Department of Computer Science Technical Reports

Department of Computer Science

2000

Validation Methodology for GasTurbnLab

S. Fleeter

Elias N. Houstis
Purdue University, enh@cs.purdue.edu

John R. Rice
Purdue University, jrr@cs.purdue.edu

C. Zhou

Report Number:
00-002

Fleeter, S.; Houstis, Elias N.; Rice, John R.; and Zhou, C., "Validation Methodology for GasTurbnLab" (2000). *Department of Computer Science Technical Reports.* Paper 1480.
https://docs.lib.purdue.edu/cstech/1480

VALIDATION METHODOLOGY FOR GASTURBNLAB

S. Fleeter
E. Houstis
J. Rice
C. Zhou

CSD TR #00-002
February 2000

Validation Methodology for GasTurbnLab[1]

S. Fleeter, E. Houstis, J. Rice and C. Zhou

Abstract

The design of the mechanism to integrate validation into the GasTurbnLab problem solving environment is described. The goal is to facilitate the comparison of simulation results with one another and with physical data or analytic solutions. The primary tool is the visualization of data sets (computed, experimental or analytic) and their differences. Since experimental data almost never comprises a full solution, provisions are made for subsets of various types and for aggregated data.

This report comprises the January 2000 deliverables for Sub Tasks C1 and C2 for the Department of Energy ASCI project.

I. APPROACH

It is assumed in this report that the reader is familiar with the GasTurbnLab project as described in references [1], [2], [3] and [4]. These reports may be viewed at

`http://www.cs.purdue.edu/research/cse/gasturbn/publications/reports.html`.

The primary approach is to support the visualization of full or partial data sets representing physical quantities in a gas turbine. These quantities include both intrinsic physical variables (pressures, velocities, densities, concentrations) and aggregated data such as the average velocity or pressure along a cross section or over a period of time, the total force (integrand of pressure), or mass of pollutant at the outlet of the turbine. Since it is extremely difficult to make accurate measurements inside an operating turbine, comparisons of aggregate data are the principal method for validating simulations with experimental data. The actual experimental measurements made are normally determined by what is feasible within the experimental setup and not by what one wants to measure. Thus the geometry and quantities that are used in validations using experimental data are governed by the experimental setup. The simulation output must be processed numerically (integrated, differentiated, etc.) to produce data to be compared to the measurements. Further, the visualizations required for this comparison might not be within the standard framework of viewing simulation results. The special cases that arise from these considerations are analyzed individually and, if warranted, the numerical processing and visualization is incorporated as a high level function within the GasTurbnLab user interface.

The GasTurbnLab PSE has a visualization system based on the IRIS Explorer for the visualization of simulation results. Explorer automatically provides a variety of tools for things like changing color maps or scales, and viewing cross sections and simple subsets. The GasTurbnLab PSE has a specialized interface to the IRIS Explorer that provides direct access to the turbine

---

[1] This work is supported by DOE contract LG-6982 at Purdue University and by the Center for Advanced Computing Research at the California Institute of Technology

geometry, combines results from different simulations (e.g., stationary blade, a rotating blade and the combustor) into a single display and similar actions.

II. VALIDATION MECHANISM DESIGN

The primary tool for validation is to compare solutions visually. Frequently the differences between solutions is so large that placing plots (or cross sections) side by side is sufficient to assess the differences. Recall that the differences normally observed between "good" simulated and measured results are the order of 25% even for aggregated values. Rarely are there differences the order of 10%. This means that local differences are often the order of 100–200% at places and very obvious to the eye. Nevertheless, one needs to visualize the differences directly so GasTurbnLab will have built-in facilities to immediately display the difference between two solutions. Numerical values can also be tabulated for one or many points if desired.

The simulated and experimental solutions are actually sets of values given on a discrete set of points, a mesh or grid. It is trivial to take differences of two sets of data given on the same mesh or grid. It is common, however, for solutions to be given on different point sets. In this case the values from one point set are used as input to interpolation formulas to produce values on the other point set. The numerical method (interface relaxation) used in GasTurbnLab already includes a complete set of interpolation formulas for data defined on the meshes used in the simulations. Thus if either one of the solutions is a simulation results, then the difference can be computed and displayed easily. Note that the solutions being compared can be either 3D (steady state) or 4D (dynamic behavior). The experimental data might only be 1D or 2D.

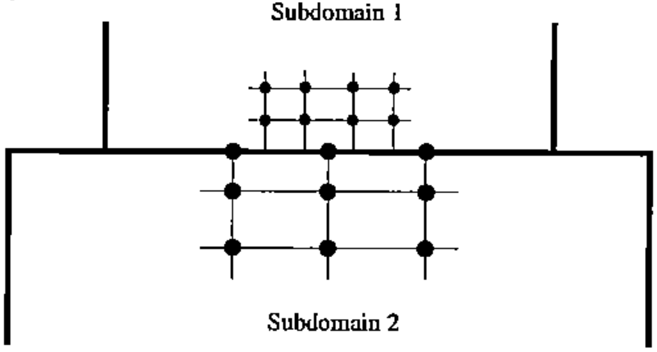
The discussion so far tacitly assumes that the solutions to be compared are for the same domains (even if they might have different meshes or grids). The GasTurbnLab PSE, however, employs multiple domains in its simulations. Thus a comparison of simulation results involves the following cases:

- The same domain, subdomains and meshes are used.
- The same domain and subdomains are used, but with some different meshes.
- The same domain is used, but with some different subdomains (and, hence, different meshes).
- The domain of one solution is a subset of another, but the subdomains of the smaller domain are also subdomains of the larger.
- The domain of one solution is a subset of the domain of another solution and some or all of the subdomains of the smaller do not match those of the larger.
- Neither domain is a subset of the other, but the subdomains in their intersection match.

- Neither domain is a subset of the other and the subdomains do not correspond. Some subdomains might be only partially in the intersection of the two domains.

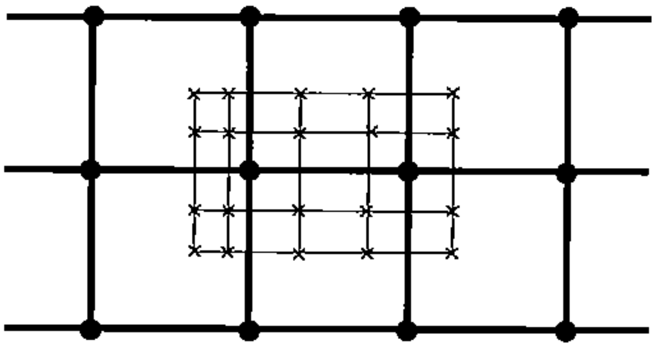
The visualization tool of GasTurbnLab will handle all of these cases automatically. The interpolation approach is directly applicable to some of these cases but in order to cover all these cases, it must be extended to handle the following situations.

(a) The interpolation involves points from two different and "incompatible" meshes. In two dimensions this might appear as follows:



At points along and near the interface of the two subdomains, one must use special interpolation formulas.

(b) The interpolation involves points from "outside" the intersection of the two domains or uses one sided interpolations. In two dimensions this might appear as follows:



No matter which of the two meshes is chosen for the plotting, there are points where the interpolation uses "unusual" points and data from outside the domain of comparison.

These situations are not difficult to handle in theory; there are methods known for interpolation for more or less arbitrary configurations of points. These situations are not difficult to handle in GasTurbnLab because low order (1st or 2nd order errors) discretizations are used in ALE3D and KIVA. Thus simple interpolations are as accurate as the simulation solutions. One must nevertheless take care to see that all the cases are recognized and handled correctly.

The comparison of simulations with analytical solutions is simpler because the analytical solution can be evaluated at arbitrary points and no interpolation is needed. There is another complication though, there are no analytical solutions for the geometry used in GasTurbnLab. Thus we introduce some truly simple cases into the set of turbine geometries that the GasTurbnLab PSE supports. For example, there are no blades in the compressor or turbines and the combustor is also a simple tube. There is no combustion either. Thus these comparisons only serve to show that the simulation does not have really gross errors.

III. AN EXAMPLE VALIDATION

We have already used visualization to validate some simulations and we describe how one error was discovered this way. Recall that a GasTurbnLab simulation normally involves several subdomains, each with a separate simulation. The subdomain simulations may be run in parallel or on separate machines or together on one machine. There are interface conditions between the subdomains derived directly from the physics of fluid flow, e.g., conservation of mass, momentum and energy. These conditions are used to derive relationships between mesh variables along the interface. Our initial trial was to have two subdomains, one with a fixed compressor blade in it and another with an adjacent (downstream) rotating compressor blade.

Two copies of ALE3D were used to simulate this situation and the output obtained is shown in Figure 1. The interface conditions used were physically plausible and the solution seen in Figure 1 corresponds well with the expected behavior of in the compressor. Namely, the axial velocity decreases in the direction of the flow (left-to-right), which indicates a corresponding pressure rise in this direction.

However, as experiments continued, it was suspected that the solution in Figure 1 was not correct. The ALE3D solution technique and interface conditions were formulated a second time, changing from an implicit to an explicit numerical approximation. The simulation then obtained is shown in Figure 2 and it too corresponds well with the expected behavior in the turbine, with the axial velocity decreasing in the direction of the flow. One might conclude that both solutions have about the same accuracy as the difference between them is in the order of the discretization error of ALE3D. The difference of the two solutions is shown in Figure 3. The difference looks larger since it is scaled differently, but it still could be argued (incorrectly) that the difference is just what one would expect between two low accuracy discretizations. But further analysis showed that the implicit approximation damped out the acoustic waves in the solution and thus made the overall turbine performance completely unrealistic. In other words, unsteady aerodynamic effects are missing in the implicit solution. It is just such incidents that underscore the necessity of continually testing and validating simulation results.

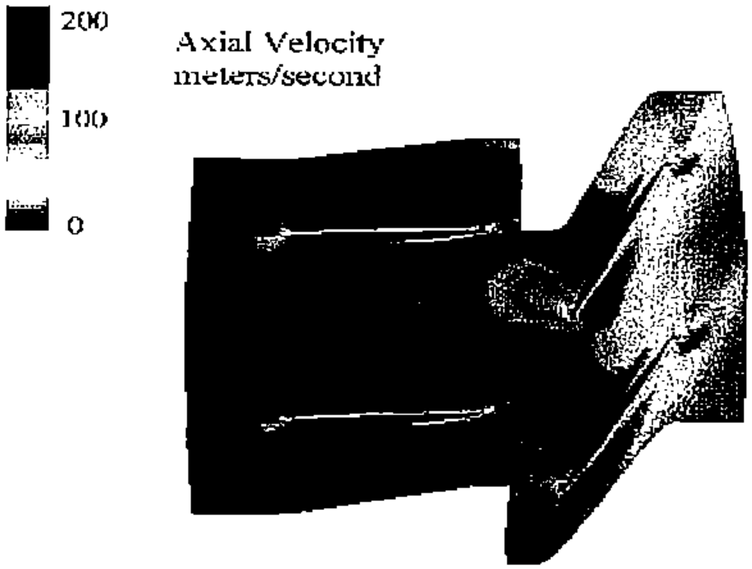


Figure 1. Parallel simulation of two rows. Fixed blades on left, rotating compressor blades on right. The solution assumes cylindrical symmetry for the 18 blades, but only two are shown. An implicit numerical approach is used.

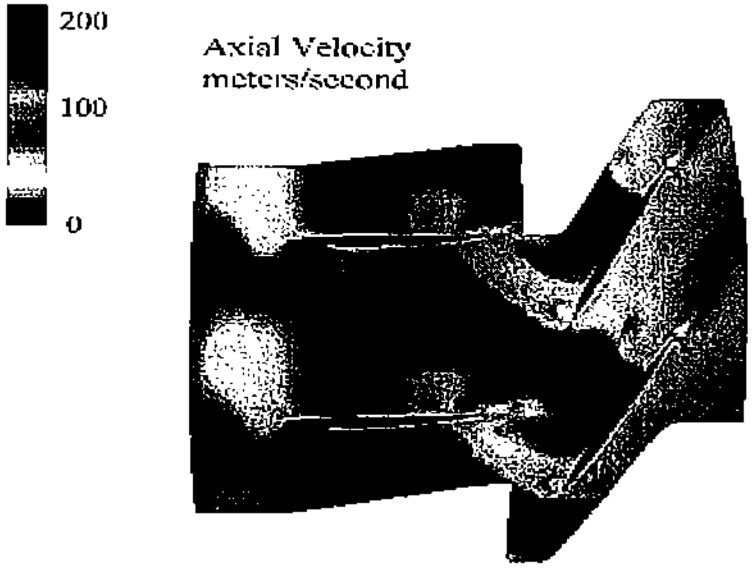


Figure 2. Parallel simulation of two rows as in Figure 1. An explicit numerical approach is used.

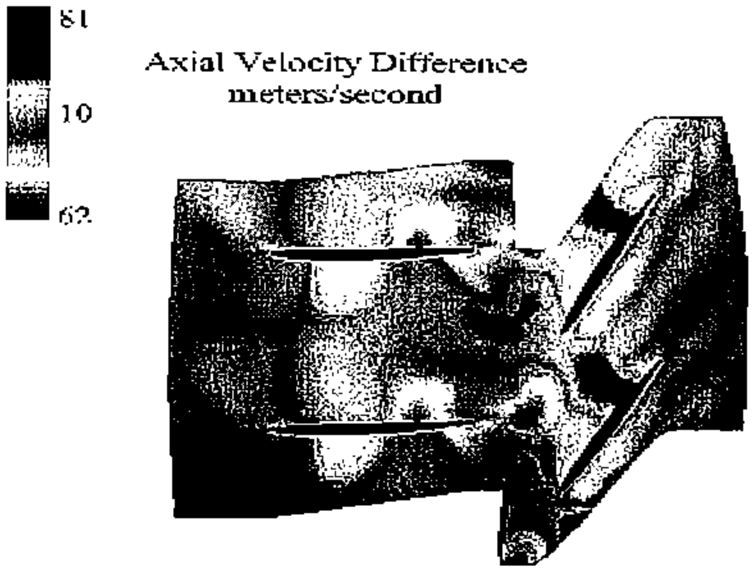


Figure 3. The difference between the solutions in Figure 1 and 2. The scale has been changed to exaggerate the difference.

IV. VALIDATION WITH EXPERIMENTAL DATA

Much of the discussion above applies to comparing simulation results with experimental data. The project is fortunate that it has a instrumented laboratory with an Allison XXX gas turbine. Thus we can coordinate the comparison of the simulation and real engines. Recall, though, the inherent difficulties in this validation.

- **The simulation involves many simplifications in the turbine model.** These are due to sources such as: (1) There is not enough computer power for an accurate simulation; (2) There is a lack of knowledge about certain phenomena in the engine such as in the fuel spray and the combustion.
- **Engines deviate slightly from their designs and, in any case, the exact engine specifications are industrial trade secrets.** Surprisingly, one cannot obtain the specifications even for an engine that one has.
- **Engines wear with age.** The one in our lab is 20 years old and it is not practical to directly measure all the parameters of the engine one has.

All these difficulties lead to the observed differences of 20–30% between simulated and measured values:

REFERENCES

1. S. Fleeter, E.N. Houstis, J.R. Rice, and C. Zhou. *GasTurbnLab PSE Design*, Tech. Rpt. 99-002, Computer Sciences Department, Purdue University, January 1999.

2. S. Fleeter, E.N. Houstis, J.R. Rice, and C. Zhou. *Gas Turbine Engine Compressor-Combustor Dynamics Simulation Design*, Tech. Rpt. 99-006, Computer Sciences Department, Purdue University, February 1999.

3. S. Fleeter, E.N. Houstis, J.R. Rice, and C. Zhou. *Gas Turbine Spray Dynamics and Combustion Simulation Design*, Tech. Rpt. 99–024, Computer Sciences Department, Purdue University, 1999.

4. S. Markus, E.N. Houstis, A.C. Catlin, J.R. Rice, P. Tsompanopoulou, E. Vavalis, D. Gottfried, and K. Su. *An Agent Based Netcentric Framework for Multidisciplinary Problem Solving Networks (MPSE)*, Tech. Rpt. 99–025, Computer Sciences Department, Purdue University, September 1999.