A Look Ahead: Ongoing Developments in SU2

1st Annual SU2 Conference
Virtual
June 12, 2020

SU2
foundation

1

Session Objectives

- Review development priorities and progress from previous years

- Preview of areas of SU2 development for coming year

- Prioritization of efforts and soliciting additional ideas:

  - Online polls

  - Write-in suggestions via GTW question window

- Enlist your help as POCs, developers, and testers!

What topics will we discuss at the 2nd Annual SU2 Conference?

A Blast from the Past...2017 Priorities

2nd SU2 Developers Meeting, 2017

- Discussion of dev priorities

- Broad participation, many topics brought up

- Pledge to look at list at <u>every</u> subsequent meeting

- How have we done?

1. Incompressible solver (SU2_FV and SU2_DG-FEM)
2. Performance (algorithmic and implementation) improvements
3. Robustness improvements (RANS solver, adjoints and optimization)
4. Adaptive mesh refinement
5. Wall functions / Wall models
6. DDES improvements
7. Multi-physics: aeroacoustics and FSI
8. Better probes / information recording
9. Graphical User Interface
10. UQ / scripting for parameter studies, surrogate models
11. Geometric constraints in design / design parameterizations
12. Better initialization for RANS solutions
13. Solution interpolation between grids
14. Non-equilibrium gas dynamics / hypersonics
15. Reacting flows
16. Reduced order modeling framework
17. Flow transition modeling

A Blast from the Past...2017 Priorities

2nd SU2 Developers Meeting, 2017

- Discussion of dev priorities
- Broad participation, many topics brought up
- Pledge to look at list at <u>every</u> subsequent meeting
- How have we done?

1. Incompressible solver (SU2_FV and SU2_DG-FEM) ✔
2. Performance (algorithmic and implementation) improvements ✔
3. Robustness improvements (RANS solver, adjoints and optimization) ✔
4. Adaptive mesh refinement ✔
5. Wall functions / Wall models ✔
6. DDES improvements ✔
7. Multi-physics: aeroacoustics and FSI ✔
8. Better probes / information recording ✔
9. Graphical User Interface ✖
10. UQ / scripting for parameter studies, surrogate models ✖
11. Geometric constraints in design / design parameterizations ✖
12. Better initialization for RANS solutions ✔
13. Solution interpolation between grids ✔
14. Non-equilibrium gas dynamics / hypersonics ✔
15. Reacting flows ✔
16. Reduced order modeling framework ✔
17. Flow transition modeling ✔

✔ Significant work completed
✖ More work remains to be done

2018 Priorities and Progress (Unordered)

✔ ▪ Config clean up / Consistent options between C++ & Python
✘ ▪ Shape Parameterizations / FFD Improvements
✘ ▪ Multigrid Improvements
✘ ▪ Transition Models
✔ ▪ Multipoint Optimization
✘ ▪ Interface w/ OpenMDAO
✔ ▪ Wall Functions
✔ ▪ Large CGNS Meshes
✘ ▪ Better references to methods/algorithms
✘ ▪ Multi-phase (VoF)
✔ ▪ Topology Optimization
✘ ▪ Lattice Boltzmann
✔ ▪ Multiphysics (enabling & integration)
✘ ▪ Snappy-Hex Mesh Compatibility / Auto-meshing
✘ ▪ MHD Capabilities
✔ ▪ Source File Organization & White space / Indentation Clean up
✔ ▪ Probes & Interpolation
✔ ▪ Linear solver performance
✔ ▪ Robustness & Convergence (Primal & Adjoint)
✔ ▪ Verification by manufactured solutions (MMS)
✘ ▪ Validation databases

✔ Significant work completed
✘ More work remains to be done
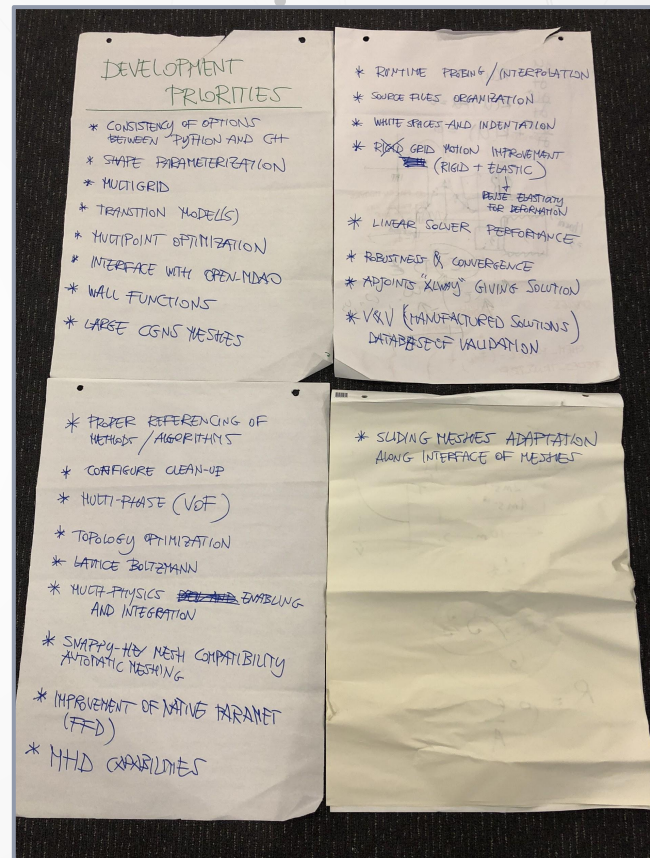
2019 Working Groups and Priorities

MDAO Working group

Outbrief and next steps

Ruben Sanchez

Flow Physics Working group outbrief

Eduardo Molina, Beckett Zhou

RPSVV Working group outbrief

Robustness, Performance, Scalability, Verification, & Validation

Edwin van der Weide, Thomas Economon

SU2

2019 RPSVV WG Priorities

Envisioned Required Code Developments

- ✗ ● Improvements to existing geometric multigrid (FAS) or new multigrid algorithm(s)
- ✓ ● New limiters/reconstruction/grid checks for robust convergence
- ✓ ● Linear solver/Jacobian improvements for robust convergence
- ✓ ● Fixes for existing CGNS reader or new native binary format for large grids (>100M cells)
- ✗ ● Algorithmic enhancements for massively parallel searches (ADT or otherwise)
- ✓ ● Devise a strategy for potential hybrid OpenMP-MPI implementation
- ✓ ● Optimize data structures for performance while balancing readability/flexibility
- ✗ ● Improve parallelization for multizone problems
- ✗ ● Expand MMS library to cover more governing equations, BCs, etc.

✓ Significant work completed
✗ More work remains to be done

9   © SU2 Foundation | Ruben Sanchez | 2019.05.10

SU2

Summary of Past Progress

Successes (major progress)!

- robustness & convergence
- interfaces to external packages
- performance improvements (hybrid, data structures, Jacobians, etc.)
- better code organization
- non-equilibrium / reacting flows
- CHT / FSI (+ adjoints)
- large mesh support
- wall functions / wall models
- MMS verification
- incompressible solver(s)
- scale-resolving simulations
- many more...

Not quite yet (repeat offenders)...

- robustness & convergence (never finished)
- interfaces to external packages (never finished)
- more docs / tutorials (never finished)
- validation databases (never finished)
- multigrid improvements (FAS & AMG)
- mesh generation integration / auto-meshing
- more physics / solvers:
  - MHD, LBM, transition, multi-phase, others?
- shape optimization framework /scripting / surrogates
- shape parameterizations
- improved multi-zone parallelization
- graphical user interface (GUI)

We can always improve, but let's not forget to celebrate the hard work of our volunteer community!

Known Ongoing Developments

- SU2 DG-FEM - Higher-order solver
- SU2-NEMO - Non-Equilibrium MOdels / Hypersonics
- Customizable Outputs / Expression Parsing / Modules
- Shape Optimization Framework Improvements
- Performance Improvements / Convergence acceleration / Hybridization
- Scalar Transport (reacting flows, transition, etc.)
- Scale-resolving simulations: DDES (SU2 FV) and WMLES (SU2 FV and SU2 DG-FEM)
- Mesh adaptation (collaboration with INRIA / pyAMG)
- Multi-physics frameworks
- Unsteady optimization
- Aeroacoustics
- Pressure-based Incompressible Solver
- Other ongoing development from the audience that we should know about?

Let us know if you want to be considered a POC or if you are interested in collaborating on a topic.

Help Us Set Priorities…Let's Do some Polls!

Numerical Methods / Robustness

- Linear / nonlinear solvers + preconditioners
- Algebraic multigrid (AMG)
- Higher-order discretizations
- Robustness to poor/adapted grids
- Advanced discretizations

Physical Modeling

- Scale-resolving methods (DES, LES, WMLES)
- Flow transition
- High-speed / non-eq / real gas
- Reacting flows / combustion modeling
- Wall functions / wall models

Performance

- Hybrid MPI / OpenMP
- GPU implementation
- Memory layout / reordering / vectorization
- Performance benchmarks
- Large mesh (> 100M cells) improvements

Multi-physics / Interfaces

- Fluid-structure improvements
- Conjugate Heat Transfer (CHT)
- Unstructured mesh oversetting
- NASTRAN / TACS
- Electromagnetics

Adjoints / Optimization

- Shape parameterization options
- Optimization methods (incl. gradient-free)
- AD improvements
- Geometric constraint framework
- Mesh deformation approaches

Code base / Support

- Better code integration
- Unit / regression tests
- V&V (incl. database) / MMS
- Docs / tutorials / events
- Python interface / automation

Tracking Our Progress

- GitHub Projects are lightweight and useful for organizing work in the various areas

- Example for the RPSVV WG (see right), which has been active.

- You can find other projects on GitHub, e.g., General Maintenance.

- Do you think we should keep this system? Expand? Try something else?
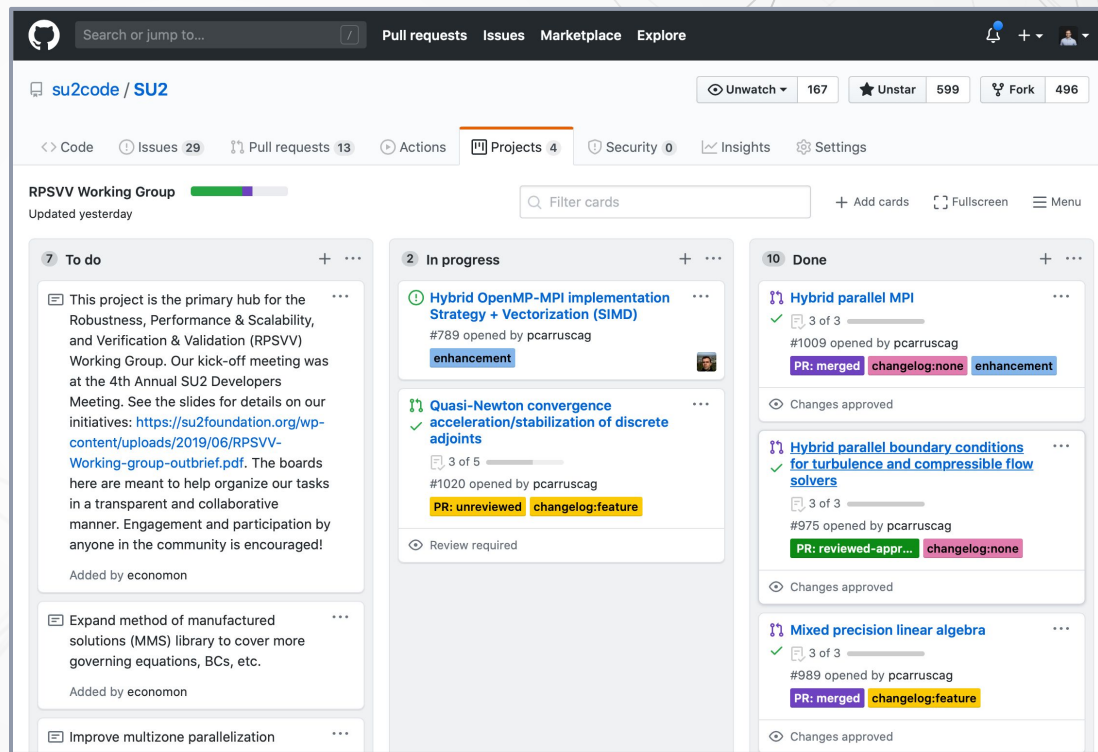
CFD Courses Using SU2

- This is a popular request from folks in the community. We are often asked:
  - Have courses already been taught using SU2?
  - Are slides / lecture materials available?
  - Does SU2_EDU still work? (yes, it can be updated)
  - Can we connect people that are interested in sharing materials for this purpose?

- The SU2 Foundation would be happy to help facilitate if we have interested volunteers.

Join us on our mission of education, research, and innovation for the benefit of all society!

Here's how you can get involved:

- Join our email list at **su2foundation.org**

- Get in on the action on GitHub: **https://github.com/su2code/SU2**

- Contact us about SU2 Foundation programs and sponsorship: **info@su2foundation.org**

Be Safe, Be Healthy, Be Well ...

and Code On!

Thank **you**!!!

SU2
foundation

2019-2020 Development Highlights

- Addition of hybrid OpenMP/MPI (#1009, #975, #861, …)
- Modernization to C++11 throughout codebase (ongoing)
- Compilation speedup of 10x (#853)
- New unit testing framework (#850)
- Automated release workflow (#813)
- Wider CI coverage on more platforms with GitHub actions (#806)
- Algorithmic improvements / performance optimizations for up to 10x speedup, more robustness, and scalability (#790, #753, #728, #652, …)
- Addition of fully user-customizable output (#728)
- New solution verification framework (#672)
- Too many more to include!

Help Us Set Priorities

- Overall prioritization of high level topics by the audience (what are they most interested in):
  - physics, numerics, meshing, time-to-solution, optimization/adjoints
- integration w external packages (what do folks want most?)
  - mesh gen/adapt, openmdao/optimizers, structural solvers, linear solver packages, etc
- physical models:
  - MHD, LBM, more combustion, transition, multi-phase, lagrangian particles, write-in?
- mesh related:
  - oversetting, improved mesh def techniques (RBF), more dynamic grid options (e.g., 6DOF, rigid+elastic, etc), tighter mesh adaptation integration
- optimization related (what do folks want most)
  - better/more optimizers, geometric constraints, better parameterizations, gradient-free methods, more customization of objectives and DVs, surrogates
- Docs / Tutorials/ Training (what is most helpful)
  - more written docs, more tutorials, V&V studies, paper references in code comments, short training videos on youtube
- Future events (preferred format):
  - workshops w/ space for discussion (of development for instance), in person or virtual meetings w/ primarily presentations, formal training courses (programming/development, CFD, etc.)