Computers & Fluids 39 (2010) 1562–1574

Contents lists available at ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

Toward optimally seeded airflow on hypersonic vehicles using control theory

Andre C. Marta *, Juan J. Alonso

Department of Aeronautics and Astronautics, Stanford University, CA 94305, USA

ARTICLE INFO

Article history:
Received 30 June 2009
Received in revised form 28 April 2010
Accepted 15 May 2010
Available online 17 June 2010

Keywords:
Magnetohydrodynamics
Hypersonics
Seeding
Control
Adjoint
Optimization

ABSTRACT

Following the renewed interest in hypersonic flight and the significant advances made recently, it is now the time to start looking at ways to optimize hypersonic vehicle designs in an efficient manner. Since the medium, in a hypersonic flow, can be locally ionized, it is possible to use electromagnetic actuators that induce an acting force to optimally control the flow. The local injection of substances that have a considerably lower ionization temperature than air into the airflow – flow seeding – leads to stronger local ionization levels at relatively low hypersonic speeds, amplifying the magnetic effects for the same imposed magnetic field intensity. Because much has been devoted to the analysis of such problems but no formal design approach as been persued to date, the main motivation for this work is to provide an efficient design framework built around high-speed magnetohydrodynamics (MHD) prediction capabilities that can be used in hypersonic control applications using magnetic effects. In particular, the design framework should provide information that leads to an optimal airflow seeding strategy in conjunction with an imposed magnetic field. The proposed framework is based on control theory, which implies developing an adjoint solver aimed to efficiently provide sensitivity analysis capability in arbitrary complex hypersonics MHD flows. Automatic differentiation tools are selectively used to develop the discrete adjoint, which make for a much shorter implementation time and greatly reduce the probability of programming errors. A generic hypersonic vehicle is used to demonstrate the sensitivity analysis capability of the implemented MHD adjoint solver. The precision of the computed adjoint-based sensitivities is established and the performance of the adjoint solver is analyzed. A sample design problem is included using a gradient-based optimizer.

1. Introduction

1.1. Hypersonic flow

During the last few years, there has been renewed interest in hypersonic flight. In July 2002, the world's first experimental flight of an air-breathing supersonic ramjet (scramjet) engine took place in Australia, as a result of the *HyShot* research project led by the University of Queensland. In March 2004, significant advances were made with the successful flight test of the NASA X-43, a hypersonic scramjet-powered research aircraft. The following year, NASA released an integrated hypersonic technology demonstration roadmap, which triggered a feasibility study on the X-43D, with the objective of developing a baseline conceptual design, assessing its performance, and identifying the key technical issues [12]. The descendant of this effort is the current hypersonics program, the X-51, being the demonstrator scheduled to fly by 2010. Simultaneously, a number of other efforts are being pursued at NASA and elsewhere.

The hypersonic regime still poses challenging problems in fluid mechanics, propulsion, materials and structures, and stability and control. There are still many technical and scientific obstacles to overcome but the community is getting closer and closer to the stage where sustained hypersonic flight may be possible. With that in mind, it is now the time to start looking at ways to optimize such designs in an efficient manner.

1.2. Magnetohydrodynamics flow control

Among all the phenomena in fluid mechanics, the one that triggered this work was that of magnetohydrodynamics (MHD), in particular that of flow seeding.

Since the medium, in a hypersonic flow, can be locally ionized, the possibility opens up to use electromagnetic actuators to optimally control the flow. These actuators can generate magnetic fields that induce an acting force on the flow, as given by the Lorentz force,

$$\mathbf{F}_{Lorentz} = q(\mathbf{E} + \mathbf{u} \times \mathbf{B}), \tag{1}$$

where $\mathbf{F}_{Lorentz}$ is the force exerted on a fluid particle moving with velocity $\mathbf{u}$, whose electric charge is q, subject to an electric field $\mathbf{E}$ and a magnetic field $\mathbf{B}$.

* Corresponding author. Tel.: +351 218419466.
E-mail address: andre.marta@ist.utl.pt (A.C. Marta).

0045-7930/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2010.05.009

Several studies have already been made to evaluate the potential use of this MHD phenomenon in hypersonic flow control applications. As early as 1967, experimental observations were made by Nowak et al. [19], revealing that the bow shock stand-off distance and drag increased, in re-entry vehicles, with the application of moderately strong magnetic fields. Four years later, one of the first numerical analysis of the MHD blunt body problem was performed by Coakley and Porter [4]. This problem has been revisited recently but some of the results are contradictory with respect to the possible to drag reduction, as seen on the work of Agarwal and Augustinus [1]. Since then, many other applications have been developed. The potential heat transfer mitigation through magnetic control in a hypersonic blunt body flow has also been analyzed by Poggie and Gaitonde [21]. Even the replacement of movable control surfaces by the use of glow discharges in plasma flows was evaluated in the work of Shang et al. [24].

Despite reasonably matured MHD analysis tools, all of these experiments have been based on either trial-and-error or parametric studies [13], no formal design approach has been employed yet and very little effort has been devoted to automate these flow control processes. However, any of these applications can be viewed as an optimization problem in which a cost function must be minimized by varying a set of control variables, while satisfying a specific set of constraints.

1.3. Flow seeding

In order for the magnetic effects to be significant, it is often necessary to produce relatively high magnetic fluxes, which imply using heavy electromagnets, because of the low ionization level of air (which translates into low electrical conductivity). This problem can be mitigated by seeding the airflow, which consists in the addition of substances that have a considerably lower ionization temperature than air. The most commonly used seed used is Potassium. Studies have shown that seeded air leads to stronger local ionization at relatively low hypersonic speeds, amplifying the magnetic effects, thus decreasing the required applied magnetic field intensity [8].

As such, the main motivation for this work is to provide an efficient design framework built around high-speed MHD prediction capabilities that can be used in hypersonic control applications using magnetic effects. In particular, the design framework should provide information that leads to an optimal airflow seeding strategy.

2. Control theory

2.1. Generic design problem

In the context of optimization, a generic design problem can be posed as the minimization of a cost function, I (also called function of interest or figure of merit) with respect to a vector of design variables, $\boldsymbol{x}$, while satisfying a set of linear or nonlinear constraints. The cost function depends directly on the design variables and on the state of the system, $\boldsymbol{w}$, that may result from the solution of the governing equations of the problem. Thus we can write the vector-valued function I as

$$I = I(\boldsymbol{x}, \boldsymbol{w}(\boldsymbol{x})). \tag{2}$$

For a given input vector $\boldsymbol{x}$, the solution of the governing equations subject to appropriate boundary conditions yields a state vector, $\boldsymbol{w}$, thus establishing the dependence of the state of the system on the design variables. We denote these governing equations by

$$\mathscr{R}(\boldsymbol{x}, \boldsymbol{w}(\boldsymbol{x})) = 0. \tag{3}$$

In mathematical terms, this design problem can be expressed as

$$\begin{aligned} &\text{Minimize } I(\boldsymbol{x}, \boldsymbol{w}(\boldsymbol{x})) \\ &\text{w.r.t.} \boldsymbol{x}, \\ &\text{subject to } \quad \mathscr{R}(\boldsymbol{x}, \boldsymbol{w}(\boldsymbol{x})) = 0 \\ &\qquad\qquad C_i(\boldsymbol{x}, \boldsymbol{w}(\boldsymbol{x})) = 0, \quad i = 1, \ldots, m, \end{aligned} \tag{4}$$

where $C_i(\boldsymbol{x}, \boldsymbol{w}(\boldsymbol{x})) = 0$ represents m additional constraints that may or may not involve the flow solution.

When using a gradient-based optimizer to solve the design problem (4), the gradient (also designated as sensitivity or first derivative) of both the cost function and the constraints with respect to the design variables are required. That is, $\frac{dI}{d\boldsymbol{x}}$ and $\frac{dC_i}{d\boldsymbol{x}}$ have to be determined. The control theory approach, also called the adjoint method, emerges as an excellent candidate in achieving that goal.

2.2. Adjoint method

Adjoint methods have been used to perform sensitivity analysis of partial differential equations (PDEs) for over three decades. These methods were first applied to optimal control problems and thereafter used to perform sensitivity analysis of linear structural finite element models. The first application to fluid dynamics was due to Pironneau [20]. The method was then used to perform airfoil shape optimization by Jameson [10] and since then it has been extended to three-dimensional problems, leading to applications such as aerodynamic shape optimization of wings [11] and complete aircraft configurations [22]. The adjoint theory has since been generalized for multi-disciplinary systems [17] and for MHD problems [15].

The adjoint method is extremely valuable because it provides a very efficient and accurate method to compute the sensitivity of a given function of interest with respect to many parameters by solving a system of equations of size equivalent to the governing equations of the flow one time only, independently of the number of design variables. These properties make a significant positive impact on the overall performance of the optimization, when compared to other methods to estimate sensitivities such as finite-differences.

The adjoint method has already been mathematically well documented [6]. It provides a way for computing the function sensitivity by formulating the *dual problem*, given by

$$\frac{dI}{d\boldsymbol{x}} = \frac{\partial I}{\partial \boldsymbol{x}} - \psi^T \frac{\partial \mathscr{R}}{\partial \boldsymbol{x}}, \tag{5}$$

$$\text{such that } \left[\frac{\partial \mathscr{R}}{\partial \boldsymbol{w}}\right]^T \psi = \left[\frac{\partial I}{\partial \boldsymbol{w}}\right]^T, \tag{6}$$

which result from the definition of the adjoint vector ψ as

$$\psi^T = \frac{\partial I}{\partial \boldsymbol{w}} \left[\frac{\partial \mathscr{R}}{\partial \boldsymbol{w}}\right]^{-1}. \tag{7}$$

The adjoint vector obtained by solving the adjoint Eq. (6) is substituted into Eq. (5) to find the function sensitivity. In contrast with other sensitivity analysis methods, the adjoint vector does not depend on the design variables, $\boldsymbol{x}$, but instead depends on the function of interest, I.

The sizes of the sensitivity matrices involved in the adjoint method are

$$\frac{\partial I}{\partial \boldsymbol{x}} \quad (N_I \times N_x), \qquad \frac{\partial \mathscr{R}}{\partial \boldsymbol{x}} \quad (N_w \times N_x), \tag{8}$$

$$\frac{\partial \mathscr{R}}{\partial \boldsymbol{w}} \quad (N_w \times N_w), \qquad \frac{\partial I}{\partial \boldsymbol{w}} \quad (N_I \times N_w), \tag{9}$$

where N_I is the number of functions of interest, N_x the number of design variables and N_w the size of the state vector, which for the solution of a large, three-dimensional problem involving a system of conservation laws, can be very large.

2.3. Hybrid approach: merging the adjoint and AD methods

Given the value of adjoint methods, it seems odd that their application to aerodynamic shape optimization is not more ubiquitous. In fact, while adjoint methods have already found their way into commercial structural analysis packages, they have yet to proceed beyond research CFD solvers. The main obstacle to the use of adjoint methods is the complexity involved in the development and implementation of the additional solver required.

This work employs a discrete adjoint formulation, that emerges as the best suitable option to deal with the complex equations, such those that govern MHD, and with the nature of the functions of interest that may be used in relevant design problems.

To enable the development of a discrete adjoint, and taking advantage of using a discrete adjoint formulation, the *Automatic Differentiation adjoint* (*ADjoint*) approach is used [16].

In this approach, automatic differentiation tools are selectively applied to the MHD solver to produce only the code that computes the individual entries in the flux Jacobian matrix and the other partial derivatives (8) and (9) that are necessary to compute sensitivities using an adjoint method.

The use of *ADjoint* method in the context of MHD brings several advantages, namely, it is generic (applicable to any PDE solver and governing equations), largely automatic, and exactly consistent with the flow solver. Because the process of automatic differentiation allows us to treat arbitrary expressions exactly, the sensitivities produced are perfectly consistent with those that would be obtained from an exact numerical differentiation of the original solver.

Details about the implementation of this approach are included in Section 5.

3. Governing equations

The hypersonic flow simulations in this work are restricted to the stratosphere so that the Knudsen number remains everywhere small and the *continuous mechanics* assumption holds.

The equations governing the three-dimensional flow of a compressible, conducting fluid with an externally imposed magnetic field are obtained by coupling the Navier–Stokes equations to the Maxwell equations. This coupling is strong, in the sense that the velocity field given by the former equations shows in the convective terms of the latter, and the electromagnetic field modeled by the latter appears as additional force and energy terms in the former equations. The resulting set of equations is designated as the full magnetohydrodynamics (MHD) equations [5].

In order to cope with the extreme complexity of the full MHD analysis, several simplified models have been developed. In this paper, the fluid flow under the influence of magnetic fields has been modeled with both the ideal and low Re_σ approximation MHD equations. The gas is assumed to be in thermodynamic equilibrium and to be calorically perfect. In addition, finite-rate chemistry models are not included, meaning that the flow is assumed to be frozen.

3.1. Ideal MHD equations

The equations governing the three-dimensional flow of an inviscid, compressible, perfectly conducting fluid in a magnetic field are called the ideal MHD equations. Their derivation follows that of the full MHD equations, obtained by coupling the Navier–Stokes equations to the Maxwell equations, but additional assumptions are made. In particular, that the flow might be considered inviscid, thus the viscous terms of the Navier–Stokes equations vanish, and that the fluid is perfectly conducting, implying that the dispersive terms of the induction equations can be neglected.

In the literature, several versions of the ideal MHD equations can be found, all resulting from the inclusion of additional magnetic terms in the Navier–Stokes equations – the momentum equation gets an additional force per unit volume of matter (Lorentz force), and the additional power delivered to matter is included in the energy equation – and the additional magnetic field transport equation, that is derived from the Faraday's law.

In the present work, however, a different formulation, recently derived by Marta [14], is used. It combines two features: the inclusion of additional source terms that enforce the solenoidal condition of the magnetic field, $\nabla \cdot \mathbf{B} = 0$, and the decomposition of the total magnetic field into the background imposed field and the induced field to achieve better numerical accuracy.

3.1.1. Magnetic field decomposition

Besides the issue of satisfying the solenoidal condition, there is also the problem of having large imposed magnetic fields. Under these circumstances, the ratio of induced to imposed components of the magnetic field becomes extremely small and the magnetic terms can dominate the system. Small errors in the magnetic field solution can cause severe difficulties in the energy equation, because the magnetic energy becomes much greater than the kinetic energy.

Following the work of Tanaka [25], this problem can be mitigated by decomposing the magnetic field into two components, the background imposed field, $\mathbf{B}_0$, and the induced field, $\mathbf{B}_i$,

$$\mathbf{B} = \mathbf{B}_0 + \mathbf{B}_i, \tag{10}$$

3.1.2. Flux vector form

Consequently, the conservative system of ideal MHD equations used in this work was custom derived in order to obtain a stable and accurate system, and its derivation is detailed in Marta [14]. The non-dimensional ideal MHD equations are then given by

$$\frac{\partial \boldsymbol{w}_i}{\partial t} + (\nabla \cdot \mathscr{F}_i) + (\nabla \cdot \mathscr{F}_m) = \mathbf{S}_i, \tag{11}$$

where

$$\boldsymbol{w}_i = \begin{pmatrix} \rho \\ \rho\mathbf{u} \\ \rho Z_i \\ \mathbf{B}_i \end{pmatrix}, \quad \mathscr{F}_i = \begin{pmatrix} \rho\mathbf{u} \\ \rho\mathbf{u}\mathbf{u} + p\mathbf{I} \\ (\rho E + p)\mathbf{u} \\ 0 \end{pmatrix},$$

$$\mathscr{F}_m = \begin{pmatrix} 0 \\ R_b(\mathbf{B}_i \cdot \mathbf{B}_i)\mathbf{I}/(2\mu_m) - R_b(\mathbf{B}_i\mathbf{B}_i)/\mu_m \\ +R_b(\mathbf{B}_0 \cdot \mathbf{B}_i)\mathbf{I}/\mu_m - R_b(\mathbf{B}_0\mathbf{B}_i + \mathbf{B}_i\mathbf{B}_0)/\mu_m \\ R_b(\mathbf{B}_i \cdot \mathbf{B}_i)\mathbf{u}/\mu_m - R_b(\mathbf{u} \cdot \mathbf{B}_i)\mathbf{B}_i/\mu_m \\ +R_b(\mathbf{B}_0 \cdot \mathbf{B}_i)\mathbf{u}/\mu_m - R_b(\mathbf{u} \cdot \mathbf{B}_i)\mathbf{B}_0/\mu_m \\ (\mathbf{u}\mathbf{B}_i - \mathbf{B}_i\mathbf{u}) + (\mathbf{u}\mathbf{B}_0 - \mathbf{B}_0\mathbf{u}) \end{pmatrix}$$

and

$$\mathbf{S}_i = -(\nabla \cdot \mathbf{B}_i) \begin{pmatrix} 0 \\ R_b\mathbf{B}/\mu_m \\ R_b(\mathbf{u} \cdot \mathbf{B}_i)/\mu_m \\ \mathbf{u} \end{pmatrix}.$$

The inviscid and magnetic flux vectors are $\mathscr{F}_i$ and $\mathscr{F}_m$, respectively.

The conservative variables composing $\boldsymbol{w}$ are the density, ρ, the momentum density, $\rho\mathbf{u}$, the total energy density, ρZ_i, and the induced magnetic field, $\mathbf{B}_i$. The MHD total energy per unit volume is composed of the usual total energy augmented by the induced magnetic energy contribution, as

$$\rho Z_i = \rho E + R_b \frac{\mathbf{B_i}\cdot\mathbf{B_i}}{2\mu_m}, \tag{12}$$

where R_b is the magnetic force number (or pressure number) $R_b = \frac{B^2}{\rho U^2 \mu_m}$ and μ_m is the magnetic permeability.

This ideal MHD model allows for environments characterized by a high magnetic force number, where the magnetic field induced by the current is of comparable magnitude to the one imposed on the flow, since the three induction equations are solved in the governing equations, as opposed to the low magnetic Reynolds number model.

3.2. Low magnetic Reynolds number MHD model

In external aerodynamics, many of the encountered hypersonic flow fields are characterized by relatively low levels of electrical conductivity, σ. The pertinent non-dimensional parameter determining the relative magnitude of σ is the magnetic Reynolds number, $Re_\sigma = LU\mu_m\sigma$.

If the environment of interest is characterized by a low magnetic Reynolds number, then the magnetic field induced by the current is much smaller than that imposed on the flow and, therefore, it can be neglected. By neglecting the distortion of the magnetic field by the flow and only assuming that the imposed field has a significant influence on the flow, there is no need to solve the three induction equations since the magnetic field is assumed to be decoupled from the velocity field. Under this conditions, the electromagnetic forces and energy show up as source terms in the Navier–Stokes equations.

3.2.1. Flux vector form

Adopting the low Re_σ model approximation, the set of non-dimensional governing equations may be written in compact form as

$$\frac{\partial \boldsymbol{w}}{\partial t} + (\nabla\cdot\mathscr{F}_i) = \mathbf{S}, \tag{13}$$

where the vector of conservative variables and the inviscid flux vector are

$$\boldsymbol{w} = \begin{pmatrix}\rho\\ \rho\mathbf{u}\\ \rho E\end{pmatrix}, \quad \mathscr{F}_i = \begin{pmatrix}\rho\mathbf{u}\\ \rho\mathbf{u}\mathbf{u} + p\mathbf{I}\\ (\rho E + p)\mathbf{u}\end{pmatrix},$$

respectively, and $\mathbf{S}$ is the source term includes the magnetic field effects,

$$\mathbf{S} = R_b\begin{pmatrix}0\\ \mathbf{j}\times\mathbf{B}\\ \mathbf{j}\cdot\mathbf{E}\end{pmatrix} = Q\begin{pmatrix}0\\ \sigma(\mathbf{E}+\mathbf{u}\times\mathbf{B})\times\mathbf{B}\\ \sigma(\mathbf{E}+\mathbf{u}\times\mathbf{B})\cdot\mathbf{E}\end{pmatrix}.$$

The electric current $\mathbf{j}$ was eliminated using the generalized Ohm's law in non-dimensional form, $\mathbf{j} = Re_\sigma\sigma(\mathbf{E}+\mathbf{u}\times\mathbf{B})$. The resulting non-dimensional scalar defined as $Q = R_b Re_\sigma$ is the magnetic interaction parameter. In the present simulations, there is no imposed electrostatic field $\mathbf{E}$.

3.3. Imposed magnetic field

In all MHD simulations made in this work, the magnetic field is externally imposed by considering the existence of a set of electromagnetic circuits that create a dipole-like magnetic field on the flow.

The magnetic field generated by a single dipole is given by the vector function

$$\mathbf{B} = \frac{\mu_m m}{4\pi r^3}(2\cos\theta\hat{e}_r + \sin\theta\hat{e}_\theta), \tag{14}$$

where $\mathbf{B}$ is the imposed magnetic field at a point located at a distance r from the origin of the dipole, of strength m, at an angle θ with respect to the dipole direction.

4. Flow solver

The flow solver used in this work was the *Navier–Stokes Stanford University Solver* (*NSSUS*) flow solver. This solver is a finite-difference, higher-order solver that has been developed at Stanford University under the Advanced Simulation and Computing (ASC) program sponsored by the Department of Energy.

This generic node-centered, multi-block, multi-processor solver uses finite-difference operators and artificial dissipation terms follow the work of Mattsson and Nordström [18] and the boundary conditions are implemented by means of penalty terms, according to the work of Carpenter et al. [3]. The additional magnetic induction equations and source terms were included in this solver so that MHD computations could be performed.

The set of coupled ODEs that result from the spatial discretization of (11) or (13) can be re-written in semi-discrete form as

$$\frac{d\boldsymbol{w}_{ijk}}{dt} + \mathscr{R}_{ijk}(\boldsymbol{w}) = 0, \tag{15}$$

where $\boldsymbol{w}$ is the vector of the flow variables at the cell vertex. The vector of residuals, $\mathscr{R}(\boldsymbol{w})$, consists of all the physical and artificial fluxes, boundary conditions and source terms.

An explicit five-stage, modified Runge–Kutta scheme was used to integrate Eq. (15) in time and drive the solution to steady-state. An adaptive local time step was used to increase the convergence rate of the algorithm, being the time step computed for each cell according to

$$\Delta t = \frac{CFL}{\lambda_\xi + \lambda_\eta + \lambda_\zeta}, \tag{16}$$

where CFL is the Courant–Friedrichs–Lewy number and λ_ξ, λ_η, λ_ζ are the maximum speed of propagation of information in each of the three computational directions, that corresponds to the maximum eigenvalue of the hyperbolic system of governing equations, also designated by local spectral radii.

For the system of Eq. (11), the spectral radius corresponds to the fast magneto-acoustic wave,

$$\lambda_{max} = |U_n| + c_f, \tag{17}$$

where U_n is the normal fluid velocity and c_f is the speed of the fast-mode MHD wave, relative to the fluid which is given by

$$c_f = \sqrt{\frac{1}{2}\left[c^2 + \frac{B^2}{\rho\mu_m} + \sqrt{\left(c^2 + \frac{B^2}{\rho\mu_m}\right)^2 - 4\frac{c^2 B_n^2}{\rho\mu_m}}\right]}. \tag{18}$$

The speed of sound, assuming a perfect gas, is defined as $c = \sqrt{\frac{\gamma p}{\rho}}$.

5. Adjoint solver

Automatic differentiation tools are used to generate code that computes the several matrices of partial sensitivities (8) and (9), according to the hybrid *ADjoint* approach introduced in Section 2.3.

5.1. Assembly of the adjoint matrix

The discrete adjoint matrix (or flux Jacobian), $\frac{\partial \mathscr{R}}{\partial \boldsymbol{w}}$, is independent of the choice of the function of interest or the design variables – it is simply a function of the governing equations, their discretization and the problem boundary conditions.

To compute it, only the routines in the flow solver that evaluate the residuals $\mathscr{R}_{ijk}$ need to be considered. To make the implementation more efficient, it was necessary to re-write the flow residual routine such that it computed the residual, including the proper boundary condition handling, for a single specified node `(i,j,k)` based on the computational stencil:

```
subroutineresidualAdj(i, j, k, xAdj, wAdj, rAdj)
```

This routine returns the N_v residuals `rAdj`, given the stencil of grid coordinates, `xAdj`, and flow variables, `wAdj`, for each node in each block on each processor. For the ideal MHD equations, $N_v = 8$, while for the low Re_σ model, $N_v = 5$. There are a total of $(N_v \times N_s)$ flow variables in the stencil, where N_s is the dimension of the computational stencil used in the flow solver.

The adjoint matrix can then be assembled by taking the derivative of the residual $\mathscr{R}$ with respect to the flow variables $\boldsymbol{w}$. The automatic differentiation tool used was Tapenade [9], that supports Fortran 95, the programming language in which the MHD solver was coded. The automatic differentiation process produced the differentiated routine

```
subroutineresidualAdj_B(i, j, k, xAdj, wAdj,
wAdjB, rAdj, rAdjB)
```

which is able to compute all the necessary derivatives for the flux Jacobian matrix.

The $N_v \times (N_v \times N_s)$ sensitivities that need to be computed for each node, corresponding to N_v rows in the $\frac{\partial \mathscr{R}}{\partial \mathrm{w}}$ matrix, are readily computed by this automatically differentiated routine in reverse mode because it yields

$$\mathtt{wAdjB(ii,jj,kk,n)} = \frac{\partial \mathscr{R}(i,j,k,m)}{\partial \mathrm{w}(i+ii, j+jj, k+kk, n)}, \tag{19}$$

where the triad `(ii,jj,kk)` spans the stencil, `m` spans the number of governing equations N_v and `n` spans the N_v flow variables. The number of non-zero blocks $(m \times n)$ matches the dimension of the stencil.

5.2. Assembly of the adjoint RHS vector

The right-hand side (RHS) vector of the discrete adjoint system, $\frac{\partial I}{\partial \mathrm{w}}$, is constructed by differentiating the discretized version of the cost function with respect to the flow variables $\boldsymbol{w}$. An approach identical to the one described in Section 5.1 was used.

5.3. Solution of the adjoint system

The adjoint solver requires the solution of a system of $N_w = (N_c \times N_v)$ Eq. (6). To solve this large sparse linear system, the Portable, Extensible Toolkit for Scientific Computation (PETSc) [2] was used.

Once the sparse data structures were filled, the adjoint system of equations was solved using a PETSc built-in Krylov subspace (KSP) method. More specifically, a Generalized Minimum Residual (GMRES) algorithm [23] was used, preconditioned with the block Jacobi method, with one block per processor, each solved with ILU(0) preconditioning.

5.4. Total sensitivity

Once the adjoint solution, ψ, is found, the gradient of the cost function is easily obtained from the sensitivity Eq. (5). This expression for the adjoint-based sensitivity requires the differentiation of the flow solver residual $\mathscr{R}$ and cost function I evaluation routines with respect to the design variables $\boldsymbol{x}$. The resulting matrix $\frac{\partial \mathscr{R}}{\partial \mathrm{x}}$ has dimensions $(N_c \times N_v) \times N_x$, where N_x is the number of design variables, whereas the right-hand side vector $\frac{\partial I}{\partial \mathrm{x}}$ has length N_x.

Using AD tools, the re-engineered routines used for $\frac{\partial \mathscr{R}}{\partial \mathrm{w}}$ and $\frac{\partial I}{\partial \mathrm{w}}$ earlier were again differentiated automatically, this time with respect to $\boldsymbol{x}$, thus providing $\frac{\partial \mathscr{R}}{\partial \mathrm{x}}$ and $\frac{\partial I}{\partial \mathrm{x}}$, respectively.

5.5. Adjoint-based optimization

The optimization problem (4) is solved by feeding the cost and constraint function values, obtained by the flow solver, and their gradients, obtained by the adjoint solver, into a gradient-based optimizer, following the algorithm depicted in the block diagram shown in Fig. 1. These values are used by the optimizer to find the search direction and to determine the step size during the line search.

The gradient-based optimizer used in this work is *SNOPT* [7], which is a software package for solving large-scale optimization problems (linear and nonlinear programs), developed in the Systems Optimization Laboratory at Stanford University. Among the several methods available, the Sequential Quadratic Programing (SQP) method was selected.

6. Results and discussion

6.1. Problem setup

The test case used to demonstrate the hybrid *ADjoint* sensitivity analysis method was a generic hypersonic vehicle inspired by the NASA X-43A experimental aircraft, and it is shown in Fig. 2. This aircraft geometry includes most of the principal airframe components, namely the vertical fins and the scramjet duct. Its total length is 20.25 m and the origin of coordinates is located at the aircraft nose.

Since the simulation was run without any side-slip angle, only half of the body had to be modeled, with a symmetry boundary condition imposed on the center plane. The body wall was set to be an impermeable Euler wall, while the outer boundaries have non-reflecting boundary conditions imposed on them. The free-stream flow conditions chosen were Mach 5 and an angle of attack of 5°.

To simulate the magnetohydrodynamics interaction, a collection of seven hypothetical dipoles was placed inside the body, at the locations indicated in Table 1, which imposed a magnetic field on the flow given by expression (14). The resulting field given by

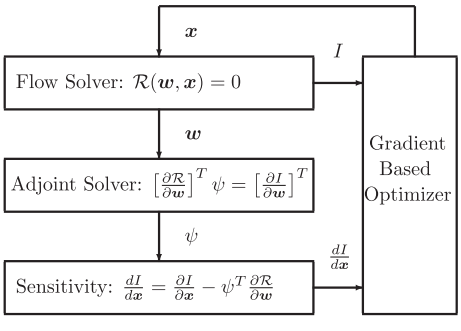
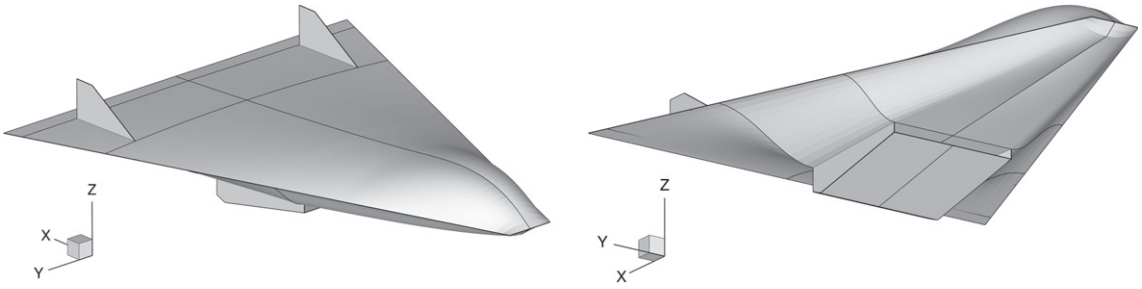


Fig. 1. Schematic of the adjoint-based optimization algorithm.



(a) Top view (b) Bottom view

Fig. 2. Generic hypersonic vehicle configuration.

Table 1
Location of dipoles.

Dipole #	Location (m)	Orientation
1	(0.7,0,0)	(−1,0,0)
2/3	(2,±1,0)	(−0.3473,±0.9378,0)
4/5	(4,±1.5,0)	(−0.3473,±0.9378,0)
6/7	(6,±2,0)	(−0.3473,±0.9378,0)

the superposition of dipoles is shown in Fig. 3. Despite the fact that only half of the domain was modeled, all dipoles were taken into account when calculating the imposed magnetic field. All dipoles were set to the same strength, and the baseline electrical conductivity was such that a magnetic Reynolds number of $Re_\sigma = 0.19$ and a magnetic pressure number of $R_b = 0.11$ were used, yielding a magnetic interaction parameter of $Q = 0.02$.

The computational mesh consisted of 15 blocks. Two versions, a coarser mesh with a total of 290,107 nodes, and a finer one with 550,109 nodes, have been used for the ideal MHD governing equations and the low Re_σ, respectively. For visualization purposes, the smaller mesh is shown in Fig. 4 with three levels of coarsening applied.

The aerodynamic coefficients were used as cost functions and the electrical conductivity, σ, in every computational node was taken as the design variables when running the low Re_σ MHD model. As seen in the results that follow, this led to a total of 550,109 design variables for the computational mesh used.

Additional design variables were tested as well, specifically, the vehicle attitude, defined by angle of attack, α, and side-slip angle, β,

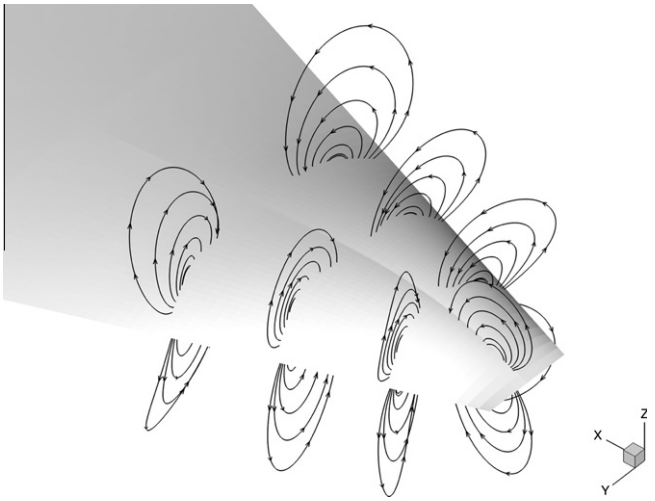


Fig. 3. Imposed magnetic field.

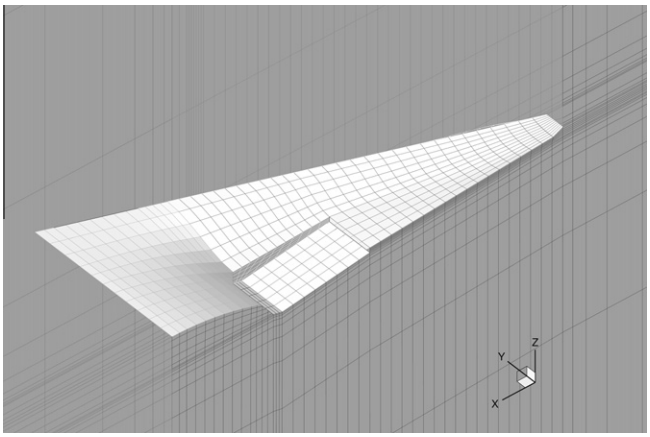


Fig. 4. Mesh, bottom view (coarsened 3 levels).

and the properties of each dipole: strength m and orientation angles θ and ϕ. These totaled 23 extra variables.

The results showed in the subsequent sections were made on a parallel processor workstation, with four 3.2 GHz nodes, 2 MB L2 cache and 8 GB of RAM.

6.2. Flow and adjoint solutions

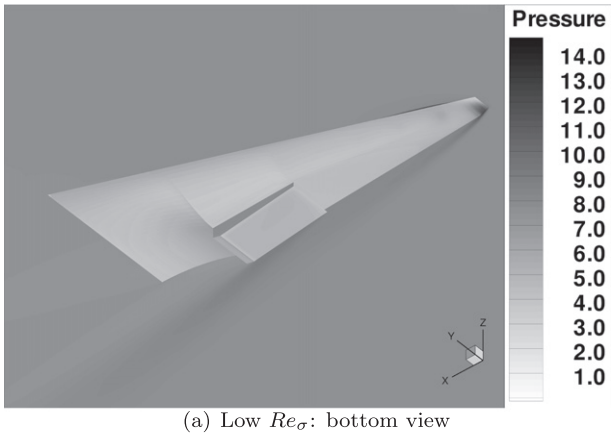
The system of ODEs (15) was integrated in time until the solution evolved to a steady-state. Convergence was assumed when the density residual dropped 10 orders of magnitude.

Fig. 5 shows the pressure contours of the flow solution on the body surface, as well as at the plane of symmetry, for the low Re_σ and the ideal MHD models. As expected, there is a large pressure increase close to the dipoles due to the imposed magnetic field. This effect is caused by the additional magnetic terms in the MHD equations.

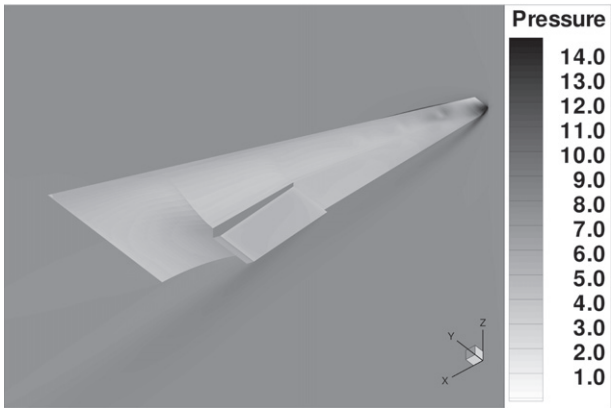
The baseline cost function values are summarized in Table 2, where the reference area was taken as 95.2 m^2, the reference length was 1 m, and the moment reference point was located 10 m behind the nose leading edge.

The assembly time of the Jacobian matrix was 32.53 s for the low Re_σ solver running on the finer mesh, and 70.76 s for the more expensive eight-equation ideal MHD solver on the coarser mesh. These included all the calls to the automatically differentiated routines. The assembly times of the adjoint vectors were negligible for both models. The detailed timings can be examined in Table 6.

Once the adjoint system of Eq. (7) was set up, the GMRES solver provided by PETSc was used. To be consistent with the flow solver, the adjoint solution residual convergence criterion was also set to 10^{-10}. The iterative solver consistently showed very good robustness and convergence properties. For the different function of



(a) Low Re_σ: bottom view



(b) Ideal MHD: bottom view

Fig. 5. Pressure contours.

Table 2
Baseline aerodynamic coefficients.

Model	C_L	C_D	C_{My}
Low Re_σ	0.08265	0.02118	−0.10308
Ideal MHD	0.07809	0.02113	−0.10087

interest tested, convergence was typically achieved after about 120 iterations, which took 263.12 and 383.55 s to run the low magnetic Reynolds number and ideal MHD models, respectively. The residual convergence history of the adjoint solution using PETSc for the different aerodynamic coefficients is plotted in Fig. 6 for the ideal MHD solver.

The adjoint solutions corresponding to the flow pressure for different cost functions, I, are shown in Fig. 7 for the ideal MHD solver. The adjoint solution, for the cost functions used in this work, resembles the flow solution, except that the flow direction looks as if it had been reversed. This is a known characteristic of the dual problem (6).

6.3. Adjoint-based sensitivities

The sensitivity of the pressure drag and pitching moment coefficients with respect to vehicle attitude are shown in Figs. 8 and 9, respectively, for different physical models. The adjoint-based values are compared with forward-FD values using a perturbation step of 10^{-3}. The agreement is within 2.2%, which is acceptable for the accuracy expected from the finite-difference approximations.

Additional sensitivities of the lift coefficient with respect to the dipole properties are summarized in Table 3, for the low

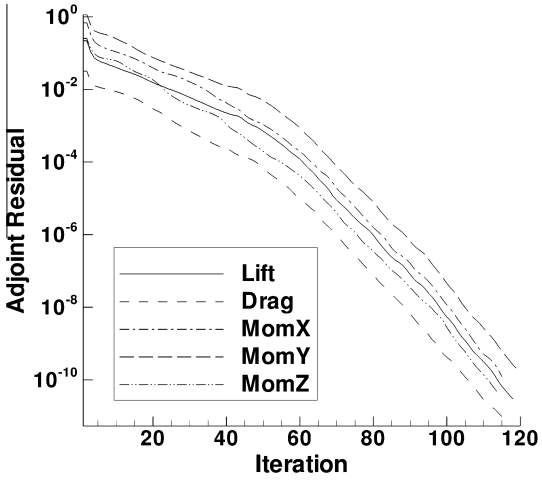


Fig. 6. Adjoint residual history (ideal MHD model).

Re_σ MHD model. A perturbation step size study was conducted for the finite-difference approximations, which showed how dependent the gradient estimates were on their proper choice. For brevity, only the step corresponding to 5×10^{-3} is shown. The overall comparison lead to very satisfactory results; the small discrepancies are mainly attributed to the lack accuracy of the finite-difference approximations.

Similar results were also obtained using the ideal MHD model, and have been included in Table 4. Once again, the matching is very good, validating the sensitivities computed using the ideal MHD adjoint solver.

When running the low Re_σ solver, the electrical conductivity, σ, was also taken as a design variable in each computational node. The total sensitivity was computed for the different cost functions and existed everywhere in the volume since the design variable σ spanned the entire problem domain. For visualization purposes, the values are only shown at the body surface and symmetry plane. Figs. 10–12 show the sensitivity of lift, drag and pitching moment coefficients with respect to the electrical conductivity, respectively.

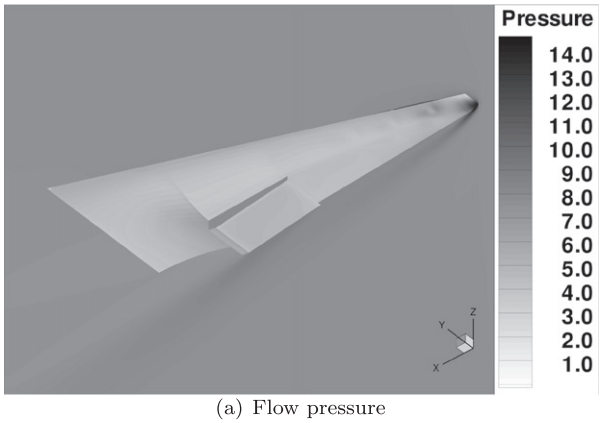
As expected, these sensitivities are greatest close to the location of the dipoles, where the imposed magnetic field intensity is stronger, and their sign depends on the cost function I. Since locally increasing the electrical conductivity σ generates stronger magnetic effects, for a given imposed magnetic field, then it also causes the local pressure to increase. Consequently, the lift sensitivity with respect to σ is positive on the bottom surface and negative on the top, the drag sensitivity is positive on the surface regions facing the incoming flow and negative on the other ones, and the pitching moment sensitivity is positive on the bottom surface behind the reference moment point and negative otherwise.

These insights can be extremely useful if local flow seeding is considered in conjunction with the imposed magnetic field. These sensitivities give the designers the tools to find where, when and how much seeding should be injected to accomplish the desired flow control.

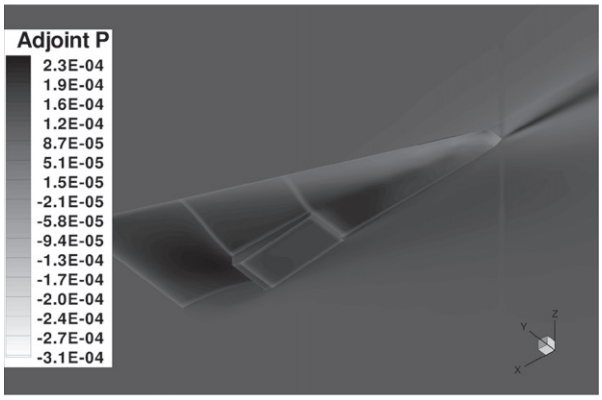
6.4. Verification of the sensitivities

Besides the comparisons performed for the gradients of the cost functions with respect to the vehicle attitude and dipole properties included in Tables 3 and 4, the sensitivities relative to the electrical conductivity were also verified.

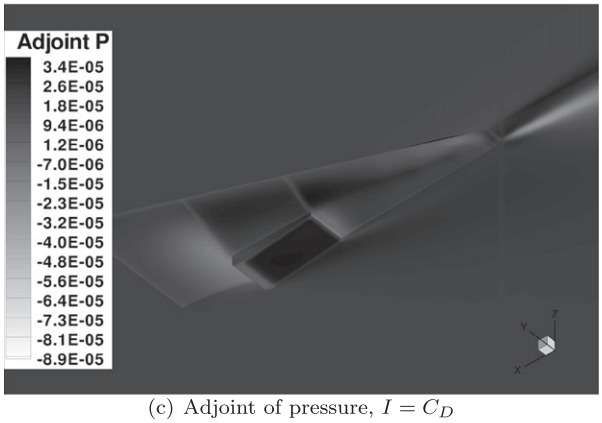
Because the adjoint-based sensitivities of the aerodynamic coefficients with respect to the electrical conductivity showed in



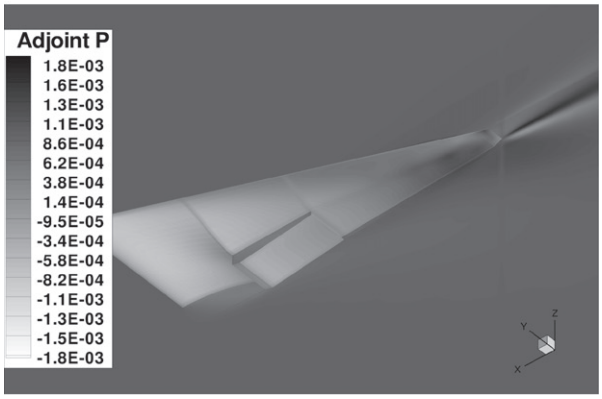
(a) Flow pressure



(b) Adjoint of pressure, $I = C_L$



(c) Adjoint of pressure, $I = C_D$



(d) Adjoint of pressure, $I = C_{My}$

Fig. 7. Adjoint solutions (ideal MHD model).

Figs. 10–12 covered the whole computational domain, the results were just spot-checked against finite-differences.

The comparison was made using five control nodes located on the body surface over the magnetic dipoles location, as graphically shown in Fig. 13.

The comparison results are summarized in Table 5 using a FD perturbation step size of 5×10^{-3} of the baseline electrical conductivity, σ. The values in Table 5 demonstrate that the agreement between the two different approaches is excellent, successfully verifying the adjoint-based gradient values. This verification also revealed that it would have been computationally prohibitive to compute the sensitivities with respect to such large numbers of design variables using anything but the adjoint method: to get the flow solver to converge (starting from the baseline solution) every time the electrical conductivity was perturbed in a single node in the domain, took roughly one and half hours. Extrapolating to all nodes, corresponding to 550,109 design variables, it would have taken almost 95 years to obtain the same results that took less than six minutes (per cost function) for the *ADjoint* method.

6.5. Run-time and memory requirements

The detailed computational costs for the different MHD models are summarized in Table 6. It is important to notice that the flow solver has not been optimized for MHD computations yet and all solutions started from a free-stream condition throughout the domain.

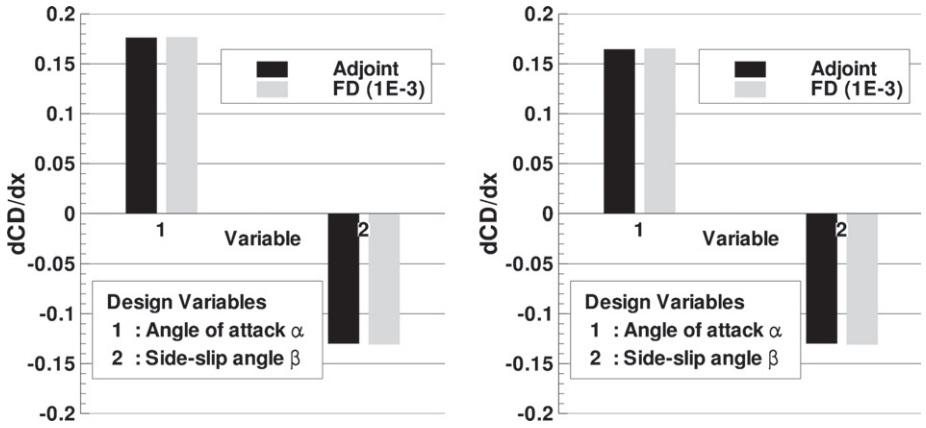
The additional magnetic terms in the MHD equations make the numerical solution much less stable, and because an explicit time integration scheme was used, the runs had to be made at significantly lower CFL numbers. Consequently, it took almost six hours for the ideal MHD flow solver residual to converge 10 orders of magnitude. This clearly rules out the use of finite-differences to compute gradients and highlights the importance of an alternative approach such as discrete adjoint-based gradients.

The solution of the adjoint equations was the component that took most of the time in the adjoint solver, whereas the automatic differentiation sections represented less than 15% of the time, proving its efficiency. The total cost of the adjoint solver, including the computation of all the partial derivatives and the solution of the adjoint system, is less than 3% of the cost of the flow solution for this case. Again, this is not truly representative of reality as the flow solver can still be optimized for MHD, but it clearly shows that the *ADjoint* approach is very efficient. Looking at the bottom line of Table 6, it can be inferred that the *ADjoint* solver runtime is proportional to the number of grid nodes N_c and the number of flow variables N_v squared, as expected from the adjoint matrix Jacobian structure.

The memory usage of the flow and adjoint solvers while running both MHD models (low Re_σ and ideal MHD) was monitored and the information is summarized in Table 7. It shows that the memory required for the adjoint solver is approximately 10 times that required for the original flow solver, when solving only for five governing equations (low Re_σ model), and increases to a sixteen fold for the eight-equation model (ideal MHD). The ratios at the bottom lines of Table 7 show that the memory usage of flow solver is proportional to the number of grid nodes N_c and flow variables N_v, whereas the adjoint solver depends on the number of grid nodes and the number of flow variables squared. This is in line with the explicit flow solver treatment, and the fully assembled adjoint matrix used in the adjoint solver.

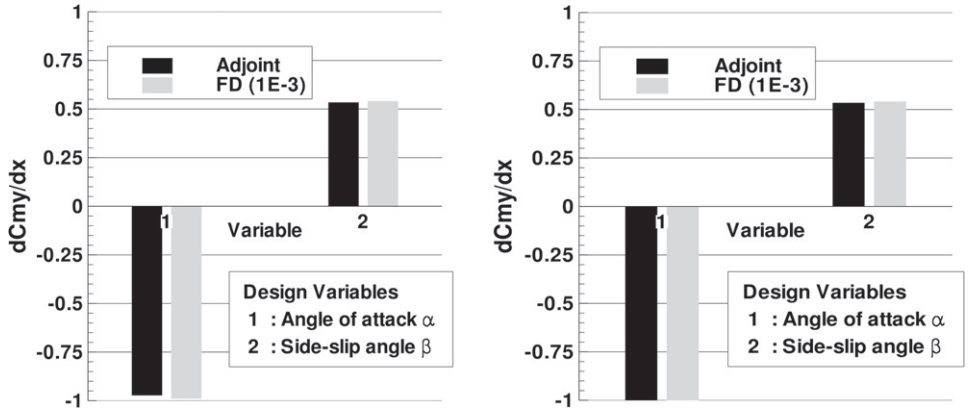
A reduction in memory would be possible, at the expense of a larger CPU cost, by handling the operations in PETSc as matrix-free, eliminating the need to assemble the matrices.

Another comparison was made for the computational cost of the *ADjoint*- and FD-based sensitivities. The values summarized in Table 8 were gathered while performing the comparison of the adjoint-based sensitivities with FD approximations shown previ-



(a) Low Re_σ MHD (b) Ideal MHD

Fig. 8. Sensitivity $dC_D/d\boldsymbol{x}$.



(a) Low Re_σ MHD (b) Ideal MHD

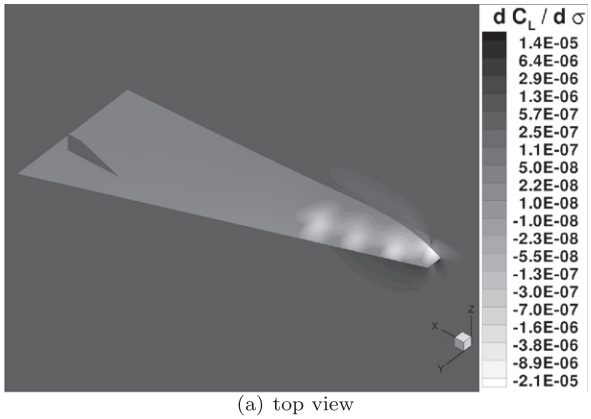
Fig. 9. Sensitivity $dC_{M\ y}/d\boldsymbol{x}$.

Table 3
Sensitivity of C_L w.r.t. magnetic field (low Re_σ MHD).

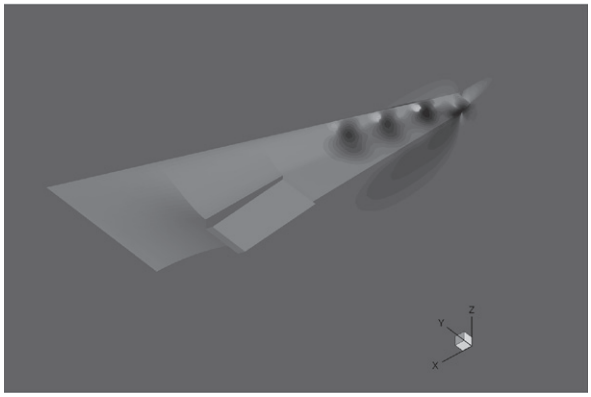
Dipole#	DV $\boldsymbol{x}$	Adjoint	Finite-diff. (step 5×10^{-3})	Δ
1	m	−3.498E−2	−3.564E−02	−1.9%
	θ	−2.337E−4	−2.345E−04	−0.3%
	ϕ	1.354E−4	1.358E−04	−0.3%
2	m	−1.065E−2	−1.123E−02	−5.4%
	θ	−2.233E−5	−2.195E−05	1.7%
	ϕ	5.589E−5	5.661E−05	−1.3%
3	m	4.882E−4	4.610E−04	5.6%
	θ	2.427E−5	2.427E−05	0.0%
	ϕ	4.370E−6	4.444E−06	−1.7%
4	m	−5.439E−3	−5.643E−03	−3.7%
	θ	−1.732E−5	−1.716E−05	0.9%
	ϕ	5.851E−6	6.318E−06	−8.0%
5	m	3.444E−4	3.364E−04	2.3%
	θ	1.008E−5	1.015E−05	−0.7%
	ϕ	2.511E−6	2.457E−06	2.2%
6	m	−1.912E−2	−1.931E−02	−1.0%
	θ	−6.968E−5	−6.985E−05	−0.2%
	ϕ	2.070E−5	2.129E−05	−2.8%
7	m	7.433E−5	7.250E−05	2.5%
	θ	3.515E−6	3.610E−06	−2.7%
	ϕ	1.611E−6	1.574E−06	2.3%

Table 4
Sensitivity of C_L w.r.t. magnetic field (ideal MHD).

Dipole#	DV $\boldsymbol{x}$	Adjoint	Finite-diff. (step 5×10^{-3})	Δ
1	m	−2.576E−1	−2.608E−1	−1.6%
	θ	−3.322E−4	−3.246E−4	2.3%
	ϕ	1.091E−3	1.091E−3	0.0%
2	m	−1.344E−1	−1.367E−1	−1.8%
	θ	−1.439E−4	−1.412E−4	1.9%
	ϕ	−9.153E−4	−9.098E−4	0.6%
3	m	−3.734E−3	−3.853E−3	−3.2%
	θ	1.763E−5	1.838E−5	−4.3%
	ϕ	8.434E−6	7.048E−6	16.4%
4	m	−6.715E−2	−6.790E−2	−1.1%
	θ	−3.764E−4	−3.757E−4	0.2%
	ϕ	−5.535E−5	−5.578E−5	−0.8%
5	m	−4.637E−3	−4.612E−3	0.5%
	θ	−3.293E−5	−3.312E−5	−0.6%
	ϕ	−8.589E−6	−8.780E−6	−2.2%
6	m	−1.043E−1	−1.053E−1	−0.9%
	θ	−6.896E−4	−6.909E−4	−0.2%
	ϕ	−2.577E−4	−2.617E−4	−1.6%
7	m	−2.349E−3	−2.334E−3	0.6%
	θ	−2.277E−5	−2.314E−5	−1.6%
	ϕ	−1.660E−5	−1.652E−5	0.5%

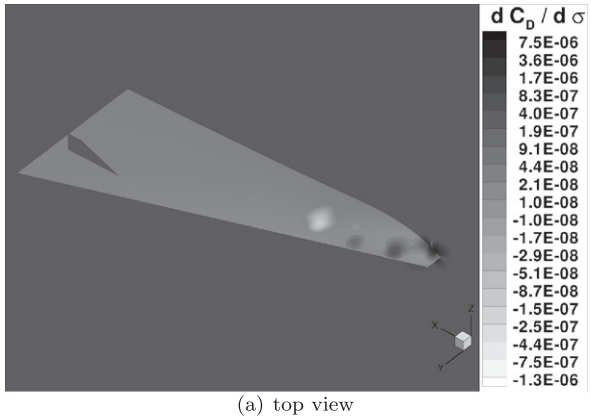


(a) top view

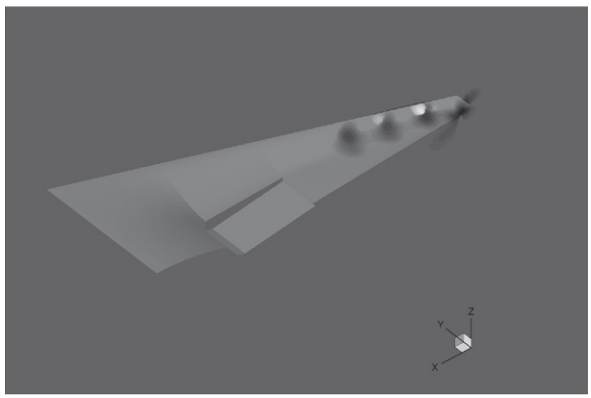


(b) bottom view

Fig. 10. Sensitivity $dC_L/d\sigma$ (low Re_σ model).

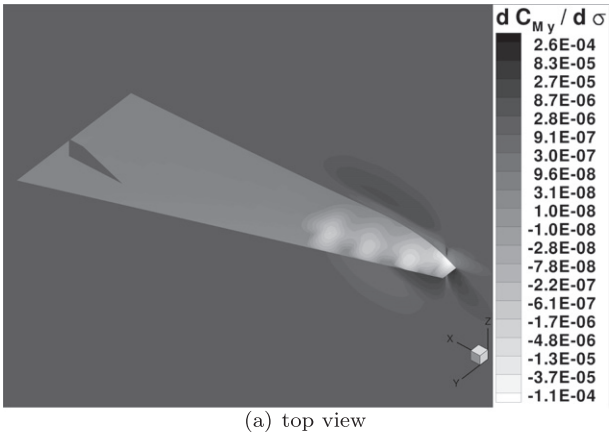


(a) top view

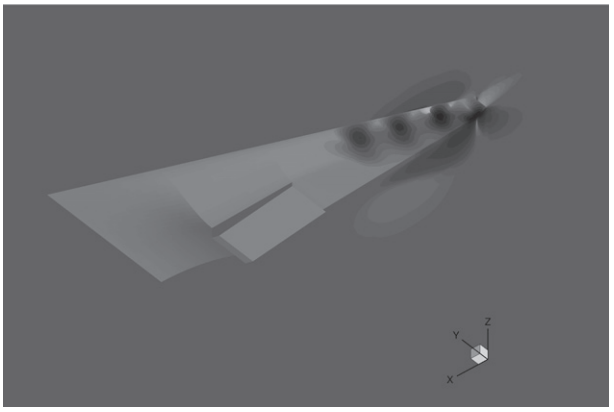


(b) bottom view

Fig. 11. Sensitivity $dC_D/d\sigma$ (low Re_σ model).



(a) top view



(b) bottom view

Fig. 12. Sensitivity $dC_{My}/d\sigma$ (low Re_σ model).

Table 5
Verification of $dI/d\sigma$.

Node #	Func. I	Adjoint	Finite-diff. (step 5×10^{-3})	Δ
1	C_L	−2.0799E−5	−2.0835E−05	−0.2%
	C_D	3.0044E−6	3.0007E−06	0.1%
	C_{My}	−1.0495E−4	−1.0462E−04	0.3%
2	C_L	1.4367E−5	1.3960E−05	2.8%
	C_D	7.5052E−6	7.4295E−06	1.0%
	C_{My}	2.5843E−4	2.6109E−04	−1.0%
3	C_L	1.3471E−5	1.3223E−05	1.8%
	C_D	1.3559E−6	1.2954E−06	4.5%
	C_{My}	1.5455E−4	1.5620E−04	−1.1%
4	C_L	4.1276E−6	4.0171E−06	2.7%
	C_D	6.6961E−7	6.5858E−07	1.6%
	C_{My}	3.5045E−5	3.5971E−05	−2.6%
5	C_L	4.0768E−6	4.0112E−06	1.6%
	C_D	8.9999E−7	8.9586E−07	0.5%
	C_{My}	2.5903E−5	2.6500E−05	−2.3%

ously. The efficiency of the adjoint solver is again evident, outperforming tremendously the traditional finite-difference sensitivity method. In this case, using a non-optimized MHD flow solver, the adjoint-based sensitivity is roughly 30 times faster, per cost function and design variable, than the FD sensitivity.

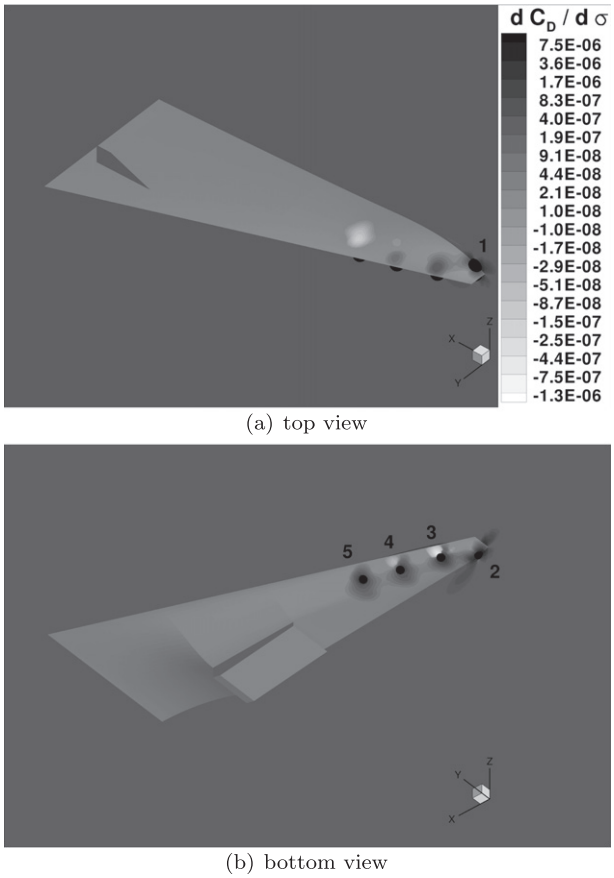
6.6. Sample design problem using the low Re_σ MHD solver

The low Re_σ MHD flow and adjoint solvers were tested on an optimization application.

The design problem was a re-entry hypersonic vehicle in the atmosphere, in which both the vehicle attitude and dipole

Table 6
Computational cost breakdown.

	Wall clock time (s)	
	Low MHD (550*k*)	Ideal MHD (290*k*)
Flow solver	15,353	20,614
ADjoint solver	322.90	476.15
Breakdown:		
Setup PETSc variables	1.48	0.38
Assemble matrix $\frac{\partial \mathscr{R}}{\partial w}$	32.53	70.76
Assemble vector $\frac{\partial I}{\partial w}$	0.01	0.01
Solve ADjoint system	263.12	383.55
Compute sensitivity	25.76	21.45
$\frac{\text{ADjoint system}}{\#\text{GridNodes}\times\#\text{FlowVars}^2}$	0.0191	0.0207



(a) top view

(b) bottom view

Fig. 13. Spot-check of $dC_D/d\sigma$ (low Re_σ model)

properties were taken as control variables, $\boldsymbol{x}$, with the objective of maximizing the inviscid drag coefficient, $I = C_D$. A total of 23 design variables, x, were considered: angle of attack, side-slip angle, dipole strengths (7) and dipole orientations (14). Their upper and lower bounds, and initial values are compiled in Table 9. The problem had a constraint on the lift coefficient, as indicated in the previously mentioned table.

The baseline flow conditions corresponded to the setup described in Section 6.1.

The design was performed using the SNOPT optimizer and its convergence history is shown in Fig. 14. To make the flow turnaround time faster, a coarser mesh with *98,288* nodes was used, and the relative L2-norm for convergence was lowered to 10^{-6}.

The optimal design found after the optimizer converged to the specified tolerances is described in the last column of Table 9. It

Table 7
Memory usage comparison (in MB).

	Virtual memory (MB)	
	Low MHD (550*k*)	Ideal MHD (290*k*)
Flow solver	697	602
ADjoint solver	7349	9782
Ratio	10.5×	16.3×
$\frac{\text{Flow memory (B)}}{\#\text{GridNodes}\times\#\text{ FlowVars}}$	253	259
$\frac{\text{ADjoint memory (B)}}{\#\text{GridNodes}\times\#\text{FlowVars}^2}$	534	527

Table 8
Cost comparison of *ADjoint* and FD gradients.

	Wall clock time (min)	
	Low MHD (550*k*)	Ideal MHD (290*k*)
Flow solution	255.9	343.6
Grad. *ADjoint* (per function)	5.4	7.9
Grad. FD (per variable)	146.3	210.2

Table 9
Baseline and optimized design variables.

	Var.	Lower bound	Upper bound	Baseline	Optim.
Vehicle	α	−0.1745	0.1745	0.0349	0.0971
Attitude	β	0.0000	0.0000	0.0000	0.0000
Dip. 1	m_1	−0.0150	−0.0001	−0.0050	−0.0150
	θ_1	−0.6981	0.6981	0.0000	−0.5357
	ϕ_1	−0.3491	0.3491	0.0000	−0.3491
Dip. 2	m_2	0.0001	0.0150	0.0050	0.0150
	θ_2	−0.6981	0.6981	0.0000	0.6981
	ϕ_2	1.5708	2.2689	1.9251	1.5708
Dip. 3	m_3	0.0001	0.0150	0.0050	0.0150
	θ_3	−0.6981	0.6981	0.0000	−0.4893
	ϕ_3	−2.2689	−1.5708	−1.9251	−1.9112
Dip. 4	m_4	0.0001	0.0150	0.0050	0.0150
	θ_4	−0.6981	0.6981	0.0000	0.4662
	ϕ_4	1.5708	2.2689	1.9251	1.5708
Dip. 5	m_5	0.0001	0.0150	0.0050	0.0001
	θ_5	−0.6981	0.6981	0.0000	0.0000
	ϕ_5	−2.2689	−1.5708	−1.9251	−1.9251
Dip. 6	m_6	0.0001	0.0150	0.0050	0.0150
	θ_6	−0.6981	0.6981	0.0000	0.6981
	ϕ_6	1.5708	2.2689	1.9251	1.5794
Dip. 7	m_7	0.0001	0.0150	0.0050	0.0001
	θ_7	−0.6981	0.6981	0.0000	0.0000
	ϕ_7	−2.2689	−1.5708	−1.9251	−1.9251
Lift	C_L	0.0750	0.0900	0.0420	0.0900
Drag	C_D	−/−	−/−	0.0133	0.0224

can be seen that the dipole strengths had their bounds active, and that the lift constraint was satisfied at its upper bound.

The number of SNOPT major iterations and functions calls done by the optimizer are included in Table 10, where the values obtained while running the problem without any magnetic effects (degenerating to the Euler equations model) were also included as reference. Notice that a function call corresponds to one flow and two adjoint (cost function C_D and constraint on C_L) solutions. Table 10 shows an improvement of 6 drag counts using MHD control.

The flow field pressure and the velocity streamlines for the optimized configuration are shown in Fig. 15 for the low Re_σ model.

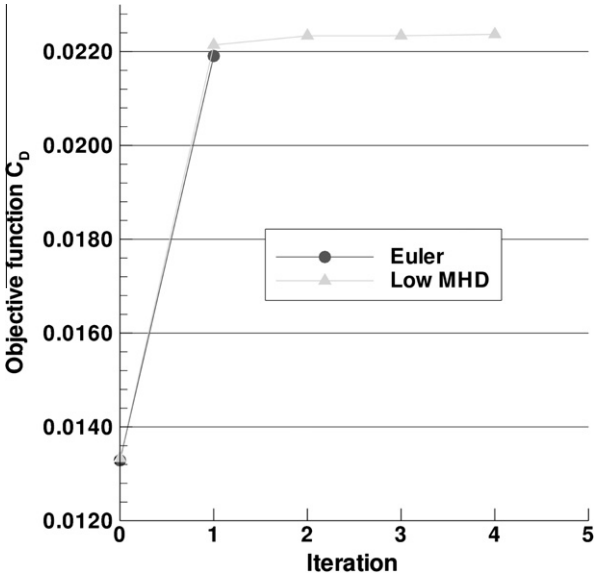
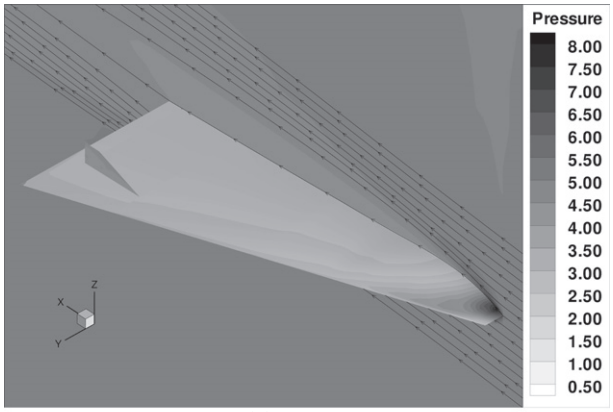


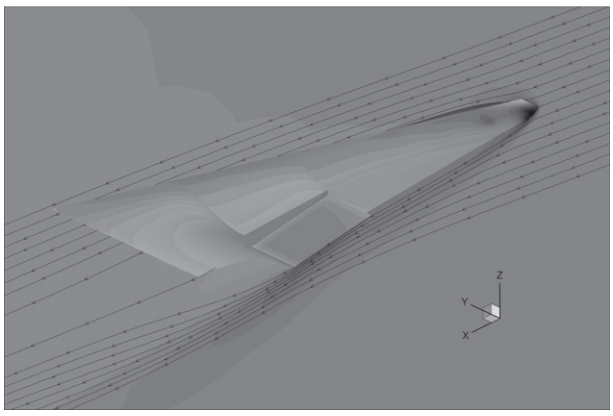
Fig. 14. Optimization problem history.

Table 10
Functions of interest of design problem.

Case	Iter.	F. Call	C_D	C_L
(Baseline)			0.0133	0.0420
Euler	1	3	0.0218	0.0894
Low Re_σ	5	9	0.0224	0.0900



(a) top view



(b) bottom view

Fig. 15. Pressure contours on optimized vehicle.

The results shown for this test case clearly attest to the validity and efficiency of the *ADjoint* approach to estimate gradient information in large, complex problems modeled by the MHD equations. This provides a solid ground to even more complex MHD flow control problems that have not been possible to grasp by the scientific community yet.

7. Conclusions

This work has pioneered the extension of the discrete adjoint approach to the control of a hypersonic flow in the presence of magnetic fields, and successfully demonstrated its feasibility in simple design problem governed by MHD governing equations and using up to a half million design variables. It represents the first step toward an automatic design framework for problems involving hypersonic flow control using electromagnetic effects.

The implementation of the MHD adjoint solver has been largely automated with the selective use of AD tools, which (1) speeds up the derivation of the adjoint system of equations considerably; (2) generates the numerically exact Jacobians (with no need for approximations that may impact the accuracy of the sensitivities); and (3) takes care of the boundary conditions automatically.

Regarding the main driver of this work, the approach described here is particularly well suited to provide the designer with the necessary information regarding the location and amount of seeding that should be injected to optimally accomplish the desired MHD flow control, since the sensitivity of any function of interest with respect to the local electrical conductivity of the medium can be easily assessed by means of the MHD adjoint solver.

References

[1] Agarwal RK, Augustinus J. Numerical simulation of compressible viscous MHD flows for reducing supersonic drag of blunt bodies. AIAA Paper 1999-0601. In: Proceedings of the 37th AIAA aerospace sciences meeting and exhibit, Reno, NV; 1999.
[2] Balay S, Buschelman K, Eijkhout V, Gropp WD, Kaushik D, Knepley MG, et al. PETSc users manual. Technical Report ANL-95/11, Revision 2.3.0, Argonne National Laboratory; 2004.
[3] Carpenter MH, Nordström J, Gottlieb D. A stable and conservative interface treatment of arbitrary spatial accuracy. J Comput Phys 1999;148(2):341–65.
[4] Coakley JF, Porter RW. Time-dependent numerical analysis of MHD blunt body problem. AIAA J 1971;9(8):1624–6.
[5] Gaitonde DV. Higher-order solution procedure for three-dimensional nonideal magnetogasdynamics. AIAA J 2001;39(11):2111–20.
[6] Giles MB, Pierce NA. An introduction to the adjoint approach to design. Flow, turbulence and combustion, vol. 65. Kluwer Academic Publishers; 2000. p. 393–415.
[7] Gill PE, Murray W, Saunders MA. SNOPT: an SQP algorithm for large-scale constrained optimization. SIAM J Optim 2002;12(4):979–1006.
[8] Gupta S, Tannehill JC, Mehta UB, Bogdanoff DW. Simulation of 3-D nonequilibrium seeded airflow in the NASA Ames MHD channel. J Thermophys Heat Transfer 2007;21(2):276–83.
[9] Hascoët L, Pascual V. Extension of TAPENADE towards Fortran 95. In: Bücker HM, Corliss G, Hovland P, Naumann U, Norris B, editors. Automatic differentiation: applications, theory, and tools. Lecture notes in computational science and engineering. Springer; 2005.
[10] Jameson A. Aerodynamic design via control theory. J Sci Comput 1988;3(3):233–60.
[11] Jameson A, Pierce NA, Martinelli L. Optimum aerodynamic design using the Navier–Stokes equations. Theoretical and computational fluid dynamics, vol. 10. Springer-Verlag GmbH; 1998. p. 213–37.
[12] Johnson DB, Robinson JS. X-43D conceptual design and feasibility study. AIAA Paper 2005-3416. In: Proceedings of the 13th AIAA/CIRA international space planes and hypersonics systems and technologies, Capua, Italy; 2005.
[13] Malmuth N, Zakharov S, Fedorov A. Conical Navier–Stokes modeling of forebody vortex symmetry plasma control. AIAA Paper 2007-0219. In: Proceedings of the 45th AIAA aerospace sciences meeting & exhibit, Reno, NV; 2007.
[14] Marta AC. Rapid development of discrete adjoint solvers with applications to magnetohydrodynamic flow control. Ph.D. Dissertation, Stanford University, Stanford, CA, USA; 2007.
[15] Marta AC, Alonso JJ. Discrete adjoint formulation for the ideal MHD equations. AIAA Paper 2006-3345. In: Proceedings of the 3rd AIAA flow control conference, San Francisco, CA; 2006.

[16] Marta AC, Mader CA, Martins JRRA, van der Weide E, Alonso JJ. A methodology for the development of discrete adjoint solvers using automatic differentiation tools. Int J Comput Fluid Dyn 2007;21(9–10):307–27.
[17] Martins JRRA, Alonso JJ, Reuther JJ. A coupled-adjoint sensitivity analysis method for high-fidelity aero-structural design. Optimiz Eng 2005;6(1):33–62.
[18] Mattsson K, Nordström J. Summation by parts operators for finite difference approximations of second derivatives. J Comput Phys 2004;199(2):503–40.
[19] Nowak R, Krank S, Porter RW, Yuen M, Cambel AB. Magnetogasdynamic re-entry phenomena. J Spacecraft 1967;4(11):1538–42.
[20] Pironneau O. On optimum design in fluid mechanics. J Fluid Mech 1974;64:97–110.
[21] Poggie J, Gaitonde DV. Computational studies of magnetic control in hypersonic flow. AIAA Paper 2001-0196. In: Proceedings of the 39th AIAA aerospace sciences meeting and exhibit, Reno, NV; 2001.
[22] Reuther JJ, Jameson A, Alonso JJ, Rimlinger MJ, Saunders D. Constrained multipoint aerodynamic shape optimization using an adjoint formulation and parallel computers, part 1. J Aircraft 1999;36(1):51–60.
[23] Saad Y, Schultz MH. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J Sci Stat Comput 1986;7(3):856–69.
[24] Shang JS, Gaitonde DV, Updike GA. Modeling magneto-aerodynamic actuator for hypersonic flow control. AIAA Paper 2004-2657. In: Proceedings of the 35th AIAA plasmadynamics and lasers conference, Portland, OR; 2004.
[25] Tanaka T. Finite volume TVD scheme on an unstructured grid system for three-dimensional MHD simulation of inhomogeneous systems including strong background potential fields. J Comput Phys 1994;111(2):381–9.