SAND2022-8746C

Sandia National Laboratories

Exceptional service in the national interest

Creating Data-Driven Turbulence Models Using PIV

Steven Beresh, Nathan Miller, Eric Parish, and Jaideep Ray

20th Lisbon Symposium

July 11-14, 2022

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Let's talk about CFD.

RANS remains the backbone of engineering predictions because it is computationally affordable.

Therefore, we keep using it even though we know it can be wildly inaccurate for many aerospace applications.

LES and DNS are much more accurate, but too expensive to run many cases.

The problem: RANS sucks.

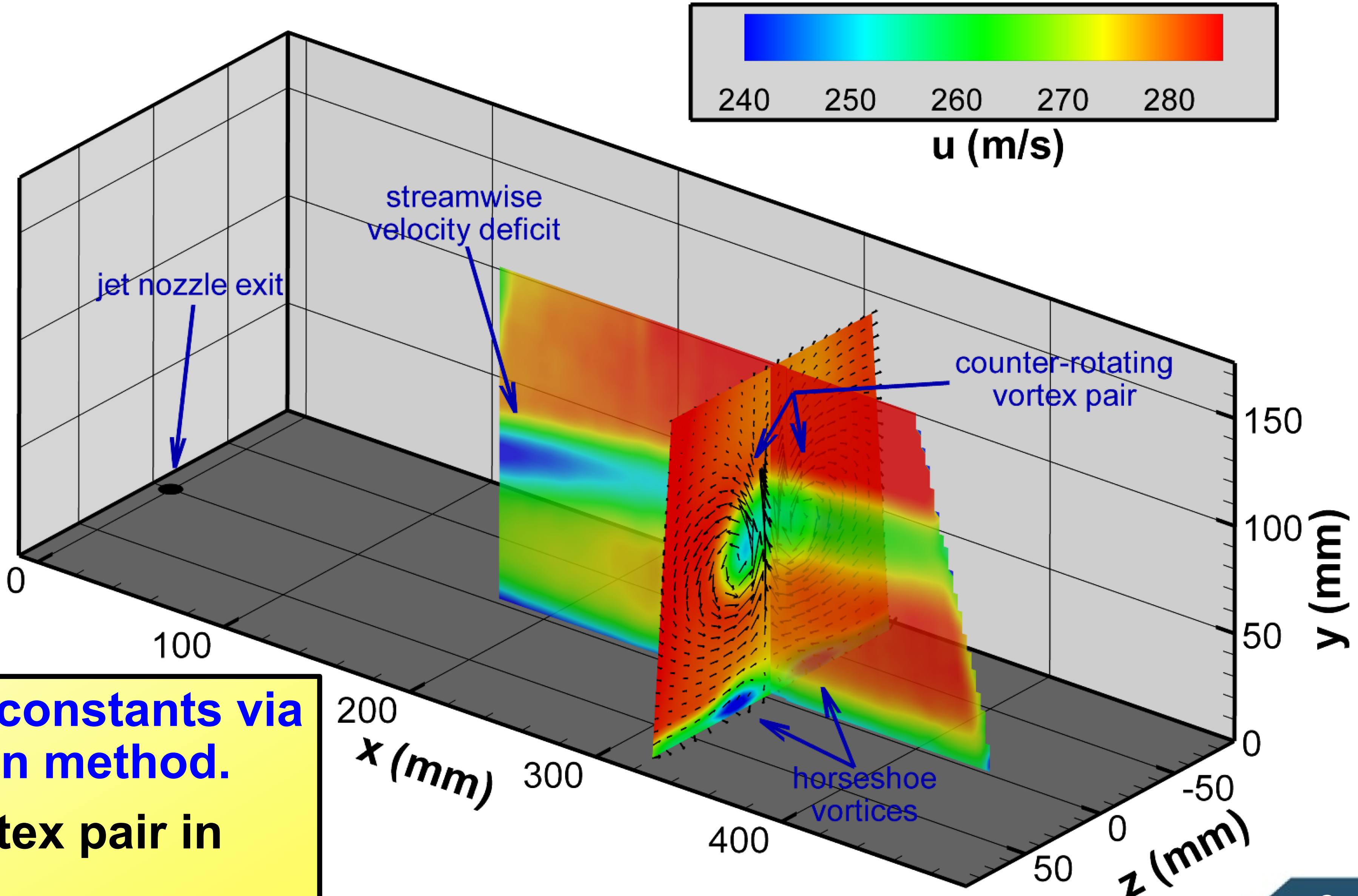
The question: How do we make RANS suck less?

The solution: Incorporate PIV data.

Application: Supersonic jet in transonic crossflow

PIV data from Sandia experiments circa 2005.

(not TR-PIV)



Redefine RANS model constants via a data-driven calibration method.

Goal is to track the vortex pair in the mean velocity field.

Look inside a typical turbulence model.

Turbulent viscosity (or eddy viscosity):

$$\nu_t = \frac{-\overline{u'v'}}{\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x}}$$

turbulent shear stress

mean strain rate, S_{xy}

In a *k*-ε model:

$$\nu_t = \frac{C_\mu k^2}{\varepsilon}$$

t.k.e.

dissipation rate

model constant

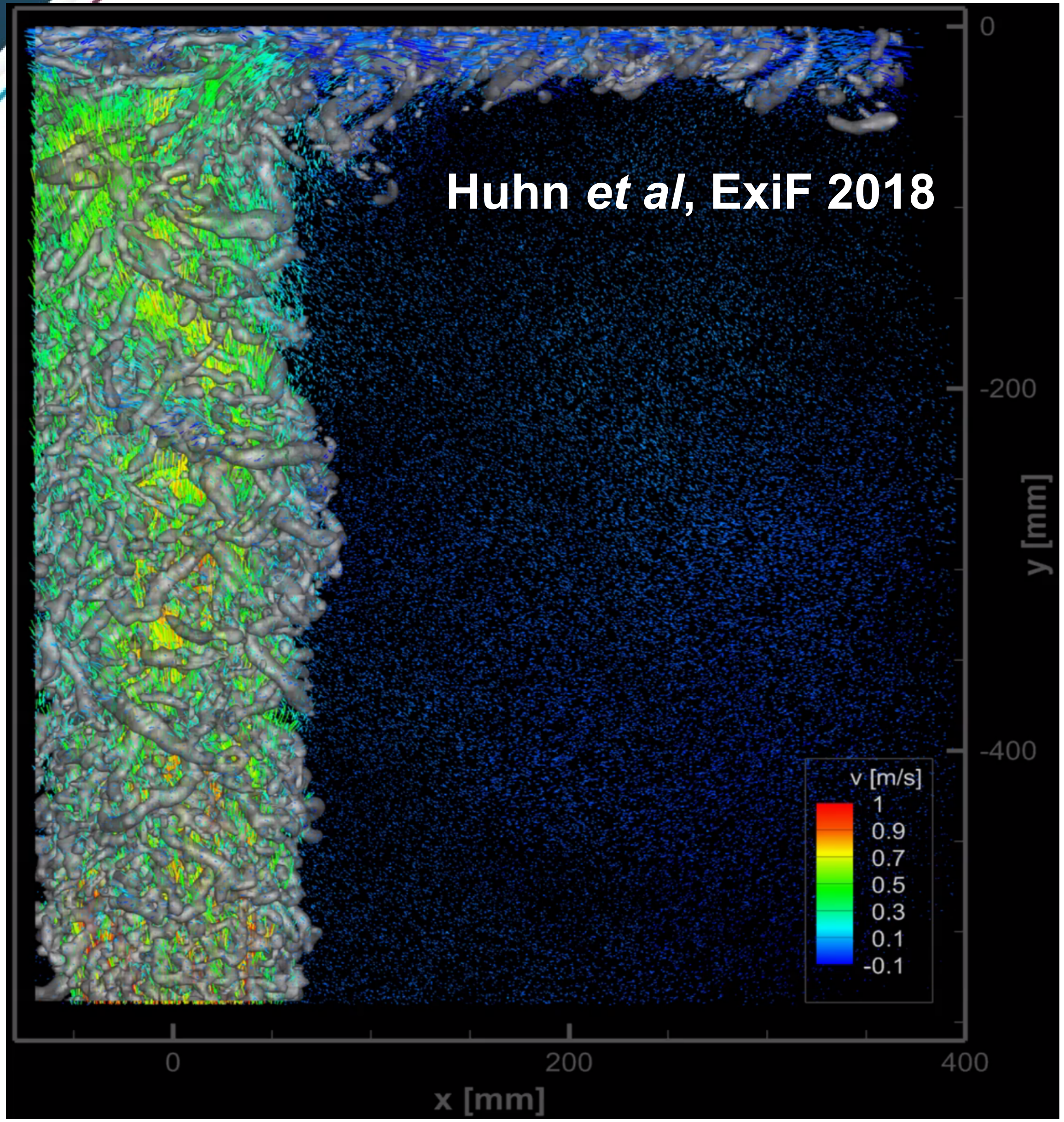
<u>We can calculate all of these terms directly from PIV!</u>

A simple computation based on the above equations will not suffice.

- Random uncertainty
- Bias error
- Spatial resolution
- Convergence….

The full story: see Miller and Beresh, *AIAA Journal*, 2021.

This is a type of PIV data assimilation.



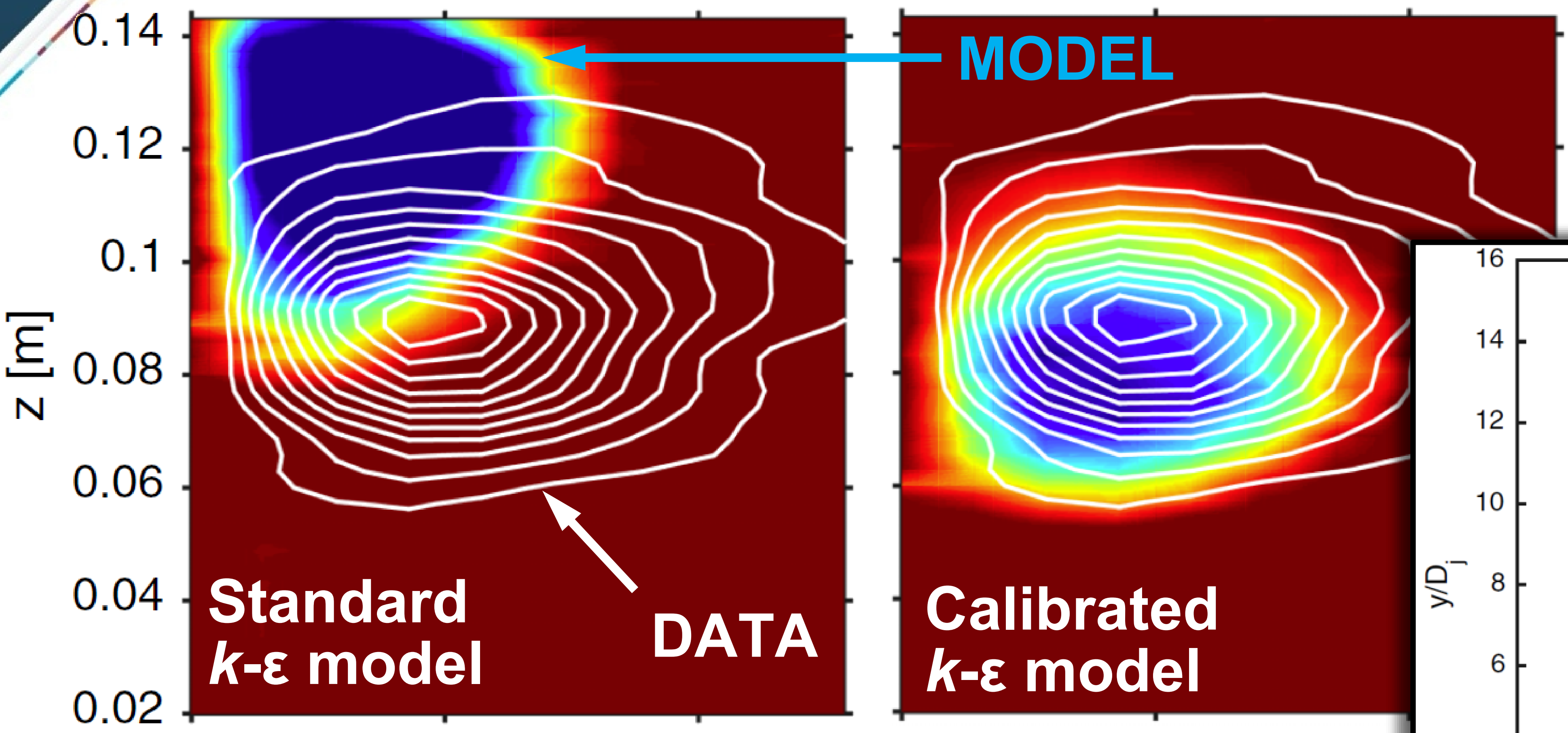
Usually associated with volumetric methods to incorporate governing equations and numerical methods.

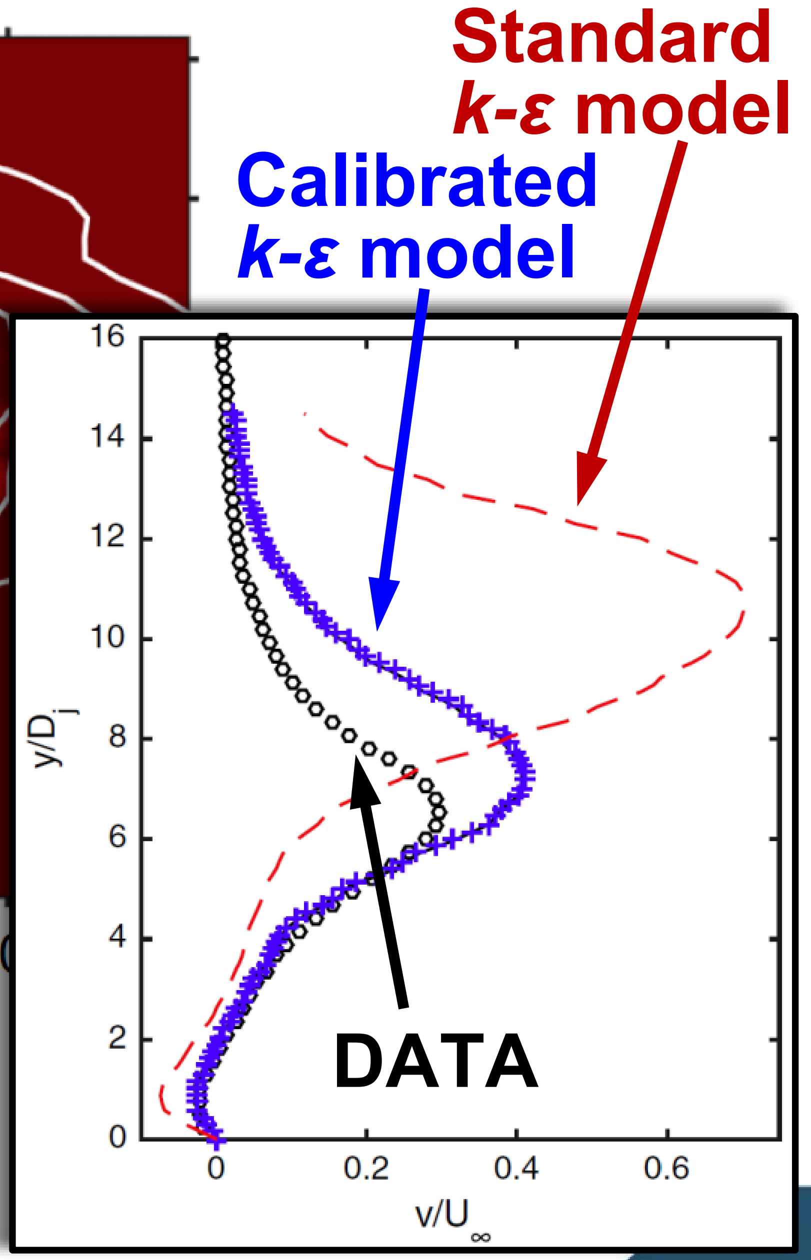
Yields higher fidelity velocity fields and pressure fields.

But they apply only to the test case measured!

Our approach yields a *generalized data-driven RANS model* with improved physical fidelity that can predict any relevant flow case.

Recalibrate RANS based on PIV data.



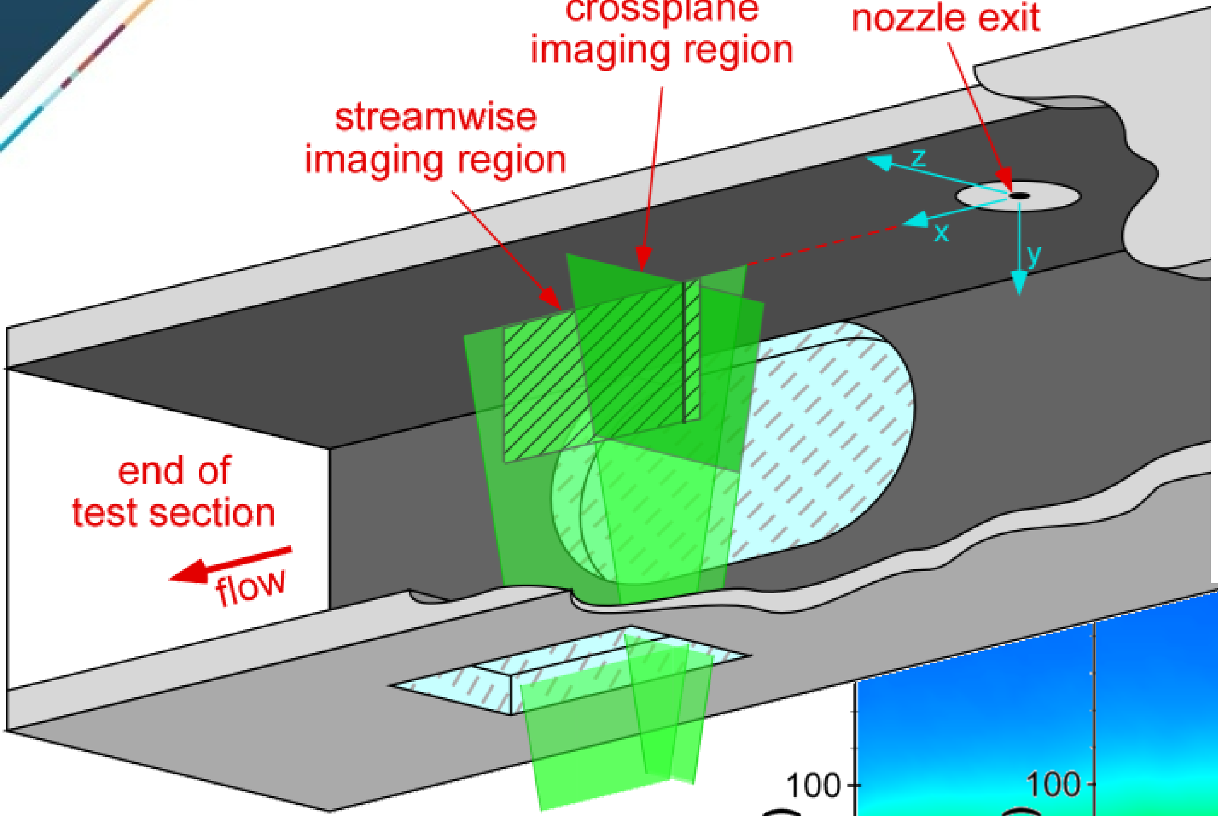


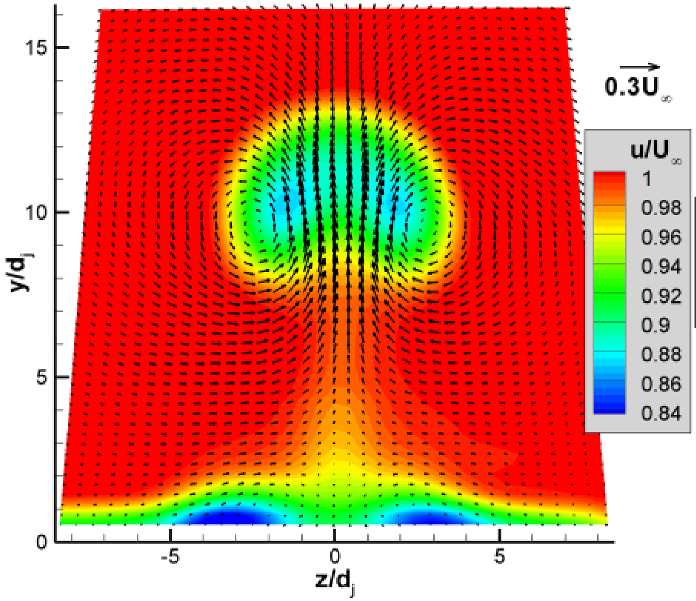
Redefine RANS model constant C_μ via a Bayesian calibration method.

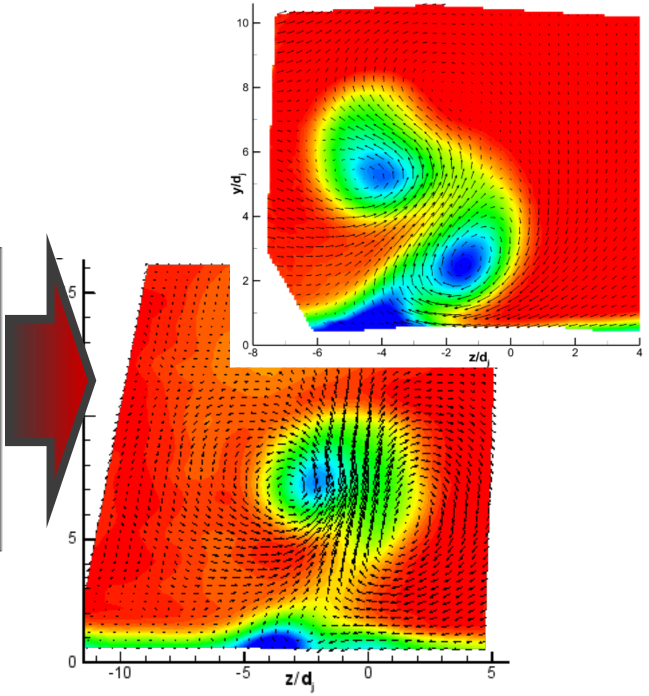
Yields calibrated C_μ = 0.1025 instead of nominal C_μ = 0.09.

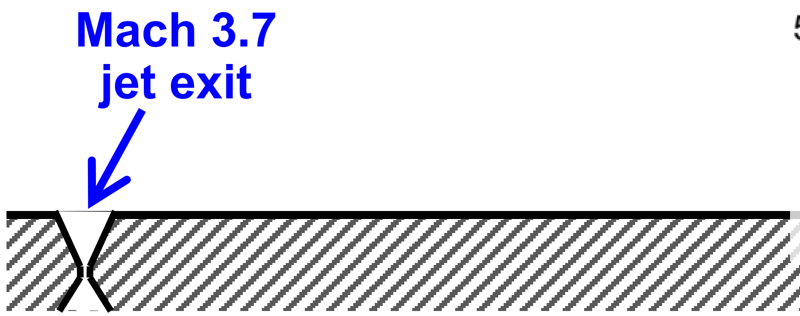
figures from Ray *et al*, *AIAA Journal*, 2016

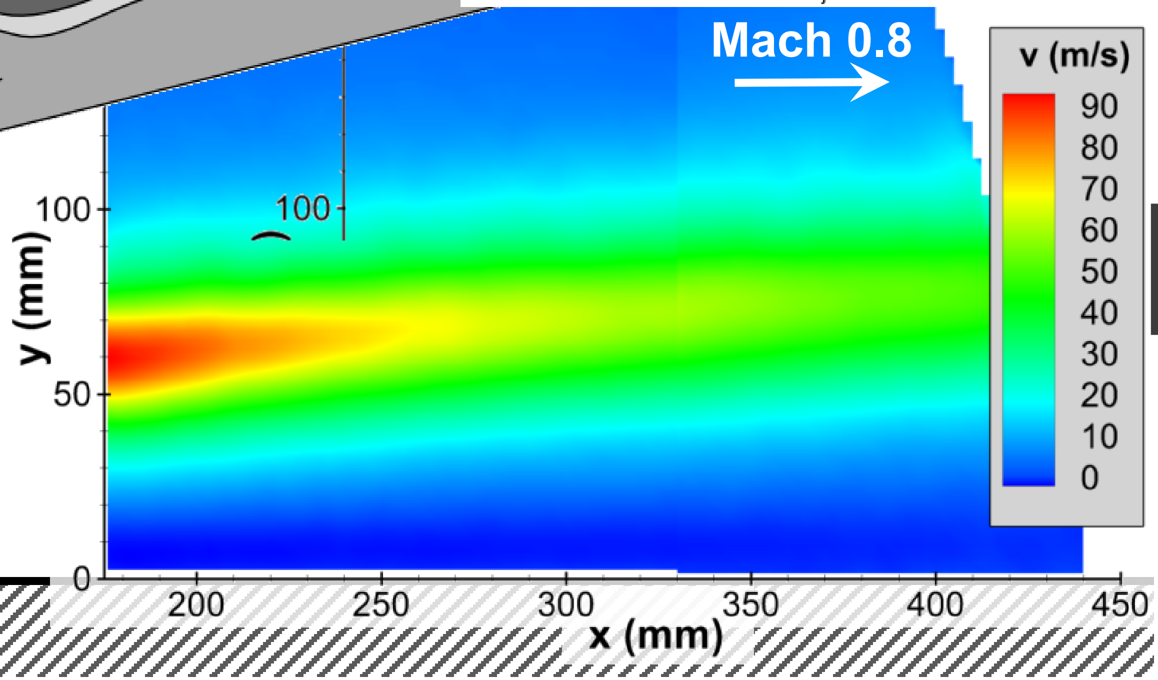
The jet interaction data set.

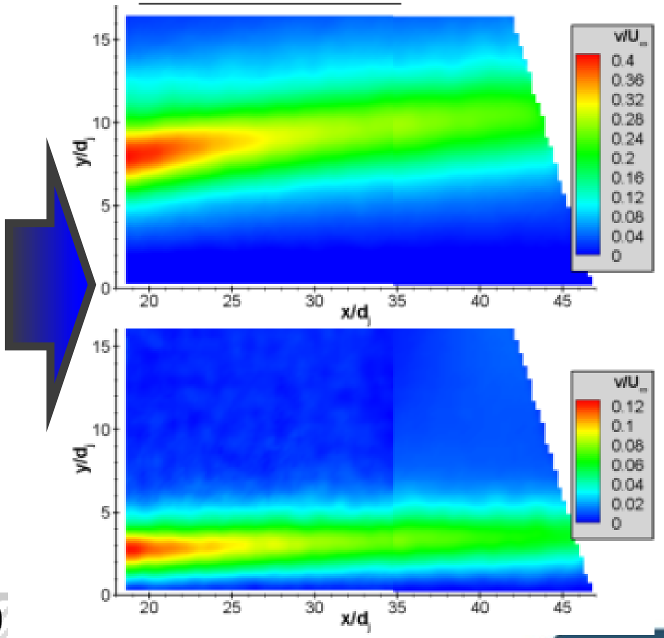












The jet interaction data set.

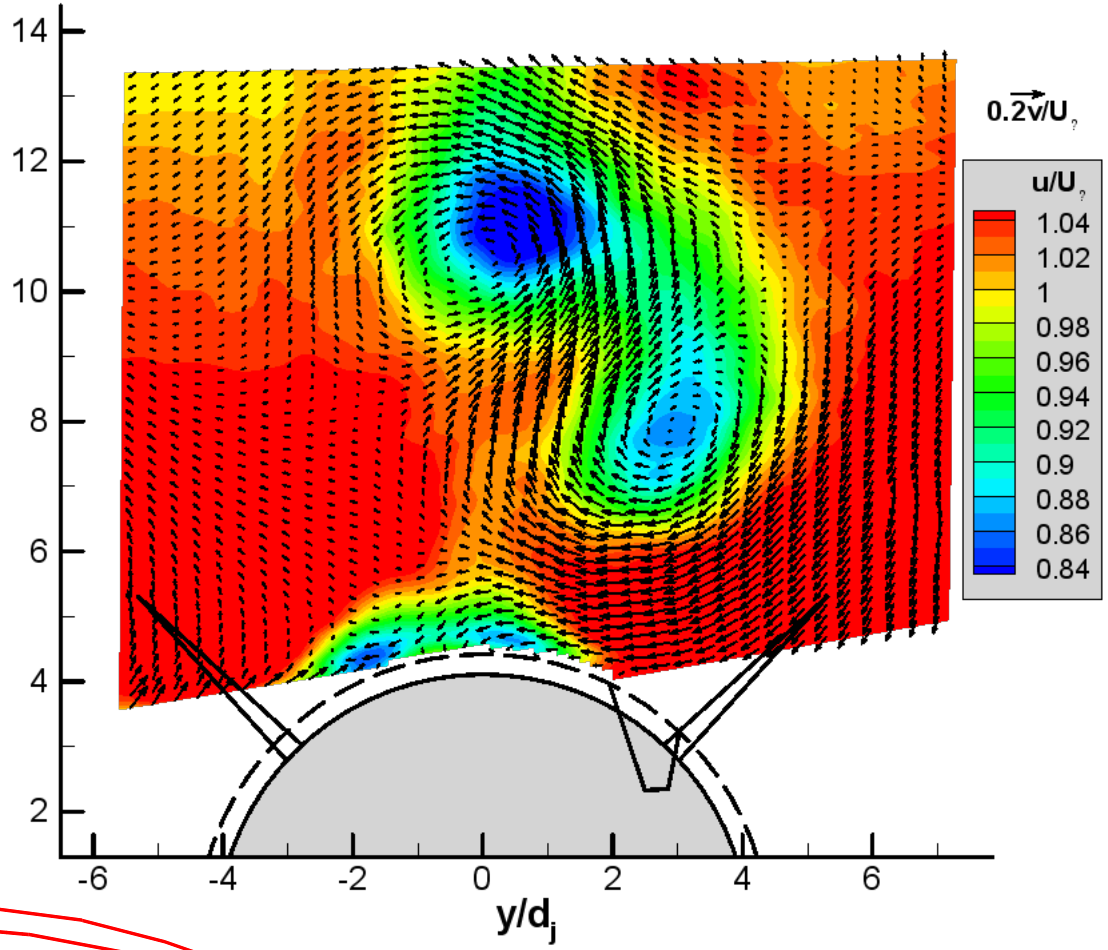
Calibrated based on only four PIV cases:

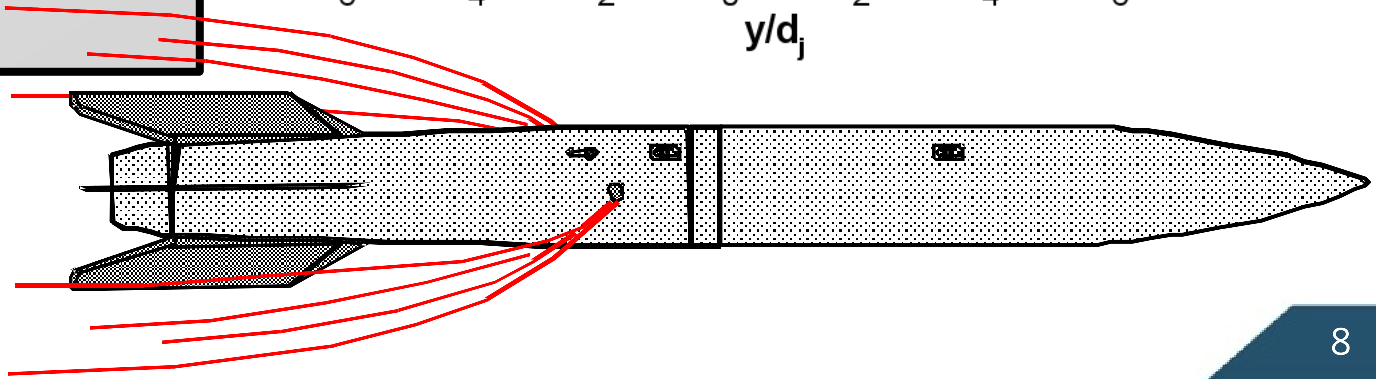
Transverse jet of varying strength.

The full data set contains 48 test cases, varying:

- **Jet strength**
- **Nozzle inclination**
- **Measurement station**

Also, PIV test case on a full-scale vehicle with spin rockets.

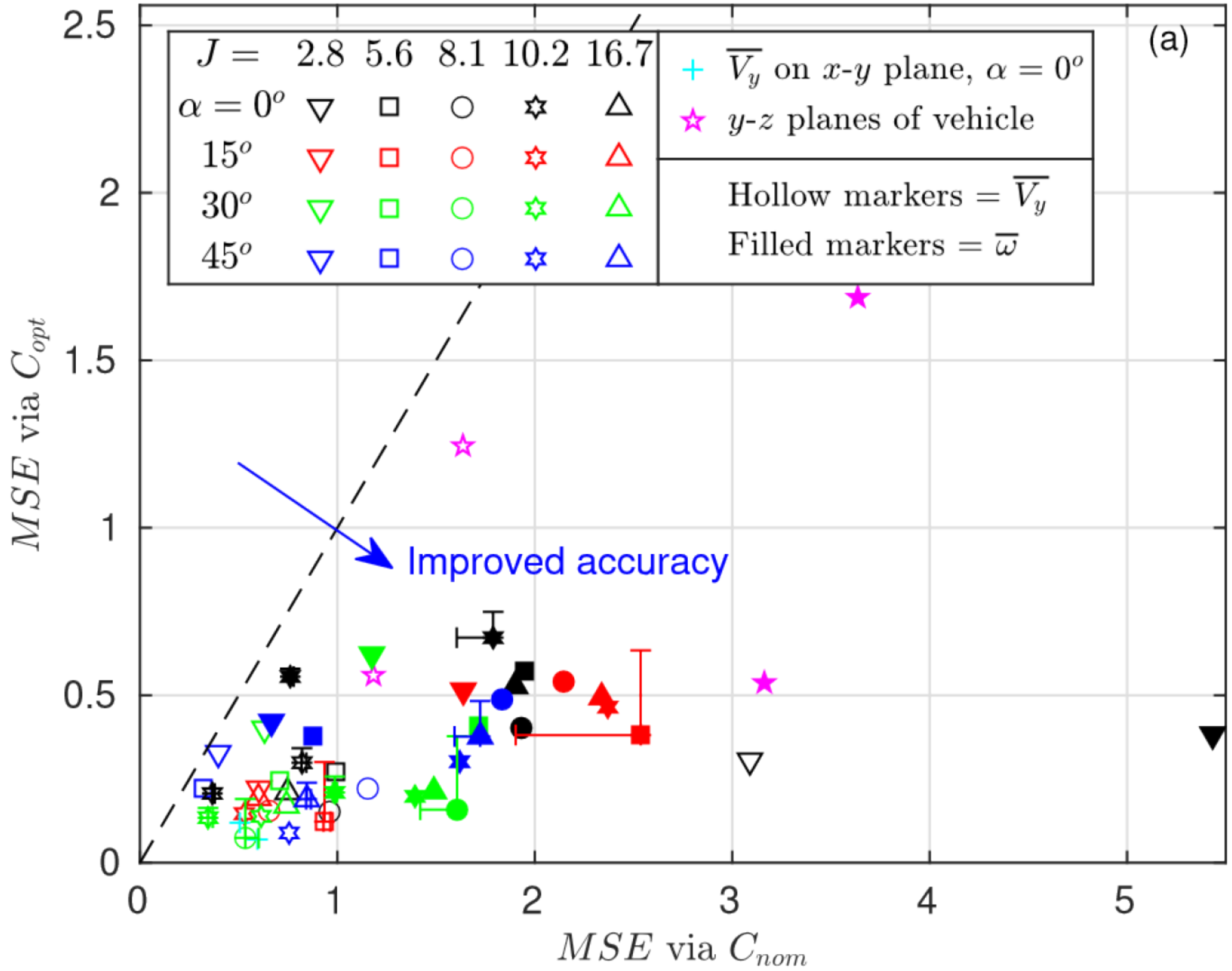




Validating the calibrated C_μ model.

We examined six quality metrics.

Here's one:



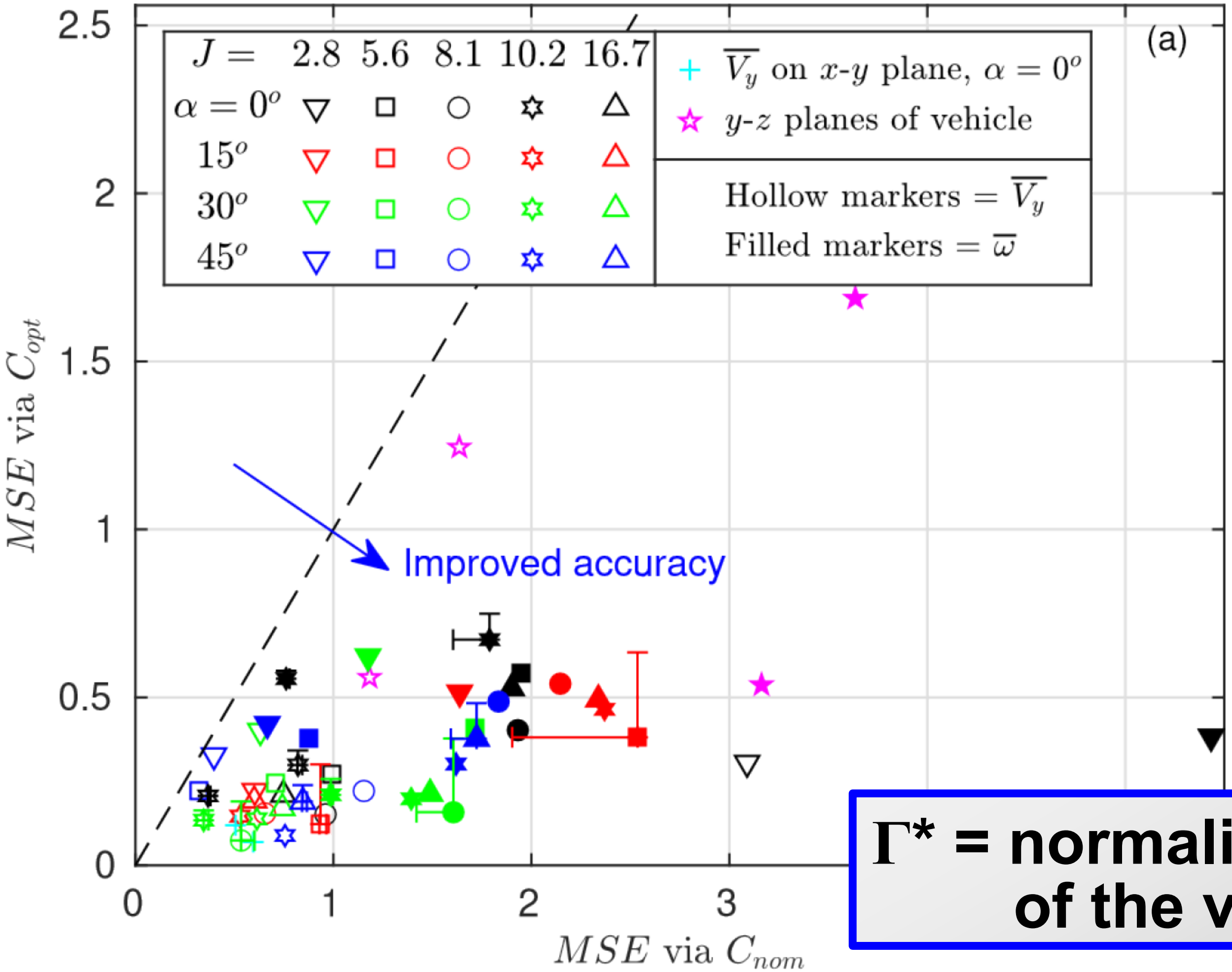
MSE = mean square error

Overall picture of the error of the CFD w.r.t. the PIV.

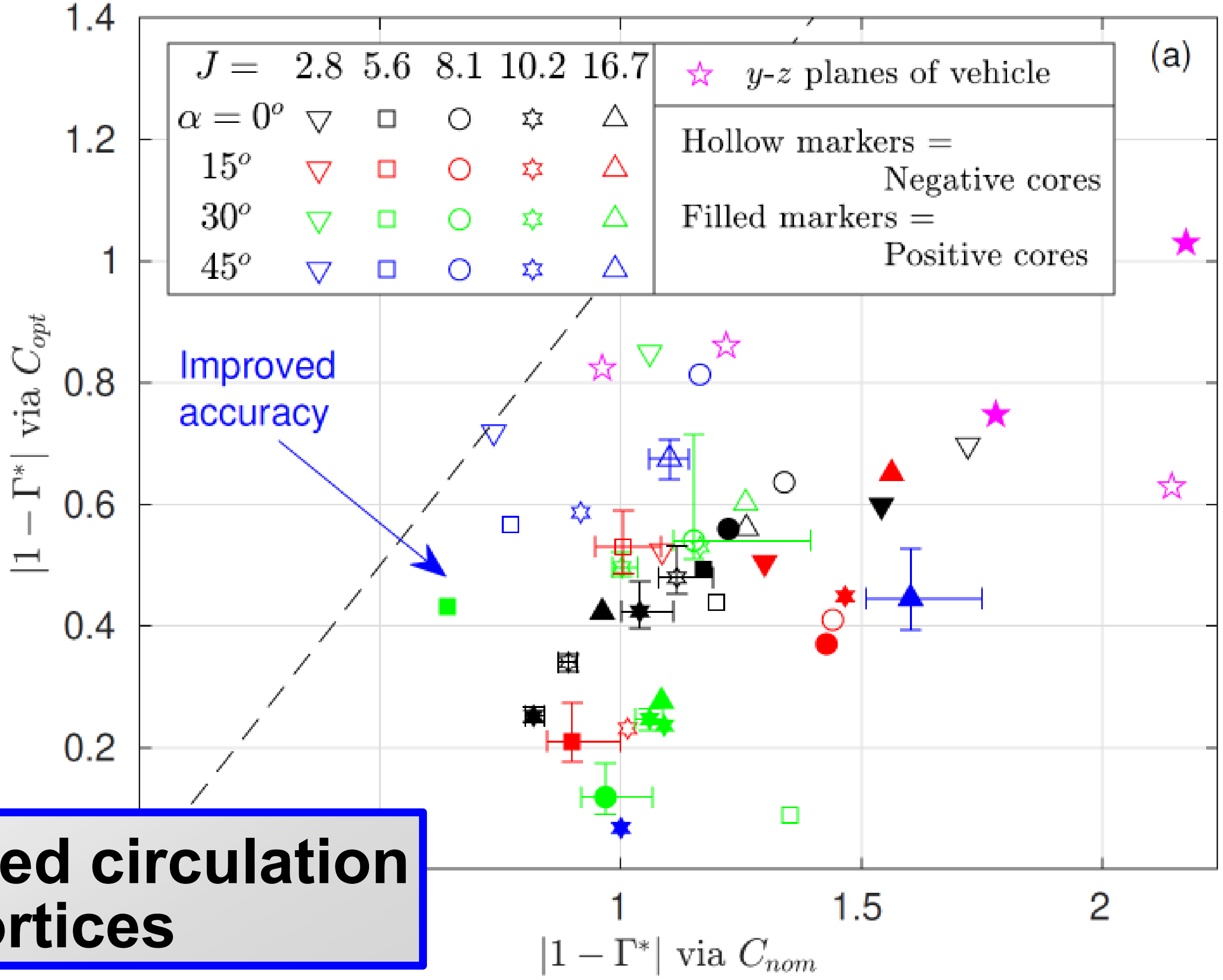
Validating the calibrated C_μ model.

We examined six quality metrics.

Here's one:



Here's another:

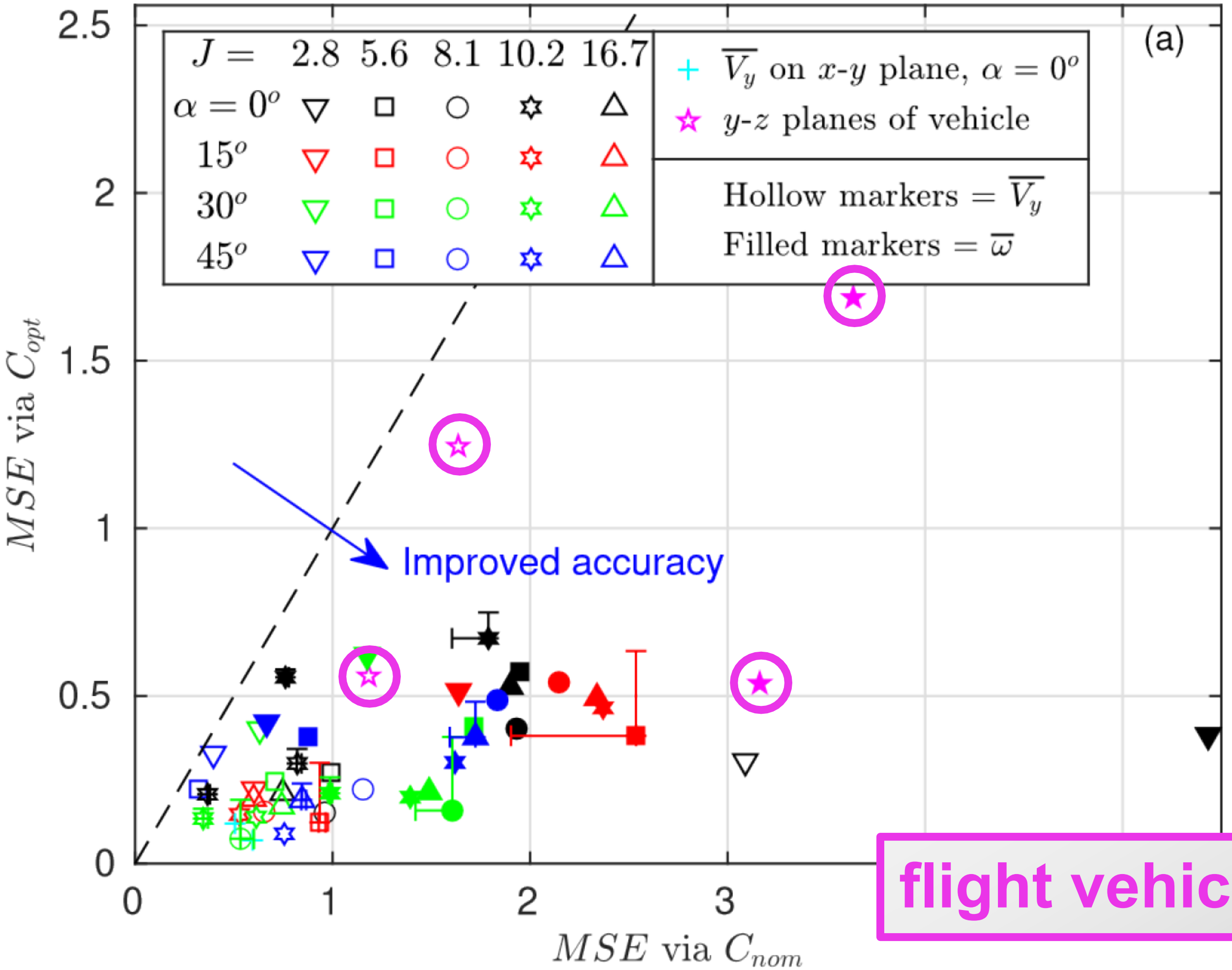


Γ^* = normalized circulation of the vortices

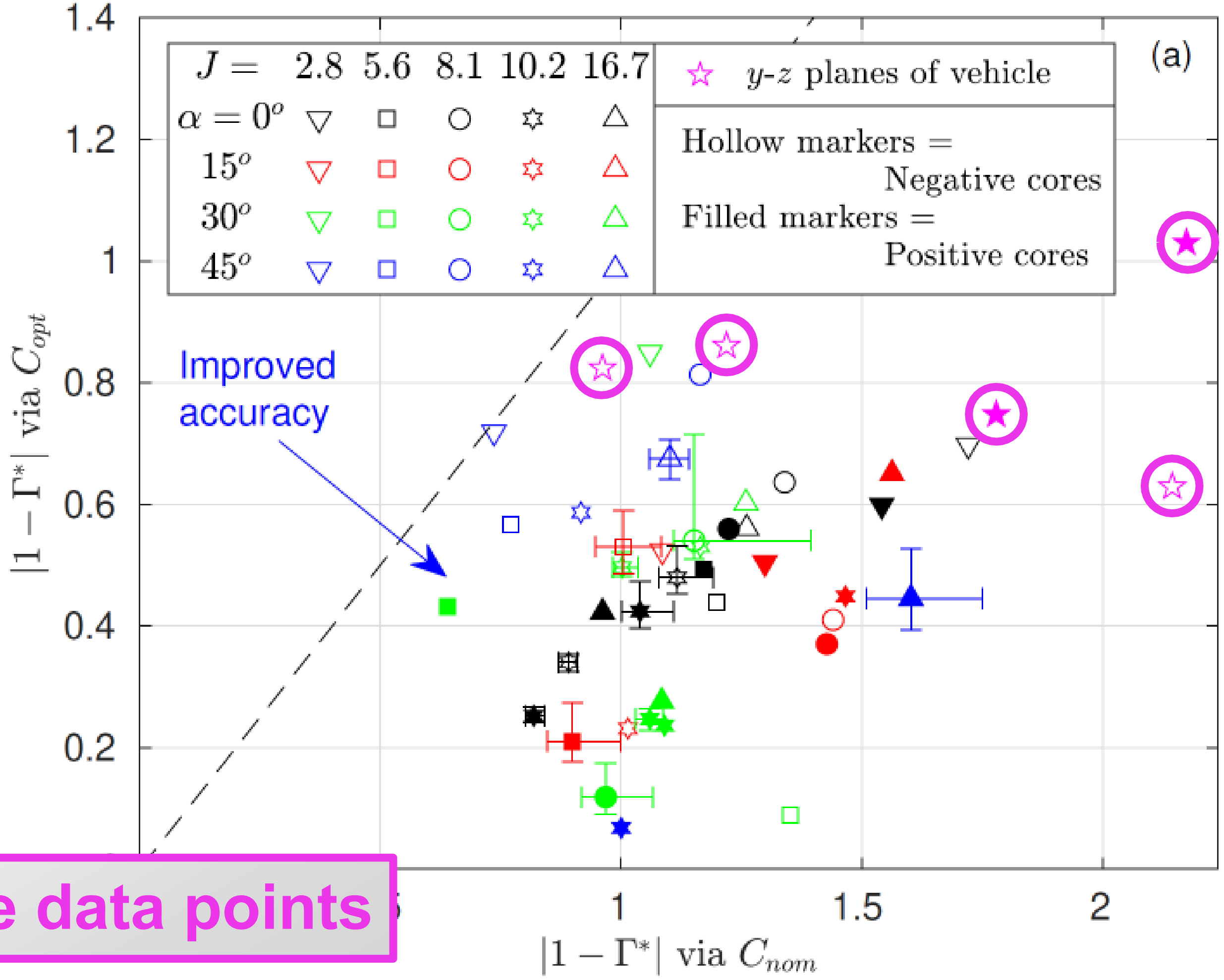
Validating the calibrated C_μ model.

We examined six quality metrics.

Here's one:



Here's another:

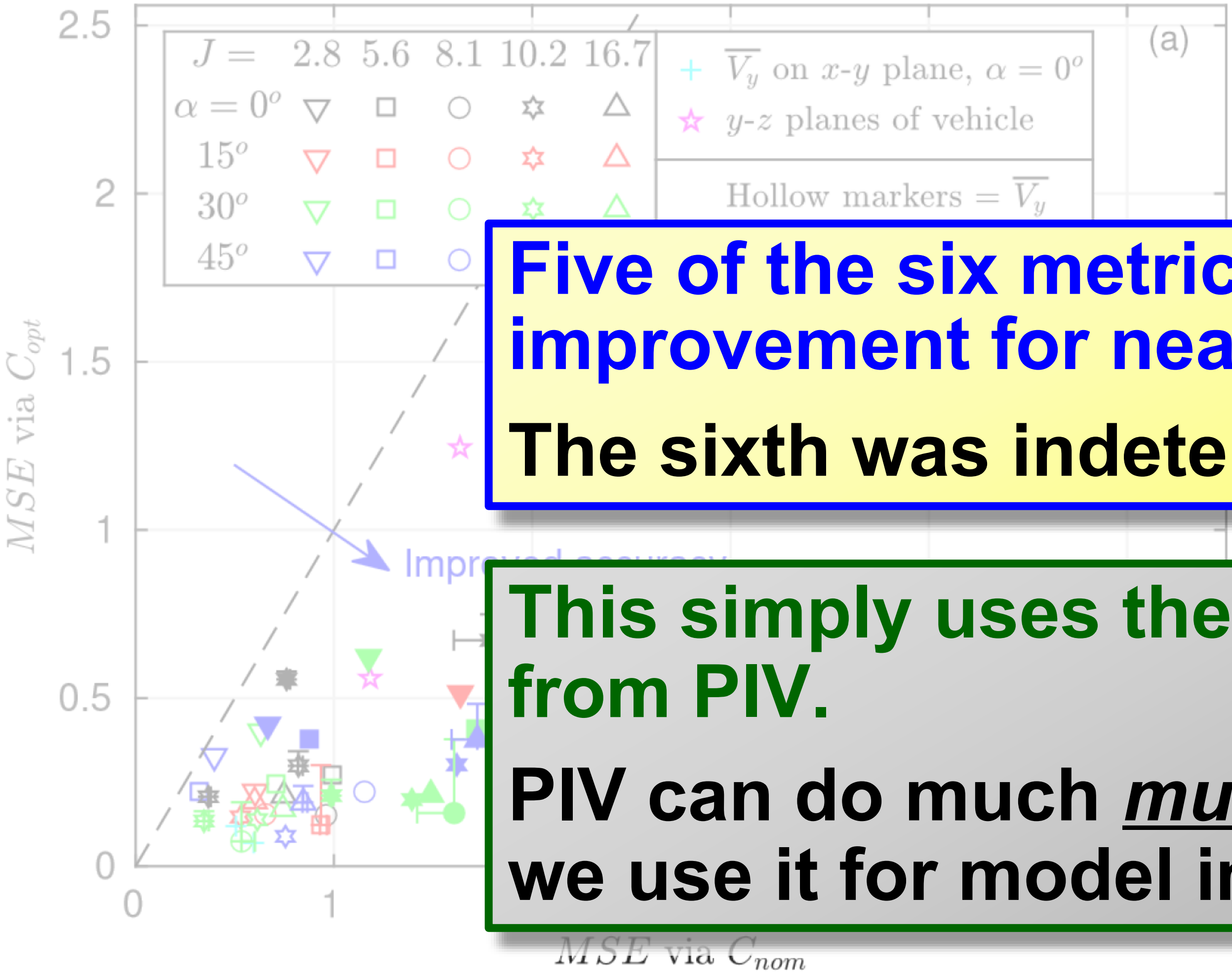


flight vehicle data points

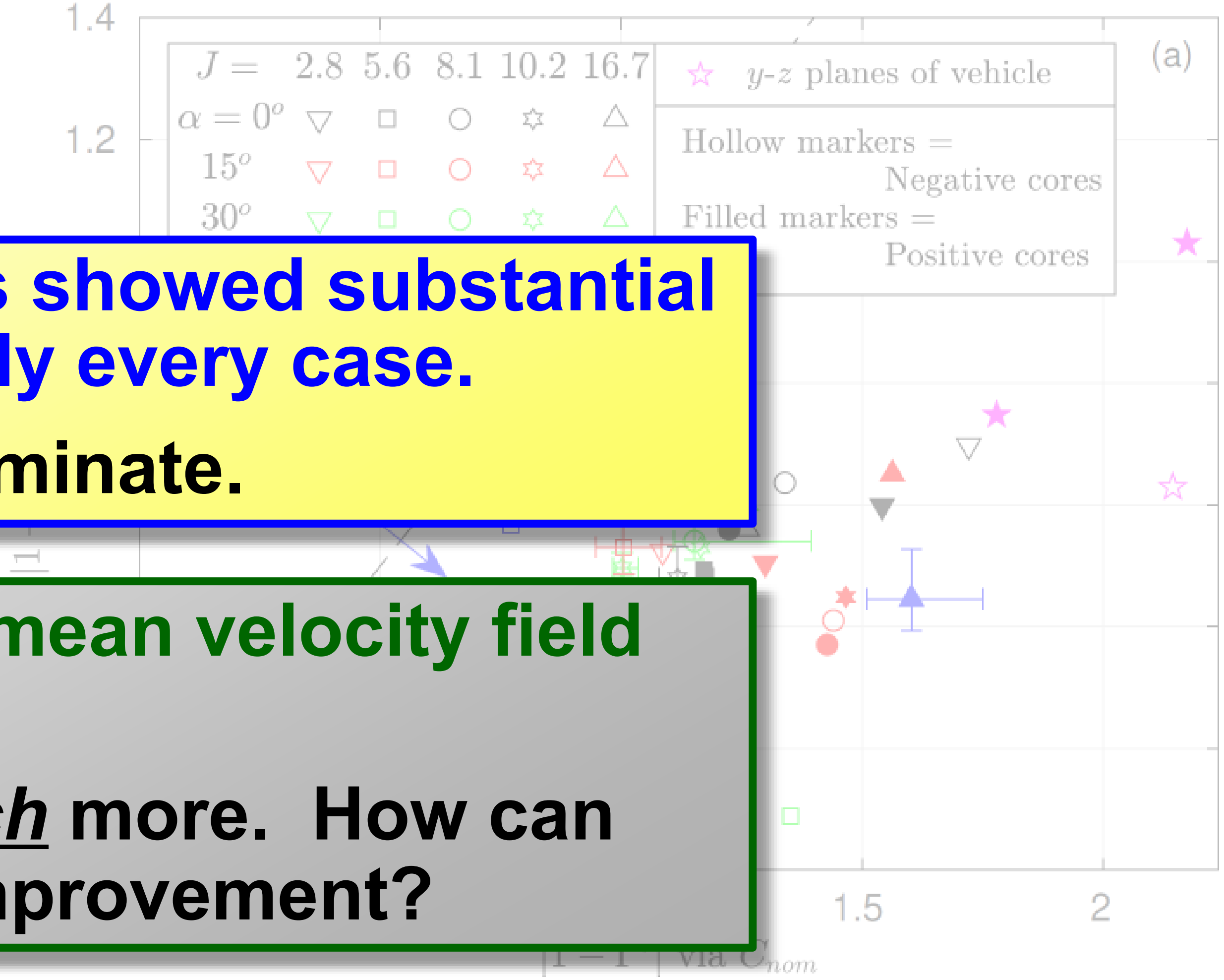
Validating the calibrated C_μ model.

We examined six quality metrics.

Here's one:



Here's another:



Five of the six metrics showed substantial improvement for nearly every case.

The sixth was indeterminate.

This simply uses the mean velocity field from PIV.

PIV can do much *much* more. How can we use it for model improvement?

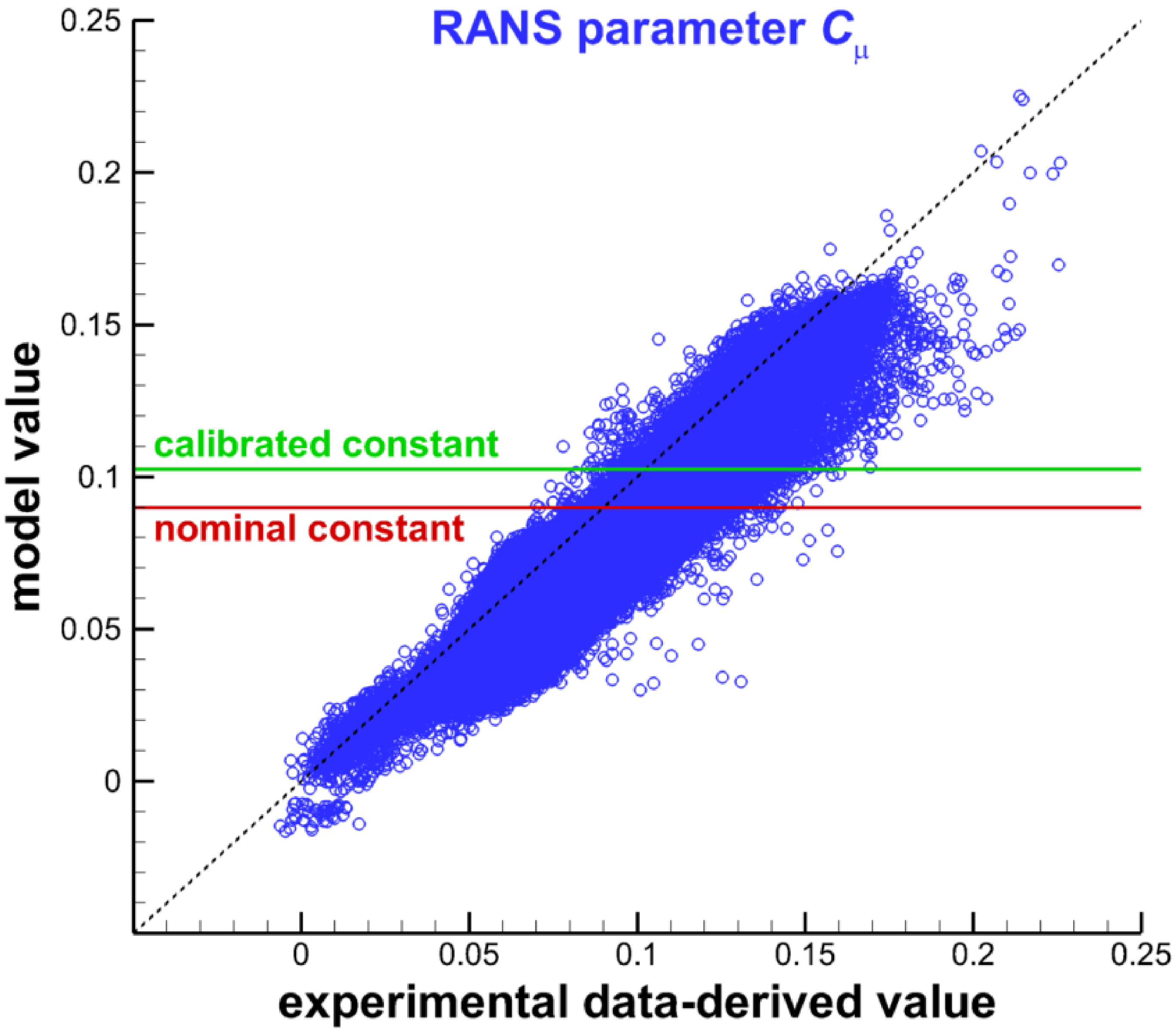
Move to a spatially variable C_μ model.

In a conventional RANS model, C_μ = 0.09.

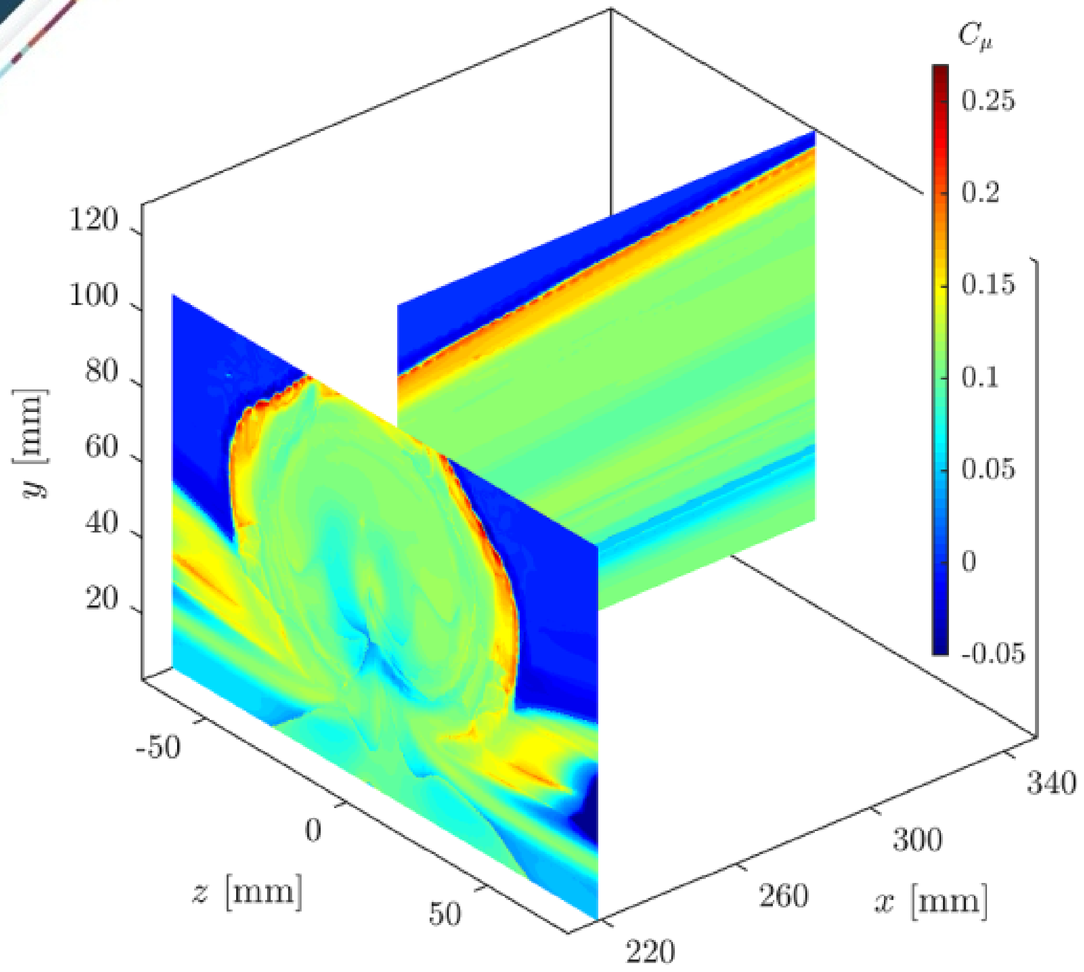
In our calibrated RANS model, C_μ = 0.1025.

New approach:

***C_μ* is allowed to vary spatially based on wind tunnel PIV data, rather than assuming a fixed constant.**



But we need C_μ everywhere!



Machine learning of C_μ from the PIV data…

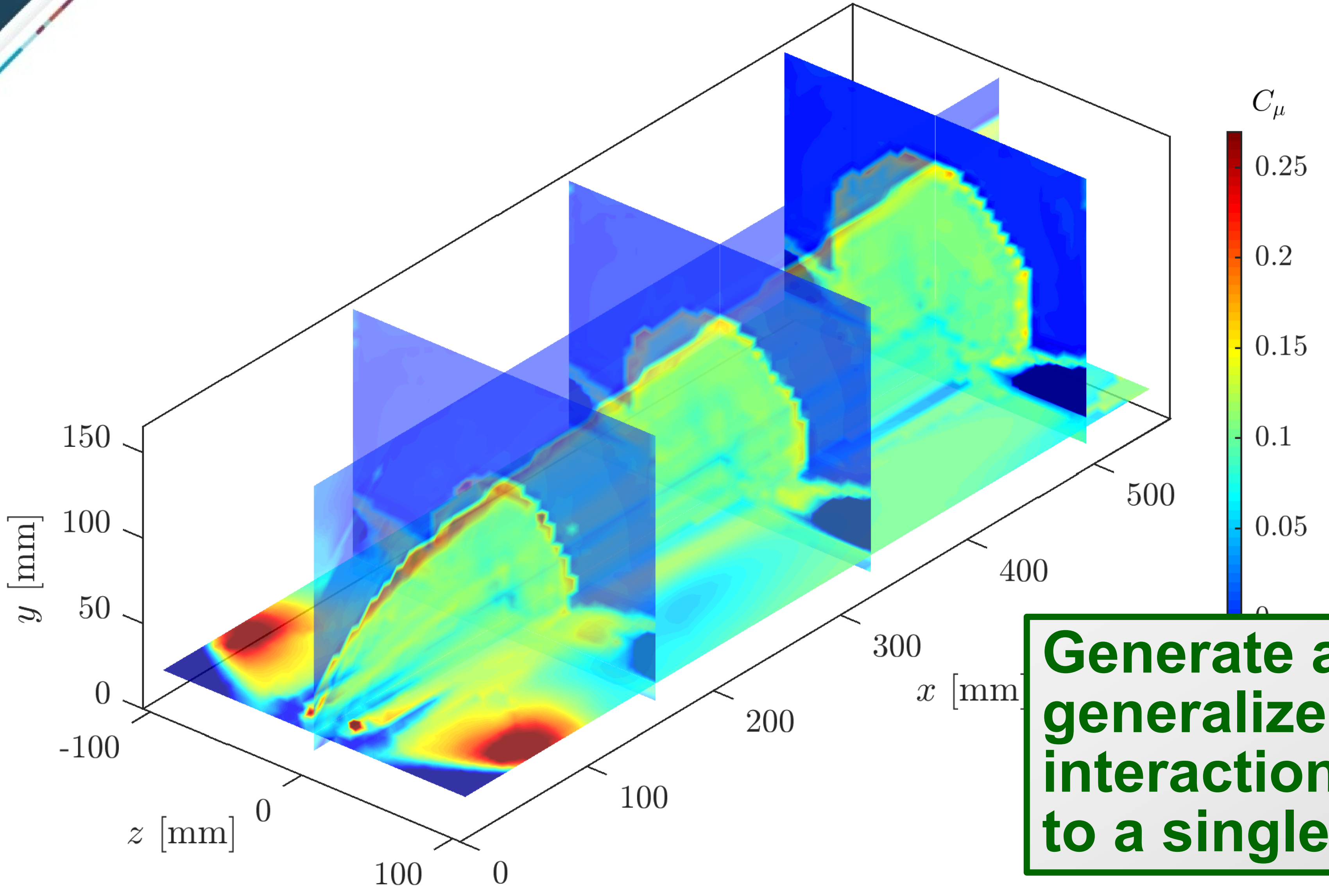
…constrained by fluid dynamics:

$$C_\mu = f\left(\hat{S}_{ij}, \hat{\Omega}_{ij}, P, \varepsilon\right)$$

The data provide C_μ in only two planes.

We must obtain C_μ over the entire computational domain.

Now we have C_μ everywhere!



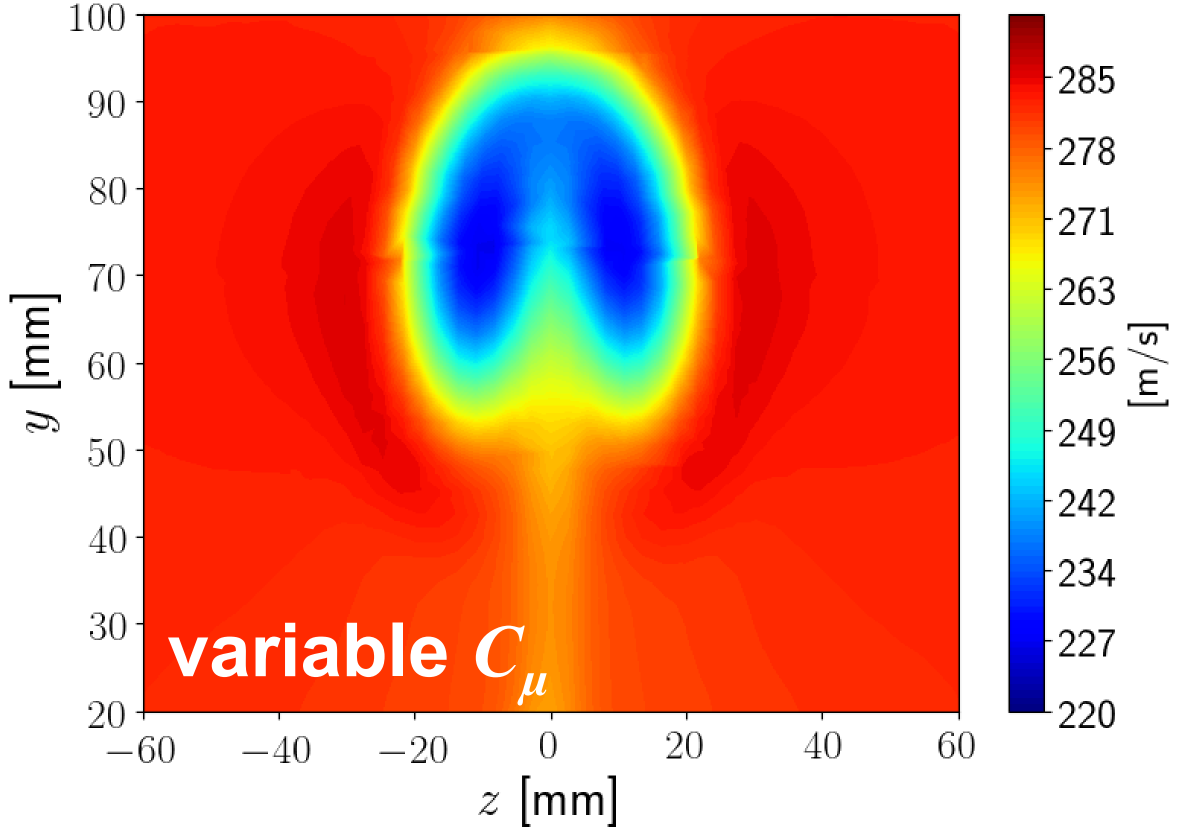
Now we have a new, sophisticated closure model for RANS!

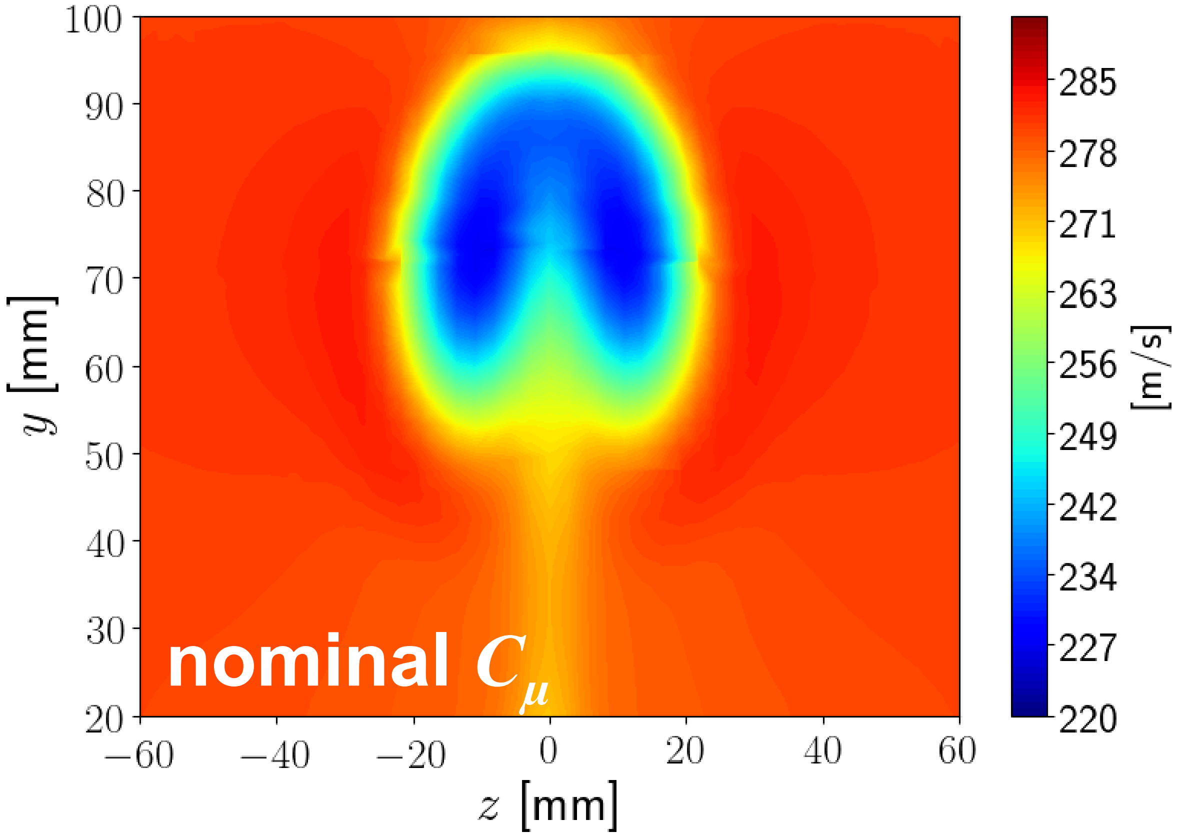
It adapts to the *local flow behavior* to map C_μ to any flow topology.

Generate a predictive capability generalized to any simulated jet interaction rather than restricted to a single test case.

How well does this work?

(Let's skip over the tedious part about building this model into a production RANS code.)





The variable C_μ simulation is nearly identical to the original *k*-ε model without any C_μ modification.

What's going on?

What is C_μ in unmeasured regions?

The PIV data miss important physics near the wall and the jet nozzle.

Default C_μ to nominal 0.09.
Avoid extrapolation that "blows up."

Result: Default C_μ dominates the result.

Using calibrated C_μ of 0.1025 shifts result to match calibrated case.

Another issue is *data consistency*.

C_μ model trained using measured *k* and ε, but RANS *k* and ε values may be in error.

Result: Inconsistent model yields erroneous C_μ.

What's next?

We find ourselves on the cutting edge of data-driven modeling and machine learning.

Testing novel techniques including inverse modeling and extrapolation detection.

Some additional digestion of the PIV data may be required before incorporation into an altered RANS model.

Data-driven CFD trained with PIV measurements of real physics rather than trained with LES/DNS.

Incorporate PIV data to make RANS suck less.