

## 9 CFD コード共通化プロジェクト UPACS の現状

山根 敬、山本一臣、榎本俊治、高木亮治、山崎裕之、牧田光正、山本 武、岩宮敏幸、中村 孝  
(航空宇宙技術研究所 UPACS チーム)

### Current Status of Common CFD Platform - UPACS

by

Takashi YAMANE, Kazuomi YAMAMOTO, Shunji ENOMOTO, Ryoji TAKAKI, Hiroyuki YAMAZAKI  
Mitsumasa MAKIDA, Takeshi YAMAMOTO, Toshiyuki IWAMIYA, Takashi NAKAMURA

*National Aerospace Laboratory*

#### ABSTRACT

UPACS, *Unified Platform for Aerospace Computational Simulation*, is a project to develop a common CFD platform since 1998 at National Aerospace Laboratory. The project aims not only to overcome the increasing difficulty in recent CFD code programming on parallel computers for aerospace applications which includes complex geometry problems and coupling with different types of simulations such as heat conduction in materials and structure analysis, but also to accelerate the development of CFD technology by sharing a common base code among research scientists and engineers. Currently the development of UPACS for compressible flows with multi-block structured grids is complete and further improvements are planned and being carried out.

## 1 はじめに

計算流体力学 (CFD) 技術が脚光を浴びるようになってから 20 年以上が経ち、CFD がさまざまな流れの解析に役立つことが実証されてきた。また CFD 分野の研究成果として市販ソフトウェアも数多く登場している。しかしながら、CFD が CFD そのものの研究という面から脱皮して、設計や研究のツールとなるにはさらなる飛躍が求められている。

その背景の一つは、「ツールとしてより使いやすく、理解しやすく」というニーズである。これまでは CFD を十分に活用するには、流体力学の知識や CFD プログラム利用に必要なノウハウだけでなく、ソフトウェアの中身にある程度知ったうえで自分の目的に合うように改造する能力も求められた。しかし専門の研究者がそれぞれの研究目的に合わせてプログラムを書いたため、CFD プログラムのバリエーションが増大するとともに複雑化し、基本的な数値解法が似ていても共通的に利用することが困難になり汎用化や CFD 技術の継承の障害となってきた。一方で CFD に対するツールとしての要求と期待はますます高まっており、中でも航空宇宙分野においては、

- より複雑な形状に適用できる
- 高精度、高信頼性
- 流体解析だけでなく、燃焼や二相流の解析、熱伝導や構造解析との連成解析

等が求められている。これらを実現するには、中心となる流体解析プログラム部分に柔軟な拡張性と汎用性が必要になっている。

もう一つの背景はコンピュータアーキテクチャの進化である。CFD の進化はコンピュータの高速化に支えられてきた。まずベクトルプロセッサによるスーパーコンピュータが登場したことで CFD の可能性が大きく広がり、並列計算機によりさらに大規模計算が可能になったことで CFD の実用性が高まった。しかしコンパイラによる自動並列化の性能が十分でない現状で並列計算機を十分に活用するには並列処理とハードウェアの知識が必要であり、また、ハードウェアに依存するコンパイラ特有のプログラムを記述することで他のハードウェアへの移

植性 (ポータビリティ) が損なわれることもある。最近では PVM や MPI など並列計算のための標準的なメッセージパッキングライブラリが普及しているが、CFD プログラマーが個々に自分のプログラムを並列化するのは基本的なプログラムの枠組みがかなり似通っているだけに多重投資となり無駄である。

このような現状を打破するために始められた UPACS (Unified Platform for Aerospace Computational Simulation) プロジェクトは今年で 3 年目を迎えた。UPACS は、マルチブロック構造格子を用いる圧縮性流体数値シミュレーションを分散メモリマルチプロセッサマシンにおいて実行することができる共通的な基盤コード (プラットフォーム) を確立し、この基盤コードを核にすることにより、従来、同じような計算手法であるにもかかわらず、適用する問題や解析を実行する計算機環境が異なるために、その開発に多重投資を強いられていた状態を改善するとともに、系統だった CFD コードの検証やノウハウの蓄積を効率的に実現できるようにすることを目標としている。現在、複雑形状まわりの圧縮性流れの解析について、いくつかの解法を使って計算できるようにプログラムが整備され、最初のバージョンとしてユーザ会形式で公開した。さらに今後の拡張予定についてもここで紹介する。

## 2 UPACS の特徴

UPACS コードは、次のような開発方針で整備を進めている。

- 複雑形状への対応と計算精度を両立させるためにマルチブロック構造格子法を利用する
- CFD ソルバー部を並列マルチブロック処理部から分離する
- 特定のハードウェアへの依存性を局所化する
- コードの階層化構造とデータおよび計算手順のカプセル化
- プログラム実行に必要な支援ツールを整備する
- Fortran77 で書かれたこれまでのプログラム資産からのスムーズな移行

これらの開発方針と特徴の大部分は、すでに昨年の航空宇宙数値シミュレーション技術シンポジウム 1999 で発表した [1] が、重要なものをここで改めて紹介する。

## 2.1 Fortran90 と MPI によるプログラミング

UPACS で目指している CFD コードは、「コードの階層化構造とデータおよび計算手順のカプセル化」という開発方針において、一部にオブジェクト指向的考え方を取り入れている。今日、オブジェクト指向プログラミングと言うと C++ が代表的なプログラミング言語とされているが、UPACS では日本におけるこれまでの CFD プログラムのほとんどが Fortran77 で書かれていることを考慮して、Fortran90 を採用した。Fortran90 は、Fortran77 よりも構造化プログラミングという面で優れており、また、行列演算などに関する拡張によりこれまでよりも少ない行数での記述が可能になり、ソースプログラムの可読性も向上している。また、ベクトル計算機での高速化チューニングについては、Fortran の方が C や C++ よりも実績があるため、新しい機能とこれまでの技術やプログラムの継承のやりやすさの両面から、Fortran90 がもっともバランスがとれていると判断した。

一方、並列処理には MPI[2] を採用した。現在、MPI はほとんどの計算機で利用可能になっており、「特定のハードウェアへの依存性をできるだけ排除する」という開発方針を満たすものである。なお、並列処理が必要ない場合は、MPI を使用しなくてもプログラムを実行できるように UPACS は設計されている。

## 2.2 マルチブロック構造格子法

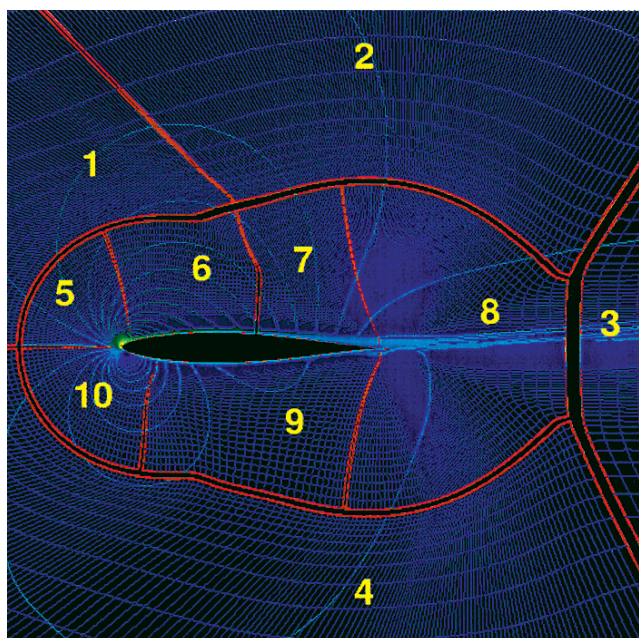


図 1: 二次元翼まわりのマルチブロックの例

これまでの CFD においては、構造格子による数値解析手法が、特に粘性計算において数多く研究されてきており、計算精度の点でも有利である。しかしながら、複雑な物体形状にあわせて単一の構造格子を作るのは難しい作業である。そこでいくつもの構造格子を組み合わせることで複雑な形状にあわせた格

子を作りやすくするのがマルチブロック構造格子法で、UPACS でもこの方法を採用した。現在の UPACS では、ブロックどうしの境界上の格子点座標は接合するブロックの格子点と一致しなければならないが、ブロックの組み合わせかたに関しての自由度は大きく、複雑物体まわりの計算格子の作成が比較的容易になっている。図 1 は二次元翼まわりのマルチブロックの例を示しており、10 個のブロックで構成されている。現在の UPACS ではブロック数の上限を 1000 個に設定しているが、このような多数のブロックの接合に関する処理をおこなうツールを備えている点が UPACS の特徴でもある。

UPACS の並列処理はこのブロック単位でおこない、ユーザがどのプロセッサにいくつのブロックを処理させるかあらかじめ指定する方式を採用している。一つのブロックを複数のプロセッサで計算することはできないので、計算の高速化のためには使用できるプロセッサ数を考慮してブロック数を決める方が有利である。

## 2.3 CFD ソルバー部を並列マルチブロック処理部から分離

前述のように、個々のブロックの処理は一つのプロセッサ内で完結している。そこで UPACS のソルバー部分は単純に単一ブロック内を解くサブルーチンとして記述することができ、並列マルチブロック処理からは分離されている (図 2)。

たとえば新しいソルバーを試したいときは、単純なシングルブロック用のイメージでサブルーチンを作って UPACS に組み込むことで、並列マルチブロック処理が可能になる。

さらに、ブロック毎にソルバーを切り替える機能の搭載も計画しており、熱伝導解析や構造解析などの流体以外の解析ルーチンとの連成・組み合わせも計画している。

## 2.4 プログラムの階層構造とデータおよび計算手順のカプセル化

ソルバー部と並列マルチブロック処理の分離を概念図で表すと、図 3 のようになる。

Main Loop Level は解析コードの骨組みを構成するメインプログラムに相当し、CFD プログラムに共通的にあられる繰り返し計算処理を実行する。並列計算の処理の一部はこのレベルにも必要だが、並列計算の初期化のほか、ブロック間転送の種類や回数、タイミングなどをプログラムで指定するためのもので、具体的なブロック間のデータ転送処理などは中間層である Multi-Block Level が行う。

この中間層 (Multi-Block Level) では、ブロックとプロセッサの割り当て処理、ブロック間通信などを行う。各ブロックの計算で境界処理に必要な接続部分のデータは、この中間層が各ブロックが持つバッファに転送しており、個々のブロック内部の計算時には接続している他のブロックとの通信は必要ない。

Single-Block Level では、ブロック単位の流体計算を行う。前述のように一つ一つのブロック内の計算は他のブロックから完全に独立して実行できるので、この部分のプログラムは単一ブロック用のイメージで書けば良い。実際の計算に必要な接続しているブロックの物理量情報は、ブロック接続情報に基づいて中間層の Multi-Block Level が処理しているので、ブロック接続に必要なデータのやり取りは Single-Block Level では必要ない。また、Single-Block Level で使用する変数全体は一つの構造体にしてあるので、中間層の Multi-Block Level では構造体の内部構成の変更に影響されない。このように、Single-Block Level のサブルーチンでは、他のレベルのプログラムに影響を与えずに変数を変更、追加することが可能になっている。

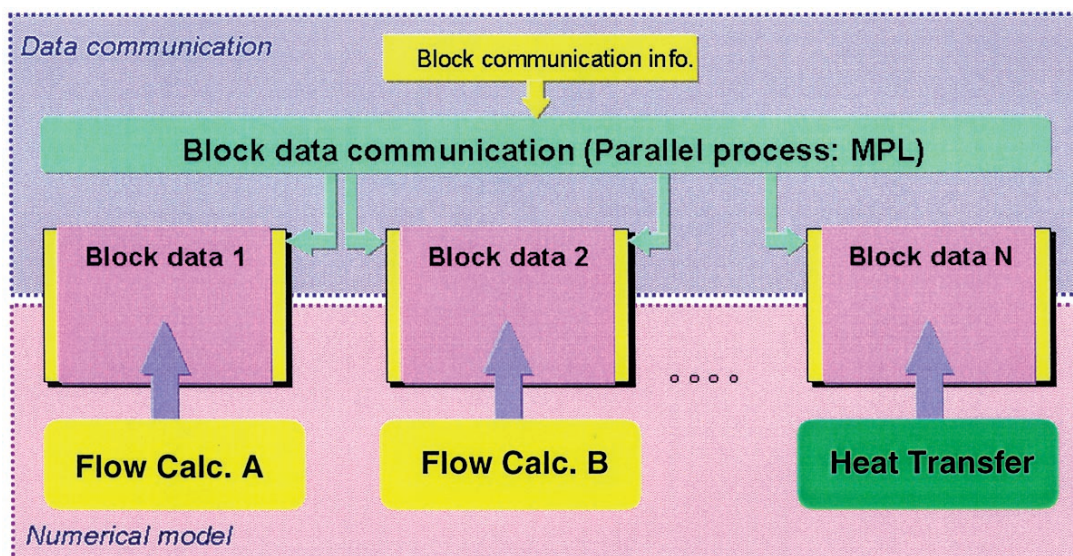


図 2: 並列マルチブロック処理とソルバー部の分離の概念図

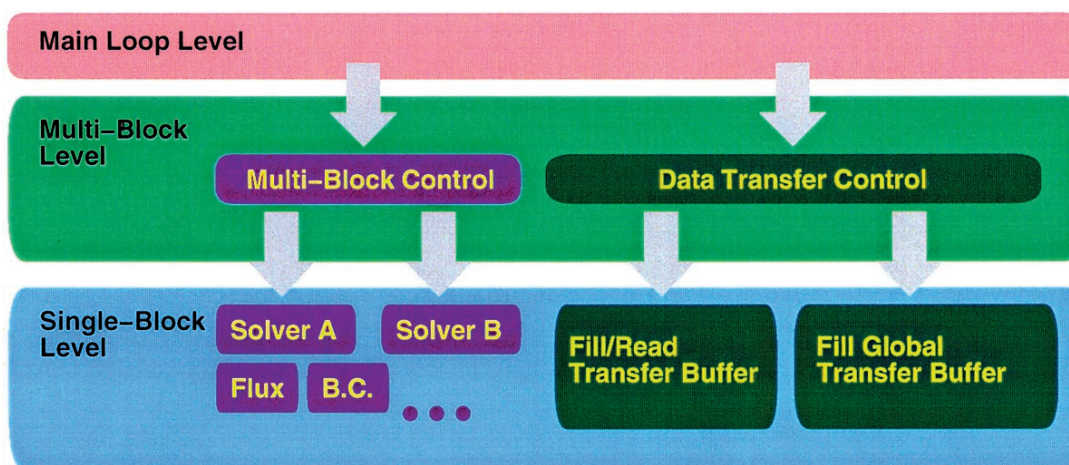


図 3: UPACS の階層構造の概念図

### 3 UPACS による流体解析手順

UPACS を使って流体解析を実行する実際の手順を示したものが図 4 である。流体解析プログラム `upacsSolver` を動かすために準備する必要があるファイルは主に次の 4 種類である。

- 計算格子データファイル (Grid data)
- ブロック接続情報ファイル (Block Connection Information)
- 境界条件設定ファイル (B.C.Information)
- パラメータデータベースファイル (Parameter Database, PDB File)

以下に順を追って説明する。

#### 計算格子データファイル (Grid data)

一つ一つのブロック内部の格子をつくる際には、通常の構造格子と同じやり方で作成できる。ただし、ブロックの境界が別のブロックと接合している場合には、境界上の格子点は接合しているブロックの境界上の格子点と一致していなければならない。

#### ブロック接続情報ファイル (Block Connection Information) 及び境界条件設定ファイル (B.C.Information)

マルチブロック格子を使った流体解析では、ブロックとブロックがどのように接続しているかの情報を、プログラム内部で指定するかパラメータとして与えるようにするのが普通であるが、計算する対象により接続形態が変わるためこの情報をいかに汎用化するかがプログラム開発の鍵の一つである。

UPACS ではこの煩雑な作業を大幅に軽減する前処理プログラム、`createConnect3D` を開発した。`createConnect3D` は、ブロック毎に分かれたグリッドファイルとユーザが作成した境界条件設定ファイルを読み込んで、ブロックどうしの接合情報をすべて出力するとともに、境界条件の指定からも接合面からももれている未定義の境界位置を出力する支援プログラムである。ユーザはこれをもとに境界条件設定ファイルを修正し、再度 `createConnect3D` を走らせ、未定義境界をチェックする。このプロセスを繰り返して未定義境界がなくなればブロック接続情報ファイルと境界条件設定ファイルが完成する。

この `createConnect3D` がマルチブロック構造格子法を採用している UPACS の汎用化に重要な役割を果たしており、特定の形状・条件に依存しないソルバー部を作ることが可能にし

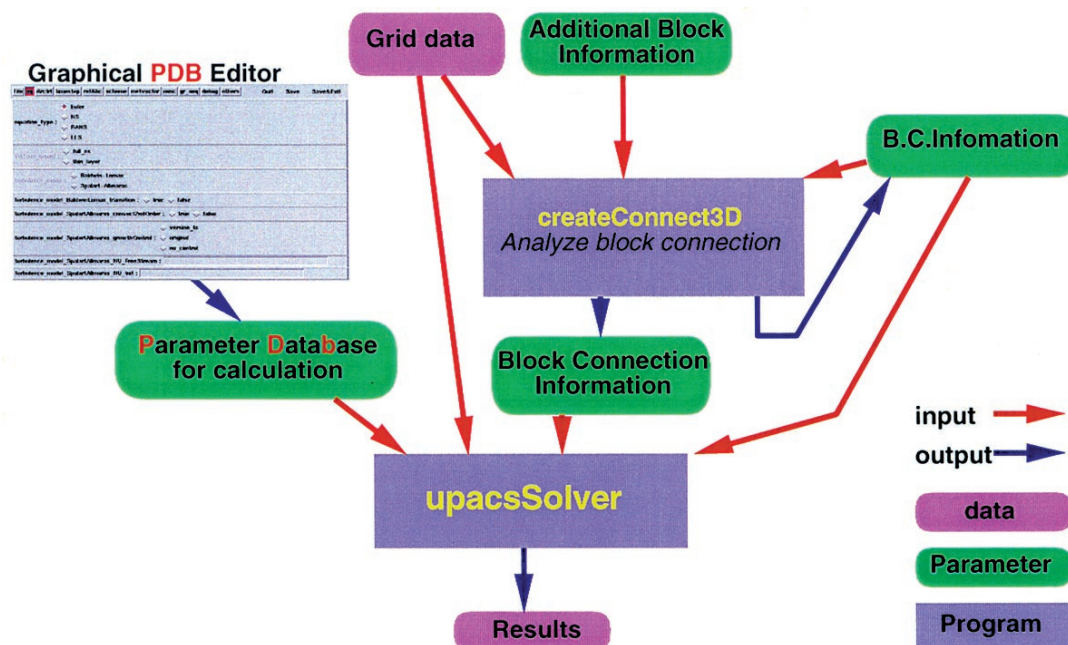


図 4: UPACS による流体解析の具体的手順

ている。

#### パラメータデータベースファイル (Parameter Database)

UPACS に限らず、流体解析では境界条件の値、計算ステップ数、時間刻みの大きさ、数値解法のパラメータ、収束判定条件など、種々のパラメータを与える必要がある。UPACS ではこれらのパラメータを記述したファイルを PDB ファイルと呼んでいる。PDB ファイルは、パラメータ名と値（数値、論理値あるいは文字列）を対にして一行に 1 パラメータずつ記述したテキストファイルで、UPACS ソルバーは実行時に PDB ファイルからパラメータ名をキーワードに必要な値を検索し、変数に代入する。ソルバープログラムは必要なパラメータ名だけを検索するので、PDB ファイルには実行に不必要なパラメータ名や値が入っていてもまったく問題なく、さらにファイル中のパラメータ名とその値は任意の順番で記述できる。

PDB ファイルに書く必要があるパラメータの数は多く、全てをテキストエディタで作成するには煩雑でわかりにくい。パラメータには相互に依存関係があるものがある。そこで PDB ファイルの作成を支援する PDB エディタを開発した。最初に作られた PDB エディタは X-Window 上で動く GUI ツールで、現在の UPACS ソルバーの機能のみに対応しており、入力する値が数値か論理値か文字列かを区別し、選択式パラメータの場合は選択肢を示すとともに、パラメータの依存関係にもある程度対応した。

しかしながら UPACS 自体が発展途上で随時更新されており、また UPACS のユーザが独自にソルバー部を改良してパラメータが変わる場合もあるため、PDB エディタも簡単に修正できなければならない。そこで PDB エディタを、定義ファイルに基づいて動作するように大幅な改良を実施すると同時に、ソルバー部のソースプログラムから PDB ファイルのパラメータ名を検索している箇所を自動的に調べて、PDB エディタで入力しなければならないパラメータとそのデフォルト値のリストを作るツールも作成した。この改良で、GUI プログラムの知識がなくてもソルバー部分を作る人自身が PDB エディタを作り変えることが可能になった。

以上のファイルの他に、並列計算を実行する場合には、どのプロセッサにどのブロックを割り当てるかテキスト形式で指定

する簡単なファイルを用意する必要がある。

以上で UPACS による数値解析の準備が完了し、ソルバープログラム、**upacsSolver** を実行することができる。実際に UPACS で数値解析を実行している状態のイメージが図 5 で、計算中も一定時間毎にあらかじめ指定しておいた断面の物理量などを表示することができる。

図 6 は UPACS による数値解析の例で、航空宇宙技術研究所で実施している次世代 SST プロジェクトにおいて飛行実験を予定している小型実験機をモデルにした [3]。計算格子は 105 個のブロックで構成されているが、その接続情報は UPACS で自動的に生成されており、解析プログラム自体にはこの形状のための特別な修正は加えられていない。

## 4 UPACS の現状と今後の計画

UPACS は現在、圧縮性完全気体の流れについて、以下の解法を選択して解析が可能である。

対流項	Roe スキーム、AUSMDV
時間積分	Runge-Kutta、Matrix Free Gauss Seidel
乱流モデル	Baldwin-Lomax、Spalart-Allmaras

さらに、次のような機能の搭載、改良を計画している。このうち、※印のものは 2000 年度（平成 12 年度）中に作業を完了する計画である。

- ※ ターボ機械などの回転機械内部流れへの適用
- ※ 航技研で開発されてきた数多くのスキームの移植
  - ・ 2 方程式乱流モデル及び LES
  - ・ 非構造格子
- ※ オーバーセット（重合）格子
- ※ 物体内部の熱伝導解析との連成
  - ・ 化学反応を含んだ解析
  - ・ 構造解析との連成
- ・ 低マッハ数流れ
- ・ Adjoint 法による最適化問題への対応

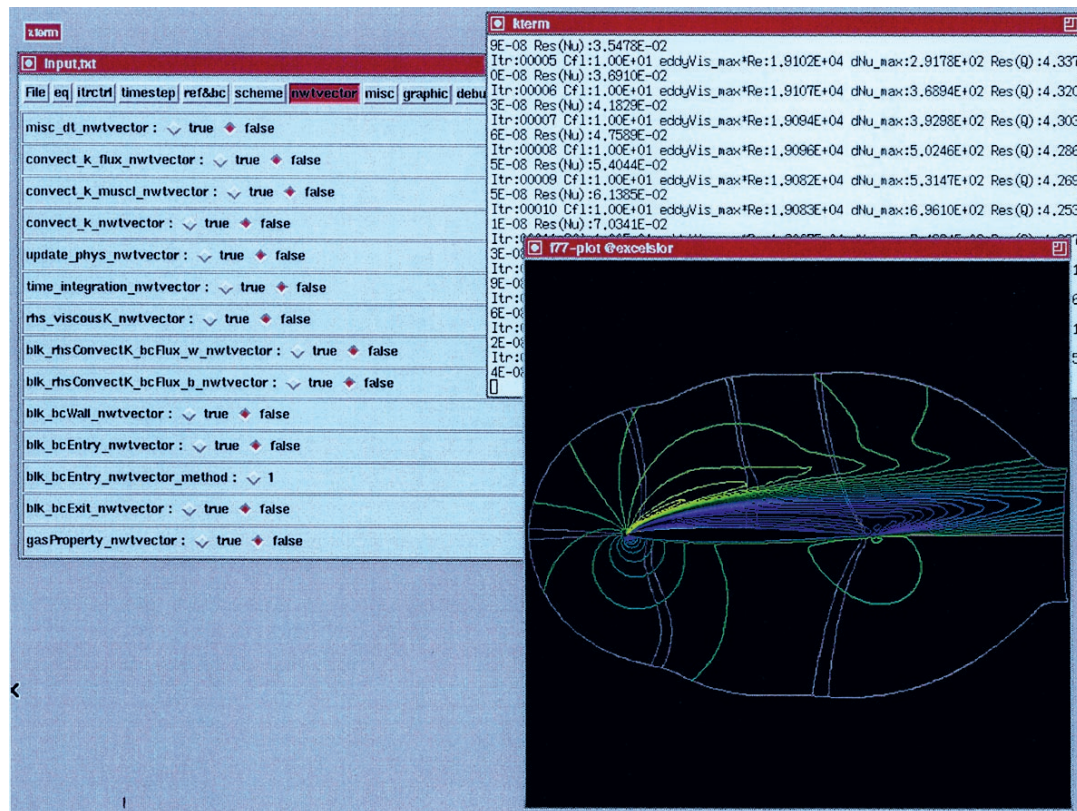
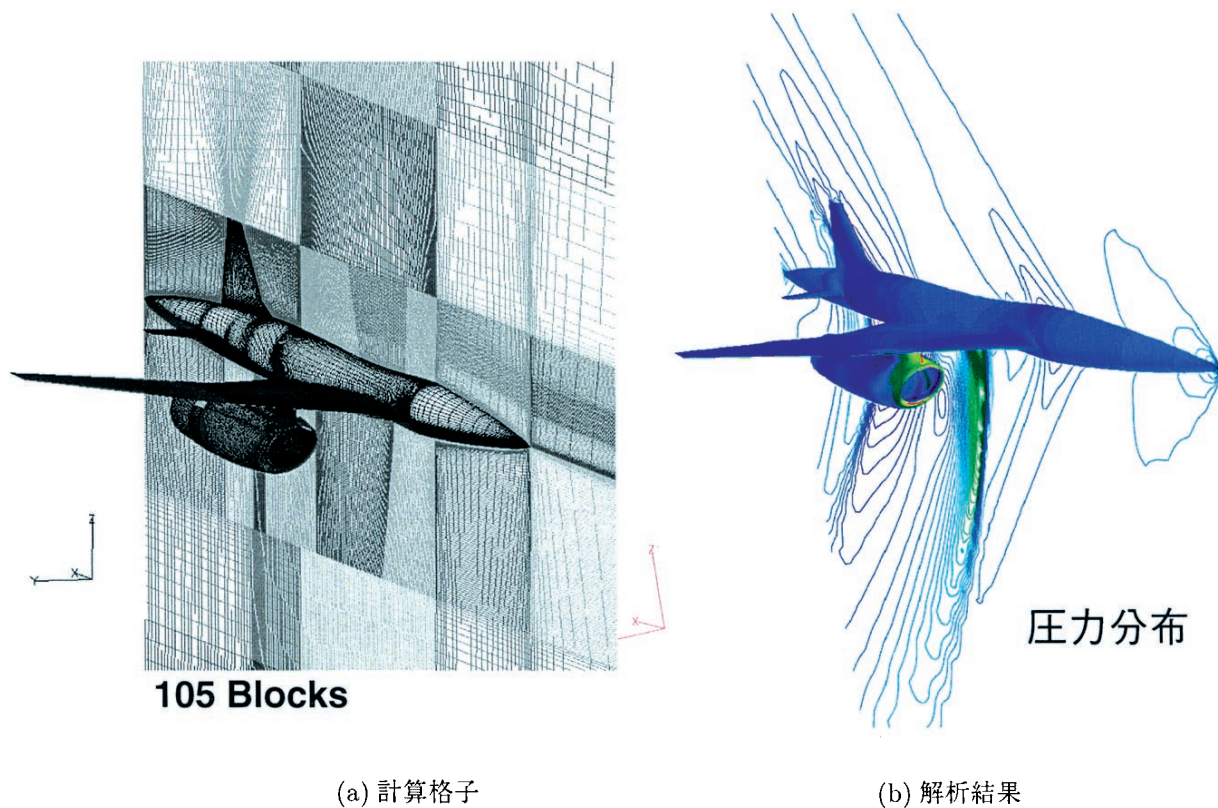


図 5: UPACS による流体解析実行中の画面



(a) 計算格子

(b) 解析結果

図 6: UPACS による数値解析例 (SST 実験機)

## 5 UPACS の公開とユーザ会

「はじめに」で述べたように、UPACS は共通的な基盤コードを核として、系統だった CFD コードの検証やノウハウの蓄積を効率的に実現できるようにすることを目標の一つとしている。そのために、航技研内部のみならず大学や民間企業などで広く利用してもらい、さらに役に立つ CFD ツールとして UPACS を進化させていきたいと考えている。

そこで UPACS を普及させるためにユーザ会を発足させた。現段階では学術的研究目的に限っているが、ユーザ会に入会することにより UPACS のソースプログラムを無償で利用することができる。

なお、UPACS の最新情報やユーザ会についての問い合わせ先は、「航技研インターネットホームページ」<http://www.nal.go.jp/>上で提供しているので参照していただきたい。

## 参考文献

- [1] 山本一臣、他、「並列計算プラットフォーム UPACS について」、航空宇宙数値シミュレーション技術シンポジウム 1999 講演集、航空宇宙技術研究所特別資料 SP-44
- [2] “The Message Passing Interface (MPI) standard”, <http://www-unix.mcs.anl.gov/mpi/>
- [3] Enomoto, S., “A Structured Grid Method in Simulating Flow around Supersonic Transports”, Proceedings of 2nd SST-CFD Workshop, 2000