Towards aerospace design in the age of extreme-scale supercomputing

Qiqi Wang
Assistant Professor of Aero & Astro, MIT

with work by Patrick Blonigan and Chaitanya Talnikar
NASA has inspired the world with its Aerospace Designs.
NASA is uniquely positioned to lead revolutions in simulation-based design.
CFD methods developed during the ICASE period led to rapid decrease in the number of wings tested in tunnel.

From Jameson, 2012
CFD has been on a plateau for the last 15 years.

– Jameson 2012

Plan and roadmap were created to help focus investments in technology development to help achieve the CFD vision in 2030

– Slotnick, Khodadoust, Alonso, Darmofal, Gropp, Lurie, Mavriplis, funded by NASA with Kleb and Malik
“Hybrid RANS/LES methods show perhaps the most promise in being able to capture more of the relevant flow physics for complex geometries at an increasingly reasonable computational cost.”

“WRLES is potentially feasible in 2030 for lower Reynolds numbers.”

‘Design is “a huge carrot out there, in terms of what we’ll be able to do in the future,” Slotnick says. “In 20 years, engineers will not simply be doing CFD analysis,” he predicts. “They’ll be sitting and doing high-fidelity design using computational methods and incorporating CFD into the design, almost in real time.”’

‘The problem, Gropp says, is that today’s well-understood CFD algorithms and software will have to be rewritten or overhauled to work on massively parallel computers or quantum computers. ’
A typical high fidelity CFD requires millions of time steps.

Talnikar, Blonigan, Bodart, Wang, 2014
The best optimization algorithm requires \( \sim 100 \) simulations

A typical high fidelity CFD requires millions of time steps. The best optimization algorithm requires ~100 simulations.

With today’s methods, a high fidelity design optimization takes weeks, if not months.

Can we shorten a high fidelity design optimization to minutes?
Future supercomputer may be built with "smartphone"s
A typical high fidelity CFD requires **millions of time steps**. The best optimization algorithm requires \(~100\) simulations.

With today’s methods, a high fidelity design optimization takes **weeks, if not months**.

Can we shorten a high fidelity design optimization to **minutes**, using billions of “smartphone” nodes?
Can we shorten a high fidelity design optimization to **minutes**, using billions of “smartphone” nodes?

- Can we shorten a high fidelity **simulation** to minutes?
- Can we shorten a high fidelity **optimization** to minutes?
- Can we shorten a high fidelity optimization with **many design parameters** to minutes?
Accelerating high fidelity simulation with extreme scale parallelism

FUN3D is one of the most scalable CFD code in the world

FUN3D reference, extracted 2015
The latency barrier

Each sub-timestep requires communication between nearest neighbor cells/elements.
Breaking the latency barrier

Classical “straight” domain decomposition
Breaking the latency barrier

"Swept" domain decomposition
Breaking the latency barrier for the Kuramoto-Sivashinsky equation

Latency barrier broken by a factor of 10

Graph showing the comparison between straight and swept schemes in microseconds per sub-timestep against grids per core.
“Swept” decomposition in 2D (1)
“Swept” decomposition in 2D (2)
“Swept” decomposition in 2D (3)
“Swept” decomposition in 2D (4)
Accelerating high fidelity simulation with extreme scale parallelism

**First attempt** broke the latency barrier by a factor of 10

Can be improved a lot by a **talented programmer**

Can be extended to **2D and 3D**, and **unstructured** mesh

Further acceleration through **hierarchical** decomposition

**Huge potential**
Accelerating high fidelity optimization with extreme scale parallelism

FUN3D is one of the few adjoint-enabled full-blown CFD codes in the world

How can extreme scale parallelism benefit optimization?
Classic optimization

- One simulation at a time
- Each costs the same
- A good scheme minimizes the number of simulations

High fidelity CFD optimization

- Concurrent simulations
- Simulations run indefinitely
- A good scheme minimizes the time-to-design

The design of experiments.

RA Fisher - 1935 - psycnet.apa.org

Abstract 1. Different types of experimentation are considered with reference to their logical structure, to show that valid conclusions may be drawn from them without using the disputed theory of inductive inferences, i.e., of arguing from observation to explanatory theory. This is ...
We found the **Bayesian framework** suitable for high fidelity CFD optimization

Design **rounded trailing edge** to minimize Pressure Loss and Heat Transfer.

Requires high fidelity simulation

Léonard, Gicquel, Gourdain & Duchaine 2014
VKI HP-IGV, Re=10^6, Ma=0.9
4% turb inflow, Span ≈ 1/7 c
CharLES solver on 15M Cells
Trailing Edge Parameterization

- **Fixed points**
- **Control points**
Trailing Edge Parameterization
Optimization and Results

Baseline Design

Optimized Design

- 4 design variables, Single objective
- 4 concurrent simulations, each running for at least 12 hours, sometimes much longer.
- Optimization completed in one week.
Optimization and Results

Baseline Design

Optimized Design

- **21% reduction** in pressure loss
- **17% reduction** in heat transfer
Accelerating high fidelity optimization with extreme scale parallelism

Talnikar and Blonigan did it in their spare time.

Can benefit a lot from better algorithm deciding when to pause and resume simulations.

Tested only on 4 design variables. Extension to more design variables likely need the adjoint.

The algorithm is still primitive; further improvements can multiply its benefit to application engineers.
Accelerating high fidelity optimization with many **design parameters** with extreme scale parallelism

FUN3D is one of the few adjoint-enabled full-blown CFD codes in the world

Adjoint alleviates the “curse of dimensionality”, thus is often essential in optimization with many (>10) design parameters.
Many high fidelity simulations in important aerospace applications are unsteady and chaotic.

High fidelity optimization with many design parameters requires adjoining chaos.

Launch abort system (NASA)

Rotorcraft (NASA)

Maneuvering vehicle (Airforce)
Divergent adjoint of 2D Navier-Stokes

Re=3900

Re=10000

Credit: T. Barth, ASA Ames.
Divergent adjoint of 2D Navier-Stokes

Re=10000

FUN3D, Blonigan, Nielson & Diskin 2012

Indicates magnitude of adjoint solution
For a “uniformly hyperbolic” dynamical systems

\[ \frac{du}{dt} = f(u, s) \]

and its ergodic time average

\[ \langle J \rangle := \lim_{T \to \infty} \left( \frac{1}{T} \int_0^T J(u) \, dt \right) \]

exists, but

\[ \frac{d\langle J \rangle}{ds} \neq \lim_{T \to \infty} \frac{d}{ds} \left( \frac{1}{T} \int_0^T J(u) \, dt \right) \]
double lorenz(r)
{
    double b = 8./3., s = 10.;
    double T = 100., dt = 0.01;
    double x = 1., y = 1., z = 1., zMean = 0.;
    for (double t = 0; t < T; t += dt) {
        double dx = s * (y - x);
        double dy = x * (r - z) - y;
        double dz = x * y - b * z;
        x += dt * dx;
        y += dt * dy;
        z += dt * dz;
        zMean += dt * z / T;
    }
    return zMean;
}
\[ \frac{d(z\text{Mean})}{dr} = -1.83 \times 10^{23} \]
Trajectories with similar parameters

- \( u(t) \) (\( \rho = 30 \)) and \( u_r(t) \) (\( \rho = 28 \)) are very different after some time!
Trajectories with similar parameters

- $u(\tau) \ (\rho=30)$ shadows $u_r(t) \ (\rho=28)$!
“Least Squares Shadowing” Works

![Graph showing the relationship between zMean and r, with labels indicating before and after changes.](image)

\[ \frac{d(z\text{Mean})}{dr} = -1.83 \times 10^{23} \]

\[ \frac{d(z\text{Mean})}{dr} = 1.01 \]
“Least Squares Shadowing” Works

Computed derivative from Initial value problem before
"Least Squares Shadowing" Works

Computed derivative from Initial value problem

before

Computed derivative from Least squares problem

after
“Least Squares Shadowing” Works

An objective function in a chaotic simulation

Before

After
Accelerating high fidelity optimization with many design parameters with extreme scale parallelism

Much of this work is done by Patrick Blonigan

Extending to full CFD applications with Boris Diskin and Eric Nielson

Combined with faster simulations and parallel optimization, we aim to accelerate high fidelity optimization with many design variables to minutes.
Towards aerospace design in the age of extreme-scale supercomputing

We can break the “latency barrier”. Performing high fidelity CFD in minutes, even seconds, is not a dream. Let’s do it.

Optimization can concurrently run many simulations. Let’s improve the method so that it finds optimal designs in just a few batches of high fidelity simulation.

Adjoint works on chaotic simulations. Let’s devise an efficiently implementation, so that optimization with many design variables will be as fast as that with a few design variables.

Let’s do these, so that high fidelity optimization can be the next slide rule for every aerospace engineer.