.M.McQueen. *Heart Simulation by an Immersed Boundary Method with Formal Second-order Accuracy and Reduced Numerical Viscosity*. In: *Mechanics for a New Millennium, Proceedings of the International Conference on Theoretical and Applied Mechanics (ICTAM) 2000*, (H.Aref and J.W.Phillips, eds.) Kluwer Academic Publishers, 2001
- [23] C.S.Peskin and B.F.Printz. *Improved volume conservation in the computation of flows with immersed elastic boundaries*. J.Comput. Phys. 105:33-46, 1993
- [24] P.G.Saffman *On the Boundary Condition at the Surface of a Porous Medium*. Studies in Applied Mathematics, V.50, 93-101, 1971
- [25] J.Sahu, G.Cooper, and R.Benney. *3-D parachute descent analysis using coupled CFD and structural codes*. AIAA Pap. 95-1580, 1995
- [26] K.R.Stein and R.J.Benney. *Parachute inflation: a problem in aero-elasticity*. US Army Tech.Rep. NATICK/TR-94/015, Natick, MA. 1994
- [27] L.Zhu and C.S.Peskin. *Simulation of a flapping flexible filament in a flowing soap film by the Immersed Boundary method*. J.Comput. Phys. 179:452-468, 2002