

# Global Artificial Boundary Conditions for Computation of External Flows with Jets \*

Semyon Tsynkov<sup>†</sup> Saul Abarbanel<sup>‡</sup> Jan Nordström<sup>§</sup>  
Victor Ryaben'kii<sup>¶</sup> and Veer Vatsa<sup>||</sup>

*NASA Langley Research Center, Hampton, VA 23681-2199, USA*

## Abstract

We propose new global artificial boundary conditions (ABCs) for computation of flows with propulsive jets. The algorithm is based on application of the difference potentials method (DPM). Previously, similar boundary conditions have been implemented for calculation of external compressible viscous flows around finite bodies. The proposed modification substantially extends the applicability range of the DPM-based algorithm. In the paper, we present the general formulation of the problem, describe our numerical methodology, and discuss the corresponding computational results. The particular configuration that we analyze is a slender three-dimensional body with boat-tail geometry and supersonic jet exhaust in a subsonic external flow under zero angle of attack. Similarly to the results obtained earlier for the flows around airfoils and wings, current results for the jet flow case corroborate the superiority of the DPM-based ABCs over standard local methodologies from the standpoints of accuracy, overall numerical performance, and robustness.

---

\*This research was supported by the National Aeronautics and Space Administration under NASA Contract No. NAS1-97046 while the first, second, third, and fourth authors were in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center. This work was also supported by Director's Discretionary Fund, NASA Langley Research Center. Originally presented as AIAA Paper # 99-3351 at the 14th AIAA CFD Conference, Norfolk, VA, June 1999.

<sup>†</sup>Consultant for ICASE. Permanently: Senior Lecturer, School of Mathematical Sciences, Tel Aviv University, Ramat Aviv, Tel Aviv 69978, Israel. Member AIAA. Phone: (972-3)640-8812, Fax: (972-3)640-9357, E-mail: [tsynkov@math.tau.ac.il](mailto:tsynkov@math.tau.ac.il).

<sup>‡</sup>Consultant for ICASE. Permanently: Professor, School of Mathematical Sciences, Tel Aviv University, Ramat Aviv, Tel Aviv 69978, Israel. Phone: (972-3)640-8951, Fax: (972-3)640-9357, E-mail: [saul@math.tau.ac.il](mailto:saul@math.tau.ac.il).

<sup>§</sup>Consultant for ICASE. Permanently: Senior Scientist, FFA, The Aeronautical Research Institute of Sweden, Box 11021, S-161 11, Bromma, Sweden. Phone: (46-8)634-1165, Fax: (46-8)634-1071, E-mail: [nmj@ffa.se](mailto:nmj@ffa.se).

<sup>¶</sup>Consultant for ICASE. Permanently: Professor, Keldysh Institute for Applied Mathematics, Russian Academy of Sciences, 4 Miusskaya Sq., Moscow 125047, Russia. Phone: (7-095)250-7839, Fax: (7-095)972-0737, E-mail: [ryab@spp.keldysh.ru](mailto:ryab@spp.keldysh.ru).

<sup>||</sup>Senior Research Scientist, Computational Modeling and Simulation Branch, Aerodynamics, Aerothermodynamics & Acoustics Competency, Mail Stop 128. Member AIAA. Phone (757)864-2236, Fax: (757)864-8816, E-mail: [vatsa@tabdemo.larc.nasa.gov](mailto:vatsa@tabdemo.larc.nasa.gov).

## Introduction

Many typical problems in aerodynamics including those that present immediate practical interest, e.g., flows around aircraft, are formulated on infinite domains. It is, however, obvious, that any discretization used for solving such problems on the computer must be finite. Therefore, any numerical solution methodology for these problems has to be supplemented (or, rather, preceded) by a special technique that helps create such finite discretizations.

A widely used approach to this problem is based on truncating the original flow domain prior to the actual discretization and numerical solution. Subsequently, one can construct a finite discretization on the new bounded computational domain using one of the standard techniques: finite differences, finite elements, or other. However, both the continuous problem on the truncated domain and its discrete counterpart will be subdefinite unless supplemented by the appropriate closing procedure at the external computational boundary. This is done by using artificial boundary conditions (ABCs); the word “artificial” emphasizing here that these boundary conditions are necessitated by numerics and do not come from the original physical formulation.

The ideal or, in other words, exact, ABCs are obviously those that would drive the error associated with domain truncation to zero. However, numerically efficient procedures of this kind cannot be attained routinely except in model (mostly one-dimensional) problems and therefore, for typical applications one uses primarily different approximate rather than exact methodologies.

The nature of the difficulties associated with constructing the exact ABCs is that in most cases such boundary conditions appear nonlocal (in space and also in time for unsteady problems). Although the corresponding computational algorithms are robust and highly accurate, they can be cumbersome and typically apply only to rather simple geometries. On the other hand, the alternative local approaches that yield inexpensive and geometrically universal numerical procedures may often lack accuracy in computations, which, in turn, necessitates choosing excessively large computational domains. Basically, higher accuracy due to boundary conditions implies that more of the nonlocal nature of exact ABCs has to be taken into consideration. As a consequence, to avoid extra complexity due to the nonlocality of boundary conditions, most of the modern production algorithms in CFD still employ local ABCs that are based on simplified flow models. The possibility to use local ABCs comes, as mentioned, at the expense of running the cases on large domains.

Generally, it has been demonstrated theoretically and computationally in both CFD and other areas of scientific computing that the treatment of ABCs may have a profound impact on the overall performance of numerical algorithms and interpretation of the results. The literature on various ABCs’ techniques is extensive, a detailed review can be found in work

by Givoli,<sup>1,2</sup> as well as in a more recent paper by Tsynkov.<sup>3</sup>

The construction of ABCs based on the difference potentials method (DPM) by Ryaben'kii,<sup>4-6</sup> was an attempt to combine in one technique the advantages relevant to both local and global methodologies, see Refs. 7-17. These boundary conditions employ finite-difference counterparts to Calderon's pseudodifferential boundary projection operators and generalized potentials that have been first proposed in work by Calderon<sup>18</sup> and then also studied by Seeley.<sup>19</sup> The DPM-based ABCs have been successfully implemented along with NASA-developed multigrid Navier-Stokes solvers for the calculation of two-dimensional (solver FLOMG by Swanson and Turkel<sup>20-22</sup>) and three-dimensional (solver TLNS3D by Vatsa, et al.<sup>23,24</sup>) compressible viscous flows around airfoils (NACA0012, RAE2822) and wings (ONERA M6).

In many numerical tests the DPM-based boundary conditions have consistently outperformed the standard local methods from the standpoints of accuracy, multigrid convergence rate, and overall robustness (they allow for a substantial reduction of the domain size while preserving the accuracy and may also speed up the convergence of multigrid iterations by up to a factor of three, i.e., they would require only about one third of the original number of multigrid cycles for reducing the initial residual by a prescribed factor). Note, the standard local boundary conditions for external flows that are referred to above are typically based on one-dimensional characteristics analysis for the front or inflow part of the artificial boundary and specification of the free-stream pressure and extrapolation of all other quantities on the rear or downstream portion of the outer boundary; this treatment may or may not be supplemented by the point-vortex correction<sup>25</sup> for the two-dimensional case; an example of geometry in three dimensions is shown on Figure 1 in the next section.

All the problems analyzed previously in the DPM framework (see the aforementioned references) can actually be characterized as "pure" external flows. *In this paper, we for the first time incorporate a new and essentially different physical element into the formulation of the problem; namely, we will consider external flows around configurations with jet exhaust.* The problems of this kind have never been studied by means of the DPM before and including this new flow phenomena into the range of admissible formulations for the DPM-based methodology substantially enlarges the scope of its capabilities. Moreover, as different flows with jets are frequently encountered in aerospace applications, the possibility of computing external aerodynamics more efficiently with jet exhaust phenomena taken into account is important for both configuration analysis and design.

The material in the paper is prepared as follows. In the next section we outline the basic DPM-based procedure as developed for pure external flows; in the section that follows we describe the changes that are necessary for incorporating the jet exhaust flows; then, we

present the numerical results and conclusions.

## DPM-based ABCs: Basic Algorithm

In this section, we outline the corresponding derivation from Ref. 15. The paper by Tsynkov<sup>16</sup> contains a substantially more detailed account on how to set the three-dimensional DPM-based ABCs.

We consider a steady-state flow of a viscous, compressible, perfect gas past a finite three-dimensional configuration. The flow is uniform and subsonic at infinity; it is also symmetric with respect to the Cartesian plane  $z = 0$ . The hydrodynamics equations are discretized and integrated on a grid generated around the immersed body(ies). The grid actually defines a bounded computational domain; the ABCs that would close the truncated problem should be set at the external coordinate surface of the grid. Let us denote this surface  $\Gamma$ ; for a one-block curvilinear C-O type boundary-fitted grid around the ONERA M6 wing the schematic geometric setup is shown in Figure 1.

The outermost coordinate surface of the grid is designated  $\Gamma_1$  (see Figure 1); it represents the ghost nodes (or ghost cells for the finite-volume formulation). Clearly, when the stencil of the scheme used inside the computational domain is applied to any node from  $\Gamma_1$ , it generally requires some ghost cell data. Unless the required data are provided, the finite-difference system solved inside the computational domain appears subdefinite, i.e., it has less equations than unknowns. Therefore, in practical framework the closure of the discretized truncated problem means specification of the solution values at the ghost cells. This will be done by means of the DPM-based ABCs so that the boundary data used for the closure admit an exterior complement that solves the problem outside the computational domain. As soon as the data in the ghost cells have been obtained as functions of the data in the interior cells ( $\Gamma_1$  as a function of  $\Gamma$ ), the corresponding relations can be incorporated into the actual solver used inside the computational domain. If, for example, this is an iterative solver (very often the case), then one has to update the ghost cells at each iteration to advance to the next “time” step.

To construct the boundary conditions, we first assume that the flow perturbations against constant free-stream background are small in the far field and consider the linearized problem outside the computational domain (i.e., outside  $\Gamma$ ). It is important to emphasize that the possibility of far-field linearization (i.e., the possibility to retain only the first-order terms with respect to perturbations in the governing equations) requires special justification, in particular, when analyzing transonic flows. We do not present the corresponding argument here; a simple asymptotic analysis in the framework of the full potential model that justifies the far-field linearization in three dimensions can be found in our previous work.<sup>15,16</sup> Of

course, even if we know that the far field is linear, we still cannot say a priori whether or not the linearization outside  $\Omega$  is possible for a particular configuration of the domains. Clearly, for a very large computational domain one can linearize the flow outside  $\Omega$ , and as we approach the source of perturbations (the immersed configuration), the validity of linearization is verified a posteriori (see, e.g., Refs. 8, 9, 12, 15, 16).

We will be considering the entire problem in the framework of the thin-layer equations (as opposed to the full Navier-Stokes equations). This simplified flow model still retains all the essential properties pertinent to the class of problems that we are studying and at the same time it is less expensive numerically. In the far field, the thin-layer equations are linearized against the constant free-stream background and supplemented by the condition of vanishing of all flow perturbation at infinity, which corresponds to the free stream limit of the solution. The actual form of the linearized governing equations in the far field is presented in Refs. 15, 16.

We discretize the linearized thin-layer system with the second order of accuracy on an auxiliary Cartesian grid; a detailed description of the resulting finite-difference scheme can be found in Ref. 16. *The DPM will provide us with the complete boundary classification of all those and only those exterior grid vector-functions that solve this discretized system outside the computational domain and meet the boundary condition at infinity (vanishing of all perturbations).* The foregoing boundary classification will be obtained as an image of a special projection operator, which can be considered a discrete analogue to Calderon's pseudodifferential boundary projection.<sup>18,19</sup> The projection operators act on the grid functions defined as boundary traces of the solution. In actual computations, the boundary conditions are set as follows. Every time we need to update the ghost cells we take an appropriate set of data from inside  $\Omega$  (see below), project it onto the subspace in the entire space of boundary data that admits the correct exterior complement, and obtain the ghost cell values by calculating this complement on  $\Omega_1$ .

The implementation of the DPM-based ABCs starts with splitting the nodes of the auxiliary Cartesian grid into two distinct groups: those that are inside  $\Omega$ , and those that are outside  $\Omega$ . Applying the stencil of the scheme that we use to each node of both groups, we consider the intersection of the grid sets swept by the stencil. This intersection is called the grid boundary  $\gamma$ ; it is a multi-layered fringe of nodes of the auxiliary Cartesian grid located near and straddling the continuous boundary  $\Gamma$ .

For any function  $\mathbf{u}$  on the Cartesian grid we define its trace  $\mathbf{Tr}_\gamma \mathbf{u}$  on  $\gamma$  as merely a restriction. For any grid function  $\mathbf{u}_\gamma$  specified on  $\gamma$  we introduce the generalized potential  $\mathbf{P} \mathbf{u}_\gamma$  with the density  $\mathbf{u}_\gamma$ ; the generalized potential is defined on the auxiliary Cartesian grid on the boundary  $\gamma$  and outside it. The generalized potential is obtained as a solution

of the special auxiliary problem (AP) driven by the right-hand side that depends on  $\mathbf{u}_\gamma$ ; the formal construction of this right-hand side is the same in two- and three-dimensional cases, see Refs. 7, 11, 12, 16 for details. The AP is formulated on a finite auxiliary domain, which is a Cartesian parallelepiped that fully contains  $\Omega_\gamma$ . The formulation of the AP involves periodization in the cross-stream and span-wise directions  $y$  and  $z$ , respectively. The periods are chosen sufficiently large to guarantee that the periodic solution *considered on a finite fixed neighborhood of  $\Omega_\gamma$  and  $\Omega_\gamma$*  provide a good approximation to the theoretical non-periodic solution; the latter can be thought of as a limit when the periods in  $y$  and  $z$  approach infinity. The approximation of a non-periodic solution by the periodic one on a fixed subinterval as the period increases is discussed in our work.<sup>7,11,12</sup> The boundary conditions for the AP in the stream-wise direction  $x$  are constructed semi-analytically using explicit mode selection that would guarantee the desired far-field behavior of the solution, see Refs. 15, 16.

The AP allows us to calculate the generalized difference potential  $\mathbf{P}\mathbf{u}_\gamma$  for any grid density  $\mathbf{u}_\gamma$  specified on  $\gamma$ . In the vicinity of the computational domain the potential approximates the solution to the linearized equations subject to the zero boundary condition at infinity (the accuracy of this approximation can be controlled by choosing the periods in  $y$  and  $z$ ). The composition of the operators  $\mathbf{Tr}_\gamma$  and  $\mathbf{P}$ ,  $\mathbf{P}_\gamma \equiv \mathbf{Tr}_\gamma \mathbf{P}$ , is a projection,  $\mathbf{P}_\gamma^2 = \mathbf{P}_\gamma$ , and it is a discrete counterpart of Calderon's boundary projection<sup>18,19</sup> for the linearized thin-layer system. The image of this projection,  $\text{Im}\mathbf{P}_\gamma$ , contains all those and only those  $\mathbf{u}_\gamma$ 's that are traces of the exterior difference solutions to the linearized thin-layer equations that satisfy the boundary conditions of the AP. In the sense of the foregoing approximation (periodization),  $\text{Im}\mathbf{P}_\gamma$  contains all those and only those  $\mathbf{u}_\gamma$ 's that are traces of the exterior difference solutions to the linearized thin-layer equations that satisfy the zero boundary conditions at infinity.

Having constructed the procedure for calculating the potentials and projections for the discrete linearized thin-layer equations, we can now close the system inside the computational domain, i.e., obtain the ABCs. First, we take  $\mathbf{u}$  and  $\partial\mathbf{u}/\partial n$  on  $\Omega_\gamma$ ,  $n$  is the normal, (these data are available from inside the computational domain) and, using interpolation  $\mathbf{R}$ , along  $\gamma$ , and the first two terms of Taylor's expansion (denoted  $\pi_\gamma$ ), obtain  $\mathbf{u}_\gamma$ :

$$\mathbf{u}_\gamma = \pi_\gamma \mathbf{R}_\gamma \left( \mathbf{u}, \frac{\partial\mathbf{u}}{\partial n} \right) \Big|_{\Omega_\gamma}. \quad (1)$$

Then, we need to calculate the potential  $\mathbf{P}\mathbf{v}_\gamma$  for the density  $\mathbf{v}_\gamma = \mathbf{P}_\gamma \mathbf{u}_\gamma$  and interpolate it to the nodes  $\Omega_\gamma$ :

$$\mathbf{u} \Big|_{\Omega_\gamma} = \mathbf{R}_{\Omega_\gamma} \mathbf{P}\mathbf{v}_\gamma \equiv \mathbf{R}_{\Omega_\gamma} \mathbf{P}\mathbf{u}_\gamma. \quad (2)$$

Finally, the ABCs are obtained in the operator form

$$\mathbf{u}|_{,1} = \mathbf{T} \left( \mathbf{u}, \frac{\partial \mathbf{u}}{\partial n} \right) |_{,1}, \quad (3)$$

where  $\mathbf{T}$  is composed of the operations of (1) and (2). Boundary condition (3) is applied every time we need to update the ghost cells when solving the interior problem (e.g., on every iteration). The implementation of ABCs (3) can either be direct or involve preliminary calculation of the matrix  $\mathbf{T}$ . In the latter case, the runtime implementation of the ABCs (3) is reduced to a matrix-vector multiplication. Numerical results for flows around the ONERA M6 wing obtained with the DPM-based boundary conditions (3) are summarized in work by Tsynkov and Vatsa<sup>15</sup> and Tsynkov.<sup>16,17</sup>

## Application to Jet Flows

The major difference between the formulation of the previous section and the flow with jet exhaust is that *in the vicinity of the jet we can no longer claim that flow perturbations against the free-stream background are small*. Indeed, inside the propulsive jet the speed of the flow is typically much higher than the one in the surrounding area, moreover, other parameters, e.g., temperature, may also differ substantially. Therefore, the linearization of the flow against a constant free-stream background everywhere is, generally speaking, not valid in this case.

However, in many applications (in particular, aerospace) one can clearly distinguish between those parts of the overall flow that contain jet(s) and the remaining areas. Therefore, the most comprehensive way to develop the far-field linearization in this situation will apparently be to use the appropriate asymptotic solutions for jets (see, e.g., Abramovich<sup>26</sup>) in the corresponding regions as a background. For flow regions outside the jet, *it is always reasonable to assume that the foregoing linearization around a uniform free-stream solution will still be valid there*.

The particular setting that we will be studying hereafter is schematically shown in Figure 2. (The meaning of the two external grid surfaces is the same as  $\sigma_1$  and  $\sigma_2$  in Figure 1.) It includes a three-dimensional slender body (symmetric with respect to the  $z = 0$  plane but not axially symmetric, i.e., not a body of revolution) with sharp nose and boat-tail aft configuration; the rearmost plane surface of the body (not shown explicitly in Figure 2) is actually a location of the nozzle outlet; the outlet is rectangular in cross section. The exterior flow is subsonic with the free-stream Mach number  $M_0 = 0.6$  and zero angle of attack, the jet that is discharged from the outlet is supersonic,  $M_j = 1.6$ , and confluent with the exterior flow.

The specific shape of the body (see Figure 2) as well as parameters of the flow have been

previously proposed for numerical study and analyzed in work by Compton.<sup>27</sup> In this original work,<sup>27</sup> Compton had calculated external flow with the propulsive jet and also considered the interior portion of the flow, namely the flow in the actual nozzle located inside the afterbody (this nozzle flow obviously produces the jet). For our study, we have generated new grids and also simplified the overall formulation by eliminating the nozzle and specifying instead the uniform supersonic flow conditions at the nozzle outlet i.e., at the place where the jet starts. Compton's goal<sup>27</sup> was to assess the performance of different turbulence models including their prediction capabilities for the flow inside the nozzle; our goal is to assess the performance of different external boundary conditions for the flows with jet exhaust. We, therefore, think that the aforementioned simplification is justified.

Our typical grid consists of two blocks: block 1 of C-O type is for the exterior flow and block 2 of H-O type is for the jet portion (see Figure 2). Of course, this subdivision can only have an approximate meaning because the jet will obviously tend to spread while propagating downstream; basically, it means that the shear layer between the jet and coflow is located in the vicinity of the block interface. On this interface, the two grid blocks are point-matched, which is a requirement for TLNS3D.

As has been mentioned, the exterior flow is subsonic and the jet is supersonic (other parameters of the flow will be pointed out later). The standard boundary conditions in TLNS3D for this two-block jet flow case include one-dimensional characteristics for external inflow (block 1, upstream portion of the boundary), specification of the free-stream pressure with extrapolation of all other quantities for external outflow (block 1, downstream portion of the boundary), extrapolation of all quantities for the jet downstream boundary (block 2) and specification of all quantities for the jet inflow boundary (block 2); the boundary conditions on the solid surface of the body are standard no-slip conditions. Extrapolation of all flow quantities at the jet outflow boundary is justified because as shown by numerous simulations the core of the jet remains supersonic even at large distances downstream of the body, at least as far as 40–50 nozzle calibers away.

The primary goal of this paper is to develop an alternative to the foregoing local boundary conditions for the jet flow case – global ABCs similar to those described in the previous section, and compare the performance of the two techniques. A direct implementation of the ABCs (3) will, however, encounter a major obstacle in this case: as has been mentioned, we cannot linearize against the free-stream background in the jet region and therefore, cannot directly implement the operator  $\mathbf{T}$  of (3) over the entire external boundary as this operator is obtained on the basis of the thin-layer system linearized against a uniform free-stream background. Of course, if we linearized the flow against a constant free-stream background outside the jet and against some approximate asymptotic solution in the jet region (see

Ref. 26) and then used the corresponding linear system (it will have variable coefficients) to construct the operator analogous to  $\mathbf{T}$  of (3), then we could have applied the boundary conditions (3) straightforwardly as done in the previous work<sup>15,16</sup> for flows with no jets. Computation of the new operator  $\mathbf{T}$  in this framework will, in turn, require a different construction of the AP, certainly more elaborate (because of variable coefficients) and possibly more expensive than the one described in the previous section. One way of largely eliminating the difficulties associated with variable coefficients is apparently to take advantage of the supersonic nature of the jet and use marching-type algorithms in a subdomain of the new AP domain. Although this may make the whole foregoing program feasible, we consider its implementation as future work. In this paper we present the algorithm based on boundary conditions (3) with minimal alterations.

As the ABCs (3) obviously cannot be applied in the jet area, i.e., on that portion of the artificial boundary where the jet exits the domain, we need another procedure. The most natural choice will be to extrapolate all flow quantities downstream at the outflow boundary because the core of the jet remains supersonic even at large distances away from the nozzle outlet. Of course, we cannot actually predict where on the downstream boundary the flow actually becomes subsonic, i.e., where the extrapolation ceases to be applicable. However, we have observed that for the particular case under study we can extrapolate at least on the entire downstream boundary of the second grid block (see Figure 2). *Thus, extrapolation of all flow quantities will be used henceforth as downstream boundary conditions for block 2.*

In the standard procedure, the downstream boundary conditions for grid block 1, i.e., on the rest of the outflow boundary, are based on the specification of free-stream pressure and extrapolation of all other quantities. Basically, these boundary conditions are relevant for subsonic outflow. In practice, some portion of the downstream boundary of block 1 may also be supersonic; in this case, however, the implementation of these pressure boundary conditions does not lead to noticeable errors because the streamwise variations of pressure away from the nozzle are small (the jet is close to design, it is slightly overexpanded, see below) and therefore, specification of the free-stream pressure and extrapolation from the interior are both procedures with acceptable accuracy.

To replace local boundary conditions on the outer boundary of block 1 (the region outside the jet) by the more accurate global ABCs, we use relation (3). However, in formula (3) both the input and output are global, i.e., not only the operator  $\mathbf{T}$  provides the ghost cell data along the entire boundary but also requires the data along the entire (penultimate) boundary as driving terms. By using extrapolation downstream in the jet core instead of using (3), we make sure that the possibly erroneous data from the global procedure are not used on this part of the boundary. However, as the global operator  $\mathbf{T}$  is constructed on the

basis of the linearization against a constant background, which is not valid in the jet area, plugging the actual flow quantities (including the jet profile) into the right-hand side of (3) may potentially generate errors along the entire outer boundary.

On the other hand, it has been verified for model examples<sup>14</sup> and also seen for more complex cases that typically, closely located boundary nodes influence one another much stronger than the remote ones. This behavior is reasonable from the standpoint of physics; the structure of operators  $\mathbf{T}$  reflects this behavior so that although the matrix of  $\mathbf{T}$  is dense (non-locality), its near-diagonal terms are much larger than the off-diagonal ones (for systems as opposed to scalar equations, it will be a similar block-wise structure). The specific rate of decay for the off-diagonal terms can probably be obtained only for the most elementary formulations (e.g., the Laplace equation with periodic boundary conditions). However, although we cannot obtain analytical estimates for the kernels involved in operators  $\mathbf{T}$  of (3), we can still make use of the actual (block-wise) off-diagonal decay in the numerical experiments. In practical terms, this implies that *although substituting the jet profile into the right-hand side of (3) violates the small perturbations assumption, the associated error on the left-hand side of (3) will mostly be concentrated again in the jet area, where the results are not used for boundary conditions anyway as they are overridden by extrapolation.*

Thus, the actual combined DPM-based ABCs that we employ for computation of the foregoing jet flow case are the following. For most of the outer boundary (except the near-jet area) we use formula (3) while substituting the actual flow profile in its right-hand side. For the jet core (outflow boundary for grid block 2) we extrapolate all flow quantities downstream. For the small intermediate portion of the downstream boundary (near the jet core) we extrapolate all quantities except pressure, the latter is prescribed from its free-stream value. In fact, we have observed that within a certain range (5 to 30 cells of the fine grid described in Numerical Results), the actual location of where to switch from the pressure boundary condition to formula (3) does not exert much influence on either the final accuracy or multigrid convergence rate. In the next subsection, we provide an additional justification for the use of this procedure.

### **Jet Outflow Boundary Conditions**

To describe and explain the specific boundary treatment in the vicinity of the jet exit through the boundary, we start by considering the model problem below, disregarding for a moment the connection to the global boundary procedure described above.

A model problem describing the error due to inaccurate outflow pressure data for the

steady Euler equations is,

$$\begin{aligned} Ae_x + Be_y + Ce_z &= 0, & x < 0, \\ p &= g(y, z), & x = 0, \\ -\infty &< y, z < \infty, \end{aligned} \tag{4}$$

where  $e = (\rho, u, v, w, p)^T$  denotes the error and  $A$ ,  $B$ , and  $C$  are constant matrices. We assume that the boundary data has compact support outside a small portion of the boundary, i.e.

$$g(y, z) = 0, \quad |y, z| \geq \delta, \tag{5}$$

We also assume that the base flow is subsonic and moves to the right. The problem (4),(5) is a model for the error in an approximate solution with correct outflow boundary data given on  $|y, z| \geq \delta$  and erroneous one's on  $|y, z| < \delta$ .

The relation of the model problem (4),(5) to the specific outflow problem in this paper can briefly be described as follows. The global boundary procedure far away from the jet and the extrapolation procedure, see Refs. 28, 29, in the supersonic part of the jet lead to very small errors, i.e.  $|g| \approx 0$ . In an intermediate domain between the supersonic part of the jet and the part where the global boundary conditions are used, pressure with erroneous data is specified, i.e.  $|g| \approx \mathcal{O}(1)$  in that part of the domain.

Note that for problems with boundary conditions in the  $x$  (or streamwise) direction it makes little difference if one consider the inviscid Euler equations instead of the viscous thin layer Navier-Stokes equations since the number and nature of the boundary conditions required in the  $x$  direction are the same for the two sets of equations.

Let  $\Omega_L^n = ([-(L+1), -L] \times \mathbf{R}^{n-1})$  where  $n$  is the number of spatial dimensions, see Figure 3. The following theorem describes the error distribution in the halfspace  $x < 0$ .

**Theorem 1** *The error  $e$  in (4),(5) satisfies:*

$$\|e\|_{L_2(\Omega_L^n)} \leq C \sqrt{\frac{1 - \bar{u}^2}{\sqrt{(1 - \bar{Q}^2)}}} |\Delta p_{max}| \left[ \frac{\delta}{\sqrt{L}} \right]^{n-1} \tag{6}$$

where  $C$  is a constant and  $|\Delta p_{max}|$  is the maximal magnitude of the pressure error in  $|y, z| < \delta$ ,  $\bar{Q}^2 = \bar{u}^2 + \bar{v}^2 + \bar{w}^2$  for  $n = 3$  and  $\bar{Q}^2 = \bar{u}^2 + \bar{v}^2$  for  $n = 2$ .

Theorem 1 means that *by measuring the error in a local  $L_2$ -norm on the fixed domain  $\Omega_L^n$ , an error decay can be observed. Note that if the error was measured by computing the  $L_2$  norm in the whole computational domain, no error decay could be observed.* The proof of theorem

1 involves a straightforward application of the theory of Ref. 30. Numerical experiments that verify the decay rate (6) can be found in Ref. 31.

For our specific outflow problem with erroneous data given on the intermediate domain between the supersonic part of the jet and the part where global boundary conditions are used, theorem 1 means that the error decays with the rate  $\delta^2/L$  away from the outflow boundary. Furthermore, in our specific flow problem we have a slightly overexpanded jet which means that the maximum pressure error  $|\Delta p_{max}|$  in (6) is rather small.

### Effective Reynolds Number

To calculate the operator  $\mathcal{T}$  of (3), we are solving the AP for the linearized thin-layer equations. These equations formally involve the molecular Reynolds number of the flow. However, as the actual flows that we are studying are turbulent, to integrate the thin-layer equations numerically one complements them with turbulence models inside the computational domain. These models may be complex and require solving some additional differential equations (see next section).

For the simplified linearized far-field representation, we do not use these accurate and sophisticated turbulence models. However, we still need to account for the corresponding turbulent mixing and dissipation, at least in an approximate way. In previous work<sup>9</sup> Tsynkov, et al. have used the concept of effective turbulent viscosity for the far field and calculated the effective turbulent Reynolds number using the fact that the laminar and turbulent plane wakes behind the body have the same asymptotic behavior.<sup>32</sup>

The asymptotic behavior of laminar and turbulent circular jets is also known to be the same.<sup>26,32</sup> It involves a linear increase in width and a decrease in center-line velocity inversely proportional to the distance from the source. The virtual kinematic viscosity (incompressible case) can be considered constant through the entire jet region. Although we do not use boundary conditions (3) in the core of the jet, the outer portions of the shear layer region are still covered by the global procedure, therefore we need to provide the effective value of  $1/Re$  for the linearized thin-layer equations.

The jet that we are studying is rectangular in its initial cross section (see next section for particular details); however, its shape will approach circular further away of the outlet. Therefore, we will use the results obtained for circular jets to find an approximate value for the effective Reynolds number. First, we notice that the universal velocity profiles in a cross section of an incompressible submerged jet (i.e., the jet that propagates through a medium at rest) are the same as those obtained for the excess velocity of the jet propagating in a coflow.<sup>26</sup> Moreover, many experimental observations corroborate<sup>26</sup> that the same universal profiles remain valid for a compressible supersonic jet spreading through either a stationary

or moving medium. Of course, while the profiles are universal, the actual spreading rate for the jet will differ for different cases. Second, for the particular case under study (the ratio of stagnation temperatures is  $T_j^*/T_0^* = 0.936$ ; the design pressure ratio is  $p_j^*/p_0|_{design} = 4.25$  at  $M_j = 1.6$  whereas the actual pressure ratio is  $p_j^*/p_0 = 4.00$ , the jet is slightly overexpanded), the initial value of the compressibility parameter<sup>26</sup> is  $\bar{\rho} = \rho_0/\rho_j = 1.41$  and the initial velocity ratio is  $m = u_0/u_j = \sqrt{T_0/T_j}M_0/M_j = 0.459$ . According to Ref. 26, these values are within the range ( $0 \leq m \leq 0.6$ ,  $0.3 \leq \bar{\rho} \leq 1.43$ ), for which the correction due to compressibility for the spreading rate  $b$  of the jet can be taken into account by calculating it as

$$b_{comp} = cx \frac{1 + \bar{\rho}}{2} \frac{1 - m}{1 + \bar{\rho}m} \quad (7a)$$

instead of the old expression

$$b_{inc} = cx \frac{1 - m}{1 + m}, \quad (7b)$$

which is relevant for the incompressible flow;  $c$  in formulae (7) is a constant and  $x$  is the distance from the source.

According to the measurements referenced by Schlichting,<sup>32</sup> for a submerged incompressible jet  $b_{1/2} = 0.0848x$ , where  $b_{1/2}$  is half width of the jet at half depth. Substituting this into the solution for laminar jet:<sup>26,32</sup>

$$\frac{b_{1/2}}{x} = 5.27 \frac{\nu}{\sqrt{K}},$$

one obtains the virtual kinematic viscosity:<sup>32</sup>

$$\nu_T = 0.0161\sqrt{K}, \quad (8)$$

here  $K$  is the total kinematic momentum flux. Since the velocity profiles are universal, for the jet with coflow we only need to multiply the spreading rate by  $(1 - m)/(1 + m)$  according to formula (7b) and for the compressibility correction we use (7a), which altogether yields:

$$\nu_T = 0.00636\sqrt{K}. \quad (9)$$

As has been mentioned, the boundary condition that we specify for the jet inflow is a uniform supersonic profile across the entire nozzle outlet. Therefore, the quantity  $K$  can be obtained by multiplying the square of the excess velocity (relative velocity of the jet with respect to the velocity of coflow) by the area of the outlet  $\sigma$ ,  $K = (u_j - u_0)^2 \sigma$ . Then, the effective turbulent Reynolds number is calculated as  $Re_T = UL/\nu_T$ , where  $U$  is the characteristic speed and  $L$

is the characteristic length. For the particular setting under study, it is reasonable to assume that  $U = |u_j - u_0|$  and  $L = \sqrt{\sigma}$ . Consequently, from (9) we conclude that

$$Re_T = 0.00636^{-1} \approx 157. \quad (10)$$

In our computations, the actual value of  $Re$  for the linearized thin-layer system was taken from (10).

## Numerical Results

The particular geometry of the body shown in Figure 2 is the following: rectangular cross section  $y \times z = 6.2 \times 6.8$  with rounded edges; sharp nose and boat-tail afterbody; total length in the  $x$  direction is 63; rectangular nozzle outlet  $y \times z = 2.62 \times 5.12$ ; full description can be found in the work by Compton.<sup>27</sup>

The geometry and the flow are symmetric with respect to the plane  $z = 0$  (zero angle of attack). For our computations we have used three different domains with successively reduced dimensions, see Figure 4; domain I (or large domain) with the diameter of about 30 calibers of the body was used for calculating the reference solutions, domain II is 0.36 or about 1/3 of the size of domain I in each direction and domain III is 0.22 or about 1/5 of the size of domain I in each direction.

As has been mentioned, to integrate the thin-layer equations on the curvilinear grid shown on Figure 2 we use the code TLNS3D by Vatsa, et al.<sup>23,24</sup> This is a central-difference code with five stage explicit pseudo-time Runge-Kutta relaxation used for obtaining steady-state solutions. The code employs local Courant step, semi-implicit residual smoothing, and multigrid for accelerating the convergence. In our computations, we used either three or two nested grid levels with V cycles (depending on the grid dimension); this multi-level V-cycle algorithm is, in fact, a final stage of the full multigrid (FMG) procedure. In addition, to improve the convergence to steady state, the solver is preconditioned according to the methodology of Ref. 33.

The DPM-based ABCs are implemented only on the finest grid for the V-cycle in the final FMG stage; the boundary data for coarser levels are provided by the coarsening procedure. Moreover, even on this finest grid we implement the DPM-based ABCs only on the first and the last Runge-Kutta stages, which has been found<sup>15,16</sup> to make very little difference compared to the implementation on all five stages; the boundary data for the three intermediate stages are provided from the DPM-based ABCs on the first stage.

To account for the turbulent phenomena, the solver is also supplemented with Menter's two-equation turbulence model.<sup>34</sup> The actual molecular Reynolds number based on unit

length is  $Re = 321000$ , Prandtl number is  $Pr = 0.72$ , specific ratio is  $\kappa = 1.4$ .

We have used several different grids to calculate the jet flow; in all cases we kept the normal spacing near the solid surface the same:  $\approx 3 \cdot 10^{-4}$ . All grids are stretched, the cell size increases away of the body in geometric progression. The dimension of the C-O grid block 1 for domain I was  $i \times j \times k = 385 \times 77 \times 33$  ( $i$  is the streamwise C-type coordinate,  $j$  is the radial coordinate, and  $k$  is the circumferential cross-stream O-type coordinate, quarter circle). The dimension of the H-O grid block 2 for domain I was  $i \times j \times k = 81 \times 77 \times 65$  ( $i$  is streamwise,  $j$  is radial, and  $k$  covers half circle). We will further refer to this grid as *fine*. On the fine grid, we have calculated two reference solutions, one with standard ABCs and another — with global ABCs. As the artificial boundary for domain I is located sufficiently far away of the body, the difference between the corresponding results is negligible. In Figures 5 and 6 we show convergence histories for this case: residual of the continuity equation is plotted vs. work units in Figure 5 and drag coefficient is plotted vs. work units in Figure 6. (One work unit is the cost of advancing one time step on the finest grid.)

From Figures 5 and 6 we conclude that multigrid convergence rates are the same for local and global ABCs on domain I. Moreover, the values of total drag coefficient per unit area  $C_D$  are summarized for this case in the right column of Table 1. They differ by about one third of one per cent, which corroborates that the type of external boundary conditions has little effect on the solution itself, as well as multigrid convergence history, for large computational domains.

For domain II, we have computed the flow on two grids with different dimensions. The first grid has the same number of nodes as the one used in domain I; it was, in fact, constructed by scaling down the original large grid by a factor of 0.36 in each direction. We will also refer to it as *fine* grid. As shown in Table 1, the coefficient  $C_D$  obtained on this grid with global ABCs differs by less than one per cent from its reference value, whereas the accuracy provided by local ABCs is not nearly as good, about 9% discrepancy; moreover, because of the poor convergence (see Figures 7 and 8) the value of  $C_D$  for local ABCs is given with the error bands indicated.

The much smaller size of domain II compared to domain I actually suggests that on domain II one can successfully compute the solution on a grid with fewer nodes. Therefore, the second grid that we have used for domain II had one half of the original dimension in two out of three directions, block 1  $i \times j \times k = 193 \times 39 \times 33$  and block 2  $i \times j \times k = 41 \times 39 \times 65$ , this grid will be referred to as *coarse*. Again, as follows from Table 1, global ABCs provide for an accurate solution whereas the accuracy of local ABCs is not sufficient and the convergence is slow (or even non-existent). Convergence histories for domain II are presented on Figures 7

and 8.

Since the node count for the coarse grid is only 1/4 of the node count for the fine grid, the convergence vs. work for the coarse grid is about four times faster (see Figures 7 and 8), although convergence rates measured vs. number of multigrid cycles will be approximately the same for both grids. Note that because of the particular grid dimensions (the issue of divisibility by 2) we have used three nested multigrid levels on the fine grid and two levels on the coarse grid. One can clearly see from Figures 7 and 8 that the DPM-based ABCs provide for a noticeably higher multigrid convergence rate than the standard local ABCs do. Moreover, it is, in fact, hard to conclude from Figures 7 and 8 whether or not the algorithm with local ABCs converges. If it does, the resulting  $C_D$  will be about 10% off its reference value.

On domain III, the computations were performed on the fine grid, which again was obtained by scaling down the grid from domain I (a factor of 0.22 in each direction). The algorithm with local ABCs for this domain/grid failed to converge, whereas the algorithm with global ABCs converged with the same rate as before. However, the actual computed  $C_D$  is about 5% off its reference value (see Table 1). This can apparently be attributed to the fact that the assumption of linearity (small perturbations) outside the computational domain is violated for such a small domain size. Convergence histories for domain III are presented on Figures 9 and 10.

Computations on a coarse grid for domain III were not performed because we did not expect to recover the accurate value of  $C_D$ . However, the fact that the algorithm with global ABCs converges on domain III corroborates the high robustness of this procedure.

Basically, the computational results presented above already allow us to see that the performance of the global DPM-based ABCs is superior to that of the standard local boundary conditions. This conclusion is supported by the faster multigrid convergence rates, as well as better accuracy of the calculated drag coefficient  $C_D$  on small domains, that we obtained through the use of the DPM-based ABCs. In terms of accuracy, we corroborate this conclusion even further by presenting a more detailed account of the flow characteristics that pertain to the computed solutions (in addition to the values of the overall drag coefficient).

In Figures 11 and 12, we present the distribution of the pressure coefficient

$$C_p = \frac{p - p_0}{\frac{1}{2}\rho_0 u_0^2},$$

where  $p_0$ ,  $\rho_0$ , and  $u_0$ , are static pressure, density, and absolute velocity of the flow at infinity, respectively, in the vertical plane of symmetry  $z = 0$  on the afterbody portion of the analyzed configuration (see Figure 2). Figure 11 shows a longer stretch for the streamwise coordinate

$x$  normalized by the total length of the body  $L$ :  $0.84 \leq x/L \leq 1.0$ ; and Figure 12 presents a “zoomed in” view of the same distribution on a shorter stretch  $0.88 \leq x/L \leq 0.94$ . Both Figures summarize the distribution of  $C_p$  on all three computational domains on fine grids.

From Figures 11 and 12 we see that on the large domain I the solutions obtained with global and local ABCs indeed deviate from one another very slightly. The solution obtained with global ABCs on the intermediate domain II practically does not deviate from the reference solution (domain I) as well. As opposed to global ABCs, the solution obtained with local ABCs on the intermediate domain II differs from the reference solution (domain I) much more noticeably. On the small domain III, the solution with local ABCs could not be computed at all, and the solution with global ABCs differs from the reference solution even less than the solution with local ABCs from domain II does. Similar behavior of the pressure coefficient  $C_p$  can be observed in the horizontal plane of symmetry  $y = 0$  on the afterbody portion of the analyzed configuration as well, see Figures 13 and 14.

Let us also note that we have picked these particular  $C_p$  distributions (see Figures 11, 12, 13, and 14) as the examples of more detailed calculated flow characteristics (as opposed to presenting only integral force coefficients) because it is known that the  $C_p$  profiles on the afterbody are rather sensitive to the type and parameters of the numerical algorithm. Besides, these particular computed profiles have also been chosen by Compton for presenting in Ref. 27; and the results of our computations display a good correlation with the results by Compton that have been obtained on a very large computational domain.

All computations described in this section were conducted on Cray Research computers, J90 and C90 series. Computational overhead due to the use of global ABCs is about 15% for the particular fine grid referenced before. This overhead is determined mostly by domain geometry and typically does not scale linearly with the dimension of the interior grid. For the aforementioned coarse grid the overhead reaches 30%.

## Conclusions

We have constructed and implemented global ABCs for calculating external flows with jet exhaust. The ABCs combine extrapolation of all flow quantities downstream in the supersonic core of the jet and nonlocal DPM-based treatment for the remaining portion of outer boundary. The overhead associated with implementation of the new technique is compensated for by the reduced grid dimension on small domains and higher convergence rate. In the series of computations performed, the DPM-based algorithm have consistently demonstrated better accuracy, faster multigrid convergence, and higher robustness compared to the standard local methodology.

## Acknowledgments

We are most grateful to Dr. E. B. Parlette of Vigyan, Inc., for his valuable help in grid generation. We would like to thank the reviewer of the paper for his/her helpful suggestions.

## References

- <sup>1</sup>Givoli, D., “Non-reflecting Boundary Conditions,” *Journal of Computational Physics*, Vol. 94, No. 1, 1991, pp. 1–29.
- <sup>2</sup>Givoli, D., *Numerical Methods for Problems in Infinite Domains*, Elsevier, Amsterdam, 1992.
- <sup>3</sup>Tsynkov, S. V., “Numerical Solution of Problems on Unbounded Domains. A Review,” *Applied Numerical Mathematics*, Vol. 27, No. 4, 1998, pp. 465–532.
- <sup>4</sup>Ryaben’kii, V. S., “Boundary Equations with Projections,” *Russian Mathematical Surveys*, Vol. 40, No. 2, 1985, pp. 147–183.
- <sup>5</sup>Ryaben’kii, V. S., *Difference Potentials Method for Some Problems of Continuous Media Mechanics*, Nauka, Moscow, 1987 (in Russian).
- <sup>6</sup>Ryaben’kii, V. S., “Difference Potentials Method and its Applications,” *Mathematische Nachrichten*, Vol. 177, Feb. 1996, pp. 251–264.
- <sup>7</sup>Ryaben’kii, V. S. and Tsynkov, S. V., “Artificial Boundary Conditions for the Numerical Solution of External Viscous Flow Problems,” *SIAM Journal on Numerical Analysis*, Vol. 32, No. 5, 1995, pp. 1355–1389.
- <sup>8</sup>Tsynkov, S. V., “An Application of Nonlocal External Conditions to Viscous Flow Computations,” *Journal of Computational Physics*, Vol. 116, No. 2, 1995, pp. 212–225.
- <sup>9</sup>Tsynkov, S. V., Turkel, E., and Abarbanel, S., “External Flow Computations Using Global Boundary Conditions,” *AIAA Journal*, Vol. 34, No. 4, 1996, pp. 700–706; also AIAA Paper 95-0564, Jan. 1995.
- <sup>10</sup>Ryaben’kii, V. S. and Tsynkov, S. V., “An Effective Numerical Technique for Solving a Special Class of Ordinary Difference Equations,” *Applied Numerical Mathematics*, Vol. 18, No. 4, 1995, pp. 489–501.
- <sup>11</sup>Tsynkov, S. V., “Artificial Boundary Conditions for Computation of Oscillating External Flows,” *SIAM Journal on Scientific Computing*, Vol. 18, No. 6, 1997, pp. 1612–1656.
- <sup>12</sup>Ryaben’kii, V. S. and Tsynkov, S. V., “An Application of the Difference Potentials Method to Solving External Problems in CFD,” *Computational Fluid Dynamics Review 1998, Vol. 1*, M. Hafez and K. Oshima (eds.), World Scientific, Singapore, 1998, pp. 169–205.

<sup>13</sup>Tsynkov, S. V., “Nonlocal Artificial Boundary Conditions for Computation of External Viscous Flows,” *Computational Fluid Dynamics’96. Proceedings of the Third ECCOMAS CFD Conference*, P. Le Tallec and J. Périaux (eds.), John Wiley & Sons, New York, 1996, pp. 512–518.

<sup>14</sup>Tsynkov, S. V., “Artificial Boundary Conditions for Infinite-Domain Problems,” *Barriers and Challenges in Computational Fluid Dynamics*, V. Venkatakrishnan, M. D. Salas, and S. R. Chakravarthy (eds.), Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998, pp. 119–138.

<sup>15</sup>Tsynkov, S. V. and Vatsa, V. N., “Improved Treatment of External Boundary Conditions for Three-Dimensional Flow Computations,” *AIAA Journal*, Vol. 36, No. 11, 1998, pp. 1998–2004; also AIAA Paper 97-2074, June 1997.

<sup>16</sup>Tsynkov, S. V., “External Boundary Conditions for Three-Dimensional Problems of Computational Aerodynamics,” *SIAM Journal on Scientific Computing*, Vol. 21, No. 1, 1999, pp. 166–206.

<sup>17</sup>Tsynkov, S. V., “On the Combined Implementation of Global Boundary Conditions with Central-Difference Multigrid Flow Solvers,” in *Proceedings of IUTAM Symposium on Computational Methods for Unbounded Domains*, July 27 – 31, 1997, University of Colorado at Boulder, T. L. Geers, (ed.), Kluwer Academic Publishers, Dordrecht, 1998, pp. 285–294.

<sup>18</sup>Calderon, A. P., “Boundary-Value Problems for Elliptic Equations,” in *Proceedings of the Soviet-American Conference on Partial Differential Equations at Novosibirsk*, Fizmatgiz, Moscow, 1963, pp. 303–304.

<sup>19</sup>Seeley, R. T., “Singular Integrals and Boundary Value Problems,” *American Journal of Mathematics*, Vol. 88, No. 4, 1966, pp. 781–809.

<sup>20</sup>Swanson, R. C. and Turkel, E., “A Multistage Time-Stepping Scheme for the Navier-Stokes Equations,” AIAA Paper 85-0035, Jan. 1985.

<sup>21</sup>Swanson, R. C. and Turkel, E., “Artificial Dissipation and Central Difference Schemes for the Euler and Navier-Stokes Equations,” AIAA paper 87-1107, June 1987.

<sup>22</sup>Swanson, R. C. and Turkel, E. “Multistage Schemes with Multigrid for the Euler and Navier-Stokes Equations. Volume I: Components and Analysis,” NASA TP-3631, Aug. 1997.

<sup>23</sup>Vatsa, V. N. and Wedan, B., “Development of a Multigrid Code for 3-D Navier-Stokes Equations and Its Application to a Grid-Refinement Study,” *Computers and Fluids*, Vol. 18, 1990, pp. 391–403.

<sup>24</sup>Vatsa, V. N., Sanetrik, M. D., and Parlette, E. B., “Development of a Flexible and Efficient Multigrid-Based Multiblock Flow Solver,” AIAA Paper 93-0677, Jan. 1993.

<sup>25</sup>Thomas, J. L. and Salas, M. D., “Far-Field Boundary Conditions for Transonic Lifting Solutions to the Euler Equations,” AIAA Paper No. 85-0020, Jan. 1985.

<sup>26</sup>Abramovich, G. N., “Applied Gas Dynamics,” 5th edition, Nauka, Moscow, 1991. Chapter 7. (in Russian)

<sup>27</sup>Compton, W. B., “Comparison of Turbulence Models for Nozzle-Afterbody Flows With Propulsive Jets,” NASA TP-3592, Sept. 1996.

<sup>28</sup>Nordström, J., “Accurate Solutions of the Navier-Stokes Equations Despite Unknown Outflow Boundary Data,” *Journal of Computational Physics*, Vol. 120, No. 2, 1995, pp. 184–205.

<sup>29</sup>Nordström, J., “On Extrapolation Procedures at Artificial Outflow Boundaries for the Time-Dependent Navier-Stokes Equations, Applied Numerical Mathematics,” Vol. 23, No. 4, 1997, pp. 457–468.

<sup>30</sup>Hagstrom, T. and Nordström, J., “The Analysis of Extrapolation Boundary Conditions for the Linearized Euler Equations,” in preparation.

<sup>31</sup>Lindberg N., Efraimsson, G., and Nordström, J., “Numerical Investigation of Extrapolation Boundary Conditions for the Euler Equations,” FFA TN 1998-13, Bromma, Sweden, 1998.

<sup>32</sup>Schlichting, H., “Boundary-Layer Theory,” McGraw-Hill, New York, 1979. Chapter 23.

<sup>33</sup>Turkel, E., Vatsa, V. N., and Radespiel, R., “Preconditioning Methods for Low-Speed Flows,” AIAA Paper 96-2460, June 1996.

<sup>34</sup>Menter, F. R., “Performance of Popular Turbulence Models for Attached and Separated Adverse Pressure Gradient Flows,” AIAA Paper 91-1784, June 1991.

## List of Table Captions

**Table 1:** Total drag coefficient per unit area  $C_D$ .

Domain geometry	III	II		I
Grid dimension	fine	coarse	fine	fine
$C_D$ for local ABCs	—	$2.77 \pm .03$	$2.74 \pm .04$	2.506
$C_D$ for global ABCs	2.365	2.495	2.484	2.497

## List of Figure Captions

**Figure 1:** Schematic geometric setup for the ONERA M6 wing; wing on the left is enlarged.

**Figure 2:** Schematic geometric setup for the slender body with jet exhaust.

**Figure 3:** The domain  $\Omega_L^2$  where  $L$  is the distance from  $\Omega_L^2$  to the boundary.  $L \gg 1$ .

**Figure 4:** Three computational domains for the jet flow, projection onto the  $z = 0$  plane.

**Figure 5:** Convergence histories for the residual of the continuity equation, domain I, fine grid.

**Figure 6:** Drag convergence, domain I, fine grid.

**Figure 7:** Convergence histories for the residual of the continuity equation, domain II, fine and coarse grids.

**Figure 8:** Drag convergence, domain II, fine and coarse grids.

**Figure 9:** Convergence histories for the residual of the continuity equation, domain III, fine grid.

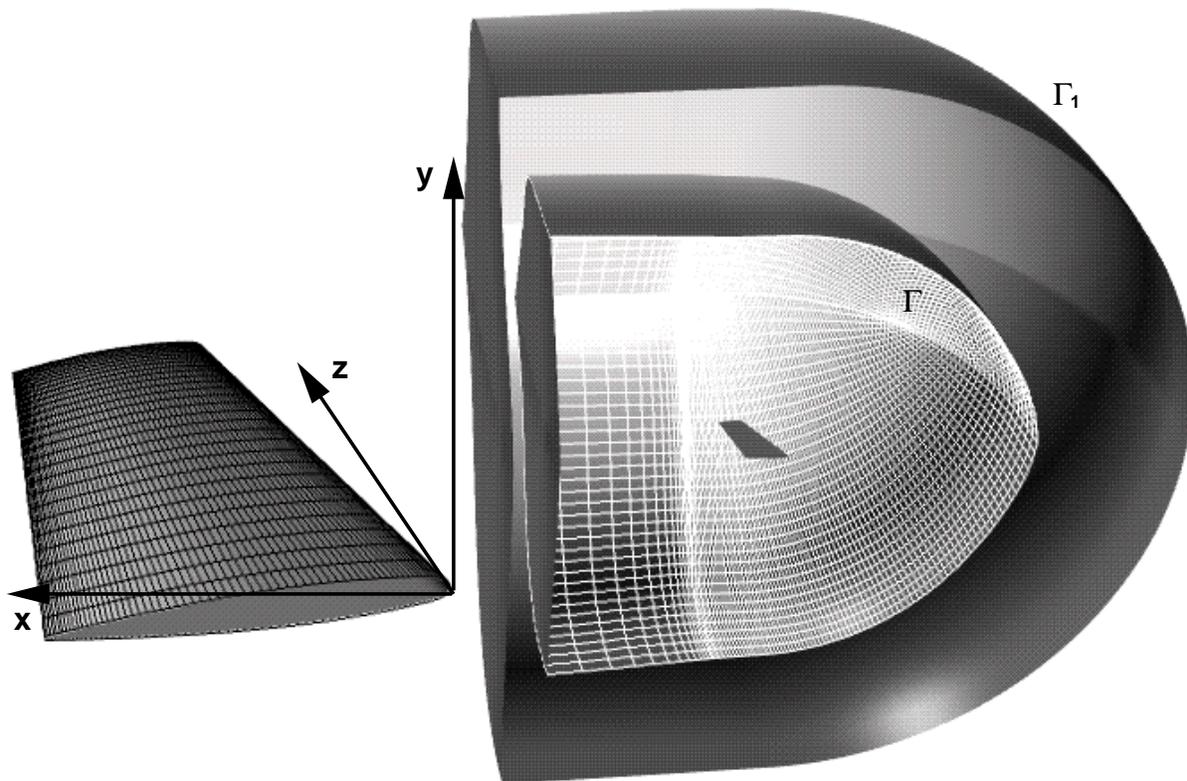
**Figure 10:** Drag convergence, domain III, fine grid.

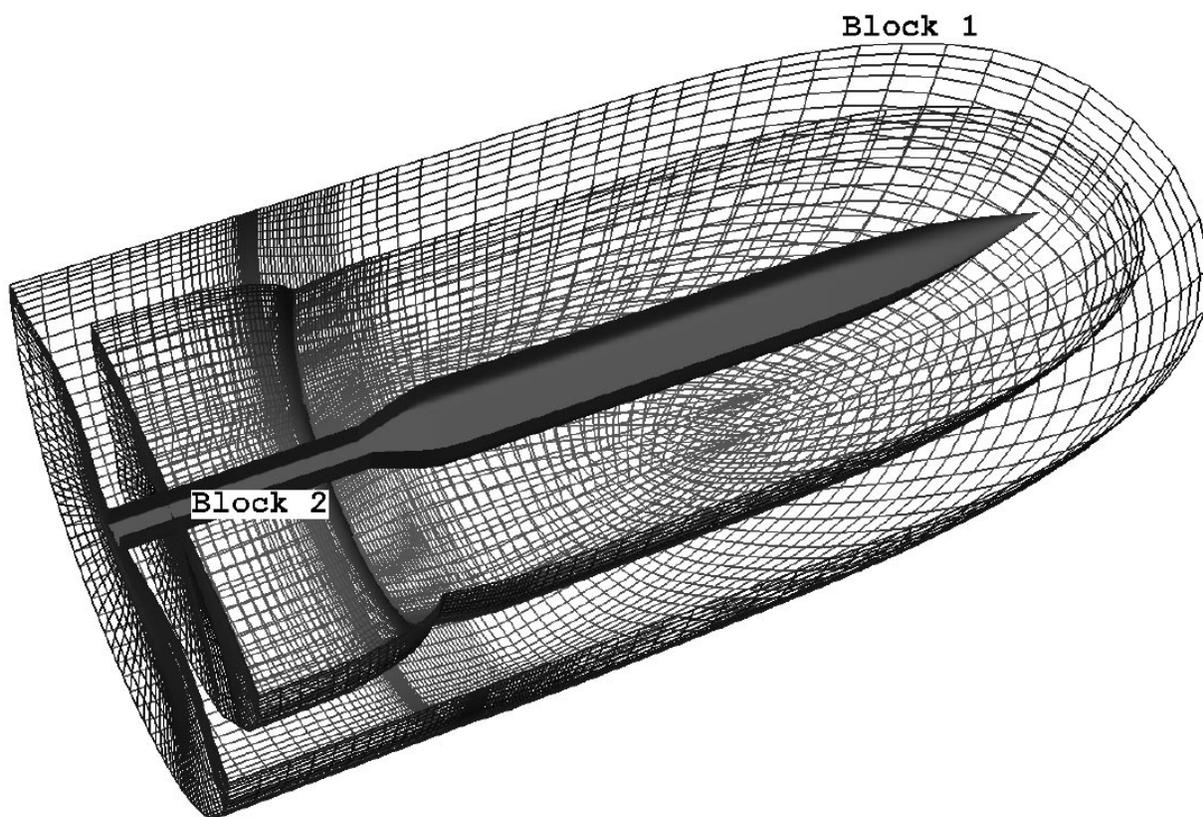
**Figure 11:** Distribution of the pressure coefficient  $C_p$  on the afterbody in the symmetry plane  $z = 0$ .

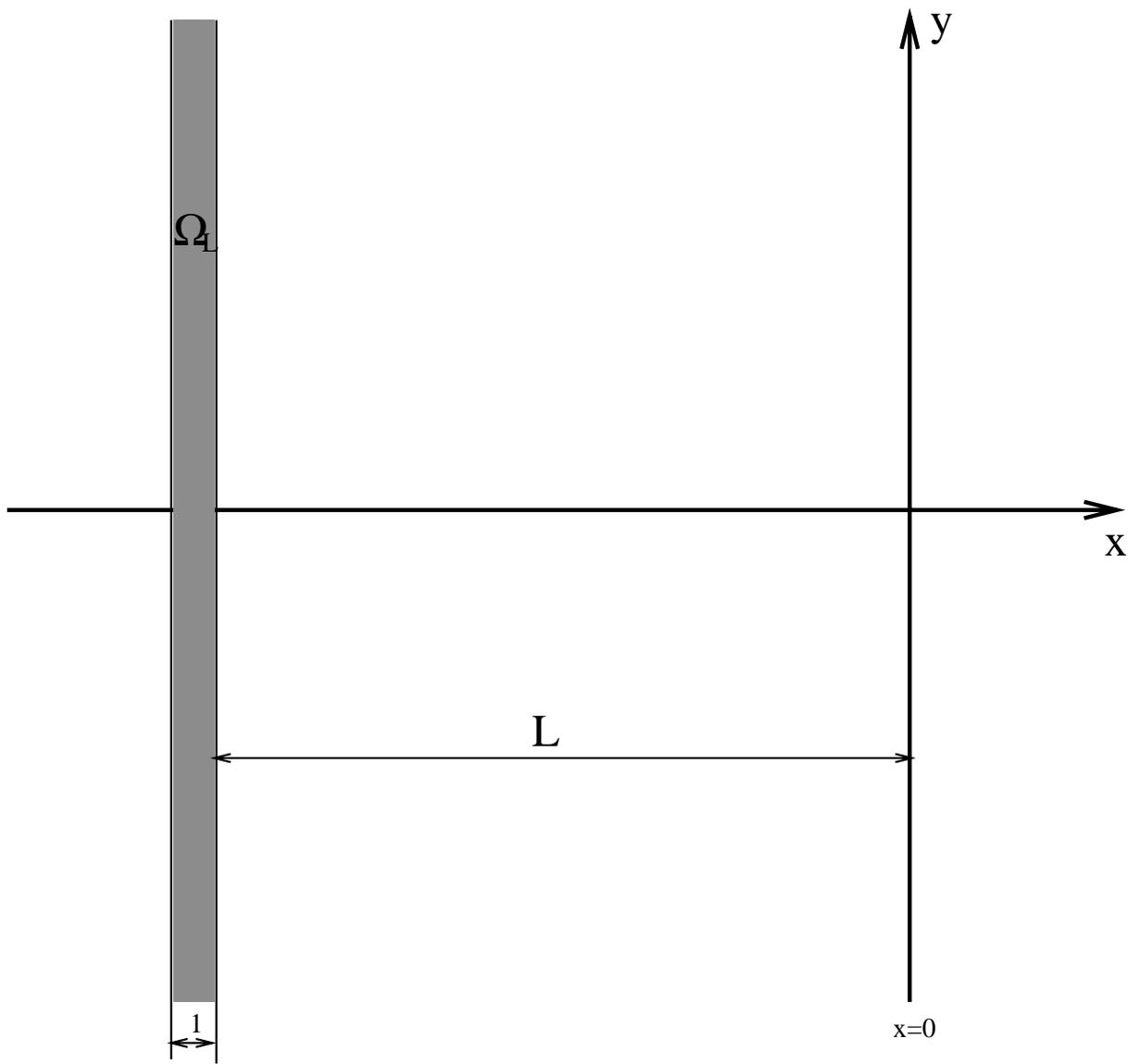
**Figure 12:** Distribution of the pressure coefficient  $C_p$  on the afterbody in the symmetry plane  $z = 0$ . Zoomed in view.

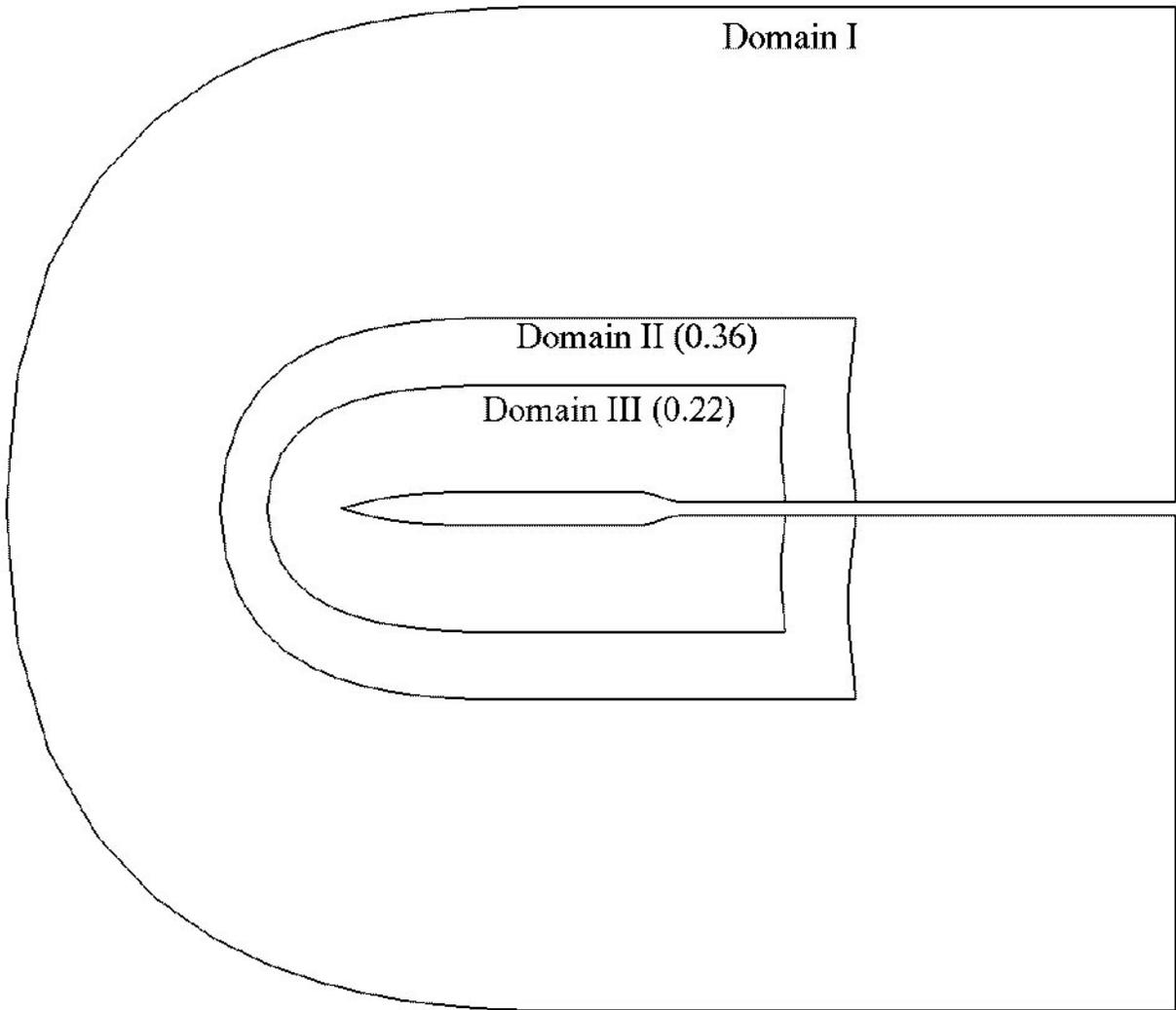
**Figure 13:** Distribution of the pressure coefficient  $C_p$  on the afterbody in the symmetry plane  $y = 0$ .

**Figure 14:** Distribution of the pressure coefficient  $C_p$  on the afterbody in the symmetry plane  $y = 0$ . Zoomed in view.

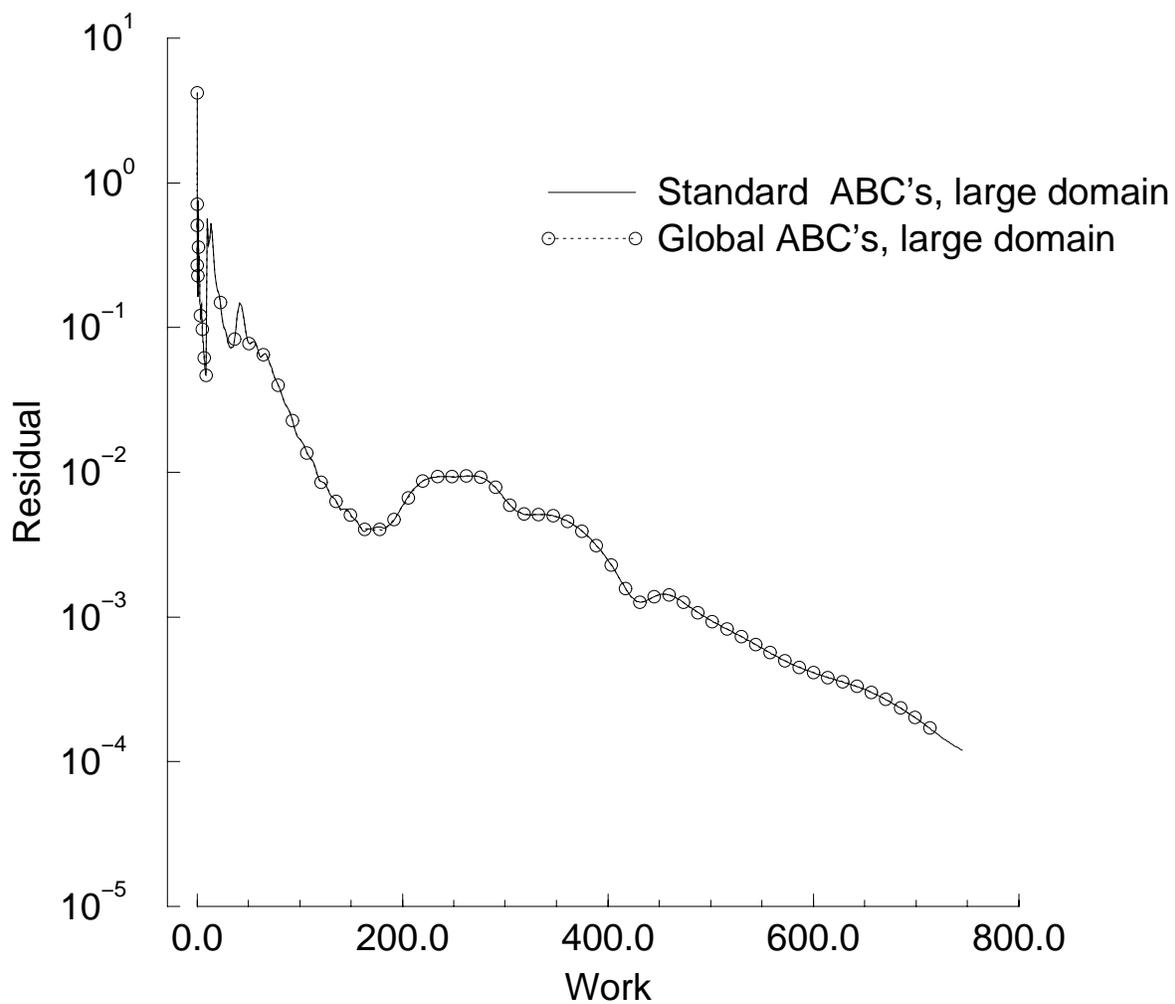




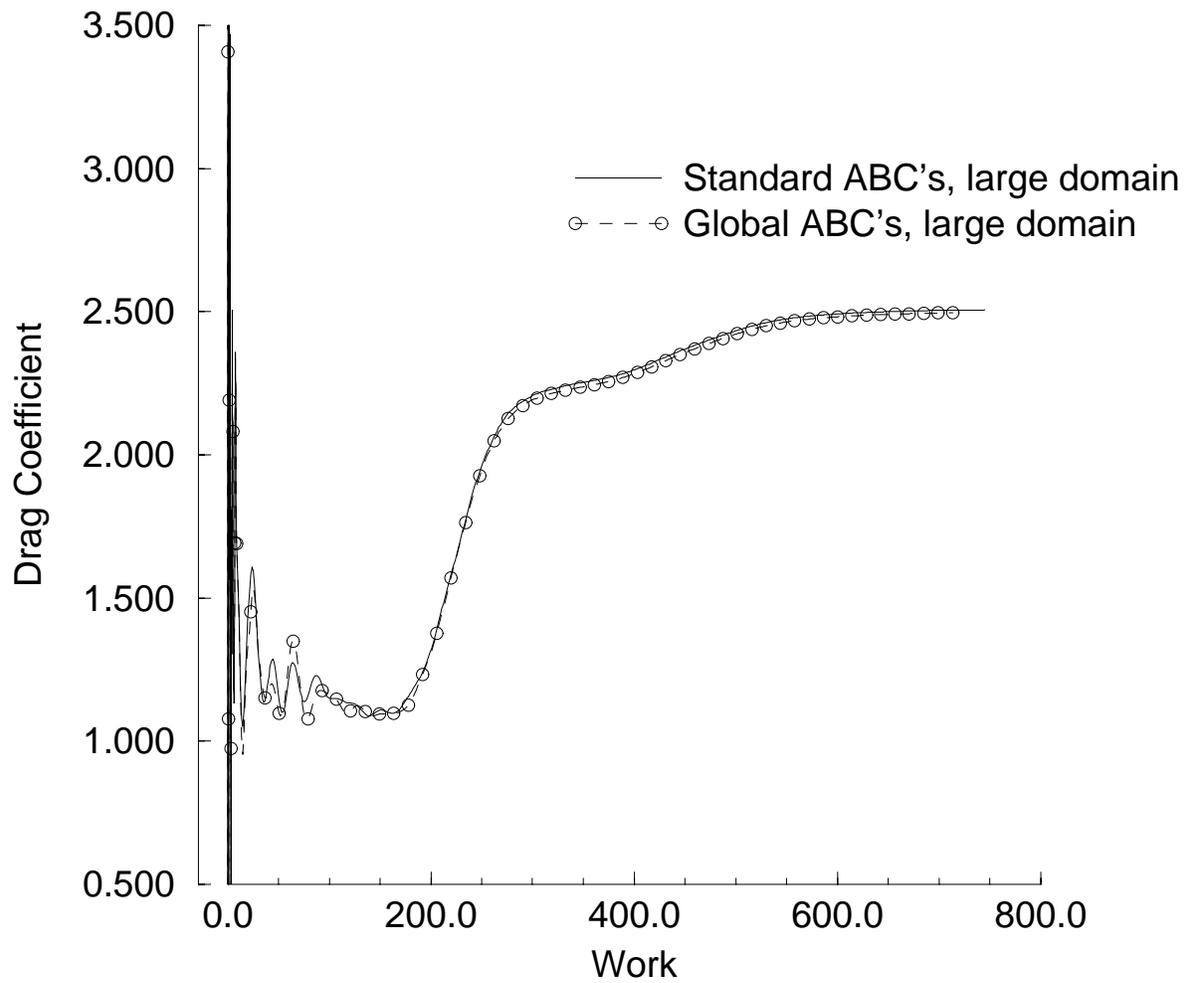




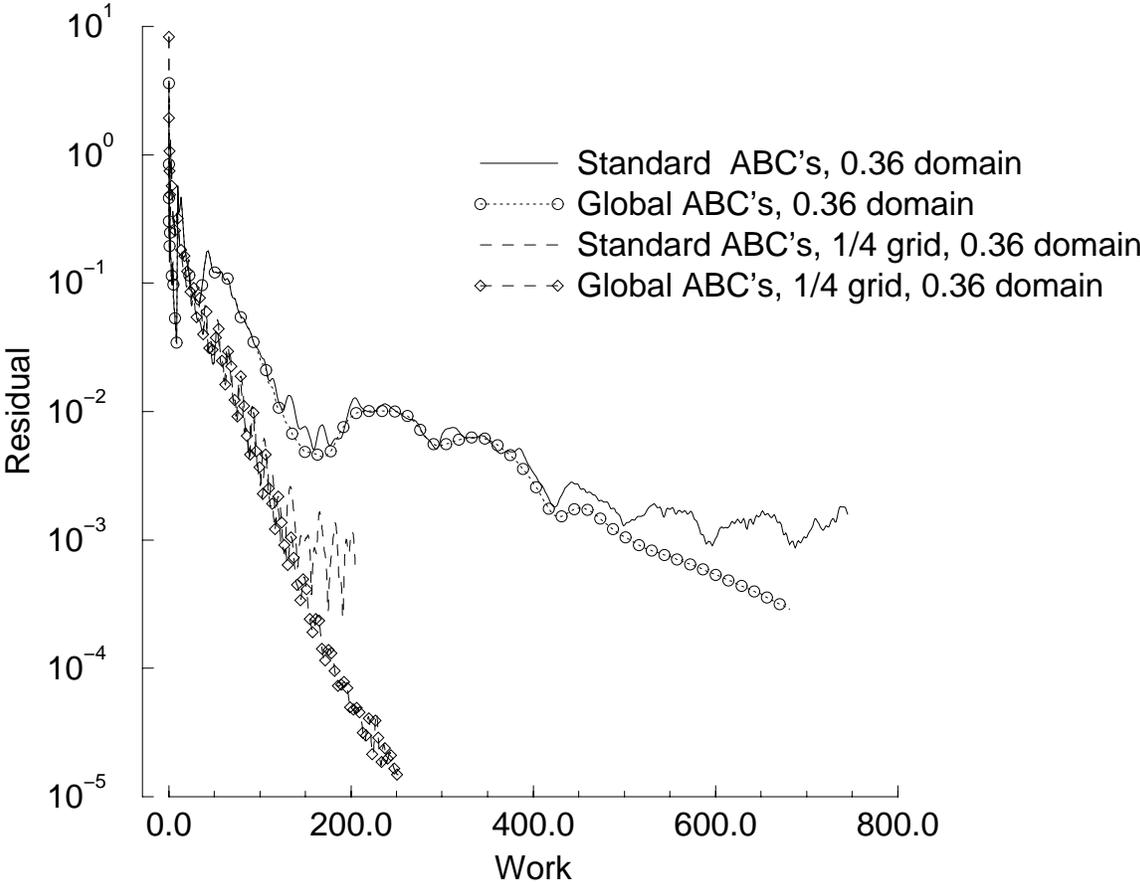
## Convergence histories for the jet flow



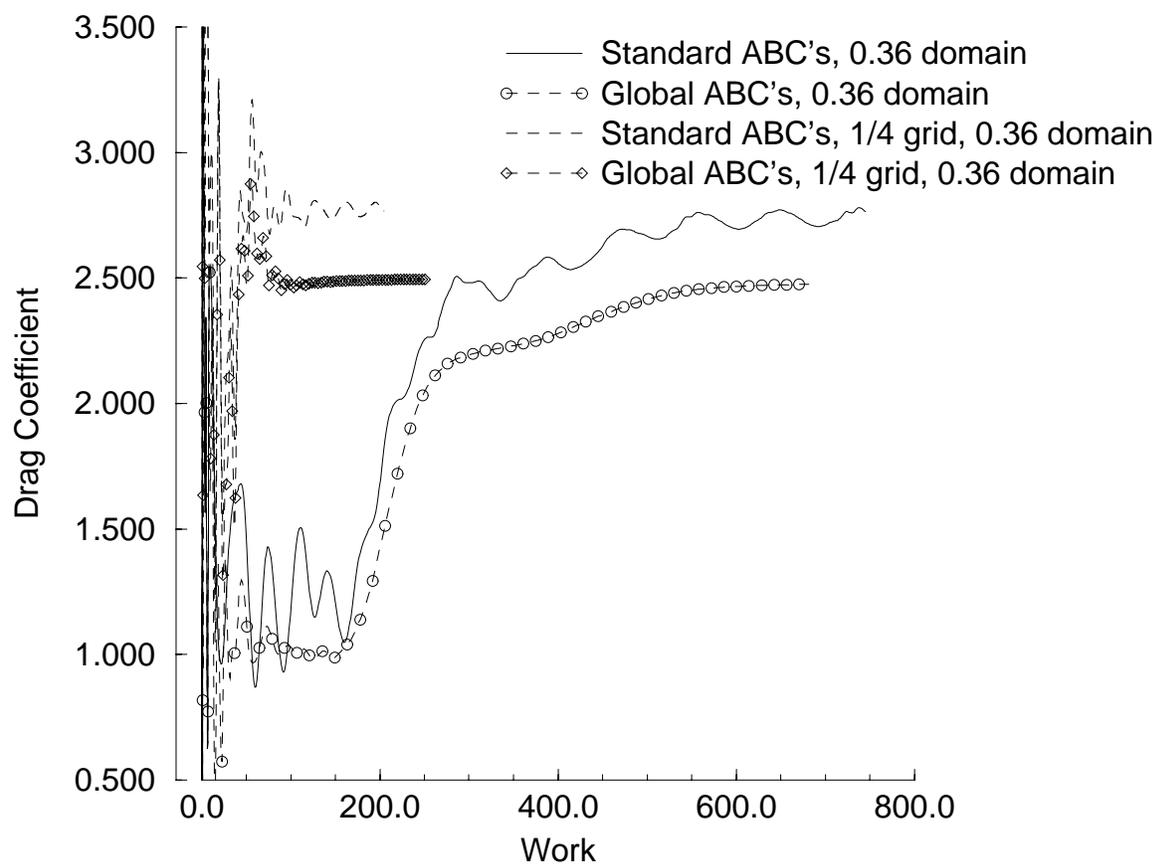
## Drag convergence for the jet flow



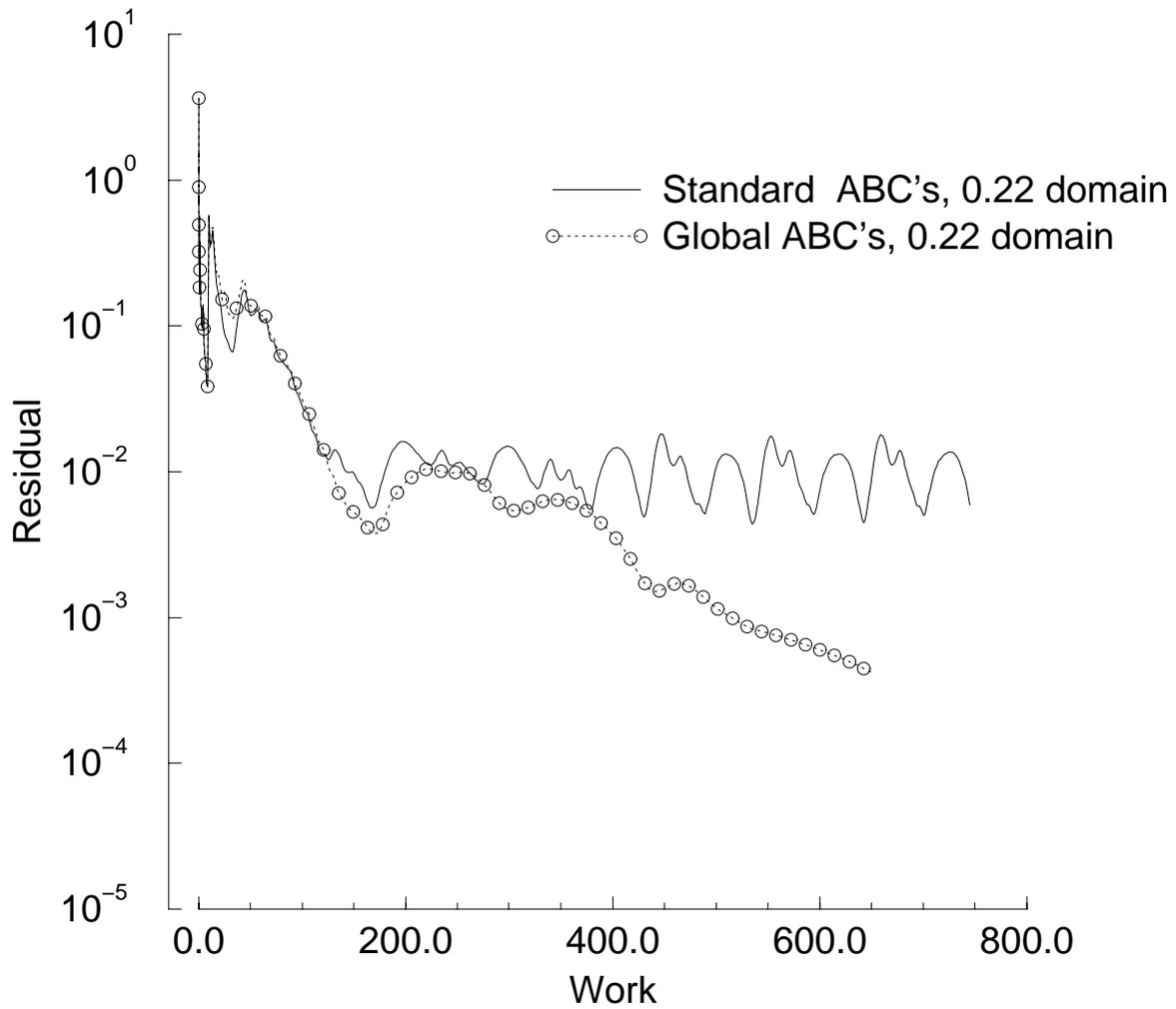
# Convergence histories for the jet flow



## Drag convergence for the jet flow



## Convergence histories for the jet flow



## Drag convergence for the jet flow

