# AFOSR Workshop 2015: Collaborative Efforts in the Development of SENSEI

**Dr. Roy and Andrew McCall**

**Department of Aerospace & Ocean Engineering**
**Virginia Tech**

# Overview



- Personnel updates
- SENSEI
- Collaborations
  - Dr. Feng: Hybrid Parallelism of SENSEI
  - Dr. Sandu: Matrix-Free Time Stepping
    - SENSEI-Lite
    - Application to SENSEI
  - Dr. de Sturler: Linear Solvers
    - Modified CUDA-ITSOL library
    - Implicit Multi-block

# Personnel

We have had significant turnover in our research group this year

- Brent Pickering has left the group for a contractor position at NASA Langley
- Andrew McCall (2nd year MS student) has taken over
- We have added a new postdoc today: Behzad Baghapour (joint AOE/Math)

# SENSEI



- Finite volume compressible CFD solver
  - Multi-block
  - Structured grid
  - Second order (MUSCL extrapolation)
- Multiple time integration methods:
  - Multi-stage Runge-Kutta
  - Euler Implicit
- Multiple upwind flux schemes:
  - Roe
  - Van Leer
  - Steger-Warming
  - AUSM

# Overview



- Personnel updates
- SENSEI
- Collaborations
  - Dr. Feng: Hybrid Parallelism of SENSEI
  - Dr. Sandu: Matrix-Free Time Stepping
    - SENSEI-lite
    - Application to SENSEI
  - Dr. de Sturler: Linear Solvers
    - Modified CUDA-ITSOL library
    - Implicit Multi-block

# Parallelization Paradigms



- MPI
  - MPICH or Open MPI
  - Distributed and/or shared memory (MPI-3.0)
- Hybrid MPI+OpenMP
  - MPI distributed, OpenMP shared
- Hybrid MPI+OpenACC
  - MPI distributed, OpenACC heterogeneous
- MPI+OpenMP/OpenACC

# Additional Studies



- Loop level vs. block level parallelization
  - Implementation decision for OpenMP
  - Currently, implemented with block level
  - Add and compare loop level implementation
- Dynamic runtime library (W. Feng)
  - Optimize CPU/GPU loading during runtime

# Additional Studies (cont'd)



- OpenACC: Parallelizing the residual assembly parts of SENSEI
  - Brent has implemented a standalone code to simulate the residual calculation in SENSEI
    - Applied OpenACC directives to parallelize calculations of cell residuals
    - Running into performance issues with offloaded code
  - Plan to finish implementation with OpenACC directives and resolve performance issues
    - Incrementally apply OpenACC to SENSEI

# Performance Metrics



- Speedup
- Parallel efficiency
- Scalability
- Memory usage

# Overview



- Personnel updates
- SENSEI
- Collaborations
  - Dr. Feng: Hybrid Parallelism of SENSEI
  - Dr. Sandu: Matrix-Free Time Stepping
    - SENSEI-Lite
    - Application to SENSEI
  - Dr. de Sturler: Linear Solvers
    - Modified CUDA-ITSOL library
    - Implicit Multi-block

# SENSEI-Lite



- Same structure as SENSEI
  - Lacks additional features implemented in SENSEI
    - Truncation error estimation
    - Multi-block
    - Manufactured solutions
    - Reconstruction schemes
- Written in C++ with MATLAB interface
  - C++ returns RHS residual

$$\frac{\partial y}{\partial t} = f(y, t)$$

  - Allows for testing of multiple time integrators
    - Implicit and explicit
    - Matrix-free methods

# SENSEI-Lite (Cont'd)



- Dr. Sandu's group has been using SENSEI-Lite to test matrix-free time stepping methods
  - Circular cylinder case
  - NACA 0012 airfoil case

# Application to SENSEI



- SENSEI currently stores and solves the primitive variables
- Conversion of primitive variables to conserved variables would allow solve in the form:

$$\frac{\partial y}{\partial t} = f(y, t)$$

- Matrix-free time stepping algorithms
  - Reduce storage
  - Can improve convergence rate
- Dr. Sandu's group has access to SENSEI and will be implementing time stepping algorithms

# Overview



- Personnel updates
- SENSEI
- Collaborations
  - Dr. Feng: Hybrid Parallelism of SENSEI
  - Dr. Sandu: Matrix-Free Time Stepping
    - SENSEI-lite
    - Application to SENSEI
  - Dr. de Sturler: Linear Solvers
    - Modified CUDA-ITSOL library
    - Implicit Multi-block

# Linear Solvers



- We have working in SENSEI a generic, object-oriented interface to linear solvers (similar to Saad's SPARSKIT)
- Also have interface to Saad's CUDA-ITSOL (in C)
- This is the same interface necessary for the work by Dr. de Sturler's group
- Their group is working to integrate their modified CUDA-ITSOL library with SENSEI (they have access to SENSEI locally)

# Full Implicit Multi-block



- Currently implemented in SENSEI using a minimal overlap Schwarz decomposition
  - A partitioning of domain cells with time-lagged inter-block Dirichlet BCs
- Seek to increase convergence rate
  - Want to maintain multi-block implementation
    1. Investigated increasing the overlap between domains (helps convergence, but not enough)
    2. We have decided to directly add off-block contributions to the full implicit solve