Efficient Interpolant-Based Spatial Error Estimation for B-Spline Collocation Solutions of 1D Parabolic PDEs

Paul H. Muir (muir@smu.ca)

Mathematics & Computing Science

Saint Mary’s University

Thanks to

Organizers of SciCADE 2011 and

NSERC, MITACS, and Saint Mary’s University for funding

Outline

- Examples
- General Problem Class
- Method-Of-Lines(MOL) Software
- Overview of B-spline based, Adaptive COLlocation software for 1D PDEs: BACOL
- Alternative Error Estimation Schemes
- Future Work: Extensions to 2D PDEs
- Joint work with:
  Tom Arsenault, Tristan Smith, Jack Pew

Burgers' Equation

See, e.g., Adjerid et al. [1995]

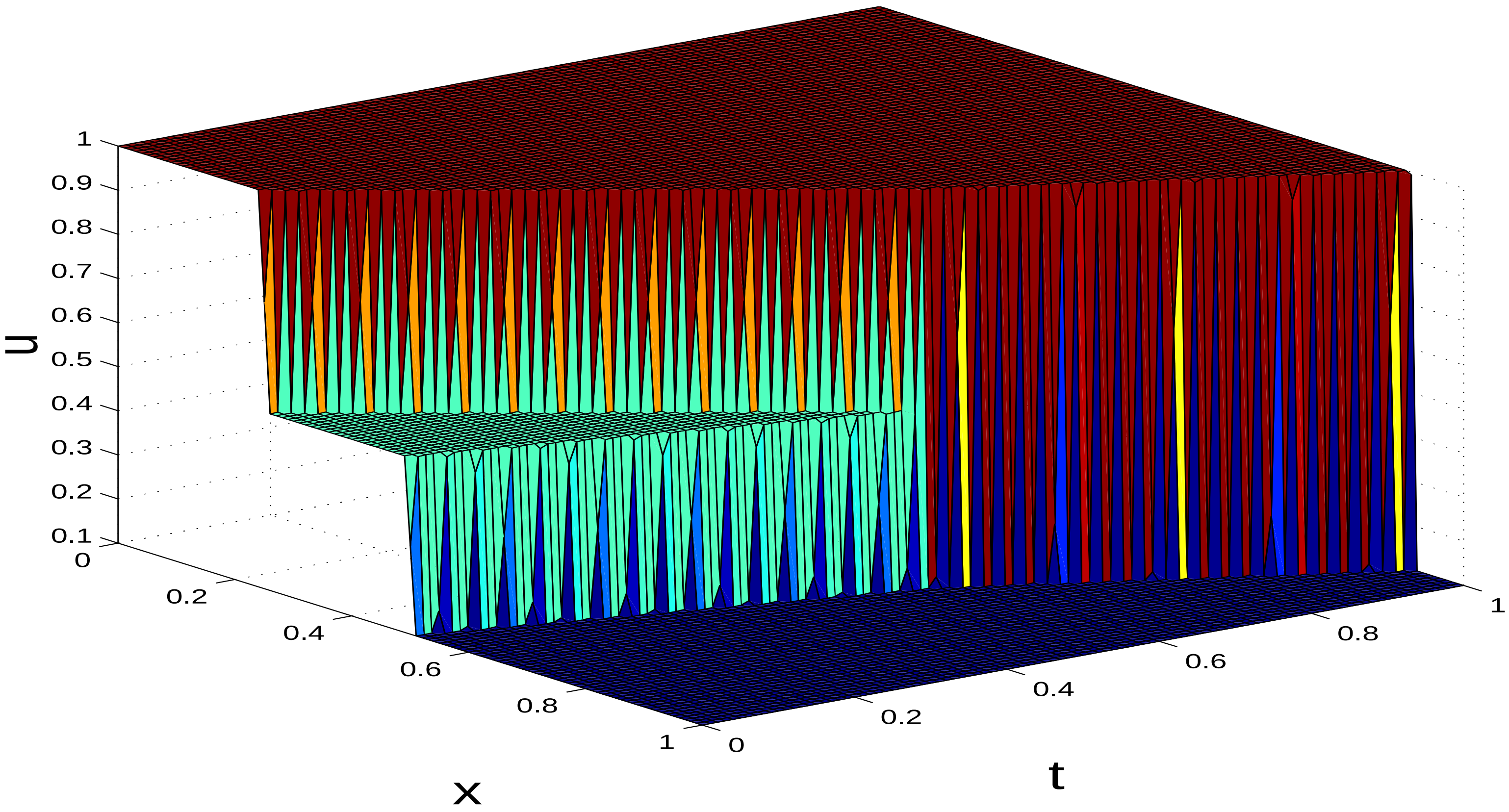
$$u_t = -uu_x + \epsilon u_{xx}, \qquad 0 < x < 1, \quad t > 0, \quad \epsilon > 0$$

Initial condition and boundary conditions chosen so that the exact solution is given by

$$u(x,t) = \frac{0.1e^{-A} + 0.5e^{-B} + e^{-C}}{e^{-A} + e^{-B} + e^{-C}},$$

where $A = \frac{0.05}{\epsilon}(x - 0.5 + 4.95t)$, $B = \frac{0.25}{\epsilon}(x - 0.5 + 0.75t)$, $C = \frac{0.5}{\epsilon}(x - 0.375)$, where ϵ is a problem dependent parameter

Burgers' Equation



Solution of Burgers' equation with $\epsilon = 10^{-4}$

Catalytic Surface Reaction

Reaction-diffusion-convection system, [Zhang, 1993]

$$\begin{aligned}
(u_1)_t &= -(u_1)_x + n(D_1 u_3 - A_1 u_1 \gamma) + (u_1)_{xx}/Pe_1, \\
(u_2)_t &= -(u_2)_x + n(D_2 u_4 - A_2 u_2 \gamma) + (u_2)_{xx}/Pe_1, \\
(u_3)_t &= A_1 u_1 \gamma - D_1 u_3 - R u_3 u_4 \gamma^2 + (u_3)_{xx}/Pe_2, \\
(u_4)_t &= A_2 u_2 \gamma - D_2 u_4 - R u_3 u_4 \gamma^2 + (u_4)_{xx}/Pe_2,
\end{aligned}$$

where $\gamma = 1 - u_3 - u_4, \quad 0 < x < 1 \quad t > 0$, and $Pe_1, Pe_2, D_1, D_2, R, A_1$, and A_2 are problem dependent parameters, with initial conditions

$$u_1(x,0) = 2 - r, \quad u_2(x,0) = r, \quad u_3(x,0) = u_4(x,0) = 0,$$

Catalytic Surface Reaction

and (mixed) boundary conditions:

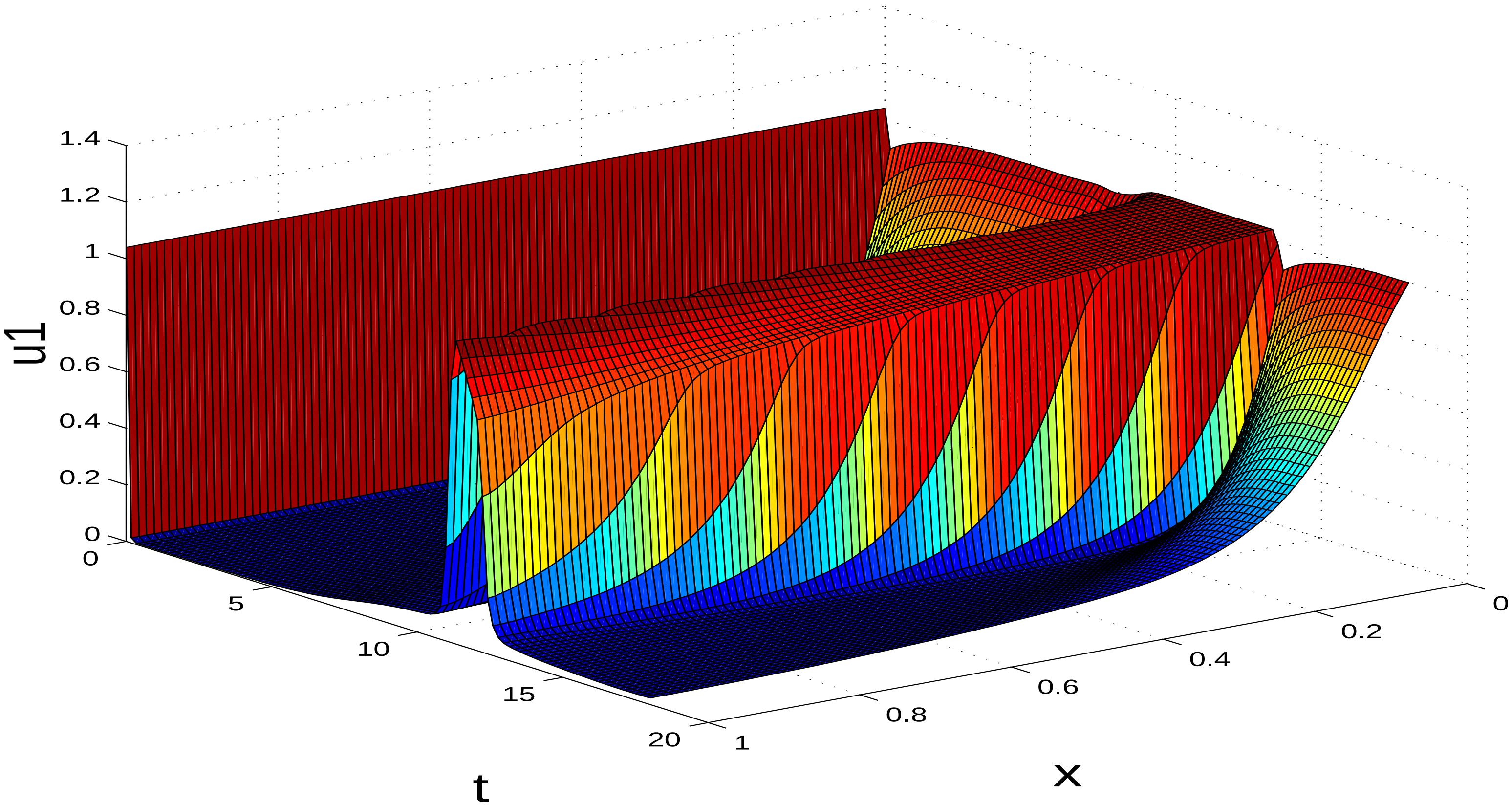
$$(u_1)_x(0,t) = -Pe_1(2 - r - u_1(0,t))$$

$$(u_2)_x(0,t) = -Pe_1(r - u_2(0,t))$$

$$(u_3)_x(0,t) = (u_4)_x(0,t) = 0$$

$$(u_1)_x(1,t) = (u_2)_x(1,t) = (u_3)_x(1,t) = (u_4)_x(1,t) = 0$$

Catalytic Surface Reaction



Catalytic Surface Reaction Model, $u_1(x,t)$

General Problem Class

$NPDE$ partial differential equations

$$u_t(x,t) = f\left(t, x, u(x,t), u_x(x,t), u_{xx}(x,t)\right),$$

$$a \leq x \leq b, \quad t \geq t_0,$$

initial conditions

$$u(x,t_0) = u_0(x), \qquad a \leq x \leq b,$$

(separated) boundary conditions

$$b_L\left(t, u(a,t), u_x(a,t)\right) = b_R\left(t, u(b,t), u_x(b,t)\right) = 0$$

MOL Software

- “Production Level" or “Library Level" software packages based on well-established algorithms, designed for a general problem class
- e.g.,
  - LINPACK, LAPACK, in numerical linear algebra,
  - QUADPACK in numerical integration,
  - IMSL, NAG, Netlib

• We focus on “Library Level" software packages for 1D time-dependent PDEs

MOL Software

- Spatial mesh which partitions spatial domain + spatial discretization of PDE by, e.g., finite differences, finite elements, collocation
  $\Rightarrow$ PDE approximated by system of ODEs
- ODEs + boundary conditions
  $\Rightarrow$ Differential-Algebraic Equations (DAEs)
- $\Rightarrow$ Takes advantage of the availability of high quality DAE solvers that adapt stepsize/order of formula to control temporal error estimate

MOL Software

Spatial Error Adaption/Control

- I: No spatial adaptation/error control
  PDECOL, [Madsen,Sincovec, 1979],
  EPDCOL, [Keast,Muir, 1991]
- II:Adaptive spatial mesh via moving mesh strategy
  (r refinement) but no spatial error control
  D03PPF, [NAG] from SPRINT,
  [Berzins,Dew,Furzeland, 1989],
  TOMS731, [Blom,Zegeling, 1994],
  MOVCOL, [Huang,Russell,1996]

MOL Software

- III: Spatial adaptation and error control
  - Computation of a high order estimate of spatial error
  - Tolerance check of spatial error estimate for every successful timestep
  - Mesh adaptation: refinement and redistribution
  - Adaptation of order of discretization method

HPNEW, [Moore,2001], hp refinement
BACOL [Wang,Keast,Muir, 2004a, 2004b, 2004c], h refinement
BACOLR [Wang,Keast,Muir, 2008] h refinement

B-spline Adaptive COLlocation

- BACOL
  - spatial discretization
  - spatial error estimation and adaptive control
  - temporal error estimation and adaptive control

Spatial Discretization

- Spatial mesh, $\{x_i\}_{i=0}^{N}$, $x_0 = a$, $x_N = b$
- Approximate solution,

$$U_s(x,t) = \sum_{i=1}^{NC} y_{i,s}(t) B_i(x), \quad NC = N(p-1)+2,$$

  s=1,. . . ,NPDE

  $\{B_i(x)\}_{i=1}^{NC}$ - B-spline basis polynomials of degree p based on B-Spline Package, [deBoor,1977]

- $y_{i,s}(t)$ are unknown time-dependent coefficients for the sth PDE component

Spatial Discretization

- $U_s(x,t)$ required to satisfy PDEs at collocation points on each subinterval $\Rightarrow$ system of ODEs
- ODEs plus boundary conditions give index-1 DAE system:

$$
\begin{aligned}
0 &= b_L(t, U(0,t), U_x(0,t)) \\
\frac{d}{dt}U_s(\xi_l, t) &= f_s(t, \xi_l, U(\xi_l, t), U_x(\xi_l, t), U_{xx}(\xi_l, t)), \\
& \quad s = 1, \ldots, NPDE,\ l = 1, \ldots, N(p-1) \\
0 &= b_R(t, U(1,t), U_x(1,t))
\end{aligned}
$$

where ξ_l is lth collocation point (Gauss points on each subinterval)

Spatial Error Estimate

- For spatial error estimate, a second (global) collocation solution, $\bar{U}(x,t)$, of degree $p+1$ is computed
- DAE systems for $U(x,t)$ and $\bar{U}(x,t)$ are integrated simultaneously
- After every successful timestep, we compute, $E_s(t)$, for sth solution component over whole problem interval:

$$E_s(t) = \sqrt{\int_a^b \left(\frac{U_s(x,t) - \bar{U}_s(x,t)}{ATOL_s + RTOL_s|U_s(x,t)|} \right)^2 dx}$$

- t is current time; $ATOL_s, RTOL_s$: absolute, relative error tolerances

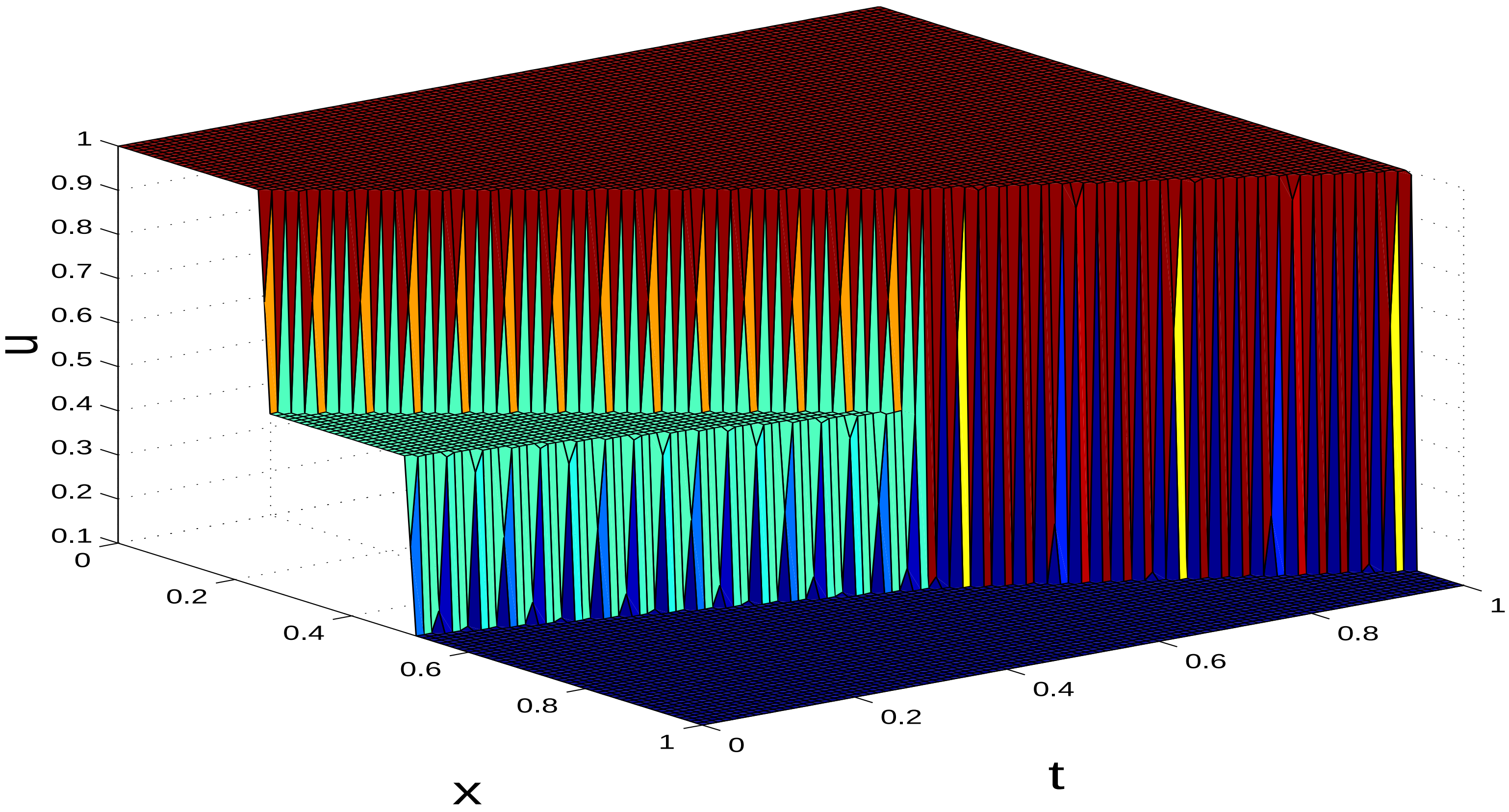
Spatial Error Estimate

- If $\max_{s=1}^{NPDE} E_s \geq 1$, (tolerance not satisfied), then
- reject current step and perform global redistribution/refinement of spatial mesh based on error estimates, $\hat{E}_i(t), i = 1, \ldots, N$, where

$$\hat{E}_i(t) = \sqrt{\sum_{s=1}^{NPDE} \int_{x_{i-1}}^{x_i} \left(\frac{U_s(x,t) - \bar{U}_s(x,t)}{ATOL_s + RTOL_s |U_s(x,t)|} \right)^2 dx}$$

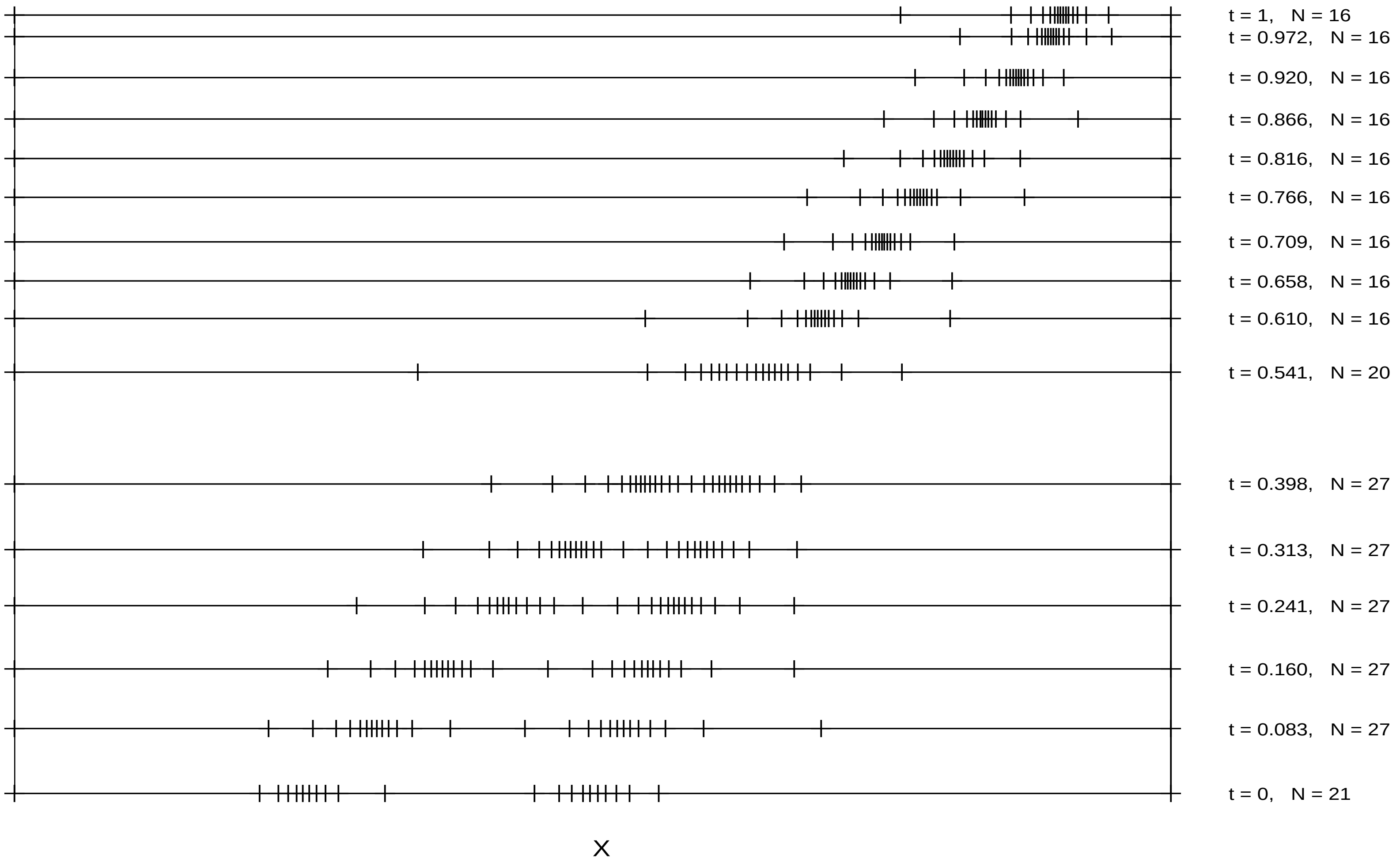
- These give a measure of the error distribution over the subintervals of the spatial mesh at time t.
- Spatial remeshing based on equidistribution principle

Spatial Mesh Adapt.



Solution of Burger's equation, $\epsilon = 10^{-4}$

Spatial Mesh Adapt.



Time Integration

BACOL/DASSL:

- “Double” DAE system treated by DASSL, [Petzold, 1982] modified to add option for COLROW package
- Family of Backward Differentiation Formulas (BDF) - Multistep Methods
- “Warm” restarts (same order, same stepsize) after remeshings, based on high order interpolation of solution values from previous mesh
- Variable order, 1 to 5

Relation to BVODE Software

- MOL Software (with Adaptive Spatial Error Control) $\approx$ Boundary Value ODE Software for spatial domain coupled with DAE software for time stepping
- In particular, spatial discretization scheme of BACOL $\approx$ discretization scheme of BVODE solver COLSYS [Ascher, Christensen, Russell 1981]
- Software consisting of COLSYS interfaced with DASSL would be similar to BACOL, although (spatial) error estimation scheme is fundamentally different

Comparisons

- BACOL: [Wang, Keast, Muir, 2004b], "A comparison of adaptive software for 1-D parabolic PDEs"
- BACOL compared with
  EPDCOL, D03PPF, TOM731, MOVCOL, HPNEW
- BACOL shown to be more efficient than these packages, especially for higher accuracy computations and problems with rapid spatial variation

Alternative Error Estimates

- Recall that BACOL error estimate involves the computation of two global collocation solutions

$$E_s(t) = \sqrt{\int_a^b \left(\frac{U_s(x,t) - \bar{U}_s(x,t)}{ATOL_s + RTOL_s|U_s(x,t)|} \right)^2 dx}$$

- Approach I: Replace higher order collocation solution, $\bar{U}(x,t)$, by interpolant of same order; uses a Superconvergent Interpolant (SCI)

- Approach II: Replace lower order collocation solution, $U(x,t)$, by interpolant of same order; uses a Lower Order Interpolant (LOI)

Superconvergent Interpolant

- Lower order collocation solution, $U(x,t)$, is the primary solution; higher order collocation solution, $\bar{U}(x,t)$, is computed only for use in error estimate
- (Auxiliary computation to obtain a higher accuracy solution for error estimation, e.g., Gauss-Kronrod quadrature, formula pairs for IVPs, etc.)
- Basic idea: replace higher order collocation solution, $\bar{U}(x,t)$, by interpolant of same order; need extra computation to obtain higher order values?
- No, higher accuracy solution info for interpolant is available for free!

Superconvergent Interpolant

- BACOL spatial discretization:
  collocation at Gauss points
- Theory from BVODEs: collocation solution has leading order error term containing the following factor:

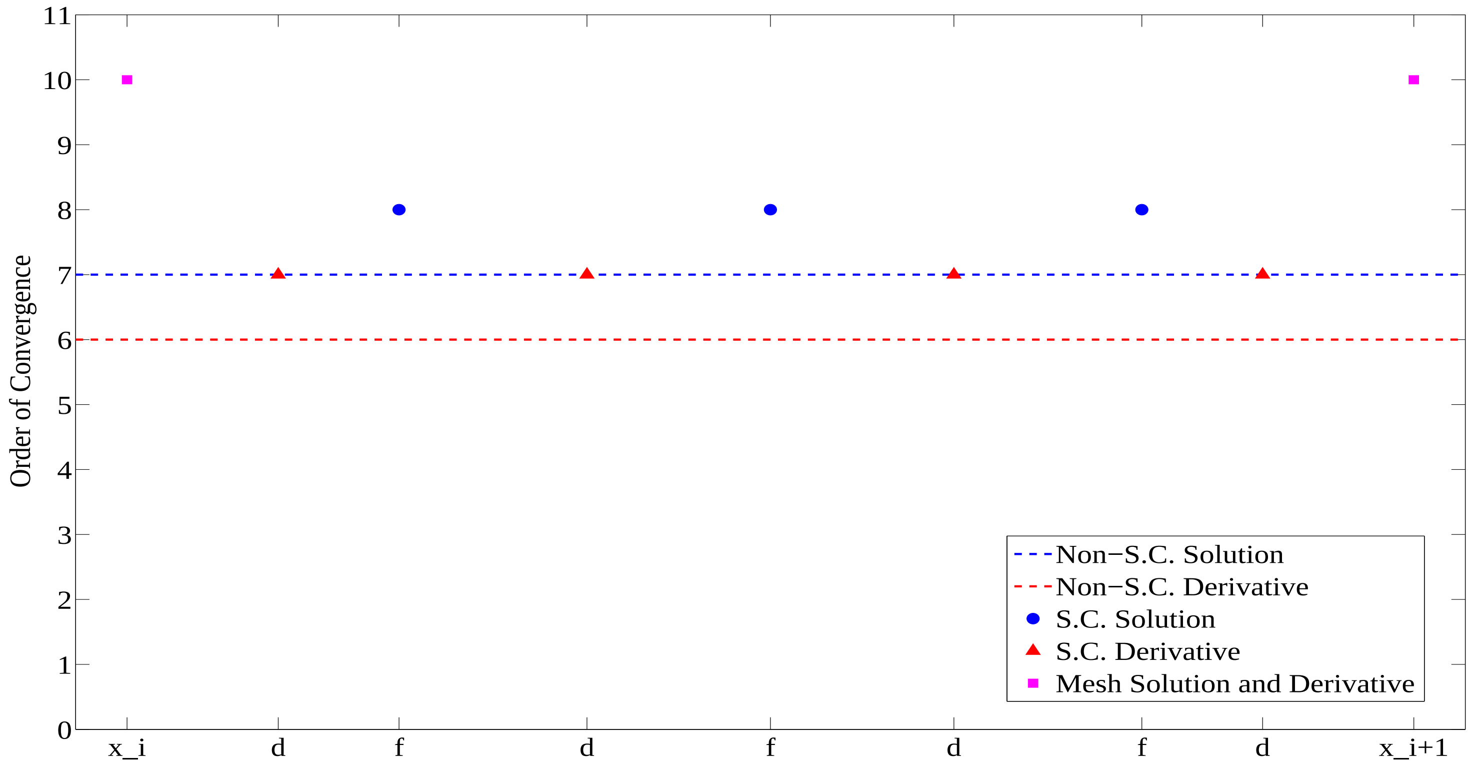
$$P(x) = \frac{1}{p!} \int_0^x (t-x) \prod_{l=1}^{p} (t-\rho_l) dt,$$

where ρ_l are Gauss points on $[0, 1]$

Superconvergent Interpolant

- Evaluation of collocation solution at points corresponding to roots of $P(x)$ on each subinterval
  - $\Rightarrow$ leading order error term is zero
  - $\Rightarrow$ collocation solution is superconvergent at such points on each subinterval
  - $\Rightarrow$ one extra order of accuracy
- Even better superconvergence at mesh points

Superconvergent Interpolant

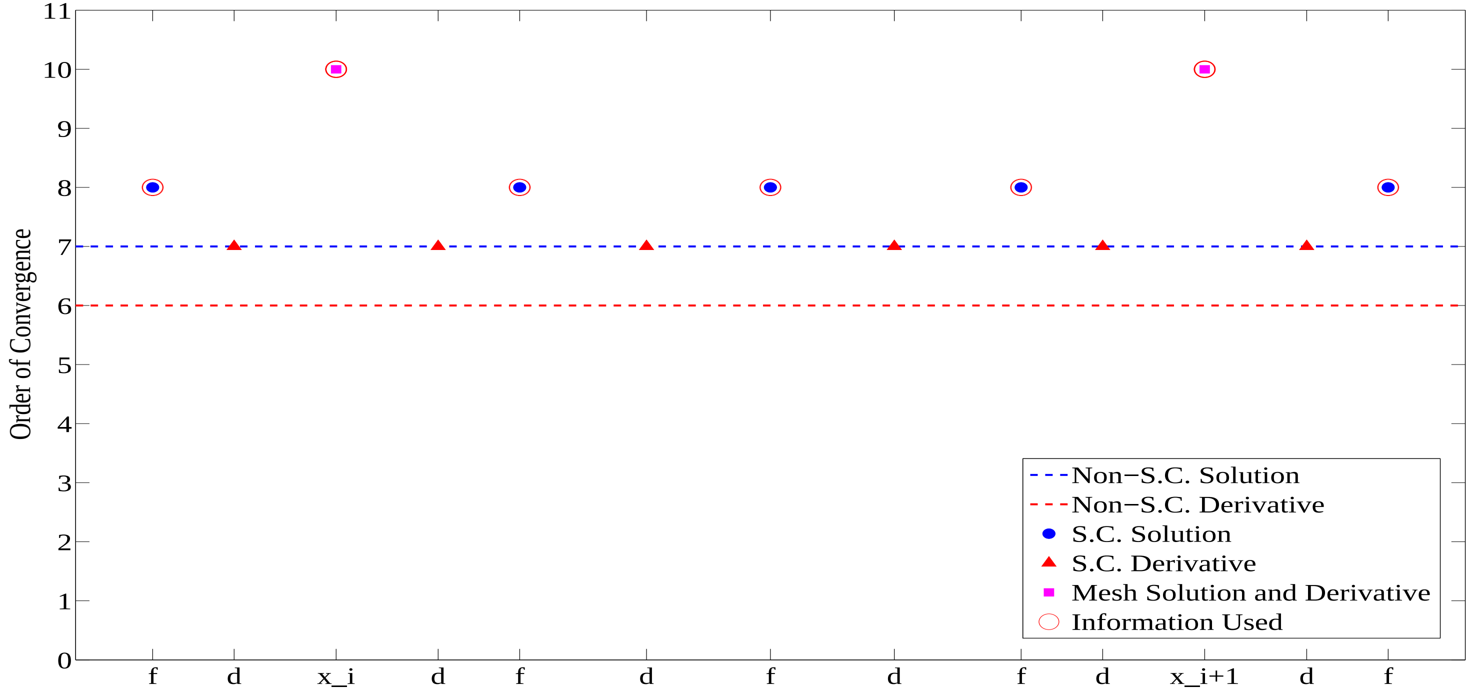


Collocation solution: superconvergent points ($p = 6$)

Superconvergent Interpolant

- Main idea:
  Replace order $p+1$ global collocation solution with local interpolant, of order $p+1$, based on superconvergent solution and derivative values
- Want data error to dominate interpolation error
- However, interpolant existence issues arise if data values are all from local subinterval
- $\Rightarrow$ Need to use two superconvergent values from outside subinterval

Superconvergent Interpolant



SCI uses mesh point solution/derivative values, all internal solution values and two external solution values

Hermite-Birkhoff Interpolant

- SCI based on Hermite-Birkhoff interpolant, using superconvergent solution and derivative values
- Interpolation error term [Finden, 2008] for $p = 6$, on ith subinterval, $[x_i, x_{i+1}]$, depends on

$$\phi(x) = \left[x^2 - (R\alpha + L\beta)x - R\alpha + L\beta + \frac{LR}{3} - 1\right]$$

  - where $\alpha = \frac{1}{2} - \frac{1}{6}\sqrt{3}$, $\beta = \frac{1}{2} + \frac{1}{6}\sqrt{3}$, $R = \frac{x_{i+2} - x_{i+1}}{x_{i+1} - x_i}$, $L = \frac{x_i - x_{i-1}}{x_{i+1} - x_i}$ are left and right adjacent subinterval ratios
  - $\Rightarrow$ Issues when adjacent subinterval ratios are large

Lower Order Interpolant

- $\Rightarrow$ A change in viewpoint: Higher order collocation solution, $\bar{U}(x,t)$, is propagated forward in time; lower order collocation solution is used only for error estimate (Local extrapolation)
- For error estimate, we replace lower order collocation solution, $U(x,t)$, by interpolant of same order - the LOI
- LOI interpolates data from higher order solution $\bar{U}(x,t)$
- Main idea: Interpolation points chosen so that leading order term in interpolation error is asymptotically equivalent to leading order term in lower order collocation solution error [Moore 2004]

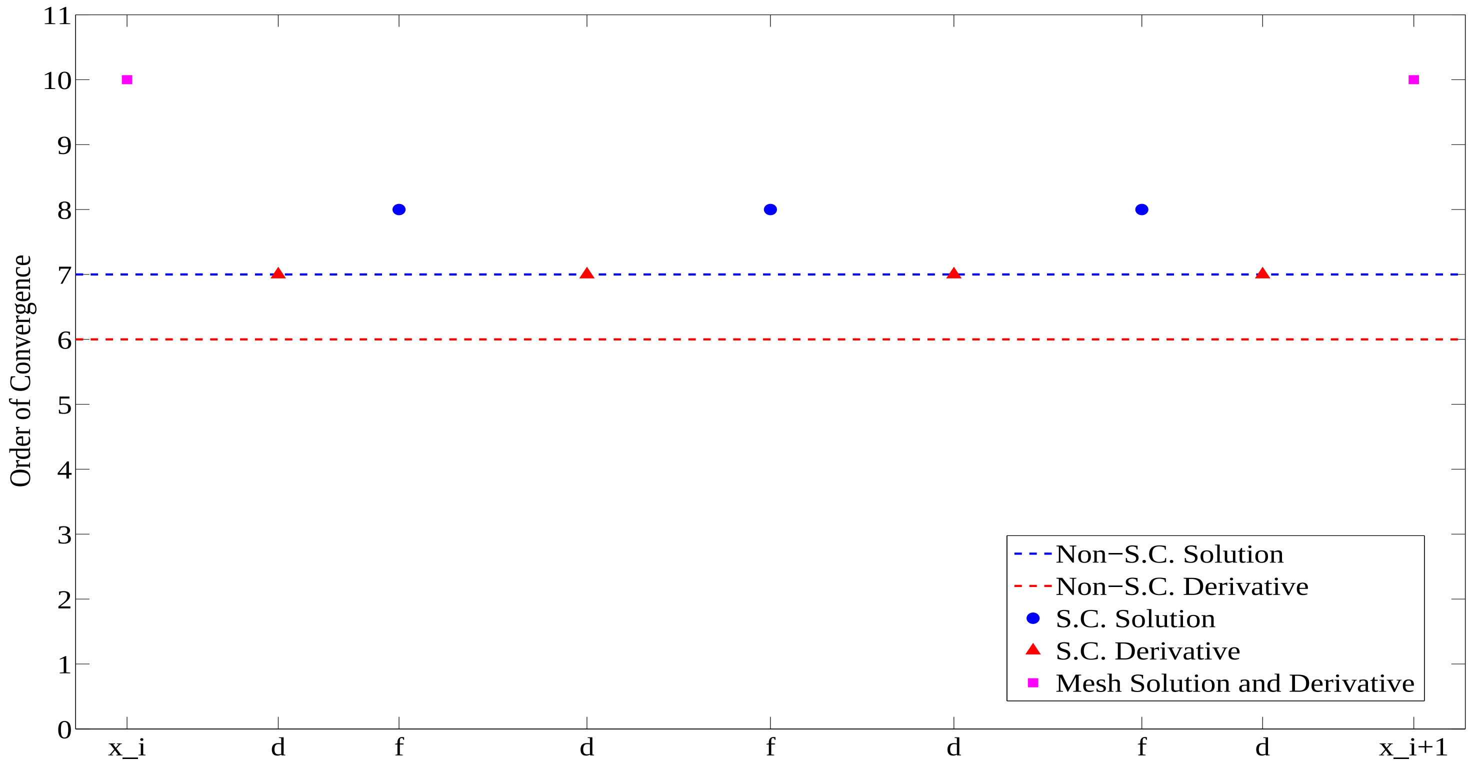
Lower Order Interpolant

- We want interpolation error to dominate data error
- Interpolation points chosen so that factor that depends on x in leading order term in interpolation error equals factor that depends on x arising in leading order term in collocation error:

$$P(x) = \frac{1}{p!}\int_0^x (t-x)\prod_{l=1}^{p}(t-\rho_l)dt,$$

- LOI based on Hermite-Birkhoff interpolant
- All interpolation points are from current subinterval $\Rightarrow$ Error does not depend on adjacent subinterval ratios

Lower Order Interpolant

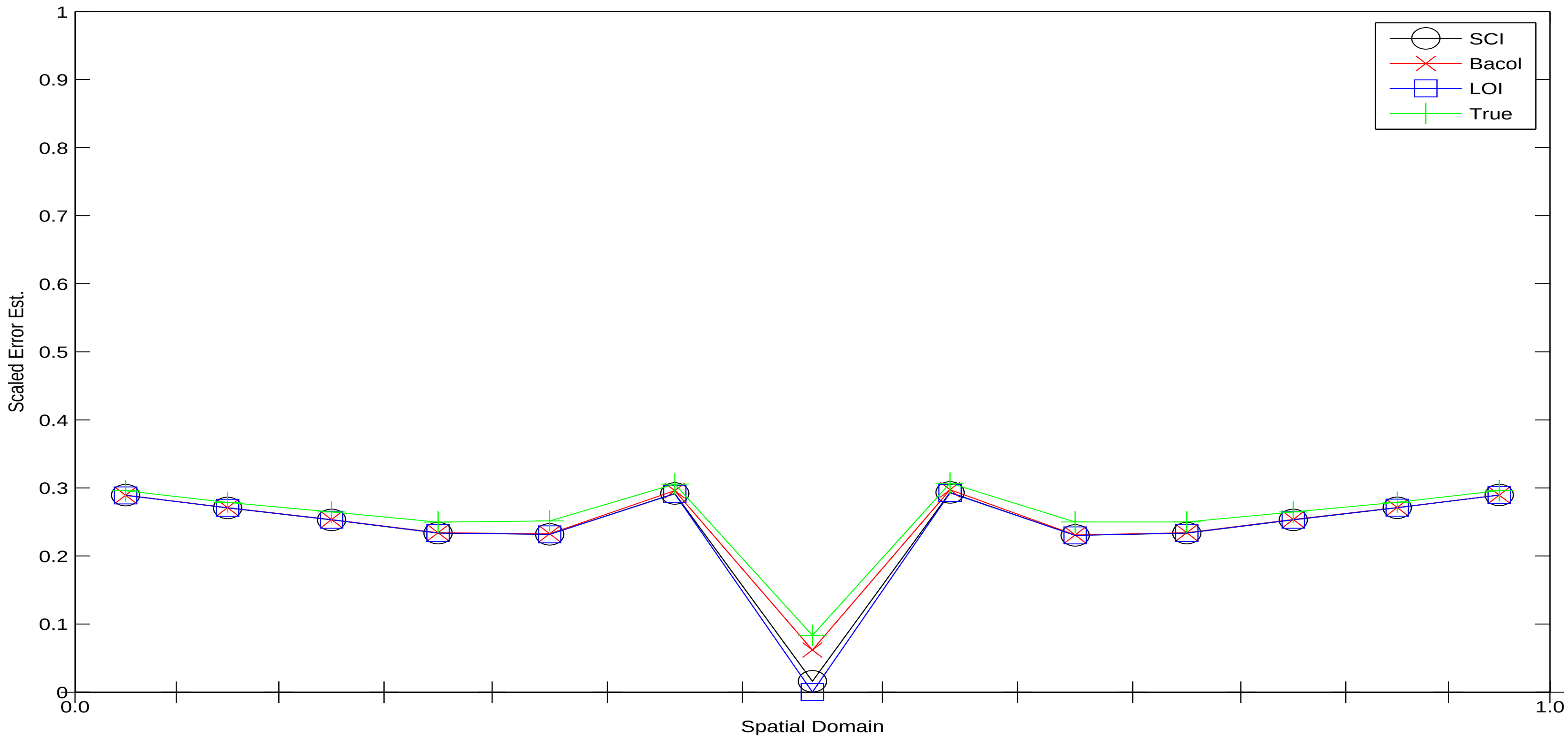


Interpolation points for LOI

Numerical Results

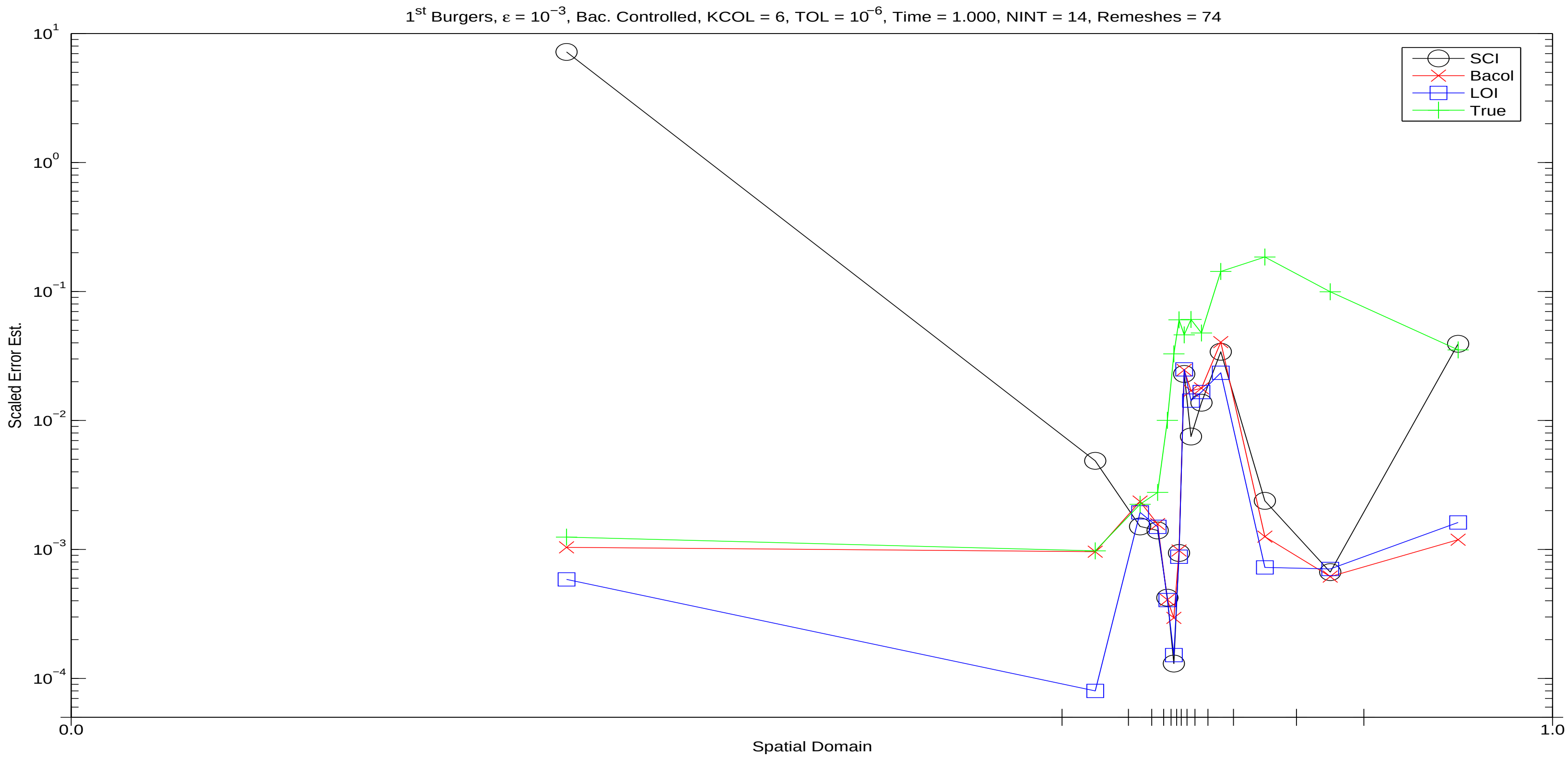
- Error estimate from SCI - $\bigcirc$, from BACOL - $\times$, from LOI - $\square$; True Error - $+$
- Mesh adaptation
  - controlled by BACOL estimate
  - controlled by SCI estimate
  - controlled by LOI estimate
- Results for simple test problem [Sincover, Madsen, 1979], with $p = 4$, $ATOL_s = RTOL_s = 10^{-8}$
- Results for Burgers' equation, with $\varepsilon = 10^{-3}$ ($p = 7$, $ATOL_s = RTOL_s = 10^{-6}$)

Numerical Results



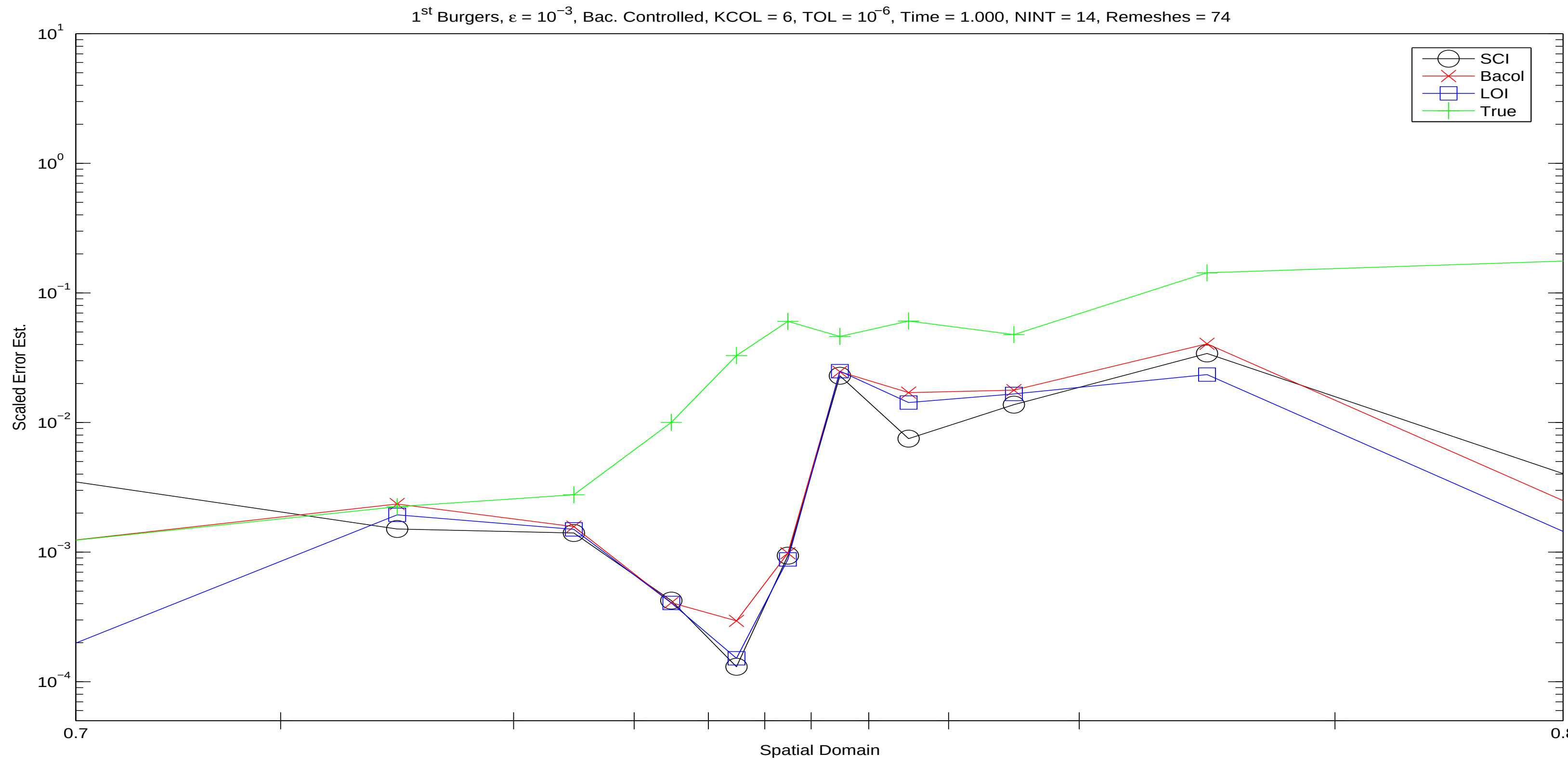
BACOL estimate controls mesh

Numerical Results



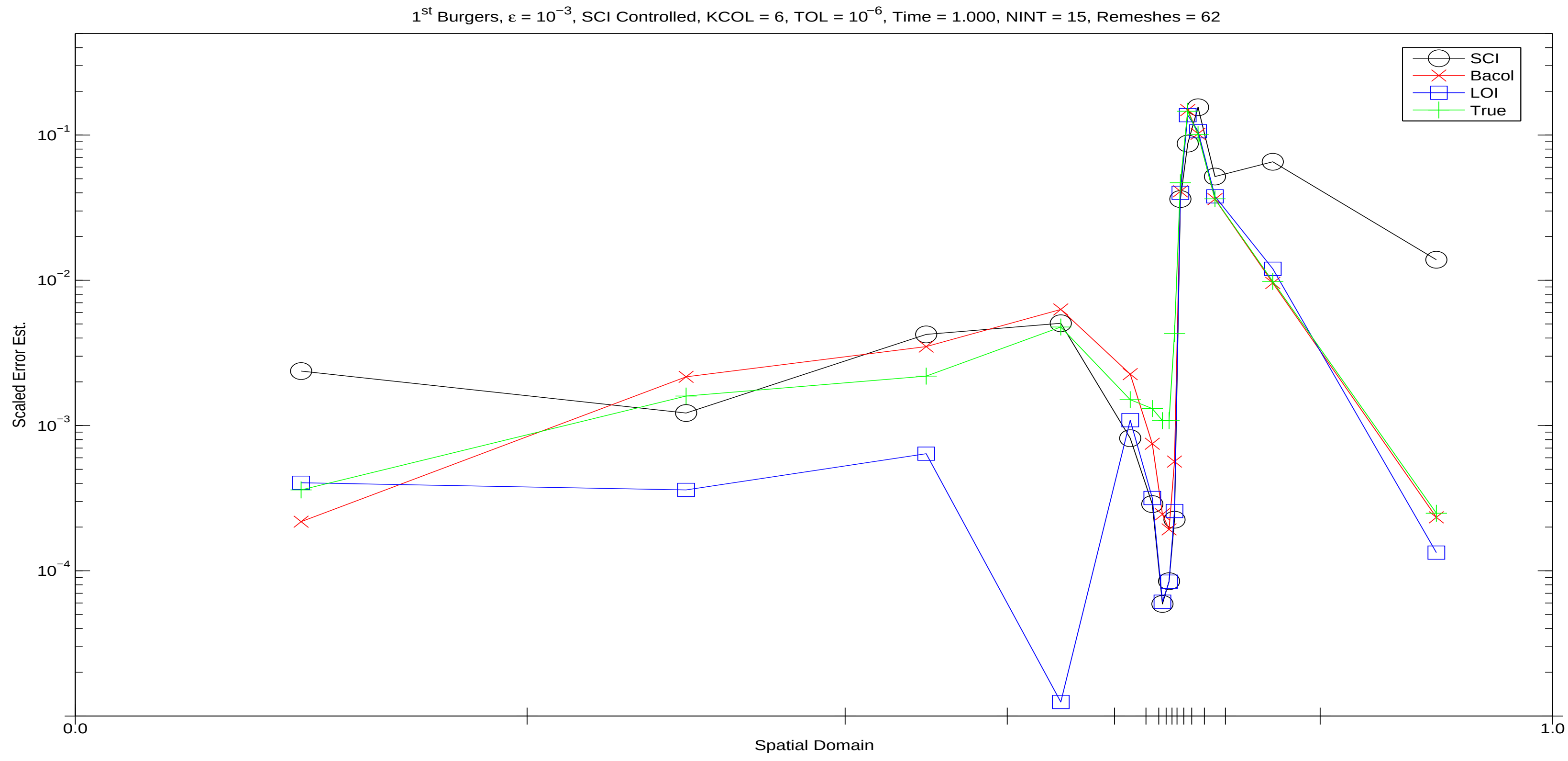
BACOL estimate controls mesh, Full Spatial Domain

Numerical Results



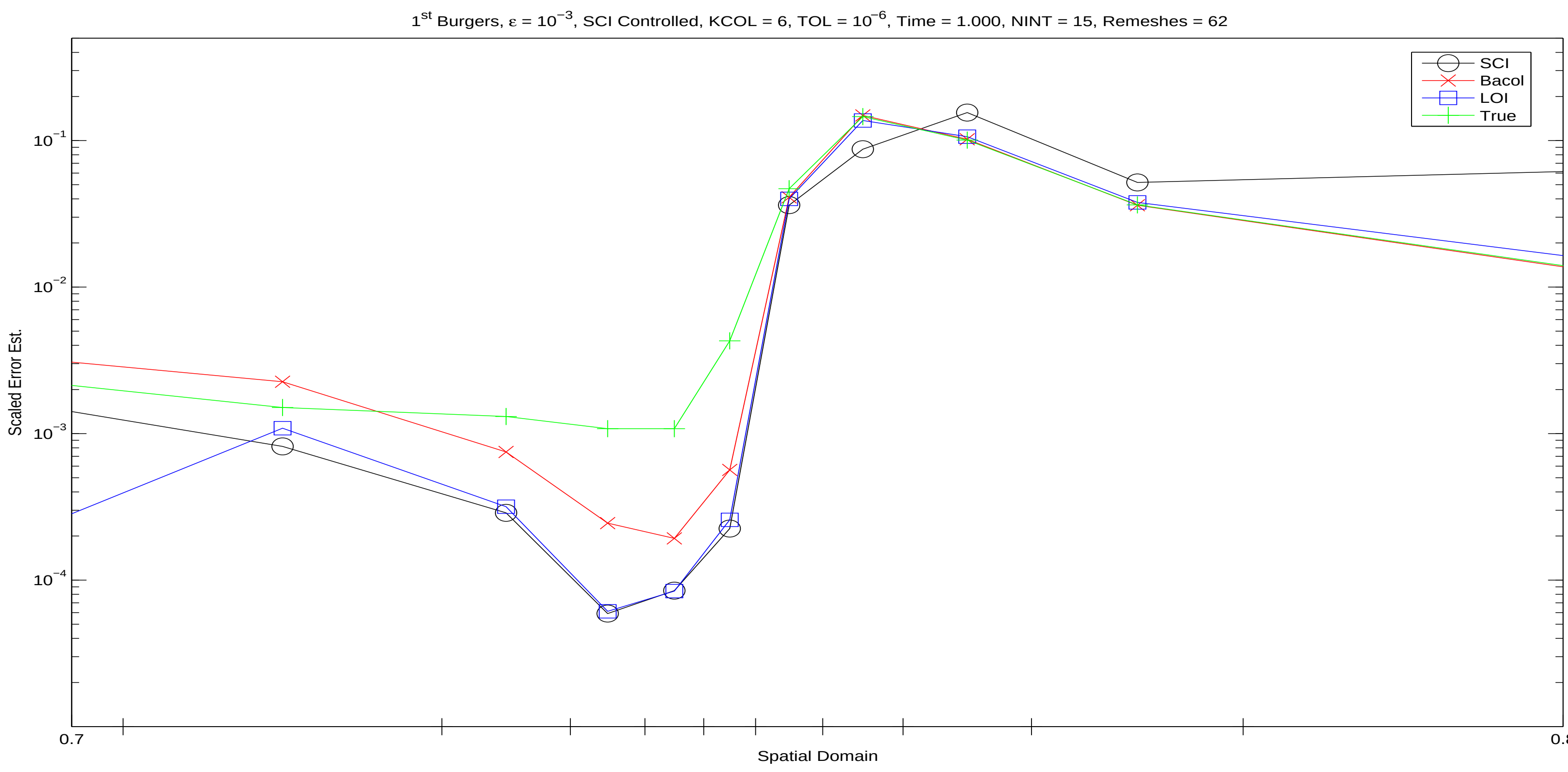
BACOL estimate controls mesh, Layer Region

Numerical Results



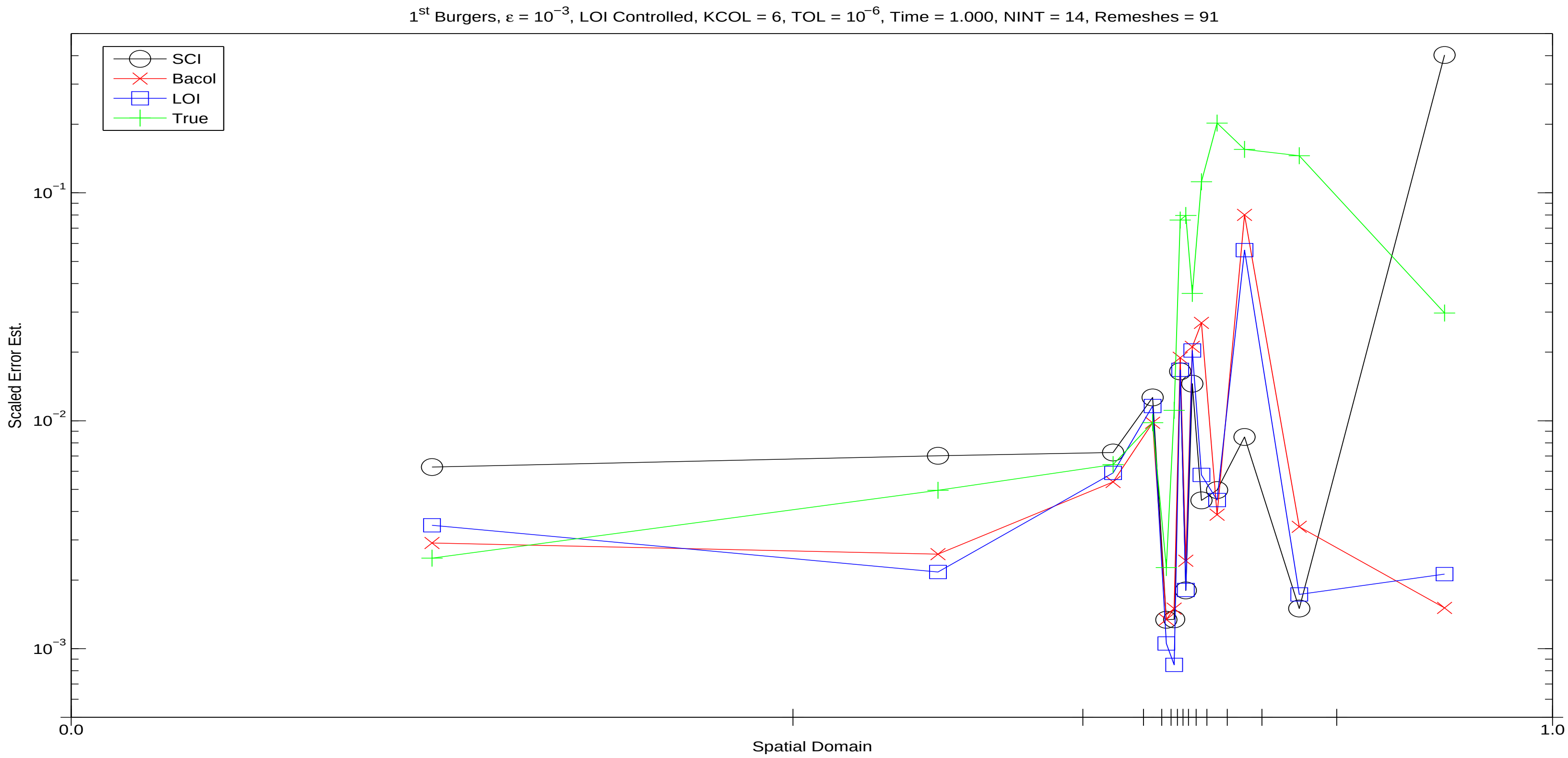
SCI estimate controls mesh, Full Spatial Domain

Numerical Results



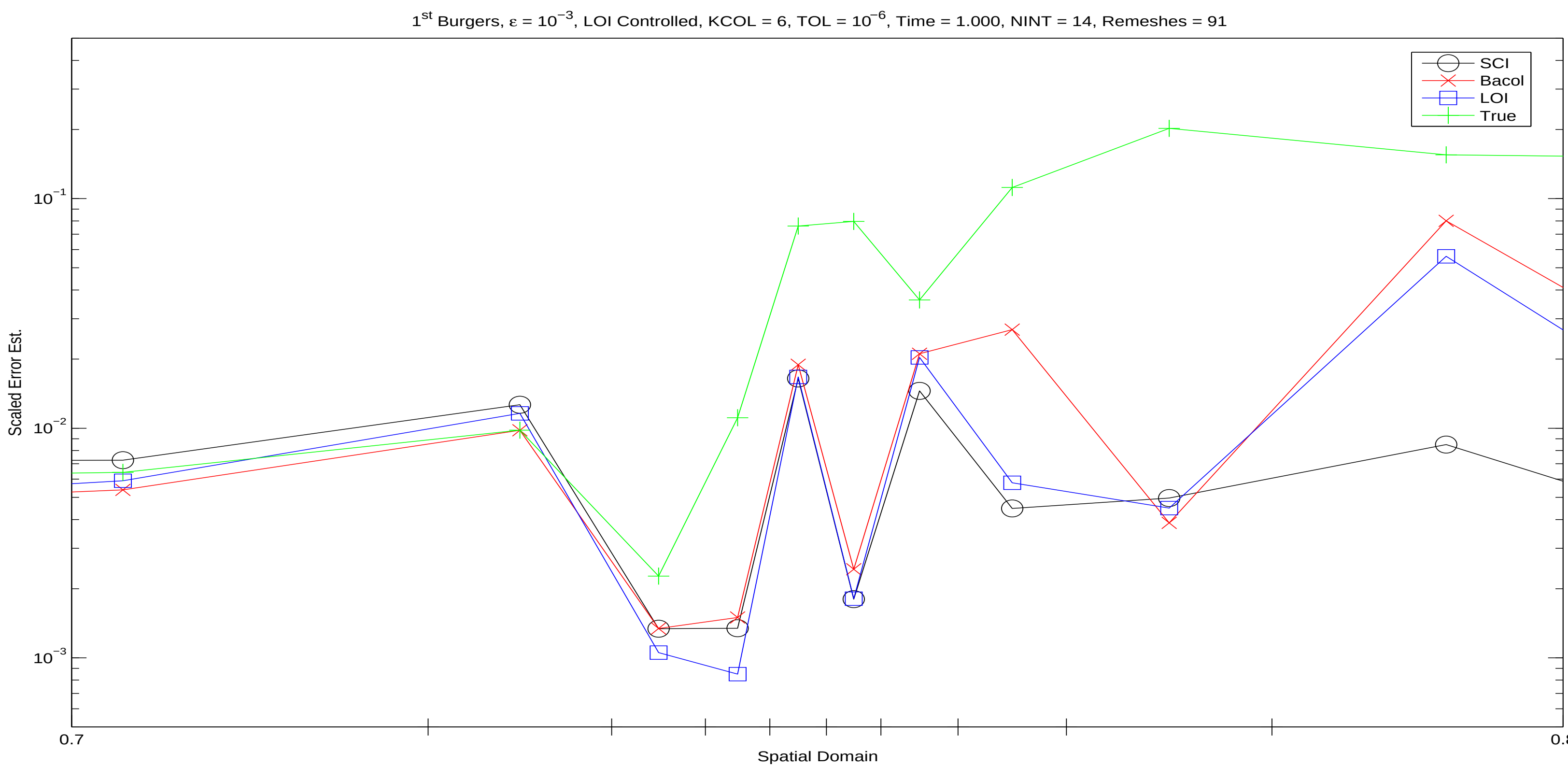
SCI estimate controls mesh, Layer Region

Numerical Results



LOI estimate controls mesh, Full Spatial Domain

Numerical Results



LOI estimate controls mesh, Layer Region

Numerical Results - Observations

- For simple problems, all estimates in good agreement with each other and true error
- For problems with sharp layer regions:
  - For BACOL controlled meshes, some SCI error estimates are too large but ...
  - SCI controlled meshes lead to “self correction”: meshpoints are moved, a few added,
  - LOI estimates are generally in good agreement with BACOL estimates (LOI control $\approx$ BACOL control)
  - All schemes underestimate error in layer region to some extent

Computational Costs

- Order $p+1$ global solution computation about same cost as order p computation: setup extra B-spline basis, solution of second DAE system $\Rightarrow$ standard BACOL error estimate doubles cost of computation
- SCI/LOI approaches involve only evaluation of global solution and evaluation of Hermite-Birkhoff interpolant
- SCI self-correction $\Rightarrow$ small number extra subintervals
- Number of remeshings $\approx$ same for all schemes
- $\Rightarrow$ SCI/LOI-based error estimates much less expensive than original BACOL error estimate

Extension to 2D PDEs

The Method of Surfaces [Zhi Li 2011]

- Takes advantage of the presence of good quality software for time-dependent 1D PDEs
- Apply a standard discretization (as in the standard MOL algorithm) to discretize the y domain, reducing the 2D PDE to a system of 1D PDEs
- Apply software for 1D PDEs to return a set of surfaces (in t and x), each of which is associated with a discrete point of the y domain
- No adaptation or error control in y domain

Extension to 2D PDEs

Generalization of BACOL to 2D: [Zhi Li 2011]

- 2D collocation (tensor product formulation)
- DASPK/sparse linear system solver
- Efficient error estimators for 2D Gaussian collocation solutions
- 2D mesh adaptation

The End

- See Technical Report (cs.smu.ca/tech_reports/txt2011_001.pdf) for many more numerical results
- SCI approach [Arsenault, Smith, Muir, CAMQ, 2011]

Thank You