

An Improved Treatment of External Boundary Conditions for Three-Dimensional Flow Computations*

Semyon V. Tsynkov[†] Veer N. Vatsa[‡]

Abstract

We present an innovative numerical approach for setting highly accurate nonlocal boundary conditions at the external computational boundaries when calculating three-dimensional compressible viscous flows over finite bodies. The approach is based on application of the special boundary operators analogous to Calderon's projections and the difference potentials method by Ryaben'kii; it extends our previous technique developed for the two-dimensional case. The new boundary conditions methodology has been successfully combined with the NASA-developed code TLNS3D and used for the analysis of wing-shaped configurations in subsonic and transonic flow regimes. As demonstrated by the computational experiments, the improved external boundary conditions allow one to greatly reduce the size of the computational domain while still maintaining high accuracy of the numerical solution. Moreover, they may provide for a noticeable speedup of convergence of the multigrid iterations.

1 Introduction

Preliminaries. External flows over finite bodies or configurations of bodies represent a wide class of important practical applications in fluid dynamics. To treat this type of problems numerically, one typically truncates the original infinite domain. The resulting truncated problem is obviously subdefinite unless supplemented by the proper closing procedure at the external computational boundary. The latter procedure is called the artificial boundary conditions (ABC's). In the ideal case, the ABC's would be specified so that the solution on the truncated domain coincides with the corresponding fragment of the original infinite-domain solution. The issue of ABC's is significant in CFD and in other areas of scientific computing; theoretical estimates and computational experiments by different authors

*Originally published in the AIAA Journal, Vol. 36, No. 11, 1998, pp. 1998–2004; and also: AIAA Paper No. 97–2074, June 1997. The work was conducted when the first author held a National Research Council Resident Research Associateship at NASA Langley Research Center.

[†]Corresponding author. School of Mathematical Sciences, Tel Aviv University, Ramat Aviv, Tel Aviv 69978, Israel; AND ICASE, MS 132C, NASA Langley Research Center, Hampton, VA 23681–2199, USA. E-mail: tsynkov@math.tau.ac.il, URL: <http://www.math.tau.ac.il/~tsynkov/>.

[‡]Computational Modeling and Simulation Branch, Aerodynamics, Aerothermodynamics, and Acoustics Competency, MS 128, NASA Langley Research Center, Hampton, VA 23681–2199, USA. E-mail: vatsa@tabdemo.larc.nasa.gov.

show that the proper treatment of external boundaries has a profound impact on the overall performance of numerical algorithms and interpretation of the results.

Different ABC's methodologies have been studied extensively over the recent two decades. However, the construction of the ideal (i.e., exact) ABC's that would provide no error associated with the domain truncation and at the same time be computationally inexpensive, easy to implement, and geometrically universal, still remains a fairly remote possibility. Among the variety of approaches proposed to date only a few can be regarded as the commonly used tools in CFD. As a rule, these approaches are based on essential model simplifications (e.g., locally one-dimensional treatment near the external boundary) and, therefore, often lack accuracy in computations. This, in turn, necessitates choosing excessively large computational domains. On the other hand, these simple methods usually provide for local ABC's, and, therefore, for cheap, geometrically universal, and algorithmically simple numerical procedures, which are attractive for practical use.

There are, of course, methods of another kind, which typically provide for highly accurate and robust numerical algorithms. These methods, however, are not used routinely because the corresponding ABC's in most cases appear global. As a consequence, these ABC's may be relatively cumbersome and expensive; moreover, they can be derived easily only for the boundaries of regular shape.

The nonlocality is inherent for exact ABC's; on the level of PDE's such boundary conditions typically involve pseudodifferential operators. As concerns the approximate approaches, the basic trend is the following: higher accuracy for the boundary procedure requires more of the nonlocal nature of the ABC's to be somehow taken into account. A survey of methods for setting the ABC's in different areas of scientific computing can be found in our recent work [1], as well as in the comprehensive reviews by Givoli [2, 3].

In this paper, we concentrate on constructing the ABC's for the three-dimensional problems of computational aerodynamics. We, however, mention that this area constitutes a fraction of the possible range of applications for the different ABC's techniques. Besides the hydro- and aerodynamic problems (external flows, duct flows, boundary layers, free surfaces, etc.), the entire range includes the flows in porous media, filtration, MHD flows, plasma (e.g., solar wind), the problems of solid mechanics (in particular, elasticity and aeroelasticity), and the problems of wave propagation (electromagnetic, acoustic, seismic), just to name a few.

Main Objective and Method of Approach. For external flow computations, our goal is to derive and implement the ABC's that would combine the advantages relevant to both global and local methodologies. Our approach to constructing the ABC's is based on usage of the finite-difference analogues to Calderon's generalized potentials and boundary projections (see the original work by Calderon [4] and also work by Seeley [5]). Ryaben'kii (see Refs. [6–8]) had modified the original Calderon's construction and proposed a numerical technique for the effective calculation of potentials and projections; this technique is known as the difference potentials method (DPM).

In Ref. [9] we describe the foundations of the DPM-based approach to setting the ABC's for computation of two-dimensional external viscous flows (Navier-Stokes equations). In Ref. [10] we implement this approach along with the multigrid Navier-Stokes algorithm of Swanson and Turkel [11–13] and present some numerical results for subsonic and transonic laminar flows over single-element airfoils. In Ref. [14] we show the results of subsequent

numerical experiments and propose an approximate treatment of turbulence in the far field. Our work in Ref. [15] delineates the algorithm for solving one-dimensional systems of ordinary difference equations that arise when calculating difference potentials and the DPM-based ABC's. In Ref. [16] we extend the area of applications for the DPM-based ABC's by analyzing two-dimensional flows that oscillate in time; we also provide some solvability results for the linearized thin-layer equations used for constructing the ABC's. In Ref. [17] we present a general survey of the DPM-based methodology as applied to solving external problems in CFD, including parallel implementation of the algorithm, combined implementation of nonlocal ABC's with multigrid, and entry-wise interpolation of the matrices of boundary operators with respect to the Mach number and the angle of attack. Additionally, in Ref. [17] one can find some new theoretical results on the computation of generalized potentials, the construction of ABC's based on the direct implementation of boundary projections (thin-layer equations), and some numerical results for various airfoil flows: laminar and turbulent, transonic and subsonic, including very low Mach numbers (incompressible limit). Finally, in Refs. [18, 19] we outline basic elements of the DPM-based ABC's for the case of three-dimensional steady-state external viscous flows; this case is undoubtedly the one most demanded by the current computational practice. Specifically, in Refs. [18, 19] we address the flows around wing-shaped configurations and show some preliminary numerical results for the subsonic regime.

Below, we report on the subsequent development of the DPM-based approach for the three-dimensional external flows. We present the results of computations for different flow regimes and compare the performance of the DPM-based ABC's with the performance of the standard boundary conditions. The assessment is conducted with respect to both the quality of the resulting solution (accuracy as it depends on the size of the computational domain) and the properties of the numerical procedure as influenced by the type of ABC's (efficiency, robustness, and convergence of the pseudo-time iterations). The results demonstrate the superiority of the DPM-based ABC's over the standard existing methods.

2 External Flow Problem

Formulation. We consider a steady-state flow of a viscous, compressible, perfect gas past a three-dimensional wing. The flow is uniform and subsonic at infinity; it is symmetric with respect to the Cartesian plane $z = 0$, which, in particular, implies that the free-stream velocity vector is parallel to this plane.

The flow equations are integrated on the grid generated around the wing. This grid actually defines the finite computational domain; the ABC's that would close the truncated problem should be set at the external coordinate surface of the grid. Let us designate this surface Γ_1 ; for a one-block curvilinear 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 Γ_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 Γ_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

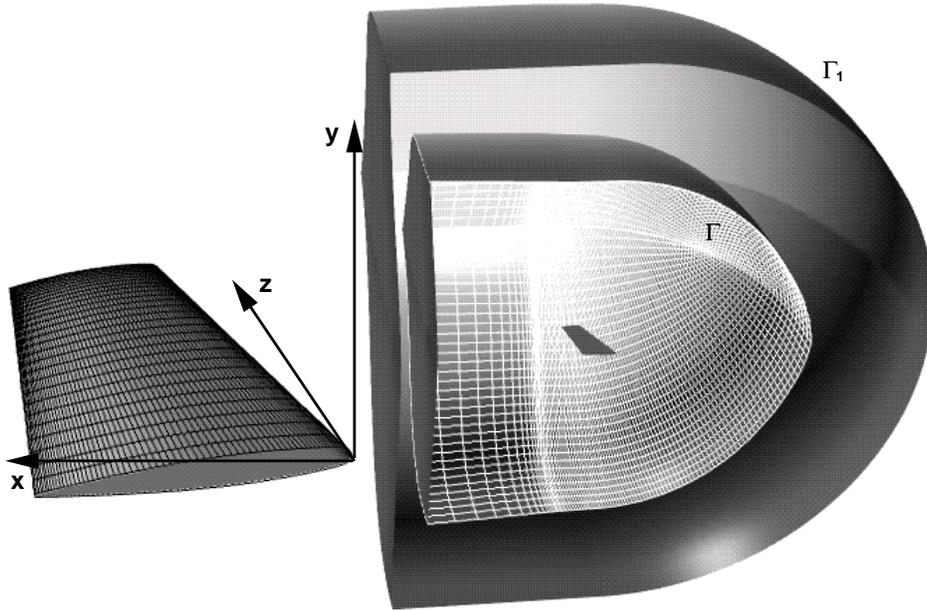


Figure 1: Schematic geometric setup for the three-dimensional case. The wing on the left is enlarged.

less equations than unknowns). From the viewpoint of what the solution technique is, one can say that when some iterative solver is employed to integrate the flow equations inside the computational domain, the values of the solution at the ghost cells should be prescribed at each iteration in order to be able to advance the next “time” step.

Therefore, in the practical framework the closure of the discretized truncated problem means the specification of the solution values at the ghost cells. This will be done by means of the DPM-based ABC’s so that the boundary data used for the closure admit an exterior complement that solves the problem outside the computational domain (see below).

First, we assume that the flow perturbations against the constant free-stream background are small in the far field and consider the accordingly simplified problem outside the computational domain (i.e., outside Ω). Clearly, for the small to moderate free-stream Mach numbers M_0 (purely subsonic flows) we can retain only the first-order terms with respect to perturbations in the governing equations. Therefore, the aforementioned simplification of the original problem will actually be the far-field linearization. However, in the transonic limit the possibility of linearization in the far field requires, generally speaking, some additional analysis. For the model full-potential formulation, we present here a simple argument that the compressible three-dimensional far field remains linear for transonic flow regimes as well.

Consider the Kármán–Guderley equation (see [20])

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial \hat{y}^2} + \frac{\partial^2 \phi}{\partial \hat{z}^2} = \frac{\kappa + 1}{K} \frac{\partial \phi}{\partial x} \frac{\partial^2 \phi}{\partial x^2}. \quad (1)$$

for the perturbation ϕ the full potential Φ of the compressible gas flow around a thin three-dimensional wing. Here

$$\begin{aligned} \frac{\Phi'_x}{u_0} &= 1 + \delta^{2/3} \phi'_x, & \frac{\Phi'_y}{u_0} &= \delta \phi'_y, & \frac{\Phi'_z}{u_0} &= \delta \phi'_z, \\ \tilde{y} &= \delta^{1/3} y, & \tilde{z} &= \delta^{1/3} z, & K &= \frac{1 - M_0^2}{\delta^{2/3}}, \\ \hat{y} &= \sqrt{K} \tilde{y}, & \hat{z} &= \sqrt{K} \tilde{z}, \end{aligned} \quad (2)$$

δ is the wing thickness ($\delta \rightarrow +0$ along with $M_0 \rightarrow 1$ in the transonic limit), K is the parameter of transonic similarity (the true linear theory corresponds to big values of K), u_0 is the flow velocity at infinity, κ is the ratio of specific heats, and the positive x direction coincides with the free stream.

The common practice [20] of developing the asymptotic expansions for solutions of the equations that involve transonic nonlinearities consists of substituting the leading linear term(s) into the nonlinear parts of the equation (right-hand side of (1)) and obtaining the corresponding corrections by solving the resulting non-homogeneous problem. In the specific case under study, the linear far-field expansion starts with the horseshoe vortex

$$\phi_1 = \frac{\hat{y}}{\hat{y}^2 + \hat{z}^2} \left(1 + \frac{x}{\hat{r}} \right), \quad \hat{r} \equiv \sqrt{x^2 + \hat{y}^2 + \hat{z}^2}. \quad (3a)$$

Substituting the expression (3a) into the right-hand side of (1) and solving the resulting Poisson equation (this involves the Fourier transform in spherical functions) one obtains the nonlinear correction

$$\phi_{1NL} \sim \hat{r}^{-3} \quad (3b)$$

that decays at infinity two orders of magnitude faster than the source term (3a). Analogously, for the general doublet term $\phi_2 \sim \hat{r}^{-2}$ the corresponding nonlinear correction can be shown to be $\phi_{2NL} \sim \hat{r}^{-5}$, which is the three orders of magnitude difference. We therefore conclude that the transonic nonlinear corrections can be neglected when analyzing the compressible far field in three space dimensions.

Of course, having established the fact of the far-field linearity, we cannot say in advance whether or not the linearization outside, is possible for every specific configuration of the domains. Clearly, for the very large computational domain one can linearize the flow outside, , and as we approach the wing (the source of perturbations), the validity of linearization can be verified a posteriori (see, e.g., our previous work [10, 14, 17]).

The linearized dimensionless thin-layer equations can be written as

$$\left. \begin{aligned} \frac{\partial \rho}{\partial x} + \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} &= 0 \\ \frac{\partial u}{\partial x} + \frac{\partial p}{\partial x} - \frac{1}{Re} \left(\frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) &= 0 \\ \frac{\partial v}{\partial x} + \frac{\partial p}{\partial y} - \frac{1}{Re} \left(\frac{4}{3} \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} + \frac{1}{3} \frac{\partial^2 w}{\partial y \partial z} \right) &= 0 \\ \frac{\partial w}{\partial x} + \frac{\partial p}{\partial z} - \frac{1}{Re} \left(\frac{4}{3} \frac{\partial^2 w}{\partial z^2} + \frac{\partial^2 w}{\partial y^2} + \frac{1}{3} \frac{\partial^2 v}{\partial y \partial z} \right) &= 0 \\ \frac{\partial p}{\partial x} - \frac{1}{M_0^2} \frac{\partial \rho}{\partial x} - \frac{\kappa}{Re Pr} \left[\left(\frac{\partial^2 p}{\partial y^2} + \frac{\partial^2 p}{\partial z^2} \right) - \frac{1}{\kappa M_0^2} \left(\frac{\partial^2 \rho}{\partial y^2} + \frac{\partial^2 \rho}{\partial z^2} \right) \right] &= 0 \end{aligned} \right\} \quad (4a)$$

where ρ , u , v , w , and p are the perturbations of density, Cartesian velocity components, and pressure, respectively, Re is the Reynolds number, and Pr is the Prandtl number. System (4a) is supplemented by the homogeneous boundary condition at infinity:

$$\begin{aligned} \mathbf{u} \equiv (\rho, u, v, w, p) &\longrightarrow (0, 0, 0, 0, 0) \\ \text{as } (x^2 + y^2 + z^2) &\longrightarrow \infty, \end{aligned} \quad (4b)$$

which corresponds to the free stream limit of the solution. Returning to the question of closing the discretized truncated system, we clarify that the boundary data provided by the DPM-based ABC's will admit an exterior complement that would solve the discrete counterpart of (4a) and meet boundary condition (4b) in a certain asymptotic sense.

DPM-based ABC's — Main Idea. We discretize system (4a) on the auxiliary Cartesian grid with second order of accuracy; we use first-order differences in x and second-order differences in y and z (see Refs. [16, 17] for the details in two dimensions). The DPM will provide us with the *complete boundary classification* (in terms of the appropriate traces) of all those and only those exterior grid vector-functions that solve the discrete counterpart of (4a) outside the computational domain and satisfy boundary condition (4b) in some approximate sense. The foregoing boundary classification will be obtained as an image of a special projection operator, which can be considered as a discrete analogue of Calderon's pseudodifferential boundary projection [4, 5]. As we solve the Navier-Stokes equations inside the computational domain iteratively, every time we need to update the ghost cells we take a certain sufficient set of data from inside Ω , (see below), project it onto the right manifold, i.e., onto the subspace in the entire space of boundary data that admit the correct exterior complement, and obtain the ghost cell values by calculating the trace of this complement on $\partial\Omega$.

DPM-based ABC's — Specific Implementation. Let us split the nodes of the auxiliary Cartesian grid into two distinct groups: those that are inside Ω , and those that are

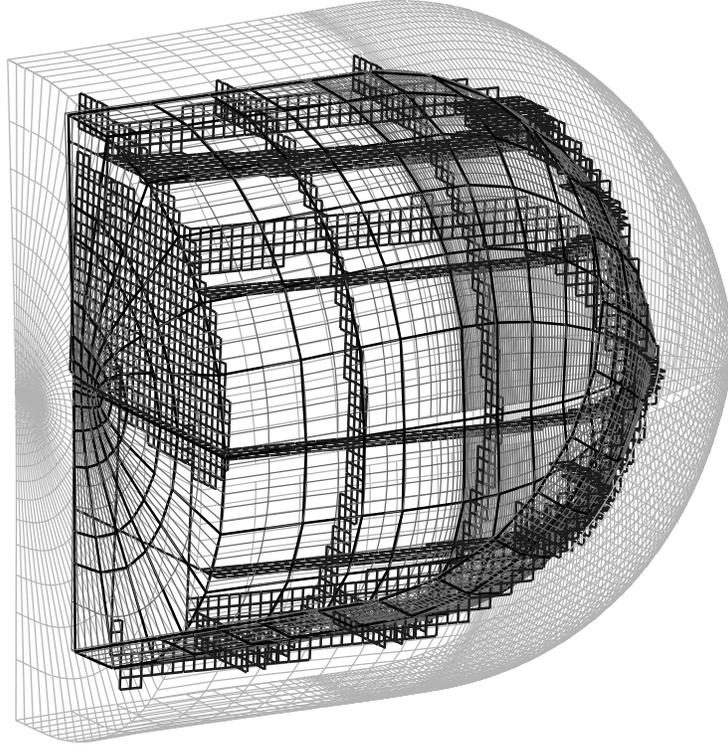


Figure 2: Grid sets for the three-dimensional case.

outside. Applying the stencil of the scheme for (4a) 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 γ ; it is a multi-layered fringe of nodes of the auxiliary Cartesian grid located near $\partial\Omega$. Figure 2 shows an example of the grid boundary γ (several cross-sections in different directions).

For any function \mathbf{u} on the Cartesian grid we define its trace $\mathbf{Tr}_\gamma \mathbf{u}$ on γ as merely a contraction. For any grid function \mathbf{u}_γ specified on γ we introduce the generalized potential $\mathbf{P} \mathbf{u}_\gamma$ with the density \mathbf{u}_γ ; the generalized potential is defined on the auxiliary Cartesian grid on γ and outside it. The generalized potential is obtained as a solution of the special auxiliary problem (AP); solution of the AP replaces and extends the operation of convolution with the fundamental solution in the classical potential theory. The AP is driven by the right-hand side that depends on \mathbf{u}_γ , the formal construction of this right-hand side is the same in two- and three-dimensional cases and we refer the reader to our previous work [9,16,17] for details. Boundary conditions for the AP should approximate boundary condition (4b) and at the same time ensure the finite-domain formulation for the AP. Therefore, we formulate the AP on a sufficiently large parallelepiped aligned with the Cartesian directions; the parallelepiped fully contains Ω . We specify periodicity boundary conditions in the y and z directions and also assume symmetry at $z = 0$. After the Fourier transform with respect to y and z the discrete counterpart to (4a) can be written as a family of one-dimensional difference equations:

$$\begin{aligned}
\mathbf{A}_{\mathbf{k}}\hat{\mathbf{u}}_{m,\mathbf{k}} + \mathbf{B}_{\mathbf{k}}\hat{\mathbf{u}}_{m-1,\mathbf{k}} &= \hat{\mathbf{f}}_{m-1/2,\mathbf{k}}, \\
m &= 1, \dots, M, \quad \mathbf{k} \equiv (k_y, k_z), \\
k_y &= 0, \dots, J_y, \quad k_z = 0, \dots, J_z,
\end{aligned} \tag{5}$$

where $\mathbf{A}_{\mathbf{k}}$ and $\mathbf{B}_{\mathbf{k}}$ are the 5×5 matrices and $M + 1$, $2J_y + 1$, and $J_z + 1$ are the numbers of grid nodes in the x , y , and z directions, respectively (symmetry is taken into account, as well as the fact that \mathbf{u} and \mathbf{f} are real-valued). Boundary conditions in the x (streamwise) direction are specified separately for each pair of wavenumbers \mathbf{k} :

$$\mathbf{S}^-(\mathbf{k})\hat{\mathbf{u}}_{0,\mathbf{k}} = \mathbf{0}, \tag{6a}$$

$$\mathbf{S}^+(\mathbf{k})\hat{\mathbf{u}}_{M,\mathbf{k}} = \mathbf{0}, \tag{6b}$$

where

$$\mathbf{S}^-(\mathbf{k}) = \prod_{|\mu_s(\mathbf{k})| > 1} (\mathbf{Q}_{\mathbf{k}} - \mu_s(\mathbf{k})\mathbf{I}), \tag{7a}$$

$$\mathbf{S}^+(\mathbf{k}) = \prod_{|\mu_s(\mathbf{k})| \leq 1} (\mathbf{Q}_{\mathbf{k}} - \mu_s(\mathbf{k})\mathbf{I}), \tag{7b}$$

$\mathbf{Q}_{\mathbf{k}} = \mathbf{A}_{\mathbf{k}}^{-1}\mathbf{B}_{\mathbf{k}}$, and $\mu_s(\mathbf{k})$ are the eigenvalues of $\mathbf{Q}_{\mathbf{k}}$. The semi-analytic boundary conditions (6a) and (6b) (the eigenvalues for (7) are calculated numerically) explicitly prohibit growing modes of the solution in the left and right directions, respectively. The periods in y and z should be chosen sufficiently large to ensure that the periodic solution considered near γ , and γ_1 is sufficiently close to the theoretical non-periodic solution; the latter can be thought of as a limit when the periods approach infinity. The approximation of a non-periodic solution by the periodic one on a finite fixed interval as the period(s) increase(s) is discussed in our work [9, 16, 17]. In Ref. [17], we also discuss the possibility of replacing the Fourier transforms by non-unitary transforms. The latter appear when the grid in y or z is stretched (which provides for a drastic cost reduction) and the corresponding eigenfunctions consequently form a skew basis.

The foregoing AP allows us to calculate the generalized difference potential $\mathbf{P}\mathbf{u}_\gamma$ for any grid density \mathbf{u}_γ specified on γ . The composition of the operators \mathbf{Tr}_γ 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 for system (4a). The image of this projection, $\text{Im}\mathbf{P}_\gamma$, contains all those and only those \mathbf{u}_γ 's that are the traces of some exterior difference solution to (4a) that satisfies the boundary conditions of the AP (periodicity in y and z and boundary conditions (6) in x). The latter boundary conditions, in turn, approximate (4b).

Having constructed the procedure for calculating the potentials and projections for the discrete version of (4a), we can now close the system inside the computational domain, i.e.,

obtain the ABC's. First, we take \mathbf{u} and $\partial\mathbf{u}/\partial n$ on Γ , n is the normal, (these data are available from inside the computational domain) and, using interpolation \mathbf{R}_Γ along Γ , and the first two terms of the Taylor expansion (denoted π_γ), obtain \mathbf{u}_γ :

$$\mathbf{u}_\gamma = \pi_\gamma \mathbf{R}_\Gamma \left(\mathbf{u}, \frac{\partial\mathbf{u}}{\partial n} \right) \Big|_\Gamma. \quad (8)$$

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 Γ_1 :

$$\mathbf{u} \Big|_{\Gamma_1} = \mathbf{R}_{\Gamma_1} \mathbf{P} \mathbf{v}_\gamma \equiv \mathbf{R}_{\Gamma_1} \mathbf{P} \mathbf{u}_\gamma. \quad (9)$$

Finally, the ABC's are obtained in the operator form

$$\mathbf{u} \Big|_{\Gamma_1} = \mathbf{T} \left(\mathbf{u}, \frac{\partial\mathbf{u}}{\partial n} \right) \Big|_\Gamma, \quad (10)$$

where \mathbf{T} is composed of the operations (8) and (9). Boundary condition (10) is applied every time we need to update the ghost cell values in the course of the iteration process. The implementation of ABC's (10) can either be direct or involve preliminary calculation of the matrix \mathbf{T} . In the latter case, the runtime implementation of the ABC's (10) is reduced to a matrix-vector multiplication.

3 Numerical Experiments

Two-Dimensional Case. For the reason of completeness, we first briefly comment here on the two-dimensional results from our previous work (see Refs. [10,14,17]). In that work, we calculated subsonic and transonic viscous flows past single-element airfoils (NACA0012 and RAE2822).

The computational domain is formed by the C-type curvilinear grid generated around the airfoil. On this grid, the Navier-Stokes equations are integrated using the code **FLOMG** by Swanson and Turkel [11–13]. The standard treatment of the external boundary in the code **FLOMG** is based on the locally one-dimensional characteristics analysis, which may or may not be supplemented by the point-vortex (p.-v.) correction [21].

Basic conclusions that could be drawn from our two-dimensional numerical experience are the following. The DPM-based ABC's are geometrically universal, algorithmically simple and easy to implement along with the existing solver. For large computational domains (30–50 chords of the airfoil), the performance of the standard methods and the DPM-based ABC's is roughly the same. As, however, the artificial boundary approaches the airfoil the discrepancy between the corresponding solutions increases. The lift and drag coefficients obtained on the basis of boundary conditions (10) deviate from their asymptotic (50 chords) values much less (within fractions of one percent) than the coefficients obtained using local ABC's do. In other words, the nonlocal DPM-based ABC's allow one to use much smaller

computational domains (as small as 2–3 chords) than the standard boundary conditions do and still maintain high accuracy of computations. Moreover, if we compare three models: DPM-based, point-vortex, and standard local, then it turns out that the DPM-based ABC’s display the best performance for small computational domains, the performance of the local characteristic boundary conditions for the small domains is very poor, and the point-vortex boundary conditions perform much better for the lift than they do for the drag coefficient. This behavior seems reasonable since the point-vortex model is a lift-based treatment.

We also note that for certain variants of the computation the DPM-based ABC’s may noticeably speed up (by up to a factor of three) the convergence of the multigrid iterations, see Refs. [9, 10, 14]. The discussion on combined implementation of the DPM-based ABC’s with multigrid is contained in Ref. [17].

Three-Dimensional Case. Here, we demonstrate some new results on computation of the external flows around the ONERA M6 wing.

We use the code TLNS3D by Vatsa, et al. [22] to integrate the thin-layer equations on the curvilinear grid (see Figures 1 and 2) generated around the wing. This code is based on the central-difference finite-volume discretization in space with first- and third-order artificial dissipation. Pseudo-time iterations are used for obtaining the steady-state solution; the integration in time is done by the five-stage Runge-Kutta algorithm (with the Courant number calculated locally) supplemented by the residual smoothing. For the purpose of accelerating the convergence, the multigrid methodology is implemented; in our computations we used three nested grid levels with V cycles; this three-level V-cycle algorithm is, in fact, a final stage of the full multigrid (FMG) procedure. In addition, we use the preconditioning technique of Ref. [23] to improve the convergence to steady state. We implement the DPM-based ABC’s 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 ABC’s only on the first and the last Runge-Kutta stages, which seems 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 ABC’s on the first stage. Unlike the two-dimensional case, the standard treatment of the external boundary in three dimensions is based on merely the locally one-dimensional characteristics analysis and extrapolation (the point-vortex model is not applicable).

Table 1: ONERA M6: $M_0 = 0.5$; $Re_0 = 11.7 \cdot 10^6$; $\alpha = 3.06^\circ$.

Domain “radius”	1.25 root chords		2 root chords		10 root chords	
Grid	193 × 49 × 33					
Type of ABC’s	standard	DPM	standard	DPM	standard	DPM
Full lift, C_L	0.2218	0.2065	0.2185	0.2065	0.2081	0.2072
Relative error	6.58%	0.34%	5.0%	0.34%	0%	0%
Full drag, $C_D \times 100$	0.817	0.791	0.793	0.791	0.787	0.788
Relative error	3.8%	0.38%	0.76%	0.38%	0%	0%

In Table 1, we present the numerical results obtained with the two types of ABC’s for the subcritical flow around the ONERA M6 wing. Note, for each type of the boundary treatment the errors in Table 1 are calculated as relative errors with respect to the values of the force coefficients C_L and C_D that correspond to the largest domain (for which the results obtained with the DPM-based and standard ABC’s differ insignificantly). As one can clearly see, the DPM-based ABC’s in three space dimensions outperform the standard approach as they do in two dimensions: the new boundary conditions are capable of producing accurate results on very small computational domains, whereas the performance of the standard technique deteriorates as the domain shrinks. Analogous to the two-dimensional case, the three-dimensional DPM-based ABC’s can be calculated for any shape of , by means of the same procedure (universality) and can be easily combined with the existing solver (TLNS3D). Let us additionally note that for this series of computations, the dimensions of all three grids are the same, grids of smaller extent are obtained by scaling down the original 10 chords grid. The concentration of nodes near the wing that results from the down-scaling obviously contributes to the general improvement in the accuracy of the solution.

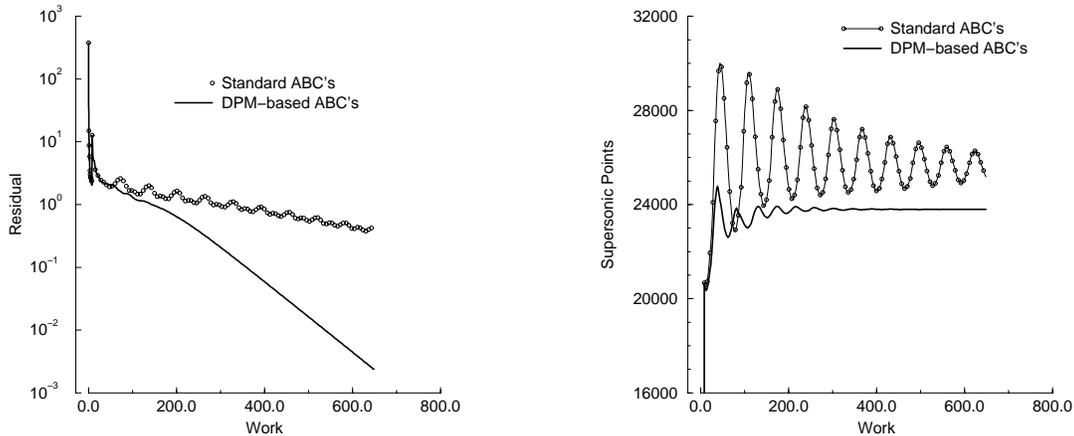
We have also conducted some three-dimensional computations for the standard transonic case for ONERA M6 wing: $M_0 = 0.84$; $Re_0 = 11.7 \cdot 10^6$; $\alpha = 3.06^\circ$. Here we could not bring the artificial boundary as close to the solid surface as in the foregoing subsonic case (see Table 1) because our far-field treatment is purely subsonic and the boundary cannot intersect the supercritical zone. We therefore calculated the solution for two computational domains of average radii of approximately 10 and 3 root chords of the wing, respectively. The results summarized in Table 2 clearly demonstrate that for small computational domains the DPM-based ABC’s generate much more accurate solutions than the standard boundary conditions do. (Similarly to Table 1, the errors in Table 2 are calculated as relative errors with respect to the corresponding values on the larger domain.) Note, in this case the grids for the different domains have different dimensions, and the smaller $193 \times 49 \times 33$ grid (3 chords) is now an exact subset of the bigger $209 \times 57 \times 33$ grid (10 chords). This is done in order to eliminate any influence that the change of the grid in the near field could possibly exert on the solution.

Table 2: ONERA M6: $M_0 = 0.84$; $Re_0 = 11.7 \cdot 10^6$; $\alpha = 3.06^\circ$.

Domain “radius”	3 root chords		10 root chords	
Grid	193 × 49 × 33		209 × 57 × 33	
Type of ABC’s	standard	DPM	standard	DPM
Full lift, C_L	0.298±0.004	0.2798	0.2805	0.2786
Relative error	6.24%±1.43%	0.43%	0%	0%
Full drag, $C_D \times 10$	0.168±0.008	0.1537	0.1542	0.1531
Relative error	8.95%±5.19%	0.39%	0%	0%

Besides the improvement of accuracy, the application of the DPM-based ABC’s to transonic flow computations on the small (3 chords) computational domain yielded a much higher convergence rate of the residual (continuity equation), as well as much faster convergence of other quantities, including those deemed as sensitive, e.g., the number of supersonic points in

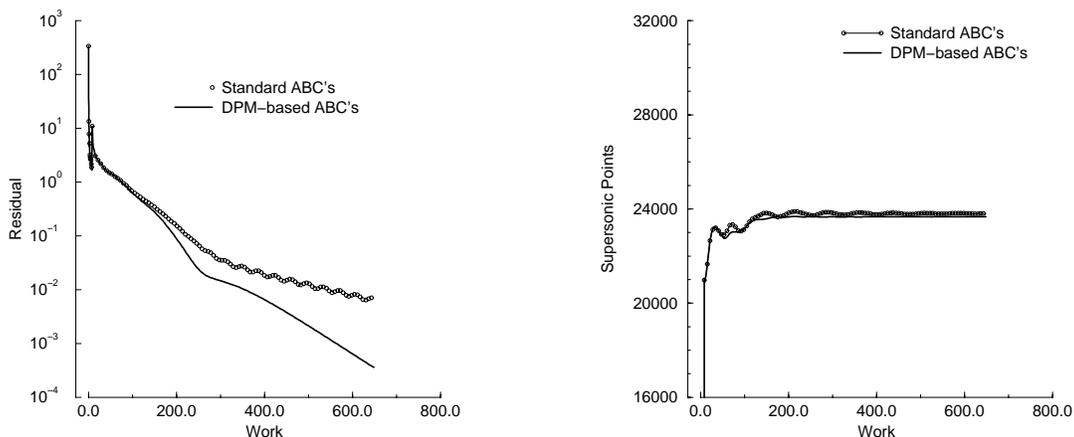
the domain. In Figure 3, we show the convergence history for this supercritical flow variant. One can see that the convergence for the standard boundary conditions is poor; therefore the corresponding force coefficients in Table 2 are given with the error bands indicated.



(a) Residual of the continuity equation

(b) Number of supersonic nodes in the domain

Figure 3: Convergence history for the supercritical case: ONERA M6: $M_0 = 0.84$, $Re_0 = 11.7 \cdot 10^6$, $\alpha = 3.06^\circ$. Average domain “radius” is 3 root chords of the wing; grid $193 \times 49 \times 33$.



(a) Residual of the continuity equation

(b) Number of supersonic nodes in the domain

Figure 4: Convergence history for the supercritical case: ONERA M6: $M_0 = 0.84$, $Re_0 = 11.7 \cdot 10^6$, $\alpha = 3.06^\circ$. Average domain “radius” is 10 root chords of the wing; grid $209 \times 57 \times 33$.

For the 10 chords domain, the DPM-based ABC's also provide for some convergence speedup, although the difference between the two ABC's techniques is less dramatic here.

This is reasonable because one could generally expect that the bigger the computational domain, the smaller is the influence that the external boundary conditions exert on the numerical procedure. The convergence history for the 10 chords computations is shown in Figure 4.

Note, from Figure 3(b) one can conclude that on the small domain the two algorithms converge to quite different solutions, whereas Figure 4(b) allows one to assume that on the big domain the final solutions are close to one another. The data from Table 2 corroborate these conclusions. This behavior again fits into the aforementioned concept that the impact of the ABC's decreases as the domain size increases. In the future, we plan on running some transonic computations for domain bigger than 10 root chords of the wing. This may provide for an experimental justification of the statement that the solutions for the two different types of ABC's approach one another as the domain enlarges.

Miscellaneous Issues. Here, we will primarily discuss the computational cost of the DPM-based ABC's and possible ways for its reduction.

The cost of the DPM-based ABC's in two space dimensions is modest; the ABC's may add about 10% to the cost of the original integration procedure. In three space dimensions, the relative cost of the DPM-based ABC's is so far higher; they typically add from 25% to 30% of extra CPU time. In many cases, however, this increase can be compensated for by the convergence acceleration, see Figures 3 and 4. Moreover, we expect that some changes to the ABC's algorithm can still be done so that the vectorization is exploited more effectively and therefore the boundary operators are computed faster. Apart from the algorithm changes there are some "indirect" ways for reducing the computational cost of the DPM-based ABC's.

Parallel implementation of the ABC's on multi-processor machines is a relatively easy task because the solution algorithm for the linear AP is readily parallelizable. For two space dimensions, an up to five times speedup in computing the nonlocal boundary operators compared to the single-processor version has been achieved on an eight-processor CRAY Y-MP.

In three space dimensions, carrying out multiple computations on the same grid is very often the case (many different flow regimes in one geometric setting). In so doing, the external geometry that influences the ABC's is fixed and the boundary conditions depend only on the aerodynamic parameters, e.g., Mach number M_0 and the angle of attack α . Therefore, the entry-wise interpolation of the matrices of boundary operators \mathbf{T} with respect to these parameters has a substantial promise from the standpoint of cost reduction. Indeed, after a noticeable startup expense for calculating the "reference" \mathbf{T} 's for some selected values of the parameters, any other matrix needed for any specific regime within the prescribed range of the parameters can be obtained for virtually no extra cost by means of the interpolation. This methodology has been tested for the two-dimensional supercritical flow around the RAE2822 airfoil [17]. The results seem very promising: for the small computational domain the accuracy provided by the operators \mathbf{T} interpolated with respect to M_0 and α is only slightly worse than the accuracy obtained using genuine DPM-based ABC's (i.e., the operator calculated for the given specific values of the parameters) and is still much better than the accuracy obtained on the basis of the point-vortex model.

Finally, we mention that the nonlocality of the DPM-based ABC's is expressed by the fact

$\rho \sim \rho$	$u \sim \rho$	$v \sim \rho$	$w \sim \rho$	$p \sim \rho$
$\rho \sim u$	$u \sim u$	$v \sim u$	$w \sim u$	$p \sim u$
$\rho \sim v$	$u \sim v$	$v \sim v$	$w \sim v$	$p \sim v$
$\rho \sim w$	$u \sim w$	$v \sim w$	$w \sim w$	$w \sim p$
$\rho \sim p$	$u \sim p$	$v \sim p$	$w \sim p$	$p \sim p$

Figure 5: Qualitative block structure of the boundary operator \mathbf{T} for three-dimensional flow computations.

that the matrix \mathbf{T} (see (10)) is dense. However, this matrix is typically structured. From the standpoint of physics, the structure of \mathbf{T} reflects the simple consideration that each specific node influences its close neighbors stronger than it influences the remote points. For system (4a), the operator \mathbf{T} from (10) is composed of 5×5 blocks as shown in Figure 5 (provided that the vectors of variables along the boundaries, and, $\mathbf{1}$ are arranged properly: first ρ 's for all the nodes, then u 's, etc., and, finally, p 's). The qualitative meaning of each block in Figure 5 is how one variable of the corresponding pair influences another one along the entire boundary. Stronger interdependence between closer nodes results in the fact that the near-diagonal entries for each block are considerably larger than the off-diagonal ones (if, additionally, the order of nodes for, and, $\mathbf{1}$ is the same). The latter observation has, in fact, been corroborated computationally (for the two-dimensional case).

To efficiently calculate matrix-vector products for the structured matrices of the foregoing type, one can use multiresolution-based data compression algorithms, e.g., the one proposed by Harten and Yad-Shalom [24]. In Ref. [19], we applied this algorithm to the model example of the matrix \mathbf{T} for the two-dimensional scalar Yukawa equation. As could be seen from the nested multiresolution representation [24] for the model operator \mathbf{T} (see Ref. [19]), all the nonzero entries are concentrated near the main diagonal of each nested block; as we recede the diagonal the corresponding values rapidly fall below the double-precision machine accuracy (10^{-13}) threshold. Therefore [24], this nested structure can be used for the efficient calculation of matrix-vector products with consecutively improving resolution. The concentration of nonzero entries near the main diagonal also implies effective localization of the boundary conditions. This feature may have certain promise for the future use in the view of the multi-block grids.

The next major challenge from the standpoint of treating numerically the infinite-domain formulations is presented by the time-dependent problems. As mentioned in the beginning, the issue of ABC's for the time-dependent problems is significant not only in CFD but also in other areas of scientific computing, e.g., those originating from acoustics and electrody-

namics.

The primary difficulty that distinguishes time-dependent problems from the steady-state ones is that here the exact ABC's are nonlocal not only in space but also in time (except in the simplest one-dimensional problems). As concerns the approximate approaches, the same basic trend that is relevant to the steady-state problems also holds for the time-dependent ones: more accurate boundary procedure requires more of the nonlocal nature of the ABC's to be taken into account.

Of course, the nonlocality of the ABC's in time presents a very serious computational obstacle, because as the solution evolves such boundary conditions would become more and more expensive from the standpoints of both memory and computer time requirements. Therefore, the most crucial numerical issue in constructing the time-dependent ABC's is how to effectively restrict the nonlocality of the boundary conditions in time. This is an interesting subject for the future research.

4 Conclusions

The new global ABC's for calculating steady-state external viscous flows in three space dimensions have been constructed on the basis of the difference potentials method. The approach generalizes and extends our previous two-dimensional results.

The new ABC's are capable of greatly reducing the size of the computational domain (compared to the standard methods) while still maintaining high accuracy of the numerical solution. This size reduction amounts to either the possibility of refining the grid in the near field, which potentially leads to the improvement in accuracy or usage of the smaller-dimension grids without compromising the accuracy. Moreover, the DPM-based ABC's may noticeably improve the convergence rate of the multigrid iteration procedure. Finally, the new boundary conditions appear geometrically universal and easy to incorporate in the structure of the existing flow solvers. The properties of the new ABC's have been corroborated experimentally by computing the subsonic and transonic flows past the ONERA M6 wing using the NASA-developed code `TLNS3D`.

A more detailed description of the theoretical foundations of the DPM-based algorithm, as well as a wider selection of the computational results, are presented in Ref. [25].

Acknowledgments

We are most grateful to Dr. E. B. Parlette of Vigyan, Inc., for his valuable help in grid generation and Dr. R. C. Swanson and Dr. J. L. Thomas of NASA Langley for the useful discussions on the subject of the paper. We are also very thankful to the reviewer of the paper for his helpful comments.

References

- [1] Tsynkov, S. V., “Numerical Solution of Problems on Unbounded Domains. A Review,” *Applied Numerical Mathematics*, Vol. 27, 1998, pp. 465–532.
- [2] Givoli, D., “Non-reflecting Boundary Conditions,” *Journal of Computational Physics*, Vol. 94, No. 1, 1991, pp. 1–29.
- [3] Givoli, D., *Numerical Methods for Problems in Infinite Domains*, Elsevier, Amsterdam, 1992.
- [4] Calderon, A. P., “Boundary-Value Problems for Elliptic Equations,” *Proceedings of the Soviet-American Conference on Partial Differential Equations at Novosibirsk*, Fizmatgiz, Moscow, 1963, pp. 303–304.
- [5] Seeley, R. T., “Singular Integrals and Boundary Value Problems,” *American Journal of Mathematics*, Vol. 88, No. 4, 1966, pp. 781–809.
- [6] Ryaben’kii, V. S., “Boundary Equations with Projections,” *Russian Mathematical Surveys*, Vol. 40, No. 2, 1985, pp. 147–183.
- [7] Ryaben’kii, V. S., *Difference Potentials Method for Some Problems of Continuous Media Mechanics*, Nauka, Moscow, 1987 (in Russian).
- [8] Ryaben’kii, V. S., “Difference Potentials Method and its Applications,” *Mathematische Nachrichten*, Vol. 177, February 1996, pp. 251–264.
- [9] 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.
- [10] 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.
- [11] Swanson, R. C. and Turkel, E., “A Multistage Time-Stepping Scheme for the Navier-Stokes Equations,” AIAA Paper 85-0035, January 1985.
- [12] 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.
- [13] Swanson, R. C. and Turkel, E., “Multistage Schemes with Multigrid for the Euler and Navier-Stokes Equations. Components and Analysis,” NASA TP-3631, August 1997.
- [14] 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, January 1995.
- [15] 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.

- [16] 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.
- [17] 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.
- [18] Tsynkov, S. V., “Nonlocal Artificial Boundary Conditions for Computation of External Viscous Flows,” *Computational Fluid Dynamics’96, Proceedings of the Third ECCO-MAS CFD Conference*, John Wiley & Sons, New York, 1996, pp. 512–518.
- [19] Tsynkov, S. V., “Artificial Boundary Conditions for Infinite-Domain Problems,” *Barriers and Challenges in Computational Fluid Dynamics*, Kluwer Academic Publishers, Dordrecht, 1998, pp. 119–138.
- [20] Cole, J. D. and Cook, L. P., *Transonic Aerodynamics*, Elsevier, Amsterdam, 1986.
- [21] Thomas, J. L. and Salas, M. D.: “Far-Field Boundary Conditions for Transonic Lifting Solutions to the Euler Equations,” AIAA Paper 85-0020, January 1985.
- [22] 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, January 1993.
- [23] Turkel, E., Vatsa, V. N., and Radespiel, R., “Preconditioning Methods for Low-Speed Flows,” AIAA Paper 96-2460, June 1996.
- [24] Harten, A. and Yad-Shalom, I., “Fast Multiresolution Algorithms for Matrix-Vector Multiplication,” *SIAM Journal on Numerical Analysis*, Vol. 31, No. 4, 1994, pp. 1191–1218.
- [25] Tsynkov, S. V., “External Boundary Conditions for Three-Dimensional Problems of Computational Aerodynamics,” *SIAM Journal on Scientific Computing*, Vol. 21, 1999, pp. 166–206.