

TECHNICAL RESEARCH REPORT

A Computational Framework for Boundary-Value Problem Based Simulations

by R. Adomaitis, Y-H. Lin, H-Y. Chang

T.R. 98-41



ISR develops, applies and teaches advanced methodologies of design and analysis to solve complex, hierarchical, heterogeneous and dynamic problems of engineering technology and systems for industry and government.

ISR is a permanent institute of the University of Maryland, within the Glenn L. Martin Institute of Technology/A. James Clark School of Engineering. It is a National Science Foundation Engineering Research Center.

Web site <http://www.isr.umd.edu>

A Computational Framework for Boundary-Value Problem Based Simulations

Raymond A. Adomaitis*, Yi-hung Lin, and Hsiao-Yung Chang
*Department of Chemical Engineering and
Institute for Systems Research
University of Maryland, College Park MD 20742*

September 4, 1998

Abstract

A framework is presented for step-by-step implementation of weighted-residual methods (MWR) for simulations that require the solution of boundary-value problems. A set of MATLAB-based functions of the computationally common MWR solution steps has been developed and is used in the application of eigenfunction expansion, collocation, and Galerkin-projection discretizations of time-dependent, distributed-parameter system models. Four industrially relevant examples taken from electronic materials and chemical processing applications are used to demonstrate the simulation approach developed.

Keywords: Chemical process simulation; boundary-value problems; method of weighted residuals; distributed parameter systems; collocation; eigenfunction expansions; Galerkin's method.

1 Introduction

Simulations of chemical manufacturing process unit operations span a range of complexity from steady-state material and energy balances used in "back of the envelope" calculations to highly resolved, supercomputer-based simulations of time-dependent reacting flows in physically complex geometries. This range of model fidelity is seen primarily in simulations of process systems modeled as distributed parameter systems, a class of systems that describe the transport of energy, material, or momentum in terms of conservation equations or contain states described by particle size, molecular weight, or other distributions. Examples of such processes include the packed towers used for physical separations, heterogeneous catalytic reactors, and fluid-mixing problems.

One set of problems that falls between these two extremes is time-dependent, boundary-value problems (BVPs) that are defined in geometrically simple domains and that contain a moderate number of independent state variables. These problems are conveniently solved by the method of weighted residuals (MWR), a class of numerical techniques that includes eigenfunction expansion, collocation, Galerkin, and other projection methods. Well-established computational procedures exist for implementing the various MWR, based on both globally and locally defined trial function expansions [4, 10, 20]. However, software developed to implement these methods tends to fall into the categories of programs written for a specific implementation of one element of an MWR solution procedure (e.g., [5, 9]), or software packages based on one of the MWR solution methods designed for solving a specific class of problems [23].

Complete simulation and analysis of an engineering simulation problem frequently can involve a combination of MWR methods and other mathematical operations. The MWR solution techniques share common mathematical operations and follow similar steps in the implementation of the solution procedures: the overall method involves elementary steps including generating trial functions from the solution of a Sturm-Liouville problem, various inner product calculations and projections, and other basic mathematical operations, most

*Corresponding author; e-mail: adomaiti@isr.umd.edu; Ph: (301) 405-2969; Fax: (301) 314-9920

of which are common to *all* other MWR. Realizing this, it became our goal to develop a set of MATLAB-based functions that have a one-to-one correspondence to these individual operations.

The methods we developed are based on two elementary principles. The first is that all functions corresponding to the different spatial, temporal, probability, and other distributions over finite domains can be represented in terms of discretized functions, specifically, Lagrange interpolation polynomials. Because many special functions can be described in terms of convergent polynomial expansions [3], this is a natural method for approximating arbitrary functions. In addition, this method makes it feasible to control the accuracy of our discrete approximations. With the trial functions represented to some known accuracy, the second principle is to make use of the many implementation steps of the MWR that can be performed with accuracy limited only by machine roundoff error. For example, the Gaussian quadrature weights assigned to the discretized domains allow exact (and efficient) computation of weighted inner products used in the different projection methods, and differentiation and other operations also can be performed to exact or highly accurate degree. Therefore, residual functions can be precisely computed, resulting in purely numerical discretization techniques with accuracy comparable to hand-calculated, discretized problems.

In this framework, all computations, including differentiation and integration, reduce to matrix operations. Evaluation of trial function expansion, computation of residuals, projection operations, and other problem solving steps follow naturally when using these discretized function representations, resulting in a coherent framework for solving engineering simulation problems.

2 The MWRtools Functions

As a result of investigating the common features of MWR applications in the context of the simulation problems discussed in this paper and other studies, we developed a set of ten elementary functions that correspond to the computationally common elements. We present these functions in the approximate order they would be applied in a typical MWR solution procedure, beginning with the problem physical domain specification and definition of the trial functions, and then proceeding with the operational methods of the MWR projections, the refinement and solution convergence analysis steps, and the functions used for the computationally efficient collocation methods.

Problem Setup

pd.m Sets up the discretized physical domain and defines the differentiation and quadrature operators according to the specified accuracy requirements and any imposed problem symmetries. This function is normally called first in the solution procedure. The inputs are the problem geometry and the number of discretization points; output consists of the discretization grid, quadrature weights, and the differentiation tensors.

The discretization positions and weights are based on the modified Gaussian quadrature method (Gauss-Radau-Lobatto method). The two endpoints (0 and 1) are preassigned, and interior positions are the roots of the $(M - 2)$ th Jacobi polynomial. The quadrature weights are computed after the discretization grid point locations. The differentiation tensors are based on the Lagrange interpolation polynomial and rely on its properties of being continuous and at least twice differentiable.

gdf.m Generates discretized representations of a sequence of trial functions Ψ according to the alphanumeric form of the trial function expansion and the grid point location vector produced by **pd.m**. The functions can be polynomials, eigenfunctions generated by a Sturm-Liouville problem, or an arbitrary sequence of functions chosen as part of a Galerkin discretization.

sl.m A Sturm-Liouville problem solver that computes the vector of eigenvalues (λ), discretized eigenfunction array (Ψ), and discretized adjoint eigenfunction array (Φ) for problems of the type described by the subset of Sturm-Liouville problems:

$$\lambda\psi = p\nabla^2\psi - q\frac{d\psi}{dx} - f(x)\psi \quad (1)$$

subject to boundary conditions

$$\begin{aligned} a(d\psi/dx) + b\psi &= 0 & \text{at } x = 0, \\ c(d\psi/dx) + d\psi &= 0 & \text{at } x = 1. \end{aligned} \tag{2}$$

The λ_n and ψ_n are computed as the eigenvalues and eigenvectors of an array defined by the discrete differentiation operations corresponding to the ∇^2 and d/dx operators (from `pd.m`) and the discretized form of the function f . In the first step of this procedure, the boundary conditions are used to define a linear relationship between the interior-point and boundary-point values of ψ , and this set of linear equations is used to reduce the *discretized* eigenvalue problem (1) to homogeneous form. After computing the $M-2$ eigenvalues and eigenvectors, the endpoint values of the discretized eigenfunctions are computed to satisfy the boundary conditions.

Operational Functions

gs.m A Gram-Schmidt orthogonalization procedure. Given a sequence of discretized functions Ψ , this function generates a sequence of discretized functions orthonormal with respect to the inner product defined by the quadrature weights computed by `pd.m`, giving an orthonormal basis for the space spanned by the original Ψ .

wip.m The weighted inner product of two sets of discretized functions. This is one of the most heavily used subprograms since inner products are used in nearly every MWR step. The array of inner products produced by this function is computed as the vector dot product of the quadrature weight array with the term-by-term product of all combinations of the discretized functions.

rdf.m Computes the roots of a discretized function. This function is used to compute eigenvalues described by nonlinear relationships; it also is used to compute collocation point locations in `colmat.m`. The inputs consist of any discretization grid and the corresponding function values. The intervals bracketing the roots are found and a linear interpolation is performed; the roots are refined using Newton-Raphson iterations (and `polyint.m`) until given tolerance is satisfied.

Collocation Methods

The computational benefits of using collocation or other pseudo-spectral methods are not lost in our solution procedure framework. The discretized trial functions used in a collocation procedure can be treated as exact representations of the trial functions when the number of discretization points used in `pd.m` is significantly larger than the number of solution trial functions. Collocation points and discrete-ordinate discretization arrays of the orthogonal collocation technique [21, 22] can then be efficiently and accurately computed using the following functions. A benefit of this approach is that the convergence properties and accuracy of the collocation solutions can be rigorously assessed [2] because of the transparent interplay between the computational MWR modules.

colmat.m Computes the collocation differentiation arrays, discrete transformation array, and quadrature weights at the collocation points based on the discretized set of trial functions Ψ and the collocation points computed with `colpts.m`.

colpts.m Computes collocation points as the roots of the highest-frequency trial function as part of an orthogonal collocation [21, 22] procedure. The inputs are the discretized trial functions with the associated fine-grid discretization point vector produced by `pd.m`. The roots are computed by the Newton-Raphson-based function `rdf.m`.

Solution Refinement

Ffilter.m Produces a vector of Fourier-space filter coefficients used in the trial function expansions to reduce the oscillations associated with the Gibb's phenomenon [13]. These oscillations result from projecting a discontinuous function (represented in discretized form) onto globally defined, smooth trial functions generated by either `gdf.m` or `s1.m`.

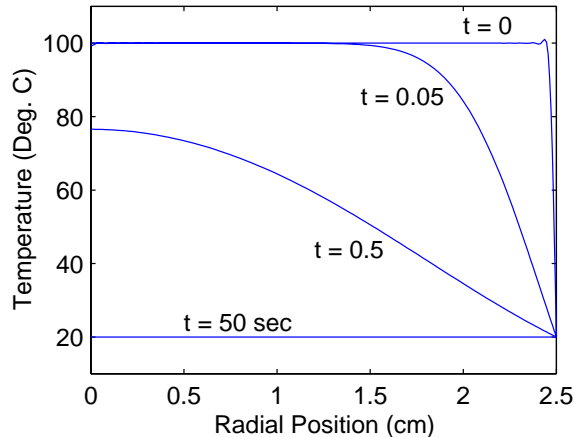


Figure 1: *The computed transient temperature profile of the quenched sphere.*

polyint.m Lagrange polynomial interpolation based Neville’s algorithm. This recursive formula is obtained by rearranging the order in which the implementation of Lagrange interpolation calculations are performed. The result is spectrally accurate and provides an error estimate.

2.1 An Implementation Example

As one example of a simulation solution procedure that makes use of the `MWRtools` functions, consider the problem of determining the time-dependent temperature profile of a 5 cm diameter steel sphere, initially at a uniform temperature of 100°C , that is quenched by dropping it into a bath of water at 20°C . Representing the temperature in terms of the truncated trial function expansion

$$T(r, t) = \sum_{n=1}^M a_n(t) \psi_n(r) \quad (3)$$

where the trial functions are defined as eigenfunctions of the dimensionless heat equation

$$\frac{\partial T}{\partial t} = \nabla^2 T, \quad (4)$$

the residual produced by substituting (3) into (4) can be projected onto each eigenfunction ψ_m to obtain a set of linear ODEs in time. Because of the modal ODE structure, the general solution is simple to compute, and the particular solution is defined by projecting the nonhomogeneous initial condition onto each eigenfunction. This completes the solution description; collocation and Galerkin solution procedures are similar.

To computationally solve this sphere-quenching example, we follow the written description and generate the sequence of computational steps consisting of intrinsic MATLAB functions and `MWRtools` functions. In this solution procedure, we choose to discretize the dimensionless physical domain using $nd_r = 120$ points and use $M = 20$ trial functions in the temperature field expansion, where the trial functions are the eigenfunctions of the heat equation subject to $\psi_n(1, t) = 0$ and symmetry at $r = 0$:

```
M = 80; nd_r = 120;
[r, wr, dr, ddr] = pd('sphe', nd_r);
[eigval eigfun] = sl(dr, ddr, r, 1, 0, 0, 1, wr);
lambda = eigval(1:M); psi = eigfun(:, 1:M);
```

The initial condition $u(r, 0) = 1$ is projected onto the eigenfunctions to compute the initial values of the mode amplitudes and is filtered to remove most of the oscillations resulting from an initial condition that

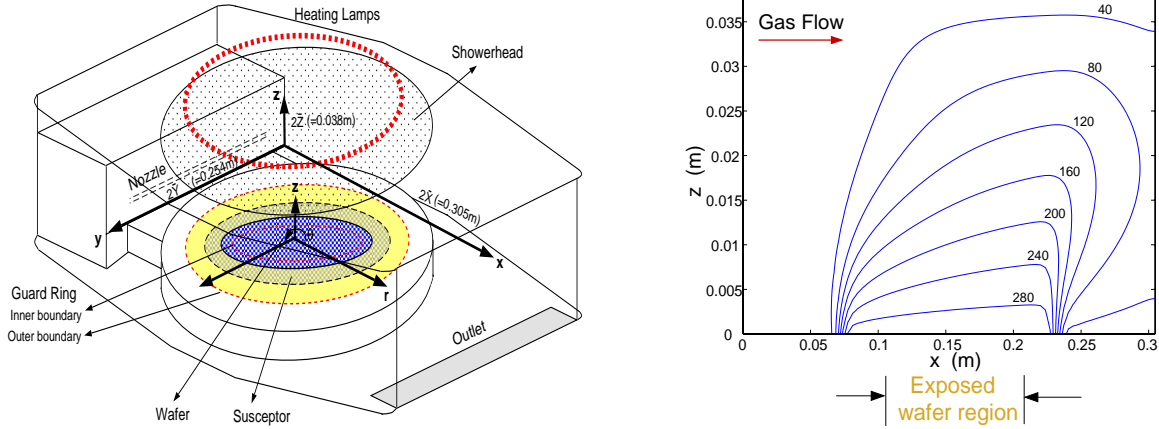


Figure 2: The geometry of the tungsten CVD reactor (left). Representative gas temperature simulation results are shown in the right plot, with contours marked in Deg. C.

does not satisfy the homogeneous boundary condition ($T(1,t) = 0$) used to compute the eigenfunctions. The solution profiles at the discrete points in time are computed by the direct integration of the ODEs that result when the residual is projected onto each eigenfunction:

```

a0 = wip( ones(nd_r,1),psi,wr ).*Ffilter(2,M);
time = [0 0.01 0.1 10];
for i = 1:4
    a(:,i) = a0.*exp(lambda*time(i));
end

```

The results are shown in Fig. 1 and are plotted (in dimensional form) using `plot(r*2.5,80*psi*a+20)`; the time is converted to dimensional form by the time constant of $\tau \approx 5$ sec.

This example was specifically chosen as a problem that can be solved explicitly so the exact solution could be compared to the results of the numerical solution procedure developed. The following case studies illustrate the use of the `MWRtools` functions for simulating industrially relevant process operations.

3 CVD System Heat Transfer – Eigenfunction Expansions

As the first example of the types of simulation problems that can be solved using the `MWRtools` functions, we consider the problem of assessing the importance of gas/wafer heat transfer in a commercial, tungsten chemical vapor deposition (CVD) reactor used for the production of microelectronic devices. Tungsten (W) is used both as a contact plug material and as a first-level metal for interconnects in microelectronic devices. Tungsten can be deposited in two different CVD modes: in the selective W CVD processes, W selectively grows on exposed silicon surfaces, producing inter-level contacts in a single step. Alternatively, W can be blanket-deposited over the entire wafer surface to fill the contact holes with a subsequent chemical-mechanical polish step to remove excess W, creating isolated plugs.

Our modeling research focuses on the BTU-ULVAC ERA-1000 selective tungsten deposition system, shown schematically in Fig. 2. Reactant gases are introduced into the reactor from two sources: a gas mixture of SiH_4 and WF_6 is injected through a slit-like nozzle on the side wall, and H_2 is pumped in through a shower head at the top of reactor chamber. Varying the feed gas mixtures allows for the different deposition modes. Gases mix in the chamber and react at the surface of a four inch diameter wafer, which is supported by a slowly rotating quartz susceptor. A portion of the wafer near its outer edge is covered by a quartz guard ring. The wafer is heated to 600 K by a ring of incoherent tungsten-halogen lamps through the transparent shower head. The CVD runs last for approximately five minutes after the operating temperature is reached.

Initial simulation work on this system has focused on gas flow and reactor/wafer thermal dynamics for the case where horizontal gas flow dominates. The ULVAC CVD system was designed for low-pressure, laminar-flow processing conditions; therefore it is possible to produce relatively accurate linear PDE models of the gas flow and temperature fields and solve these models using Galerkin and eigenfunction expansion methods based on globally defined trial functions. In our solution procedure, we discretize the three-dimensional gas temperature field using a collocation procedure in the span-wise direction and a two-dimensional eigenfunction expansion in the remaining physical dimensions; the eigenfunctions are generated by `sl.m` after three calls to `pd.m`. The residual produced by the eigenfunction expansion solution procedure is computed by projecting the nonhomogeneous and spatially dependent terms onto the eigenfunctions using `wip.m`. Solving the set of linear equations produced by projecting the residual onto the eigenfunctions gives a temperature field solution (Fig. 2) that must be spectrally filtered using the coefficients computed with `Ffilter.m`. The filtering is required because of the Gibb’s-phenomenon oscillations caused by the temperature boundary condition discontinuity at the heated-wafer edge.

This analysis gives an accurate first look at gas/wafer energy transport mechanisms and sheds light on the relative importance of the different heat transfer mechanisms ([7] and Fig. 2). Reduction of the gas-phase temperature solutions to overall wafer/gas heat transfer coefficients (using the differentiation arrays produced by `pd.m`) are made possible by this solution procedure; it was also interesting to find that the solution is relatively insensitive to the details of the gas flow field under these low-pressure conditions, justifying the assumption that detailed, CFD-type computations are unnecessary for this system. We can also conclude that because of the significant volume of the heated-gas plume seen in Fig. 2, the possibility of gas phase reactions is high. Finally, the wafer thermal dynamics have been modeled and discretized using the collocation and Galerkin discretization methods described in the following section. Preliminary results show that for some operating conditions, large temperature variations can result across the wafer leading to thermally induced wafer stress and plastic deformation.

4 Fractionation Tower Simulation – Orthogonal Collocation

Fractionation towers represent one of the most widely used separation processes in the chemical processing industry. We considered the problem of simulating the startup dynamics and steady-state performance of a packed tower designed to concentrate a specific chemical component of a liquid stream, where the desired species occurs in low concentrations in the feed stream (Fig. 3). The tower we considered is representative of the first stage in a heavy-water or other stable-isotope production process [6, 11], and this specific application was used to assess tower packing performance and alternative separation cascade sequences.

Development of the process model begins with a material balance that accounts for interphase mass transfer, convective transport, and holdup of the desired species:

$$(1 - \epsilon_p - \epsilon)\rho_g \frac{\partial y}{\partial t} = \frac{V}{A_c} \frac{\partial y}{\partial z} + k \left[\frac{x}{\alpha - \alpha x + x} - y \right] \quad (5)$$

$$\epsilon\rho_l \frac{\partial x}{\partial t} = -\frac{L}{A_c} \frac{\partial x}{\partial z} + k \left[y - \frac{x}{\alpha - \alpha x + x} \right] \quad (6)$$

where x and y denote species mole fractions in the column liquid and gas phases, respectively, k is the interfacial mass transfer coefficient, L and V are the column internal liquid and vapor flows, ϵ and ρ refer to gas, liquid, and packing phase volume fractions and densities, A_c is the column cross section, and α is the separation factor, which is close to unity and temperature dependent for this difficult separation.

The boundary conditions for the column differential equation model depend on its configuration; for this example (see Fig. 3), the top ($z = 0$) boundary condition is the feed concentration of the key component, and the bottom ($z = 1$) boundary condition is defined by the reboiler model. Thus, this model constitutes a two-point, nonlinear boundary value problem, readily solved with a semi-discrete collocation-based method. In the `MWRtools` framework, the physical domain is defined using `pd.m` and the orthonormal, polynomial trial functions are generated with `gdf.m` and `gs.m`. A four-mode trial function expansion was used for this problem; `colmat.m` was used to compute the discretization arrays corresponding to the solution evaluated at the collocation points computed with `colpts.m`. The (spatially) discretized problem was integrated with one

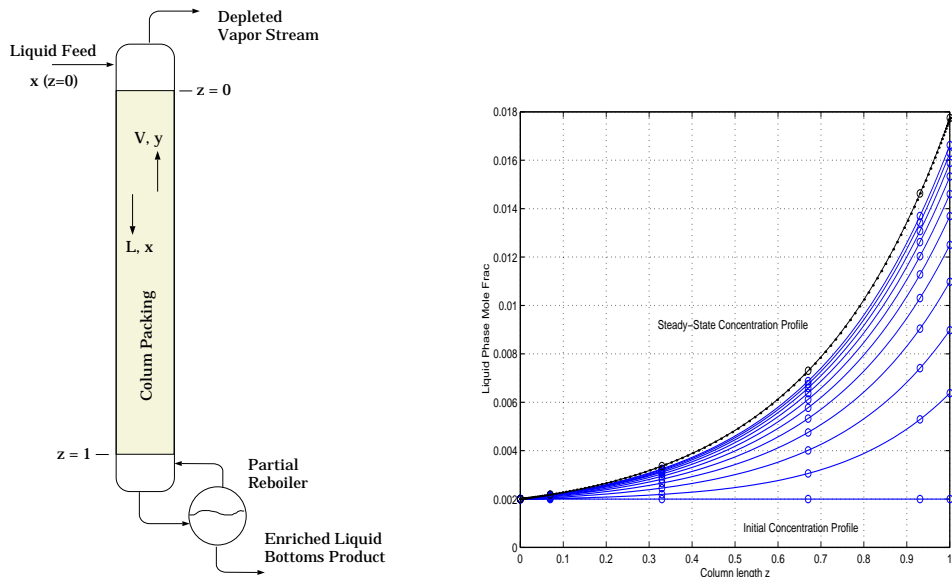


Figure 3: *Geometry of the stripping column (left). Representative simulation results (right) show the column profile snapshots taken at 2-hour intervals.*

of the MATLAB ODE solvers, subject to the nonlinear boundary condition constraints. Discretized solution accuracy was assessed by evaluating the residual on the highly resolved discretization grid generated by `pd.m`. Representative simulation results are presented in Fig. 3, where one of the most significant features of this column is observed: the extremely slow dynamic response of the column during startup transients. This is directly attributable to the long period of time required to collect the desired species in the column’s steady-state concentration profile, a feature reflected in the extremely long start-up times observed in operating separation processes [11].

5 Process Gas Compression – Galerkin Pseudo-Spectral Method

Axial compression systems are used in chemical process industry gas compression applications requiring large volume through-puts with relatively low pressure rise. Chemical and bulk material processing axial compressor applications include providing feed air for butadiene production by oxidative dedehydrogenation of *n*-butene, nitric acid production by ammonia oxidation, air-separation plants, blast furnaces, co-generation combustion applications, and providing air service for fluid catalytic cracking units (FCCUs). A schematic diagram illustrating the cross-section of a multistage FCCU axial compressor is shown in Fig. 4. Air enters at atmospheric pressure through the inlet diffuser and is compressed by a sequence of stages consisting of alternating rows of stationary and rotating blades. Flow control is provided by a combination of adjusting rotor speed, angle of the first rows of stator and inlet guide vanes, and throttle valve opening (located downstream of the exit diffuser). Overall pressure rise is determined by downstream process requirements and throttle valve pressure drop.

A model for the dynamics of the axial component of the gas flow is discussed in [1]; we use it here to demonstrate a discrete analog to the Galerkin projection, based on a multiple-grid implementation of a pseudo-spectral technique. The motivation for this simulation is to describe the growth of small (spatial) perturbations in the flow field to fully-developed stall cells and examine the form of the fully-developed rotating stall cell when the system is operated beyond the stall-bifurcation point (Fig. 4).

The model describes the dynamics of the mean axial flowrate $V(t)$ and the flow perturbation term $v(\eta, \theta, \tau)$ measured at the compressor face $\eta = 0$ and is derived from the Navier-Stokes equations under certain assumptions for the flow fields in the compressor ducting and power input from the compressor

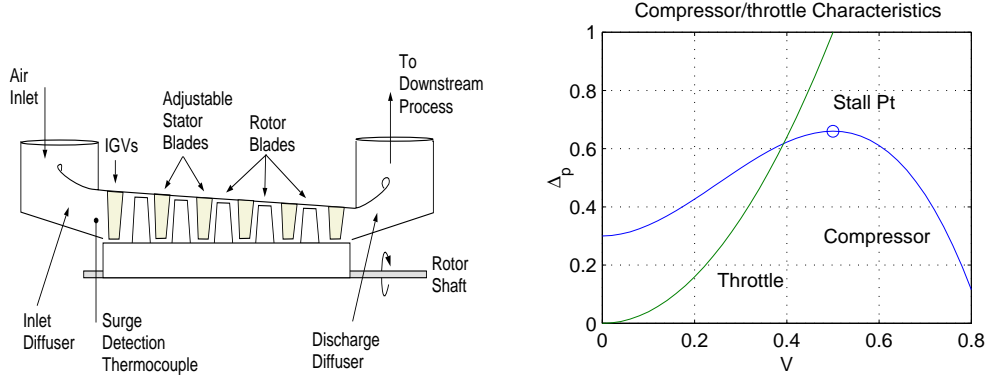


Figure 4: *Chemical process axial gas compression system including the inlet and outlet duct and compression sections (left) and the compressor/throttle characteristics (right). Steady, uniform-flow solutions are found at the intersection of the two curves; these solutions are unstable left of the stall point.*

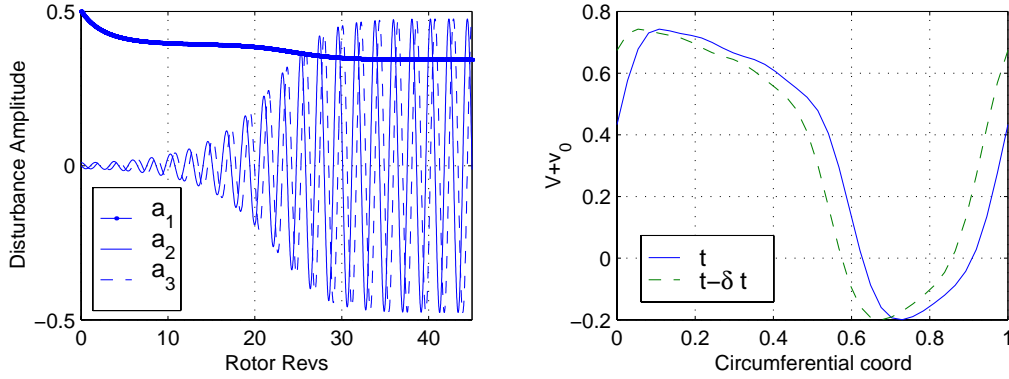


Figure 5: *Simulation results showing the growth of an infinitesimal spatial disturbance to a fully developed rotating stall cell (left) and snapshots of the rotating stall cell (right) showing the traveling-wave behavior of the stall cell.*

blading [19]:

$$\Delta_p = f(V + v_0) - l_c \frac{dV}{d\tau} - m_d \frac{\partial}{\partial \tau} \int_{-\infty}^0 v d\eta - \frac{1}{2\alpha} \left[2 \frac{\partial v_0}{\partial \tau} + \frac{\partial v_0}{\partial \theta} - \mu \frac{\partial^2 v_0}{\partial \theta^2} \right].$$

V denotes the annulus-averaged (mean) gas axial velocity; v_0 is the axial velocity perturbation evaluated at $\eta = 0$ (the inlet face of the compressor); Δ_p is the discharge-to-atmosphere pressure rise; η, θ are the axial and angular coordinates, respectively; $f(V + v_0)$ is the compressor static pressure rise characteristic, normally determined by rig tests. Additional modeling details can be found in [19]. A material balance on the gas in the diffuser and plenum chamber gives $l_c(d\Delta_p/d\tau) = [V - \gamma\sqrt{\Delta_p}]/(4B^2)$ where the throttle characteristic is used to represent the overall resistance to flow generated by the compressor and throttle valve (see Fig. 4). The parameter γ is proportional to the throttle opening and shifts the curve from right to left.

The flow perturbation is represented by the truncated trial function expansion

$$v(\eta, \theta, \tau) = \sum_{n=1}^N e^{2\pi n \eta} [a_{2n}(t) \cos 2\pi n \theta + a_{2n+1}(t) \sin 2\pi n \theta]. \quad (7)$$

The constant $B \rightarrow 0$ when the combined volume of the discharge diffuser and plenum is small relative to the

compressor volume and prevents surging flow [19] in this model.

A pseudo-spectral implementation of an aliasing-error-free formulation of the Galerkin projection is implemented using `MWRtools` functions by first defining a discretization array $\theta_i = i/M$. The trial functions used in the definition of V_{loc} are represented as $\psi = [1, \cos 2\pi\theta, \sin 2\pi\theta, \dots, \cos 2N\pi\theta, \sin 2N\pi\theta]$; therefore, because of the cubic nonlinearity of the compressor characteristic, we choose $M = 6N + 1$ to obtain an exact spectral decomposition of the residual. After computing the discrete differentiation arrays and transformation arrays that relate values on the “trial function” grid to the finer “residual” grid using the algorithms of `colmat.m`, the discretized residual is evaluated on the finer grid at each point in the time integration, and is projected onto the first $2N + 1$ trial functions *exactly* using the transformation array. An exact value of the discretization error can be computed in this solution framework with little additional computational cost. Representative simulation results are presented in Fig. 5; details of solution convergence assessment are presented in [17]. The simulation results presented here corroborate with published experimental observations of axial compressor stall [19]. The numerical discretization procedure presented provides a convenient method for testing flow instability control methods and alternate compressor models.

6 Plasma Etch – High-Degree Pseudo-Spectral Methods

Low pressure, weakly ionized, glow-discharge plasmas are used extensively for manufacturing integrated circuits (ICs) in both etching and deposition of thin films [14]. The discharges used in IC device manufacturing processes are characterized by interacting, nonlinear transport and reaction phenomena making first-principles simulation of a complete etch system virtually impossible. Therefore, researchers approach etch-system modeling problems by focusing on three submodels: the plasma physics (glow discharge), surface reaction, and electrically neutral transport (plasma chemistry) models [18]. Among them, the plasma physics submodel is considered to be the bottleneck in the development of detailed simulators. This model focuses on the ionization reaction kinetics and transport of the charged particles using modeling equations derived from the Boltzmann equations (e.g., [8]). The first three moments of the Boltzmann equations are similar in structure to the Navier-Stokes equations, and can be numerically stiff because of the differences in magnitude of the contributions of the electric drift versus diffusion terms.

The glow discharge modeling equations can be written in terms of four differential and partial differential equations. The first is the electric field (Poisson’s) equation, derived from Coulomb Law’s for a continuous media where the potential field results from the relative motion of electrons and positive ions:

$$\nabla \cdot \vec{E} = \frac{q}{\epsilon_0}(n_+ - n_e)$$

subject to the boundary conditions

$$V(0, t) = 0 \quad V(1, t) = V_{DC} + V_{RF} \sin(ft).$$

Therefore, the electrodes at $z = 0$ and $z = 1$ are the grounded and powered electrodes, respectively.

Two continuity equations describe the electron and ion species densities and are derived from a mass conservation law based on the rate of particle generation and accumulation being balanced by the divergence of the relative particle fluxes:

$$\begin{aligned} \frac{\partial n_e}{\partial t} + \nabla \cdot \vec{J}_e &= k_i N n_e \\ \frac{\partial n_+}{\partial t} + \nabla \cdot \vec{J}_+ &= k_i N n_e \end{aligned}$$

subject to boundary conditions

$$n_e(0, t) = 0 \quad \vec{J}_e(1, t) = -\gamma \vec{J}_+(1, t),$$

and

$$\nabla n_+(0, t) = 0 \quad \nabla n_+(1, t) = 0$$

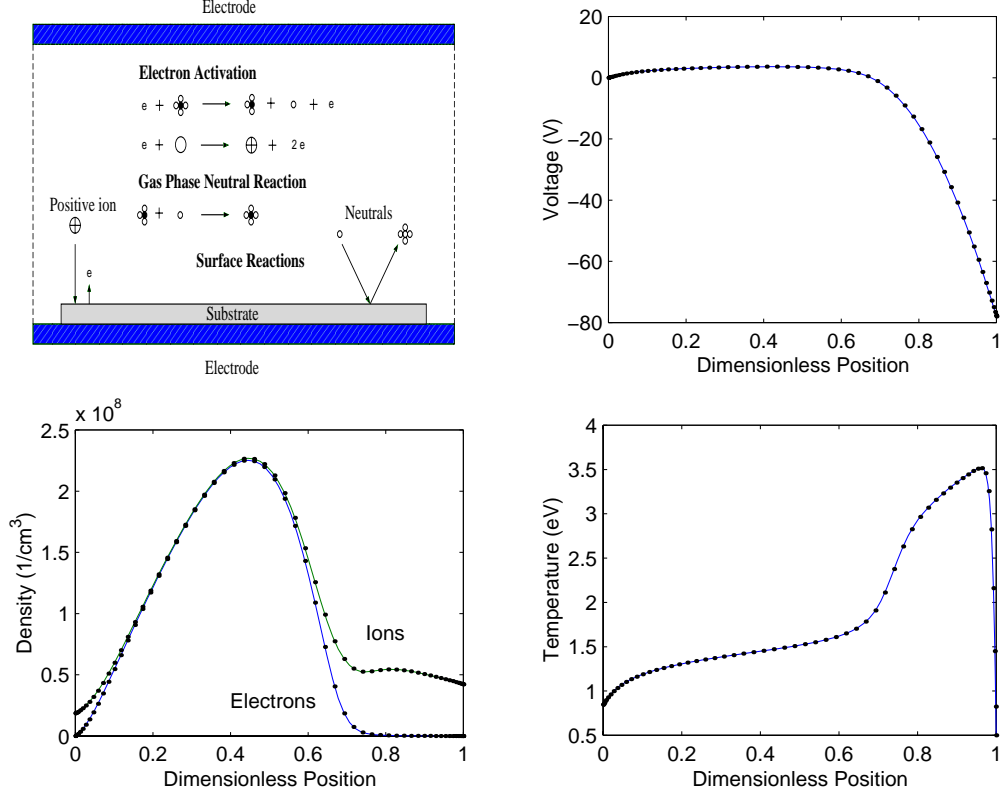


Figure 6: The parallel-plate etch reactor schematic diagram (top left), and glow discharge solutions including voltage profile (top right), electron and ion density (bottom left), and electron temperature (bottom right).

The electron temperature equation

$$\frac{\partial}{\partial t} \left(\frac{3}{2} n_e k_b T_e \right) + \nabla \cdot \vec{q}_e = (-e \vec{J}_e) \cdot \vec{E} - (k_i N n_e) H_{ei}$$

with boundary conditions

$$T_e(0, t) = T_{ec} \quad \nabla \cdot \vec{q}_e(1, t) = [-e \vec{J}_e(1, t)] \cdot \vec{E}(1, t)$$

suggests that for DC discharge solutions, the divergence of the electron energy flux is a result of balance between the electron ohmic heating $(-e \vec{J}_e \cdot \vec{E})$ and reaction cooling $(k_i N n_e H_{ei})$. The auxiliary quantities are defined by:

$$\begin{aligned} \vec{E} &= -\nabla V \\ \vec{J}_e &= -D_e \nabla n_e - \mu_e n_e \vec{E} \\ \vec{J}_+ &= -D_+ \nabla n_+ + \mu_+ n_+ \vec{E} \\ \vec{q}_e &= -K_e \nabla T_e + \frac{5}{2} k_b T_e \vec{J}_e. \end{aligned}$$

In the above equations, \vec{E} is the electric field, V is the potential, and n_e and n_+ are electron and ion densities, respectively. \vec{J}_e and \vec{J}_+ are electron and ion density fluxes, and \vec{q}_e is the electron enthalpy flux. The transport and kinetics parameters and other modeling details including the gas property constants and scaling parameters used are listed in [16].

The solution procedure calls `pd.m` to set up the physical domain; for this example, we discretize the four states using trial function expansions based on Jacobi polynomials, thus discretizing the modeling partial

differential equations directly with the differentiation arrays produced by `pd.m`. The four sets of residuals evaluated at the collocation points are set to zero, resulting in either (1) a large set of algebraic equations to be solved with a Newton-Raphson method for (steady-state) DC discharges, or (2) a large set of time-dependent ODEs constrained by the algebraic equations generated by the Poisson's equation and boundary conditions for the case of RF discharge simulations. Solution profiles between the collocation points are computed using `polyint.m`.

Representative results are presented in Fig. 6 for a self-sustained, stable DC glow discharge simulation. To date results of these simulations have been used primarily as a tool for understanding the role physical assumptions and boundary conditions play in the stability of numerical algorithms developed for glow discharge simulation [16]. The results presented can, however, be translated into factors important for determining wafer etch rates, making these simple-to-implement discretization methods valuable for quick assessment of related glow discharge models and etch process optimization methods.

7 Conclusions

In this paper, we presented a set of MATLAB-based computational elements that simplify the implementation of weighted residual methods for the solution of boundary-value based simulation problems. Presented in the context of four industrially-relevant problems, we find that the `MWRtools` functions constitute a solution procedure framework that is made possible by the one-to-one correspondence of the functions with the computationally-common operations of the the different MWR. The functions can be used for eigenfunction expansion, Galerkin-projection, and collocation-based discretization procedures, and allow rigorous discretization error assessment. The current library of functions can be obtained from the `MWRtools` project website located at <http://www.ench.umd.edu/software/MWRtools>.

Acknowledgments

This work was partially supported by a gift from the Isonics Corporation; additional support was provided by the National Science Foundation through grant NSF EEC 95-27 576 and the Petroleum Research Fund of the American Chemical Society through grant 31391-G9.

References

- [1] Adomaitis, R. A. and E. H. Abed 1996, Bifurcation analysis of nonuniform flow patterns in axial-flow gas compressors. *Proc. 1st World Congress of Nonlinear Analysts '92*, Lakshmikantham, V., Ed. 1597-1608.
- [2] Adomaitis, R. A. and Y. -H. Lin 1998, A technique for accurate collocation residual calculations. *Chem. Engng. J.*, accepted for publication; also ISR TR 98-6.
- [3] Andrews, L. C. 1985, *Special Functions for Engineers and Applied Mathematicians* Macmillan, New York.
- [4] Articolo, G. A., 1998, *Partial Differential Equations and Boundary Value Problems with Maple V* Academic Press, San Diego, CA.
- [5] Ascher, U., J. Christiansen, and R. D. Russell 1981, Collocation Software for Boundary-Value ODEs, *ACM Transactions on Mathematical Software*, **2**, 209-222.
- [6] Benedict, M., T. H. Pigford, and H. W. Levi 1981, *Nuclear Chemical Engineering* 2nd. Edition. McGraw-Hill, New York.
- [7] Chang, H.-Y. and R. A. Adomaitis 1997, Analysis of heat transfer in a chemical vapor deposition reactor: An eigenfunction expansion solution approach. *Int. J. Heat and Fluid Flow*, accepted for publication; also ISR TR 97-84.

- [8] Chen, F. F. 1980, *Introduction to Plasma Physics* Plenum Press, New York.
- [9] Childs, B., M. Scott, J. W. Daniel, E. Denman and P. Nelson 1979, *Codes for Boundary-Value Problems in Ordinary Differential Equations* Lecture Notes in Computer Science, No. 76, Springer-Verlag, New York.
- [10] Cooper, J. M. 1998, *Introduction to Partial Differential Equations with MATLAB* Birkhauser, Boston.
- [11] Dostrovsky, I. and M. Epstein 1982, The production of stable isotopes of oxygen. *Stable Isotopes* H.-L. Schmidt, H. Forstel, and K. Heinzinger, Eds. Elsevier Scientific, 693-702.
- [12] Gottlieb, G. and S. A. Orszag 1977, *Numerical Analysis of Spectral Methods: Theory and Applications* Society for Industrial and Applied Mathematics, Philadelphia.
- [13] Gottlieb, G. and C.-W. Shu 1997, The Gibbs phenomenon and its resolution. *SIAM Review* **39** 644-668.
- [14] Graves, D. B. 1989, Plasma processing in microelectronics manufacturing. *AIChE J* **35**, 1-29.
- [15] Graves, D. B. and K. F. Jensen 1986, A continuum model of DC and RF discharges. *IEEE Trans. on Plasma Sci.* **14**, 78.
- [16] Lin, Y. -h. and R. A. Adomaitis 1998, A global basis function approach to DC glow discharge simulation, *Phys. Lett. A*, **243** 142-150; also ISR TR 97-81.
- [17] Lin, Y. -h., Chang, H. -Y., and R. A. Adomaitis 1998, MWRtools: A library for weighted residual methods. *ISR TR 98-24*.
- [18] Lymberopoulos, D. P. and D. J. Economou 1994, Modeling and simulation of glow discharge plasma reactors. *J. Vac. Sci. Technol. A*. **12**, 1229.
- [19] Moore, F. K. and E. M. Greitzer 1986, A theory of post-stall transients in axial compression systems: Part I-Development of equations. *ASME J. Eng. Gas Turbines and Power*, 68-76.
- [20] Mac Cluer, C. R. 1994, *Boundary Value Problems and Orthogonal Expansions - Physical Problems from a Sobolev Viewpoint* IEEE Press, New York.
- [21] Villadsen, J. and M. L. Michelsen 1978, *Solution of differential equation models by polynomial approximation* Int. Series in Phys. and Chem. Engng. Sci., Prentice-Hall, Englewood Cliffs, N. J.
- [22] Villadsen, J. V. and W. E. Stewart 1967, Solution of boundray-value problems by orthogonal collocation, *Chem. Eng. Sci.*, 1483-1501.
- [23] *Partial Differential Equation Toolbox - For Use with MATLAB User's Guide* The MathWorks, Inc., Natick, MA, 1995.