MCAT Institute
Final Report
93-004

NASA-CR-192289

GRANT
IN-16-CR

(NASA-CR-192289) MULTIPLE-BODY SIMULATION WITH EMPHASIS ON INTEGRATED SPACE SHUTTLE VEHICLE Final Report (MCAT Inst.) 15 p

N93-20234 3099

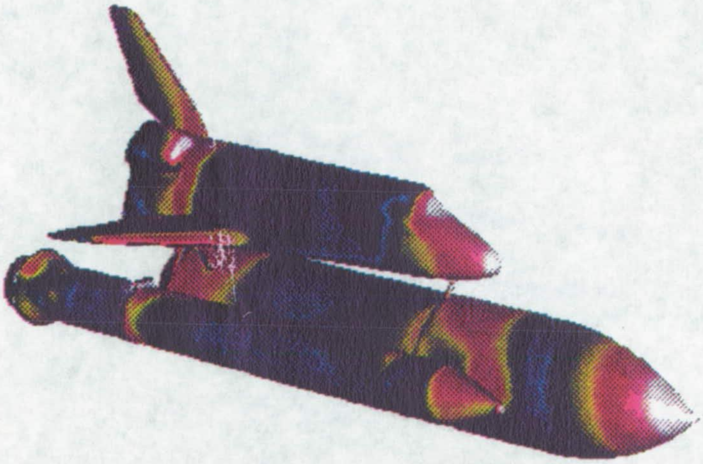
Unclas / 15
493335

G3/16 0148099

# MULTIPLE-BODY SIMULATION WITH EMPHASIS ON INTEGRATED SPACE SHUTTLE VEHICLE

Ing-Tsau Chiu

ORIGINAL PAGE
COLOR PHOTOGRAPH



February 1993 NCC2-692

**MCAT Institute**
**3933 Blue Gum Drive**
**San Jose, CA 95127**

ORIGINAL CONTAINS
COLOR ILLUSTRATIONS

**Multiple-Body Fl... Simulation with Emphasis on Integrated Space Shuttle Vehicle**

**Ing-Tsau Chiu**

## Abstract

The program to obtain intergrid communication — Pegasus — was enhanced to make better use of computing resources. Periodic block tridiagonal and penta-diagonal diagonal routines in OVERFLOW were modified to use a better algorithm to speed up the calculation for grids with periodic boundary conditions. Several programs were added to collar grid tools and a user friendly shell script was developed to help users generate collar grids. User interface for HYPGEN was modified to cope with the changes in HYPGEN.

ET/SRB attach hardware grids were added to the computational model for the space shuttle and is currently incorporated into the refined shuttle model jointly developed at Johnson Space Center and Ames Research Center. Flow simulation for the integrated space shuttle vehicle at flight Reynolds number was carried out and compared with flight data as well as the earlier simulation for wind tunnel Reynolds number.

# Multiple-Body Flow Simulation with Emphasis on Integrated Space Shuttle Vehicle

Ing-Tsau Chiu

## I. Introduction

This project is part of the long term computational effort to simulate the time dependent flow over the integrated Space Shuttle vehicle (orbiter (ORB), solid rocket boosters (SRBs), external tank (ET) and attach hardware) during its ascent mode for various nominal and abort flight conditions. Due to the inadequacies of experimental data such as wind tunnel wall effects and scaling effects and the difficulty of safely obtaining valid flight data, numerical simulations have been undertaken to supplement the existing aerodynamic data base. This data can then be used to predict the aerodynamic behavior over a wide range of flight conditions. Existing computational results [Buning et al, 1988, 1989] show relatively good overall comparison with experiments but further refinement is required to reduce numerical errors.

One of the major goals of this project is to obtain better agreement between numerical simulations and experiments. In the simulations performed so far, the geometry has been simplified in various ways to reduce the complexity so that useful results can be obtained in a reasonable time frame due to limitations in computer resources. In the past year, members of shuttle flow simulation team at Johnson Space Center and Ames Research Center were putting together the finer details of the shuttle computational model. Major components (ET, SRB and ORB) were modeled based on the CAD model from manufacturers of different components. Smaller components, e.g. attach hardware, thrust strut, oxygen feed line, etc., were added to account for the true blockage effect in the shuttle flow.

As the complexity of the space shuttle model increases (117 grids and 16 million points), a much larger computing resource is needed to obtain intergrid communication using Pegasus 3.x [Dietz and Suhs, 1989]. Pegasus 4.0 [Suhs and Tramel, 1991] was then adapted and modified for this task. Several major changes were made to make better use of computing resources and to increase the robustness of the code.

A minor improvement for periodic block tridiagonal and scalar penta-diagonal routines in OVERFLOW [Buning et al., 1991] was implemented. New algorithm reduces operation count and runs faster for grids with periodic boundary conditions.

User interface (UI) [Chiu, 1991] for HYPGEN [Chan and Steger, 1991] continued to evolve with new versions of HYPGEN and some changes were made to make better use of computing resources.

Several programs were developed or enhanced to help in generating collar grids. The procedure of generating collar grids is cumbersome and confusing for users not familiar with the concept. A script was developed to guide users through out the various steps of generating collar grids.

## II. Code Development

### Intergrid Communication — Pegasus

Over the years, Pegasus version 3.x has been the work horse in obtaining the intergrid communication for the integrated space shuttle configuration. However, with the complexity and size (117 grids and 16 million points) of the current geometric model for the integrated space shuttle approaching the real shuttle launch vehicle, a faster version is needed to obtain the intergrid communication. Therefore, Pegasus version 4.0 was adapted for this task. Although, Pegasus 4.0 in its original form is much faster than Pegasus 3.x, the required run time disk space is huge and without certain degree of modification, it is not possible to use it for the space shuttle configuration since the required run time disk space would have been so large (around 1400 million words) that even on the NAS Crays, one would have difficulty allocating enough disk space to run this program.

The modification made to reduce run time disk space usage was to write out the true size of the grids or surfaces. This reduces the run time disk space requirement by a factor of 10 and also reduces the I/O time since much less information needs to be written out.

Other improvements aimed at providing better performance and more robust stencil searching include:

- Use of Cray scientific routines whenever possible.
- Inserting inlining directive for calls to routines reported with big inlining factor by Cray performance utility programs.
- Implementing a more robust searching strategy to deal with axis boundary condition.
- Getting rid of gather/scatter type of array addressing to speed up the routine for nearest point search since this routine uses a major portion of CPU times.
- Using interpolation stencil found earlier as the starting point of stencil walk.
- Fixing bugs in reading Pegasus input file and adding meaningful error messages for users to pin-point errors in Pegasus input file.

The table below shows the performance improvement over the original Pegasus 4.0. The orbiter backend 19-grid configuration was used as the testcase. All tests were conducted on NAS YMP (Reynolds). Although the speed up resulted from the modifications made on the original version of Pegasus 4.0 varies from case to case, it can be concluded that the modified version is faster and is more useful in the environment with limited disk space.

| *Pegasus 4.0* | *Original* | *Modified* |
|---|---|---|
| *User Time (sec)* | 328.55407 | 96.71181 |
| *System Time (sec)* | 71.79700 | 14.66805 |
| *Disk usage (MW)* | 121.28 | 14.45 |

The following table shows the performance comparison of Pegasus 3.x and the modified Pegasus 4.0. The testcase is a 117-grid (16 million points) refined shuttle model for the modified Pegasus 4.0 and a 14-grid (2.05 million points) simplified shuttle model for Pegasus 3.x. All tests were performed on Cray XMP. It can be seen that even with a much bigger grid, the modified Pegasus 4.0 runs 34% faster than Pegasus 3.x with a smaller grid.

| | *Pegasus 3.x* | *Modified Pegasus 4.0* |
|---|---|---|
| *# of grids* | 14 | 117 |
| *# of points (million)* | 2.05 | 16 |
| *User Time (sec)* | 11922.6520 | 7609.0669 |
| *System Time (sec)* | 727.4653 | 700.7622 |

**Flow Solver — OVERFLOW**

The algorithm of the periodic block tridiagonal and periodic scalar penta-diagonal routines in OVERFLOW was changed and optimized . The memory requirement for each subroutine is reduced. Though this may not necessarily reduce the overall memory requirement for the OVERFLOW code, it does speedup the performance for these subroutines.

The algorithm used is to minimize the fill-ins when eliminating the off-diagonal blocks or elements. The floating operation count of this new algorithm is reduced by 21.8% and 33.3% for the periodic block tridiagonal and periodic scalar penta-diagonal respectively as compared to those originally implemented in OVERFLOW. Parentheses were also used in the code to enforce chaining of addition/subtraction and multiplication to squeeze out the last bit of the speedup from the code. This enforcement of chaining operation was also used for the non-periodic block tridiagonal and scalar penta-diagonal routines. Following table shows the CPU time per call for routines tested on Cray YMP and SGI workstations. The test used a 55 by 55 array and each routine was called from 55 to 2000 times.

| | Cray YMP | | SGI 4D-210 | |
|---|---|---|---|---|
| | *Original* | *New* | *Original* | *New* |
| *bltri* | 1.23e-2 | 1.17e-2 | 0.6719 | 0.6182 |
| *bltrip* | 3.11e-2 | 2.61e-2 | 1.5166 | 1.2587 |
| *vlpen* | 3.48e-4 | 3.40e-4 | 1.27e-2 | 1.19e-2 |
| *vlpen3* | 6.96e-4 | 6.31e-4 | 2.46e-2 | 2.19e-2 |
| *vlpenp* | 9.32e-4 | 7.02e-4 | 2.69e-1 | 3.13e-2 |
| *vlpenp3* | 1.51e-3 | 1.07e-3 | 3.53e-1 | 5.67e-2 |

Where:

| | |
|---|---|
| bltri: | non-periodic block tridiagonal |
| bltrip: | periodic block tridiagonal |
| vlpen: | non-periodic scalar penta-diagonal |
| vlpen3: | non-periodic scalar penta-diagonal with 3 RHSs |
| vlpenp: | periodic scalar penta-diagonal |
| vlpenp3: | periodic scalar penta-diagonal with 3 RHSs |

The speedup for periodic block tridiagonal and periodic scalar penta-diagonal routines ranges from 19% for bltrip to 41% for vlpenp3 due to reduction of operation count in the new algorithm. For non-periodic routines the speedup resulted from chaining of addition/subtraction and multiplication is very minor, from 2% to 5%, since the underlying algorithm stays the same.

## User Interface for HYPGEN

The user interface (UI) for HYPGEN continued to evolve with new versions of HYPGEN and new operating system for SGI workstations. Support for three additional boundary conditions introduced in HYPGEN version 1.3 were added. The frequency of mouse focus check was reduced for SGI workstations running IRIX 4.0 so that the system resource is better utilized. More error trapping — coincident points, grid dimensions, etc. — and online help for boundary conditions were added.

## Collar Grid Tools

Collar grid is a concept first proposed by Steger to treat intersecting grids in Chimera scheme. A collection of programs were developed over the years by Parks, Buning, Steger and Chan [1991]. Three additional modules to deal with surface-surface intersection, surface grid redistribution, and surface grid concatenation were added to the existing collar grid tools.

- Surface-surface intersection: The algorithm used in the surface-surface intersection program is based on line-surface intersection algorithm. Due to the simplicity of this algorithm, the speed of this program is usually two orders of magnitude faster than the existing version using a more general algorithm. For practical applications, the line-surface intersection is most often the cases that users want. However, the original version is still included in the distributed collar grid tools for those special occasions that the line-surface intersection algorithm does not give satisfactory surface-surface intersection.

- Surface grid redistribution: 2-D bilinear interpolation is used to redistribute grid points on a surface. This is an alternative to the true 2-D surface grid generator, SURGRD, developed by Chan.
- Surface grid concatenation: This program can be used to concatenate two arbitrarily oriented surface grids and relieves users from the need to orient two surface grids in the same orientation before concatenation.

Also implemented in the collar grid tools is a shell script to control all the modules in collar grid tools. Though the shell script is generic, it guides users to step through all the procedures needed to generate collar grids. A collection of examples are also included in the distributed collar grid tools to help users gain hands on experience in generating collar grids.

## III. Applications

### Shuttle Flow Simulation:

In the past, the shuttle flow simulation were mostly carried out for wind tunnel Reynolds numbers. The numerical solutions for the flight Reynolds numbers are needed to supplement the existing aerodynamic data base. Flows of two different freestream Mach numbers — 1.25 and 1.05 — were calculated. The Mach number at which the maximum pressure loading on the space shuttle occurs during ascent is usually close to these Mach numbers. The flow conditions as listed below were taken from the flight data.

| | |
|---|---|
| *Reynolds number (Re)* | $27 \times 10^6/1000in$ |
| *Angle of attack ($\alpha$)* | $-3°$ |
| *Elevon deflection (inboard/outboard)* | $10°/9°$ |

The computational model of the shuttle configuration consists of 14 grids and $2.05 \times 10^6$ points. Major component grids as seen in Fig. 1 include ET, ORB, SRB, orbiter tail, aft and forward ET/ORB attach hardware, seventeen-inch feedline and vertical strut. Two different SRBs — regular solid rocket motor (RSRM), and advanced solid rocket motor (ASRM) — were computed to check if ASRM reduces the wing load on orbiter. As can be seen from the Cp plot in Fig. 2, the pressure coefficient for the top surface of ORB in both RSRM and ASRM configurations show good agreement with the flight data (colored in open circle). Since both RSRM and ASRM differs only in the diameter of SRB, the pressure distribution on the top surface of ORB for both configurations is similar. As for the lower surface of ORB, ASRM shows smaller high pressure area near aft attach hardware than RSRM (see Fig. 3). This indicates that the wing load for ORB wing surface does reduce in the ASRM configuration. Fig. 4 shows the Reynolds number effect for the RSRM configuration. The increase in Reynolds number significantly increases the wing load on the ORB as can be seen from the larger red area for the flight Reynolds number case in Fig. 4.

### Grid Generation:

The CAD models of ET/SRB attach hardware (forward, aft upper and aft lower) were built based on the dimensions obtained from ICD (Interface Control Document) data book and Space Shuttle External Tank Layout Drawings. A commercial CAD software package, ICEM (Integrated Computer-aided Engineering and Manufacturing), was used to build the CAD models and generate the surface grids. The surface grids for all the attach hardware include parts of ET/SRB surfaces to allow intergrid communication between attach hardware and ET/SRB. Since the attach hardware intersects ET and SRB at an acute angle, the outer boundary of the volume grid generated is not far away from the surface and without a much refined ET or SRB grids, it is not possible to have adequate intergrid communication. Thus, each piece of the attach hardware is cut into two halves (see Fig. 5) and the volume grid of each half was generated separately. Since each half sees only one concave corner, the outer boundary of the volume grids can be as far as one desires. This not only provides enough overlap with ET/SRB grids to allow adequate intergrid communication but also makes a better grid. All the volume grids were generated using HYPGEN and later incorporated into the refined shuttle model developed jointly by members of shuttle flow simulation team at Johnson Space Center and Ames Research Center.

## IV. Conclusions

The modified Pegasus 4.0 was found to be faster, more robust and using much smaller computing resources than the original version. Collar grid tools are now easier to use due to the added shell script and the alternative surface-surface intersection program adds speed in obtaining surface intersection lines for a major class of grids. The new algorithm implemented in periodic block tridiagonal and scalar penta-diagonal routines in OVERFLOW enhances the performance for grids with periodic boundary conditions.

The added ET/SRB attach grids to the shuttle model account for the blockage effect between ET and SRB in shuttle flow and are currently used in shuttle flow simulation at Johnson Space Center. The shuttle flow solutions for flight Reynolds numbers at Mach 1.05 and 1.25 show reasonably good agreement in both flow structure and surface pressure to the available flight test results. However, major discrepancies can still be traced back to the use of geometric simplifications such as ET/ORB attach hardware grids that accounts for only about half the blockage of the true hardware and fuel feed lines. The comparison between the solutions of ASRM and RSRM configurations also indicate that ASRM can potentially reduce the wing load on the orbiter.

## V. References

Buning, P. G. and Steger, J. L. (1985). "Graphics and Flow Visualization in Computational Fluid Dynamics." AIAA Paper 85-1507.

Buning, P. G., Chiu, I. T., Obayashi, S., Rizk, Y. M. and Steger, J. L. (1988). "Numerical simulation of the integrated space shuttle vehicle in ascent." AIAA Paper 88-4359-CP.

Buning, P. G., Chiu, I. T., Martin, F., Meakin, R., Obayashi, S., Rizk, Y. M., Steger, J. L. and Yarrow, M. (1989). "Flowfield simulation of the space shuttle vehicle in ascent." Proceedings of the Fourth International Conference on Supercomputing, Santa Clara, California. 20–28.

Buning, P. G., Chan, W. M., Renze, K. J., Sondak, D., Chiu, I. T. and Slotnick, J. P., "OVERFLOW User's Manual, Version 1.6." NASA Ames Research Center, Moffett Field, CA, 1991.

Chan, W. M. and Steger, J. L. (1991). "A Generalized Scheme for Three-Dimensional Hyperbolic Grid Generation." AIAA Paper 91-1588.

Chiu, I. T. "UI User's Guide, Version 1.2a." NASA Ames Research Center, Moffett Field, CA, 1992.

Dietz, W. E. and Suhs, N. E. (1989). "PEGSUS 3.0 User's Manual." AEDC-TR-89-7.

Parks, S., Buning, P. G., Steger, J. L. and Chan, W. M. (1991). "Collar Grids for

Intersecting Geometric Components Within the Chimera Overlapped Grid Scheme." AIAA Paper 91-1587.

Suhs, N. E. and Tramel, W. E. (1991). "PEGSUS 4.0 User's Manual." AEDC-TR-91-8.

Figure 1. Geometry and surface pressure coefficient for shuttle configuration

ORIGINAL PAGE
COLOR PHOTOGRAPH

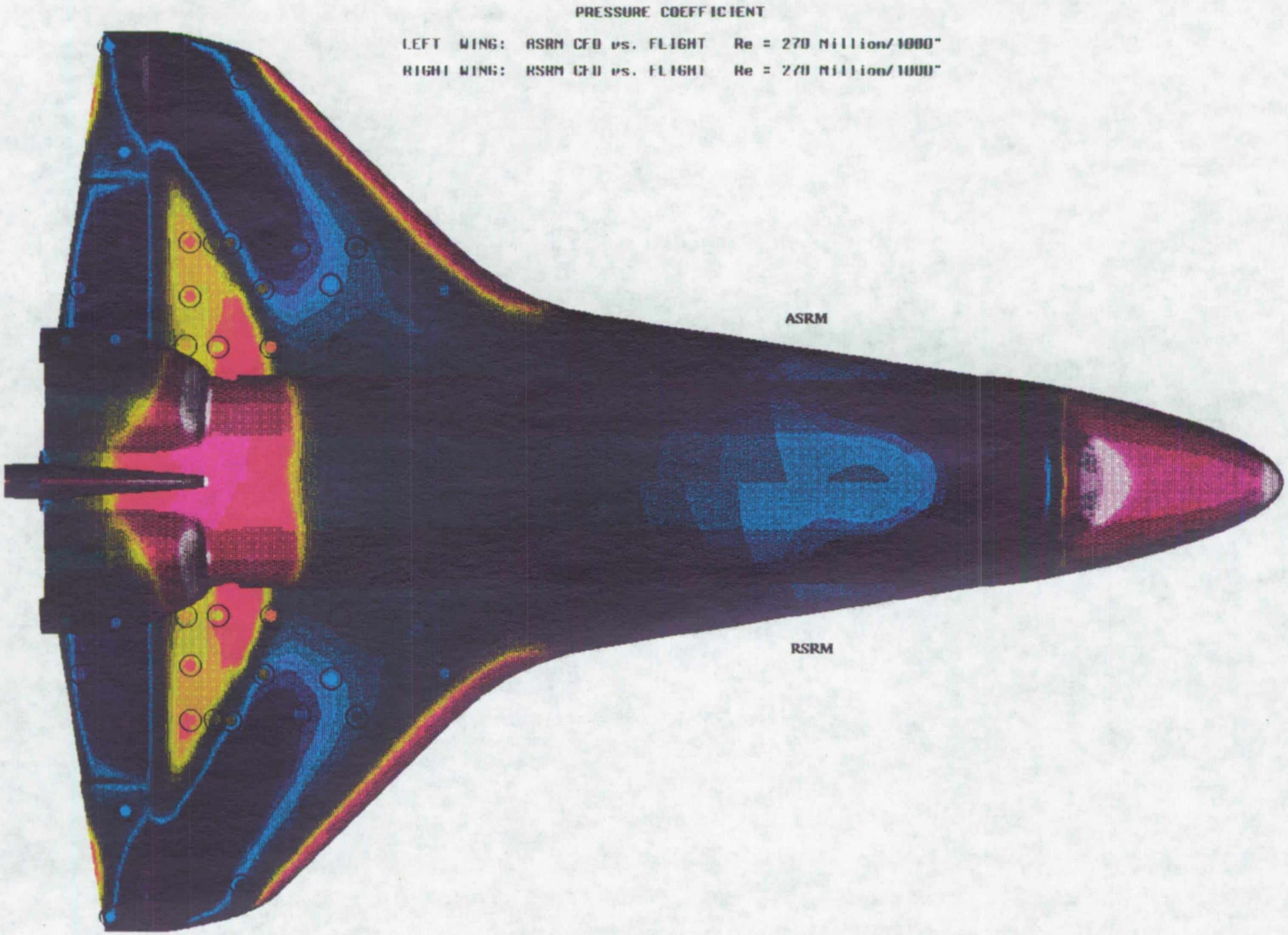



Figure 2. Comparison of surface Cp between ASRM and RSRM (top ORB surface)

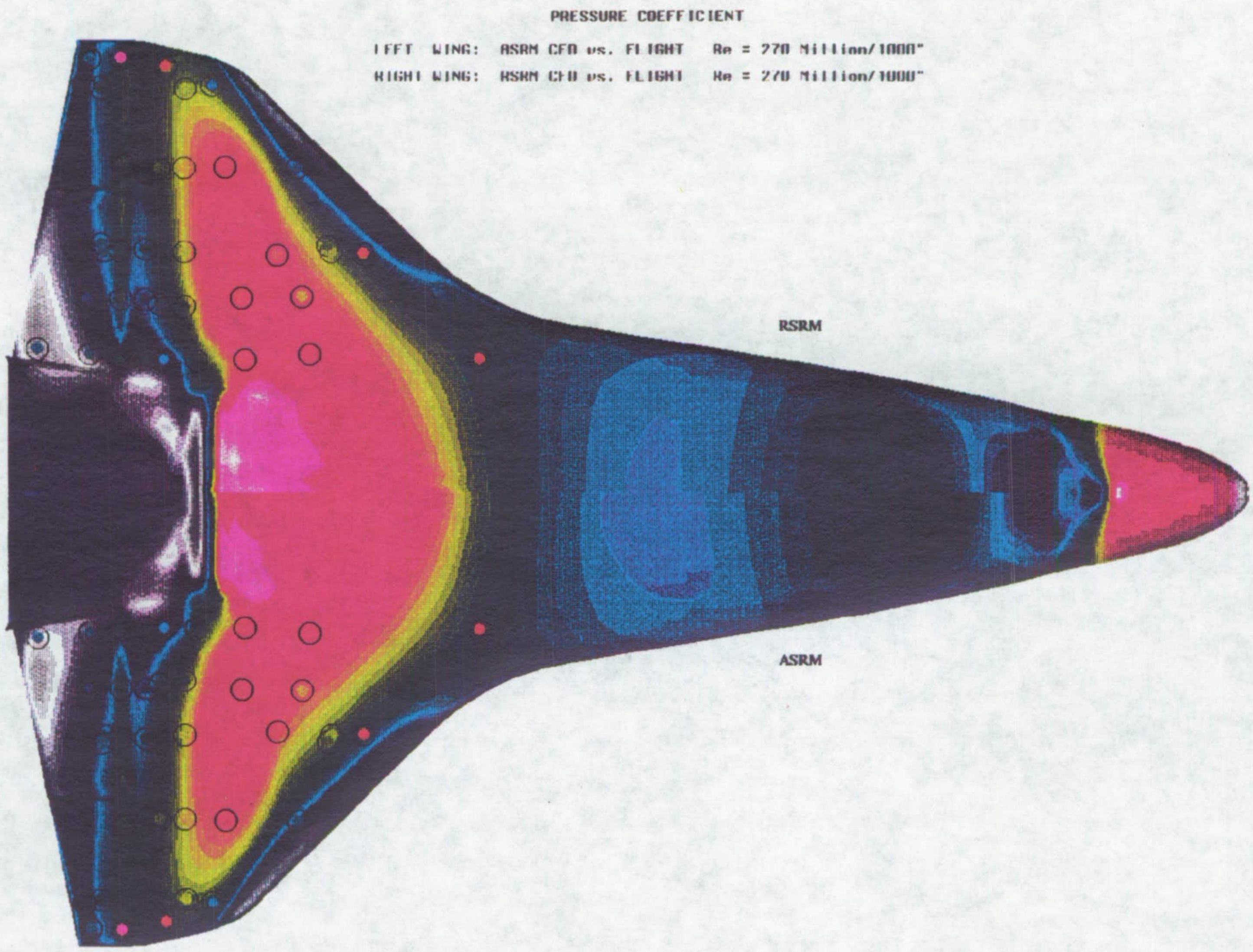



Figure 3. Comparison of surface Cp between ASRM and RSRM (bottom ORB surface)

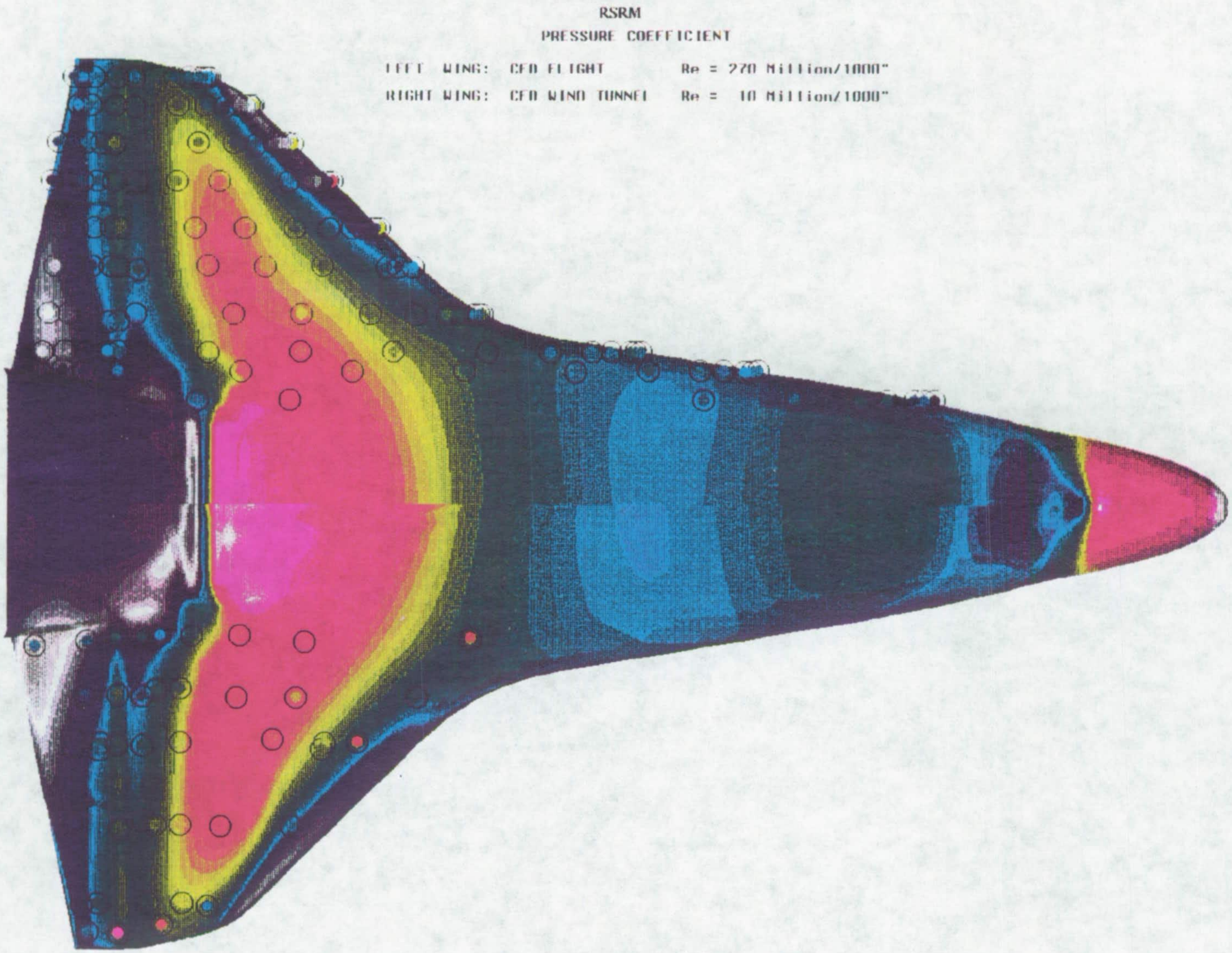



Figure 4. Comparison of surface Cp between flight and wind tunnel Reynolds numbers (bottom ORB surface)

GEOMETRY

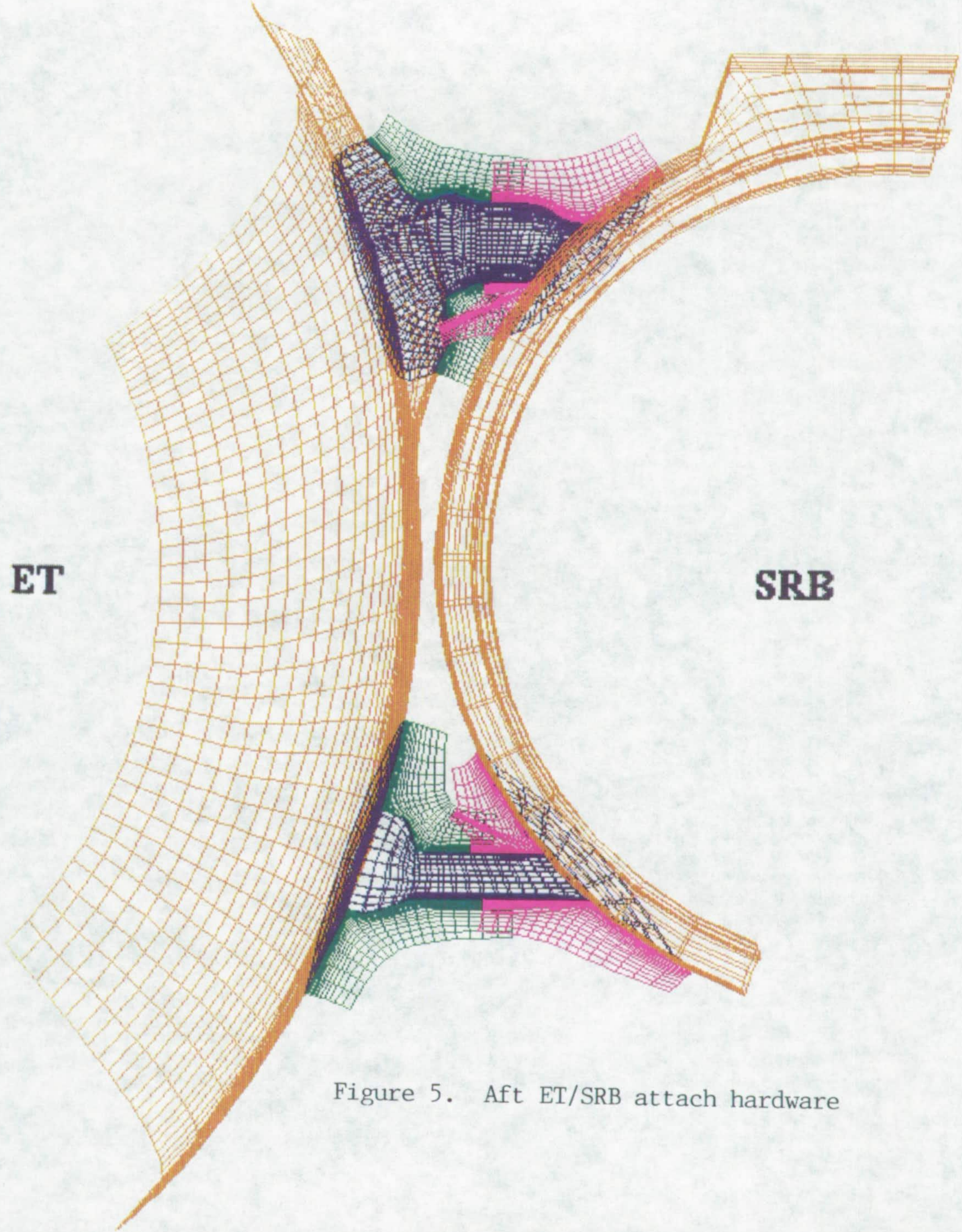



Figure 5. Aft ET/SRB attach hardware