SUPERSONIC TRANSPORT DESIGN ON THE IBM PARALLEL SYSTEM SP2

A. Jameson, Princeton University, Princeton, NJ
S. Cliff, NASA Ames Research Center, CA
S. Thomas, Sterling Software, Palo Alto, CA
T. Baker, Princeton University, Princeton, NJ
W. Cheng, IBM, Kingston, NY

THE COMPUTATIONAL AEROSCIENCES WORKSHOP

SANTA CLARA CONVENTION CENTER
SANTA CLARA, CA

March 7 - 9, 1995

SUPERSONIC TRANSPORT DESIGN ON THE IBM PARALLEL SYSTEM SP2

Antony Jameson, Princeton University, Dept. of Mechanical and Aerospace Engineering, Princeton, NJ 08544, jameson@cougarxp.princeton.edu,(609)258-5138

Susan Cliff, NASA Ames Research Center, MS 227-6, Moffett Field, CA 94035, cliff@ra-iris.arc.nasa.gov, (415)604-3907

Scott Thomas, Sterling Software, 1121 San Antonio Road, Palo Alto, CA 94303, thomas@ra-iris.arc.nasa.gov, (415)604-6387

Timothy Baker, Princeton University, Dept. of Mechanical and Aerospace Engineering, Princeton, NJ 08544, baker@cougarxp.princeton.edu, (609)258-5205

Wu-Sun Cheng, EA Segment/POWER Parallel System, MS ML7A/078, Enterprise Avenue, Kingston, NY 12401, wu-sun_cheng@vnet.ibm.com, (914)432-7931

The High Speed Aerodynamics Branch at NASA Ames has been using computational methods to develop improved High Speed Civil Transport (HSCT) configurations. These studies include reducing sonic boom intensity, improving aerodynamic performance, and optimizing nacelle-airframe integration [1,2]. The accuracy of these computations and the effectiveness of the design approaches have been confirmed by recent wind tunnel tests [3].

The AIRPLANE code has been used extensively in these investigations to treat complete HSCT configurations including nacelles and diverters. This program solves the Euler equations on an unstructured tetrahedral mesh [4,5]. An accurate computation requires a mesh of around 400,000 points and 2.5 million tetrahedral cells. 1000 iterations are usually required to produce a well converged solution for supersonic flow compared with 400 for transonic flow. A run of 1000 iterations on a 400,000 point mesh takes 2.9 hours on one processor of the Cray C-90 (10.5 seconds per iteration).

An example of a typical computation is presented in Figure 1 which shows pressure contours on the surface of a baseline HSCT configuration. Results obtained from AIRPLANE computations provide details of nacelle effects on the airframe that can be incorporated into a well validated wing/fuselage design code [1]. After reaching an improved design, AIRPLANE is run again to generate performance data for the new configuration. Figure 2 compares lift-to-drag ratios for the baseline configuration and an optimized design obtained by this procedure. At cruise condition the optimized configuration shows a dramatic improvement over the baseline. This improvement in HSCT aerodynamic performance was realized by modifications to the camber and twist distribution of the wing and the camber of the forward fuselage. Accurate prediction of the nonlinear effects plays an essential role in the design process.

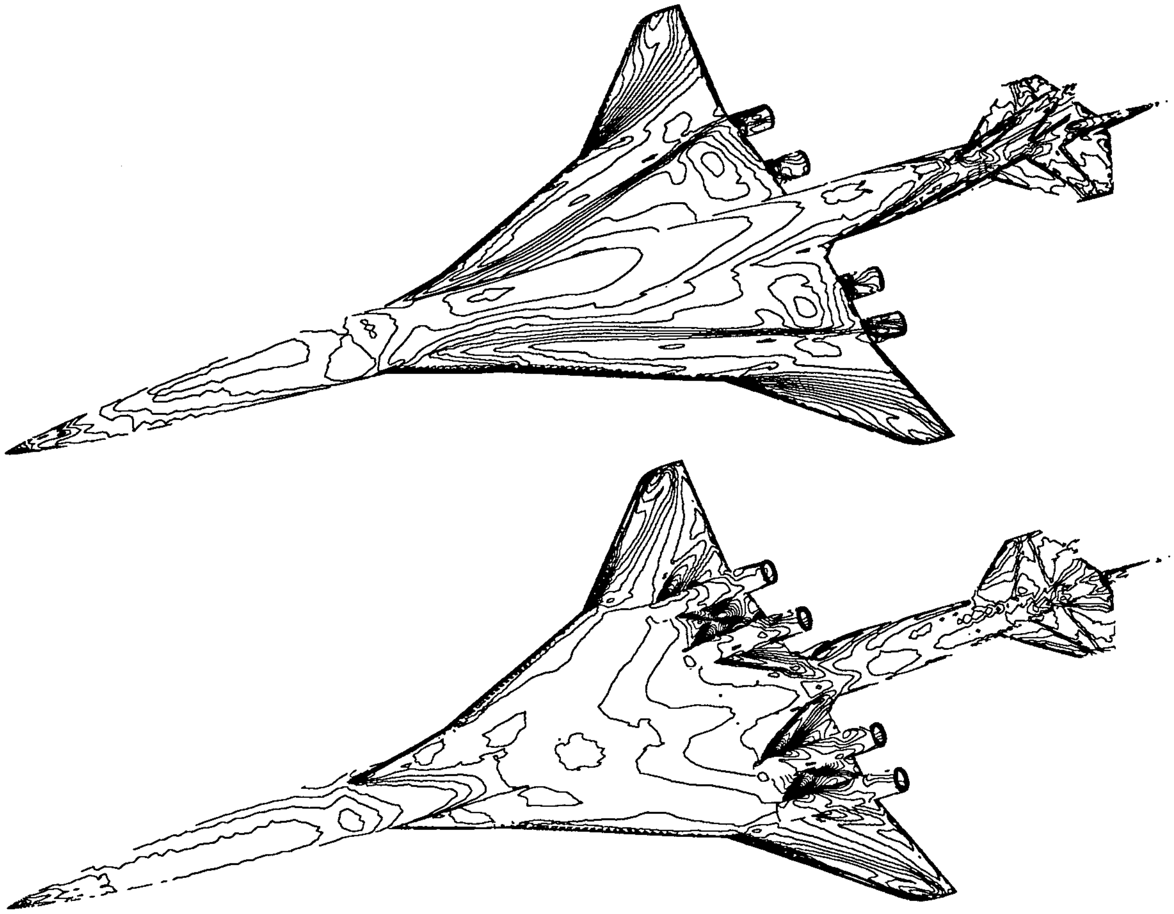

Figure 1: Upper and lower views of flow solution on Ref H

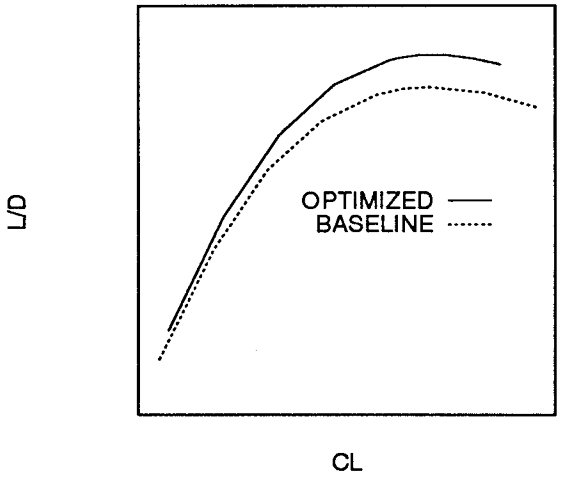

Figure 2: HSCT design validation L/D versus CL

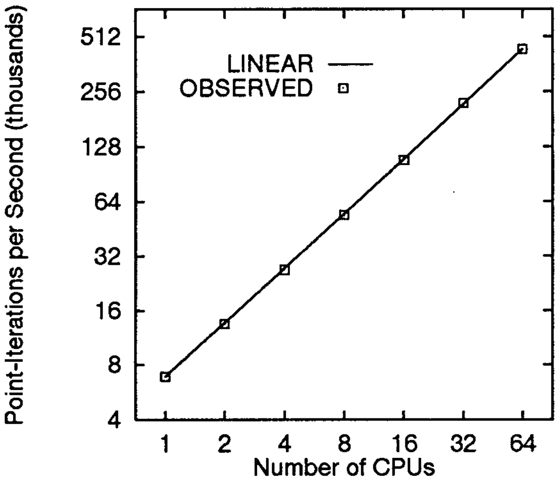

Figure 3: Parallel performance on the SP2

In order to improve the turn-around time and achieve greater efficiency in the design process, it was decided to establish a parallelized version of the AIRPLANE code on the IBM SP2 system at NASA Ames.

The AIRPLANE code had initially been parallelized with the PARMACS message passing interface [6]. Subsequently, a new parallelization was implemented on the IBM SP1 using the IBM developed Message Passing Library (MPL) in a Single Program Multiple Data (SPMD) strategy [7]. Under the SPMD strategy, the program is replicated and stored on each processor. Identical pre-processing of the mesh information is performed on each processor, and then only the data needed for the assigned subdomain is retained. The division of the mesh into subdomains is accomplished by means of a modified recursive coordinate bisection (RCB) domain decomposition algorithm which achieves an equi-distribution of points among the different processors. This procedure generates very efficient load balancing; the equi-distribution of points leads to a distribution of edges that varies only a few percent between processors.

In order to get optimal use of the cache memory, it is necessary to renumber the nodes and edges within the subdomain that is assigned to each processor. This significantly reduces the data retrieval from main memory since the proximity of data points in physical space corresponds to a small stride between memory locations. Because the nodes and edges are numbered independently, it is necessary to employ indirect addressing. The main loops of the flow solver are over the edges, with the result that each point will be referenced every time it appears as a vertex of an edge. On the average roughly 14 edges meet at each node, and the possibility of recursion may inhibit optimization by the compiler. To overcome this problem, the edges are sorted into groups so that no point is referenced more than once in each group. The compiler can then be directed to generate more efficient instructions, and to vectorize loops over edge groups.

During each time step, messages must be passed between processors to update flow variables for the evaluation of fluxes and artificial dissipation. On unstructured meshes the implicit residual smoothing is approximated by a two-step Jacobi iteration [4,5]. Considerable interprocessor communication time was measured on the SP1 in this part of the flow solver if it was faithfully implemented with message passing. By restricting the smoothing to each subdomain, the need for message passing in the smoothing routine was eliminated, and this improved parallel efficiency without significantly altering the convergence rate. Scheduled communication, whereby messages sent from one processor are synchronized with messages received from another, was built into the code using the IBM-developed MPL calls. It is apparent that the MPI message passing protocol is rapidly becoming accepted as the standard, and work is underway to replace the MPL calls by the corresponding MPI calls.

The IBM SP2 system consists of high speed, large memory RS/6000 class processors connected via a high speed switch. This permits heavy computational loading on individual processors and rapid interprocessor communication. An average parallel speed-up of 13.1 over 16 processors (82% parallel efficiency) was achieved previously on the earlier SP1 system with the AIRPLANE code [7]. The IBM SP2 has both faster processors (approximately twice the speed of the SP1) and a faster switch.

It therefore seemed likely that the parallelized version of AIRPLANE would perform even better on the SP2.

The parallelized version of AIRPLANE was recently ported to the IBM SP2 at NASA Ames. On the SP2, the first processor must have enough memory to handle domain decomposition, while the other processors only need enough memory to handle a subdomain. The results confirm a significant improvement in parallel performance over the SP1. Figure 3 presents the performance in point-iterations per second versus number of processors for a typical HSCT calculation. Almost perfect scalability is attained for up to 64 processors. This case runs at 1 iteration per elapsed second on 64 processors, and therefore a 1000 iteration flow solution can be completed in only 1000 seconds.

For the first time, flow-field computations for complete aircraft can be performed in 15 minutes, or less in the case of transonic flows. Such an improvement in computational performance provides the aerodynamicist with the opportunity to rapidly analyze a series of flow-field conditions for each design iteration. This significantly enhanced capability is now being exploited by the High Speed Aerodynamics Branch to derive even better designs for High Speed Civil Transport configurations. It also opens up the possibility of directly linking numerical optimization methods with flow analyses of complete aircraft.

1. S.E. Cliff, J.J. Reuther, M.J. Mann, R.M. Hicks, and T.J. Baker, "Supersonic Transport Wing Design by Numerical Optimization with Superimposed Nacelle Pressures," Proceedings NASA Workshop on Propulsion Airframe Integration, Cleveland, Ohio, October 1993.

2. S.E. Cliff, T.J. Baker, and R.M. Hicks, "Design and Computational/Experimental Analysis of Low Sonic Boom Configurations," NASA CDCP-1001 High Speed Research: 1994 Sonic Boom Workshop, NASA Langley, June 1994.

3. S.E. Cliff and T.J. Baker, "Nacelle Integration Studies for HSCT Configurations Using AIRPLANE," NASA TM 108865, March 1995.

4. A. Jameson, T.J. Baker, and N.P. Weatherill, "Calculation of Inviscid Transonic Flow over a complete Aircraft," AIAA 24th Aerospace Sciences Meeting, Reno, Nevada, AIAA Paper 86-0103, January, 1986.

5. A. Jameson, and T.J. Baker, "Improvements to the Aircraft Euler Method," AIAA 25th Aerospace Sciences Meeting, Reno, Nevada, AIAA Paper 87-0452, January 1987.

6. E.A. Gerteisen, and A. Jameson, "Massive Parallel Implementation of the Aircraft Euler Method and Performance Tests on Different Computational Platforms," Procs. parallel CFD '93/ GAMNI Conference, Paris, May 1993.

7. W.S. Cheng, A. Jameson, T.J. Mitty, "AIRPLANE on the IBM Parallel System SP1," Parallel CFD '94 Conference, Kyoto, Japan, May 1994.