1996

# A finite-volume Navier-Stokes solver for multiblock structured meshes.

Alexandre Yuri. Boglaev
*University of Windsor*

# A Finite - Volume Navier - Stokes Solver for Multiblock Structured Meshes

by

A.Yu. Boglaev

A Thesis
Submitted to the Faculty of Graduate Studies and Research
Through the Department of Mathematics and Statistics
in Partial Fulfillment
of the Requirements for the Degree of
Master of Science
at the University of Windsor

Windsor, Ontario, Canada
1996

ISBN 0-612-10979-8

Name ALEXANDER YURI BOGDANOV

*Dissertation Abstracts International* and *Masters Abstracts International* are arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation or thesis. Enter the corresponding four-digit code in the spaces provided.

MECHANICAL ENGINEERING
SUBJECT TERM

0548
SUBJECT CODE

UMI

Subject Categories

## THE HUMANITIES AND SOCIAL SCIENCES

### COMMUNICATIONS AND THE ARTS
Architecture ..... 0729
Art History ..... 0377
Cinema ..... 0900
Dance ..... 0378
Fine Arts ..... 0357
Information Science ..... 0723
Journalism ..... 0391
Library Science ..... 0399
Mass Communications ..... 0708
Music ..... 0413
Speech Communication ..... 0459
Theater ..... 0465

### EDUCATION
General ..... 0515
Administration ..... 0514
Adult and Continuing ..... 0516
Agricultural ..... 0517
Art ..... 0273
Bilingual and Multicultural ..... 0282
Business ..... 0688
Community College ..... 0275
Curriculum and Instruction ..... 0727
Early Childhood ..... 0518
Elementary ..... 0524
Finance ..... 0277
Guidance and Counseling ..... 0519
Health ..... 0680
Higher ..... 0745
History of ..... 0520
Home Economics ..... 0278
Industrial ..... 0521
Language and Literature ..... 0279
Mathematics ..... 0280
Music ..... 0522
Philosophy of ..... 0998
Physical ..... 0523

Psychology ..... 0525
Reading ..... 0535
Religious ..... 0527
Sciences ..... 0714
Secondary ..... 0533
Social Sciences ..... 0534
Sociology of ..... 0340
Special ..... 0529
Teacher Training ..... 0530
Technology ..... 0710
Tests and Measurements ..... 0288
Vocational ..... 0747

### LANGUAGE, LITERATURE AND LINGUISTICS
Language
- General ..... 0679
- Ancient ..... 0289
- Linguistics ..... 0290
- Modern ..... 0291

Literature
- General ..... 0401
- Classical ..... 0294
- Comparative ..... 0295
- Medieval ..... 0297
- Modern ..... 0298
- African ..... 0316
- American ..... 0591
- Asian ..... 0305
- Canadian (English) ..... 0352
- Canadian (French) ..... 0355
- English ..... 0593
- Germanic ..... 0311
- Latin American ..... 0312
- Middle Eastern ..... 0315
- Romance ..... 0313
- Slavic and East European ..... 0314

### PHILOSOPHY, RELIGION AND THEOLOGY
Philosophy ..... 0422
Religion
- General ..... 0318
- Biblical Studies ..... 0321
- Clergy ..... 0319
- History of ..... 0320
- Philosophy of ..... 0322

Theology ..... 0469

### SOCIAL SCIENCES
American Studies ..... 0323
Anthropology
- Archaeology ..... 0324
- Cultural ..... 0326
- Physical ..... 0327

Business Administration
- General ..... 0310
- Accounting ..... 0272
- Banking ..... 0770
- Management ..... 0454
- Marketing ..... 0338

Canadian Studies ..... 0385
Economics
- General ..... 0501
- Agricultural ..... 0503
- Commerce-Business ..... 0505
- Finance ..... 0508
- History ..... 0509
- Labor ..... 0510
- Theory ..... 0511

Folklore ..... 0358
Geography ..... 0366
Gerontology ..... 0351
History
- General ..... 0578
- Ancient ..... 0579
- Medieval ..... 0581
- Modern ..... 0582
- Black ..... 0328
- African ..... 0331
- Asia, Australia and Oceania ..... 0332
- Canadian ..... 0334
- European ..... 0335
- Latin American ..... 0336
- Middle Eastern ..... 0333
- United States ..... 0337

History of Science ..... 0585
Law ..... 0398
Political Science
- General ..... 0615
- International Law and Relations ..... 0616
- Public Administration ..... 0617

Recreation ..... 0814
Social Work ..... 0452
Sociology
- General ..... 0626
- Criminology and Penology ..... 0627
- Demography ..... 0938
- Ethnic and Racial Studies ..... 0631
- Individual and Family Studies ..... 0628
- Industrial and Labor Relations ..... 0629
- Public and Social Welfare ..... 0630
- Social Structure and Development ..... 0700
- Theory and Methods ..... 0344

Transportation ..... 0709
Urban and Regional Planning ..... 0999
Women's Studies ..... 0453

## THE SCIENCES AND ENGINEERING

### BIOLOGICAL SCIENCES
Agriculture
- General ..... 0473
- Agronomy ..... 0285
- Animal Culture and Nutrition ..... 0475
- Animal Pathology ..... 0476
- Food Science and Technology ..... 0359
- Forestry and Wildlife ..... 0478
- Plant Culture ..... 0479
- Plant Pathology ..... 0480
- Plant Physiology ..... 0817
- Range Management ..... 0777
- Wood Technology ..... 0746

Biology
- General ..... 0306
- Anatomy ..... 0287
- Biostatistics ..... 0308
- Botany ..... 0309
- Cell ..... 0379
- Ecology ..... 0329
- Entomology ..... 0353
- Genetics ..... 0369
- Limnology ..... 0793
- Microbiology ..... 0410
- Molecular ..... 0307
- Neuroscience ..... 0317
- Oceanography ..... 0416
- Physiology ..... 0433
- Radiation ..... 0821
- Veterinary Science ..... 0778
- Zoology ..... 0472

Biophysics
- General ..... 0786
- Medical ..... 0760

### EARTH SCIENCES
Biogeochemistry ..... 0425
Geochemistry ..... 0996

Geodesy ..... 0370
Geology ..... 0372
Geophysics ..... 0373
Hydrology ..... 0388
Mineralogy ..... 0411
Paleobotany ..... 0345
Paleoecology ..... 0426
Paleontology ..... 0418
Paleozoology ..... 0985
Palynology ..... 0427
Physical Geography ..... 0368
Physical Oceanography ..... 0415

### HEALTH AND ENVIRONMENTAL SCIENCES
Environmental Sciences ..... 0768
Health Sciences
- General ..... 0566
- Audiology ..... 0300
- Chemotherapy ..... 0992
- Dentistry ..... 0567
- Education ..... 0350
- Hospital Management ..... 0769
- Human Development ..... 0758
- Immunology ..... 0982
- Medicine and Surgery ..... 0564
- Mental Health ..... 0347
- Nursing ..... 0569
- Nutrition ..... 0570
- Obstetrics and Gynecology ..... 0380
- Occupational Health and Therapy ..... 0354
- Ophthalmology ..... 0381
- Pathology ..... 0571
- Pharmacology ..... 0419
- Pharmacy ..... 0572
- Physical Therapy ..... 0382
- Public Health ..... 0573
- Radiology ..... 0574
- Recreation ..... 0575
- Speech Pathology ..... 0460
- Toxicology ..... 0383

Home Economics ..... 0386

### PHYSICAL SCIENCES
Pure Sciences
Chemistry
- General ..... 0485
- Agricultural ..... 0749
- Analytical ..... 0486
- Biochemistry ..... 0487
- Inorganic ..... 0488
- Nuclear ..... 0738
- Organic ..... 0490
- Pharmaceutical ..... 0491
- Physical ..... 0494
- Polymer ..... 0495
- Radiation ..... 0754

Mathematics ..... 0405
Physics
- General ..... 0605
- Acoustics ..... 0986
- Astronomy and Astrophysics ..... 0606
- Atmospheric Science ..... 0608
- Atomic ..... 0748
- Electronics and Electricity ..... 0607
- Elementary Particles and High Energy ..... 0798
- Fluid and Plasma ..... 0759
- Molecular ..... 0609
- Nuclear ..... 0610
- Optics ..... 0752
- Radiation ..... 0756
- Solid State ..... 0611

Statistics ..... 0463

Applied Sciences
Applied Mechanics ..... 0346
Computer Science ..... 0984

Engineering
- General ..... 0537
- Aerospace ..... 0538
- Agricultural ..... 0539
- Automotive ..... 0540
- Biomedical ..... 0541
- Chemical ..... 0542
- Civil ..... 0543
- Electronics and Electrical ..... 0544
- Heat and Thermodynamics ..... 0348
- Hydraulic ..... 0545
- Industrial ..... 0546
- Marine ..... 0547
- Materials Science ..... 0794
- Mechanical ..... 0548
- Metallurgy ..... 0743
- Mining ..... 0551
- Nuclear ..... 0552
- Packaging ..... 0549
- Petroleum ..... 0765
- Sanitary and Municipal ..... 0554
- System Science ..... 0790

Geotechnology ..... 0428
Operations Research ..... 0796
Plastics Technology ..... 0795
Textile Technology ..... 0994

### PSYCHOLOGY
General ..... 0621
Behavioral ..... 0384
Clinical ..... 0622
Developmental ..... 0620
Experimental ..... 0623
Industrial ..... 0624
Personality ..... 0625
Physiological ..... 0989
Psychobiology ..... 0349
Psychometrics ..... 0632
Social ..... 0451

# A Finite - Volume Navier - Stokes Solver

# for Multiblock Structured Meshes

by

A.Yu. Boglaev

**Abstract**

In this thesis numerical solution of 2D steady laminar incompressible viscous Navier-Stokes equations has been considered. For such flows a problem occurs with preserving mass flow through the system. Primitive variables were chosen to perform computations. The solver is based on the finite-volume approach with artificial compressibility. The code was written to accomplish numerical computations based on the suggested approach. This code is capable of handling multiblock meshes and does not require coordinate transformations, due to the finite-volume approach. The artificial compressibility approach allows the calculation of pressure and at the same time preserves mass flow through the system at the steady-state. This code was validated against known results for the driven cavity problem and rapidly expanding channel problem. The problem of a moving road vehicle was studied for different mesh arrangements to investigate the influence of boundary conditions together with mesh quality on the computational results. The results of these calculations were also compared to those obtained by STARCD and found to be in reasonably good agreement.

# ACKNOWLEDGEMENTS

The author thanks Dr. R. M. Barron for his encouragement and support. The author also thanks the Natural Sciences and Engineering Research Council of Canada for the Industrial Postgraduate Scholarship (IPGS) and Engineering Technology Associates (ETA) of Madison Heights, MI for their participation in the NSERC Industrial PGS program and their financial support, as well as Department of Mathematics and Statistics of the University of Windsor for support through Graduate Assistantships during the period of study.

# Contents

# Nomenclature

$x, y$: Cartesian coordinates

$t$: time

$U, V$: velocity components

$P$: pressure

$\rho$: density

$\nu$: molecular viscosity

$Re$: Reynolds number

$\beta$: artificial compressibility parameter/artificial speed of sound

$A$: cell area

$\epsilon$: pressure switch parameter

$\delta$: convergence parameter

$R$: function to determine convergence

$N_i$, $N_j$ : number of nodes in i-direction, j-direction

$N_b$: number of blocks

$I_{max}$, $J_{max}$: maximum number of nodes in i, j directions

$i, j$: index of cell in i, j directions

# Chapter 1

# Introduction

In practice, engineers must deal with "complex geometry" problems in fluid mechanics. These geometrical complexities have encouraged the use of finite element methods [1] since they can more easily handle problems with complicated irregular geometry. On the other hand, finite difference methods work more efficiently on rectangular computational domains. However, these methods lose their attractiveness because of the necessity to make coordinate transformations, which sometimes can be computationally expensive. The finite-volume approach [2-6] combines the best features of these two different approaches. It has the geometrical flexibility of finite element methods and the efficiency of finite difference methods. It also inherits such characteristic features of finite element methods as the very important preprocessing part, namely mesh generation. A realistic and accurate numerical solution of a problem depends upon choice and correct implementation of boundary conditions as well as a right design of the mesh. The choice of a mesh plays a main role in getting a detailed numerical solution. A uniformly fine mesh represents an easy solution but cannot be employed because of its high computational cost. A solution to the problem is to use

a clustered multiblock grid which leaves the option open as to where to make the mesh finer or coarser [7].

The primary objectives of this study are:

1. to develop a multiblock finite-volume Navier-Stokes solver, and

2. to investigate the dependence of the numerical solution of the Navier-Stokes equations on different computational meshes.

In this thesis, the 2D Navier-Stokes equations are solved using the finite-volume method with artificial compressibility. The code is written which is capable of handling multiblock meshes. The different meshes are generated using an elliptic mesh builder. The physical problem of interest is chosen to be an airflow around a car. The airflow is laminar, viscous and incompressible. Solutions of the Navier-Stokes equations on H - type and H - O - type meshes are investigated.

A finite-volume 2D incompressible viscous non - turbulent flow code has been written based on the artificial compressibility approach. This approach was first suggested by Chorin [8], and validated against known numerical results for a set of problems. The discussion of the choice of "pseudo - Mach number" for the artificial compressibility approach can be found in [9]. The code in this thesis was validated against results obtained with finite - volumes codes [10, 11], finite - elements codes [10, 12] and finite difference codes [10, 13], using streamfunction - vorticity formulations or primitive variables formulations.

Research works and technical reports related to flow calculation of airflow around a road vehicle have lead to a better understanding of the most common problems encountered while obtaining solution for the airflow [7, 14-19].

Computing time restriction, absence of a turbulence model and the differencing schemes implemented in the code made it practically impossible to complete computations for high Reynolds numbers with the present code. However, computations with high Reynolds numbers were carried out with the commercial code STARCD. Nevertheless, computations with Reynolds numbers near the critical level are quite interesting due to the more complex flow behavior with numerous recirculation regions.

# Chapter 2

# Formulation of Equations

## 2.1 Theoretical Background

This thesis deals with the numerical solution of the 2D Navier-Stokes equations. The system of partial differential equations governing laminar unsteady flow of an incompressible viscous fluid, in primitive variables and conservative form, is [6]

$$\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = 0 \tag{2.1}$$

$$\frac{\partial U}{\partial t} + \frac{\partial (U^2)}{\partial x} + \frac{\partial (UV)}{\partial y} = -\frac{1}{\rho}\frac{\partial P}{\partial x} + \nu\left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right) \tag{2.2}$$

$$\frac{\partial V}{\partial t} + \frac{\partial (UV)}{\partial x} + \frac{\partial (V^2)}{\partial y} = -\frac{1}{\rho}\frac{\partial P}{\partial y} + \nu\left(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2}\right) \tag{2.3}$$

In order to solve this system of partial differential equations, an appropriate set of boundary conditions (B.C.) for velocity and pressure must be prescribed for the domain of the system. However, as often happens, boundary conditions may be unavailable for some part of the boundary. In this case some approximations should be made, or the physical domain may be changed in such a way as to allow the use of some known boundary conditions. This is possible only if we can get correct results

for the original problem after such modifications. Typical boundary conditions which can be applied are:

1. Inlet.

2. Outlet.

3. No-slip.

4. Constant pressure.

5. Periodic.

6. Symmetry.

The meanings and some explanation of these boundary conditions are as follows:

**Inlet:** the inflow conditions are imposed directly, including velocity components and density (for compressible flows), but not the pressure.

**Outlet:** this type of boundary treatment is applied at locations where the flow is everywhere directed outwards, but the conditions are otherwise unknown (they are mainly determined by what is happening upstream). The outlet conditions are estimated in two stages. First, the distributions of the variables on the outlet plane are evaluated by extrapolation from upstream, on the assumption of zero gradient along the mesh lines intersecting the outlet plane. Then, the velocities are adjusted to conserve the mass flow.

**No-slip:** the velocity components are set to zero on the boundary. If the mesh system is fine enough to resolve any boundary layer, pressure can be determined on the boundary from the expression for pressure using the boundary - layer equation

$$U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} = -\frac{1}{\rho}\frac{dP}{dx} + \nu\frac{\partial^2 U}{\partial y^2} \tag{2.4}$$

On the wall $U = 0$, $V = 0$ Hence, the following expression for the pressure gradient (for flow over a flat plate with nonzero pressure gradient) is obtained

$$\frac{dP}{dx} = \mu \frac{\partial^2 U}{\partial y^2} \tag{2.5}$$

Generalizing the above formula (2.5), one may write

$$\frac{dP}{ds} = \mu \frac{\partial^2 U}{\partial n^2} \tag{2.6}$$

where $s$ is a boundary-fitting coordinate and $\frac{\partial}{\partial n}$ means a derivative in the outward normal direction to the boundary curve. This relation is valid provided that the thickness of the boundary layer in the region where equation (2.6) is applied is greater than the curvature of the surface in the region.

**Constant pressure:** or prescribed pressure assumes prescribed pressure distribution on the boundary. The corresponding flow magnitude and direction are then determined as part of the analysis.

**Periodic:** or cyclic boundaries consist of pairs of geometrically identical boundaries at which the flow repeats itself. This can be exploited to reduce the size of the computational domain.

**Symmetry:** boundary condition implies normal gradients of all variables to the boundaries are zero.

## 2.2 Artificial Compressibility Approach

The steady Navier-Stokes equations are deduced from (2.1), (2.2) and (2.3) and are written as

$$\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = 0 \tag{2.7}$$

$$\frac{\partial(U^2)}{\partial x} + \frac{\partial(UV)}{\partial y} = -\frac{1}{\rho}\frac{\partial P}{\partial x} + \nu\left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right) \tag{2.8}$$

$$\frac{\partial(UV)}{\partial x} + \frac{\partial(V^2)}{\partial y} = -\frac{1}{\rho}\frac{\partial P}{\partial y} + \nu\left(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2}\right) \tag{2.9}$$

The main numerical difficulty in obtaining a solution to these equations lies in satisfying the mass conservation equation, which serves as a constraint, i.e.

$$\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = 0 \tag{2.10}$$

This difficulty can be overcome by using the "artificial compressibility method" suggested by Chorin [8].

The main idea of Chorin's method is to consider the solution of the steady-state equations (2.7)-(2.9) of fluid motion as the limit $t \to \infty$ for unsteady equations (2.1)-(2.3), modified by adding an additional time dependent term to the mass conservation equation (2.1):

$$\frac{1}{\beta^2}\frac{\partial P}{\partial t} + \frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = 0 \tag{2.11}$$

where $\beta^2$ is an arbitrary parameter. The term "artificial compressibility" was adopted because equations (2.11), (2.2) and (2.3) can be obtained from the compressible Navier-Stokes equations with state law

$$P = \beta^2 \rho$$

with $\beta^2 = constant$. This perturbed set of equations can be solved by standard numerical methods. Equation (2.11) has physical meaning only when a steady state is reached, *i.e.* $\frac{\partial P}{\partial t} = 0$, which implies $\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = 0$. The constant $\beta$ is called artificial speed of sound and chosen in such a way to obtain fast convergence. The artificial speed of sound should be restricted to assure that artificial Mach number $M < 1$ [8]. There are upper and lower bounds for $\beta$ which can be derived from stability

conditions. Since the magnitude of $\beta$ controls the speed of the pressure wave, it plays a very important role in determining convergence speed, accuracy and stability. Most of the problems numerically solved here are similar to duct flow. The lower bound on $\beta$ for these kinds of flows can be written as [8]

$$\beta \gg \left(1 + \frac{4}{Re}\left(\frac{x_{ref}}{x_\delta}\right)^2 \left(\frac{x_L}{x_{ref}}\right)\right)^2 - 1 \tag{2.12}$$

$$M = \frac{1}{\beta} \max_D |u| < 1 \tag{2.13}$$

where $x_\delta$ is half the distance between the two walls of the duct, $x_L$ is equal to the total length of the duct, $M$ is Mach number and $|u|$ is velocity magnitude taken at some point of the flow domain. The artificial speed of sound affects the stability condition which, for the case of duct flow and velocities prescribed on boundaries, can be written as [8]

$$\Delta t \leq \frac{2}{n^{1/2}(1 + 5^{1/2})} \frac{1}{\beta} \min_i \Delta x_i \tag{2.14}$$

where $n$ is the number of space dimensions and $\Delta x_i$ is the space step in $i$-direction (if the derivatives of the velocities are prescribed at the boundary, one has to ensure no instabilities arise due to boundary effects). During numerical experiments it has become clear that the artificial Mach number should not be too small, otherwise it causes instability or very slow convergence of the numerical scheme. This can be explained as trying to solve an incompressible flow problem, as a low Mach number compressible flow.

## 2.3 Finite - Volume Approximation

The conservation laws of fluid mechanics may be expressed in either differential or integral form. When a numerical scheme is applied to the differential form, the com-

putational domain is represented by a set of discrete points, upon which the finite difference equations are solved. In the case of integral equations, the computational domain is divided into volumes (or areas in the case of two dimensions) and the conservation laws are applied to these volumes. Finite difference equations which approximate partial differential equations are solved within a rectangular domain at equally spaced discrete points. Almost all practical problems have a quite complicated physical domain which is highly irregular in shape, requiring the implementation of a coordinate transformation. At the same time, the finite-volume method does not require such transformation. The governing equations can be solved if the physical domain can be successfully discretized into elements. This discretization can be unstructured or structured.

Finite-volume schemes can be divided into two groups: "cell-centered" schemes and "nodal point" schemes. The scheme used in this thesis is "cell-centered". To illustrate the "cell-centered" finite-volume approach consider the following model equation:

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} = 0 \tag{2.15}$$

First equation (2.15) is integrated over a quadrilateral element *abcd* as shown in Figure 2.1, giving

$$\int_{abcd} \left( \frac{\partial Q}{\partial t} \right) dxdy = - \int_{abcd} \left( \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} \right) dxdy \tag{2.16}$$

Then, Green's Theorem is applied to the right-hand side of equation (2.16), which leads to

$$\int_{abcd}\left(\frac{\partial Q}{\partial t}\right)dxdy = -\oint_{abcd}(Edy - Fdx) \tag{2.17}$$

Equation (2.17) is used to develop a cell-centered scheme. The integrals of equation (2.17) are to be approximated over the element *abcd*, i.e. cell 5, shown in Figure 2.1. The dependent variable $Q$ is to be solved at cell 5. Equation (2.17) is approximated as

$$\begin{aligned}\left(\frac{Q_5^{n+1} - Q_5^n}{\Delta t}\right)A_{abcd} = &-[E_i\Delta y_{ab} + E_j\Delta y_{bc} + E_m\Delta y_{cd} + E_n\Delta y_{da}] \\ &+[F_i\Delta x_{ab} + F_j\Delta x_{bc} + F_m\Delta x_{cd} + F_n\Delta x_{da}]\end{aligned} \tag{2.18}$$

where $A_{abcd}$ is the area of the cell *abcd* and points $i, j, m, n$ are midpoints of edges $ab, bc, cd, da$ respectively. Points $1, 2, 3, 4$ are cell centers of four adjacent cells. The increments of $x$ and $y$ (*e.g.* $\Delta x_{ab}$) are determined by following the cell border in a counter-clockwise direction. The values of functions $E$ and $F$ at the midpoints of the edges are determined by the following formulas:

$$\begin{aligned} E_i &= \frac{1}{2}(E_5 + E_1), \quad E_j = \tfrac{1}{2}(E_5 + E_2), \\ E_m &= \frac{1}{2}(E_5 + E_3), \quad E_n = \tfrac{1}{2}(E_5 + E_4) \\ F_i &= \frac{1}{2}(F_5 + F_1), \quad F_j = \tfrac{1}{2}(F_5 + F_2), \\ F_m &= \frac{1}{2}(F_5 + F_3), \quad F_n = \tfrac{1}{2}(F_5 + F_4) \end{aligned} \tag{2.19}$$

The values of functions $E$ and $F$ in the centers of cells can be evaluated at $n+1$ or $n$ time level, which determines whether an implicit or explicit scheme is used. It should be noted that the values of the functions at the midpoints of the edges can be determined in a variety of different ways other than by formulas (2.19).

# Chapter 3

# Mesh Generator

## 3.1 Basic Concepts

In order to numerically solve the governing partial differential equations (2.11), (2.2) and (2.3), approximations of the partial derivatives are required. Using these approximations, the differential equations are converted to a system of algebraic equations. These equations are subsequently solved at desired points within the domain. In practice, physical domains usually have nonrectangular, complex shape. A transformation from physical space to computational space can be introduced to generate a structured mesh covering the flow region. On the other hand, the irregular physical domain can also be covered by an unstructured mesh, usually generated by Delaunay's triangulation method or the advancing front scheme [20]. The finite-volume method does not require a rectangular computational domain and computation can be performed on any grid (structured or unstructured) in the physical domain. This grid system should satisfy certain requirements, such as:

1. It should not have adjacent cells with a large difference in aspect ratio - this can cause difficulties in solving the system of algebraic equations.

2. Orthogonality or near-orthogonality of the grid lines, especially near boundaries.

3. Grid lines of the same family should not cross each other.

4. The grid should be clustered in regions of high flow gradients.

Figure 3.1 shows a mapping of the physical domain into the computational one, where $x$ and $y$ are Cartesian coordinates in the physical domain, while $\phi$ and $\psi$ are coordinates in the computational domain. Solving a system of differential equations is the most popular approach to generate structured grid systems. A system of elliptic partial differential equations is usually solved, because of the nature of elliptic equations to smooth boundary data [21].

An analogy can be drawn from two dimensional, steady, inviscid, incompressible flow which is governed by Laplace equations for the velocity potential $\phi$ and the stream function $\psi$:

$$\phi_{xx} + \phi_{yy} = 0 \tag{3.1}$$

$$\psi_{xx} + \psi_{yy} = 0$$

If $\phi$ and $\psi$ are considered as coordinates in a computational plane, then solution of these equations produces smooth and orthogonal mesh lines.

Another advantage of using elliptic grid generation is that, when applied with care, it almost always produces a one-to-one transformation between the computational and physical domain, i.e. grid lines of the same family do not cross each other. Local extreme values may occur inside the domain if "very bad" boundary conditions

are possible which may lead to intersection of grid lines of the same family. This problem can be resolved by dividing the domain into several blocks and solving elliptic equations for each individual block.

## 3.2 Outline of Elliptic Grid Generation

A 2D physical domain must be mapped to a rectangular computational domain. This can be expressed mathematically as the following transformation

$$x_i = x_i(\xi^1, \xi^2), \;\; i = 1, 2. \tag{3.2}$$

Suppose that the coordinate points are specified on the entire boundary curves of the physical domain. Then elliptic grid generation may be performed. Under the assumption of finite and nonzero Jacobian, the inverse transformation exists, i.e.,

$$\xi^i = \xi^i(x_1, x_2), \;\; i = 1, 2. \tag{3.3}$$

In order to transform Cartesian coordinates to curvilinear, Thompson et al [22] used Poisson equations

$$\nabla \xi^i = g^{ii} P_i, \;\; i = 1, 2 \tag{3.4}$$

where $\nabla = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2}$ is the Laplacian, $g^{ij}$ is the contravariant metric tensor of the transformation and $P_i$ are control functions. Inversely transforming equation (3.4), one can get a system of differential equations for the Cartesian coordinates as unknown functions of the curvilinear coordinates (using notations: $\xi^1 = \xi$, $\xi^2 = \eta$, $x_1 = x$ and $x_2 = y$):

$$a x_{\xi\xi} - 2b x_{\xi\eta} + c x_{\eta\eta} = P_1 \tag{3.5}$$

$$a y_{\xi\xi} - 2b y_{\xi\eta} + c y_{\eta\eta} = P_2 \tag{3.6}$$

where the metrics are given by

$$a = x_\xi^2 + y_\xi^2$$

$$b = x_\xi x_\eta + y_\xi y_\eta$$

$$c = x_\eta^2 + y_\eta^2$$

The system of nonlinear elliptic equations (3.5) and (3.6) is solved in the rectangular domain in $(\xi, \eta)$ space in order to provide the grid point locations $(x, y)$ in the physical space. This task can be accomplished effectively using any iteration process such as point successive-over-relaxation (PSOR) or line successive-over-relaxation (LSOR). These equations are nonlinear and therefore a linearization procedure must be performed. For example coefficients a, b, and c can be evaluated at the previous iteration level, so that they just become constants on the current iteration level.

Control functions play a very important role in the system of elliptic equations. These control functions are used to attract grid points and/or lines to required boundaries, to cluster grid points and to exert some effect on angle of intersection of grid lines. Appendix A.1 contains a detailed explanation on evaluating the control functions, as suggested by Barron [23].

## 3.3 GUI and Mesh Generation Code

In this work, a 2D multiblock elliptic grid generator developed by Barron [23, 24] has been used to create the necessary meshing of the physical domain for all problems considered. The 2D profile of a real car has been digitized, in order to get geometry data necessary for defining the physical domain and generating the mesh.

After the car geometry has been established, the physical domain must be subdivided into blocks. Points on the boundaries of the blocks should be distributed in such a way to provide necessary clustering of grid points. This task can be accomplished by using a curve-fitting program which was available. The user is required to supply some points along a curve in order to generate new points according to specification of the type of curve fit (eg. spline) and the required clustering. Several options are available for clustering including packing at one or both endpoints of the curve, at the centre, or at some internal point. Spacing between points can be decreased by a prescribed factor if clustering is required near the first or the last point. If clustering has to be done near both end points then either *arcsinh* or *ln* functions are used.

Generating the mesh inside the domain was accomplished by a 2D multiblock version of the program EGG3D referred to as "MBEGG2D" [24]. The user must specify necessary parameters to select the method to use for mesh generation (algebraic/elliptic), number of blocks used to cover the whole domain, choice of the control functions, format of the output file and some constants required for mesh clustering if this option is used.

For convenience, a user-friendly graphical user interface (GUI) was developed to run with MBEGG2D. Figures 3.2 and 3.3 show some snapshots of some of the pop-up menus from the GUI.

# Chapter 4

# Finite - Volume Code Description

## 4.1 Introduction

The code presented in this work is an explicit Navier-Stokes equations solver for multiblock meshes. The grid system in each of the blocks has a rectangular topology, i.e. each block can be transformed to a rectangular computational domain consisting of $Ni \times Nj$ cells, where $N_i$ and $N_j$ are the number of cells in the $\xi$ and $\eta$ directions, respectively. Two adjacent blocks must have the same number of cells along a common boundary and the cells must be joined face to face with coincident vertices. For future reference, $Nb$ denotes the total number of blocks covering a physical domain.

## 4.2 Discretization of the Navier-Stokes Equations

First, discretization of the Navier-Stokes equations (with artificial compressibility) must be done. Integrating equation (2.11) over the cell *abcd* (Figure 2.1) yields

$$\frac{1}{\beta^2}\int_{abcd}\left(\frac{\partial P}{\partial t}\right)dxdy+\int_{abcd}\left(\frac{\partial U}{\partial x}+\frac{\partial V}{\partial y}\right)dxdy=0 \tag{4.1}$$

Applying Green's Theorem, the following equation is obtained

$$\frac{1}{\beta^2}\int_{abcd}\left(\frac{\partial P}{\partial t}\right)dxdy+\oint_{abcd}(Udy-Vdx)=0 \tag{4.2}$$

and, after discretization, the resulting equation can be rewritten as

$$\frac{1}{\beta^2}\frac{P^{n+1}-P^n}{\Delta t}A_{abcd}+ \tag{4.3}$$

$$(U_{ab}(y_b-y_a)+U_{bc}(y_c-y_b)+U_{cd}(y_d-y_c)+U_{da}(y_a-y_d)-$$

$$V_{ab}(x_b-x_a)-V_{bc}(x_c-x_b)-V_{cd}(x_d-x_c)-V_{da}(x_a-x_d))=0$$

where $U_{ab}, U_{bc}, U_{cd}, U_{da}, V_{ab}, V_{bc}, V_{cd}$ and $V_{da}$ are average values along edges $ab$, $bc$, $cd$ and $da$, respectively. Furthermore, any variable with subscript denoting an edge will refer to an average value of the variable along the edge. Now, for the momentum equations (2.8) and (2.9), taking the integral over the same cell $abcd$ one can write

$$\int_{abcd}\left(\frac{\partial U}{\partial t}\right)dxdy+\int_{abcd}\left(\frac{\partial(U^2)}{\partial x}+\frac{\partial(UV)}{\partial y}\right)dxdy= \int_{abcd}\left(-\frac{1}{\rho}\frac{\partial P}{\partial x}+\nu\left(\frac{\partial^2 U}{\partial x^2}+\frac{\partial^2 U}{\partial y^2}\right)\right)dxdy \tag{4.4}$$

$$\int_{abcd}\left(\frac{\partial V}{\partial t}\right)dxdy+\int_{abcd}\left(\frac{\partial(UV)}{\partial x}+\frac{\partial(V^2)}{\partial y}\right)dxdy= \int_{abcd}\left(-\frac{1}{\rho}\frac{\partial P}{\partial y}+\nu\left(\frac{\partial^2 V}{\partial x^2}+\frac{\partial^2 V}{\partial y^2}\right)\right)dxdy \tag{4.5}$$

Applying Green's Theorem, the above equations become

$$\int_{abcd}\left(\frac{\partial U}{\partial t}\right)dxdy+\oint_{abcd}\left(U^2dy-UVdx\right)= -\frac{1}{\rho}\oint_{abcd}Pdy+\oint_{abcd}\nu\left(\frac{\partial U}{\partial x}dy-\frac{\partial U}{\partial y}dx\right) \tag{4.6}$$

$$\int_{abcd}\left(\frac{\partial V}{\partial t}\right)dxdy+\oint_{abcd}\left(UVdy-V^2dx\right)=$$
$$\frac{1}{\rho}\oint_{abcd}Pdx+\oint_{abcd}\nu\left(\frac{\partial V}{\partial x}dy-\frac{\partial V}{\partial y}dx\right) \tag{4.7}$$

Discretization of the two momentum equations over the cell $abcd$ leads to the following system of equations

$$\frac{U^{n+1}-U^n}{\Delta t}A_{abcd}+$$
$$(U_{ab}^2(y_b-y_a)+U_{bc}^2(y_c-y_b)+U_{cd}^2(y_d-y_c)+U_{da}^2(y_a-y_d)-$$
$$(UV)_{ab}(x_b-x_a)-(UV)_{bc}(x_c-x_b)-(UV)_{cd}(x_d-x_c)-(UV)_{da}(x_a-x_d))=$$
$$-\frac{1}{\rho}(P_{ab}(y_b-y_a)+P_{bc}(y_c-y_b)+P_{cd}(y_d-y_c)+P_{da}(y_a-y_d))+$$
$$\nu\left(\left.\frac{\partial U^*}{\partial x}\right|_{ab}(y_b-y_a)+\left.\frac{\partial U^*}{\partial x}\right|_{bc}(y_c-y_b)+\left.\frac{\partial U^*}{\partial x}\right|_{cd}(y_d-y_c)+\left.\frac{\partial U^*}{\partial x}\right|_{da}(y_a-y_d)-\right.$$
$$\left.\left.\frac{\partial U^*}{\partial y}\right|_{ab}(x_b-x_a)-\left.\frac{\partial U^*}{\partial y}\right|_{bc}(x_c-x_b)-\left.\frac{\partial U^*}{\partial y}\right|_{cd}(x_d-x_c)-\left.\frac{\partial U^*}{\partial y}\right|_{da}(x_a-x_d)\right) \tag{4.8}$$

$$\frac{V^{n+1}-V^n}{\Delta t}A_{abcd}+$$
$$((U)V_{ab}(y_b-y_a)+(UV)_{bc}(y_c-y_b)+(UV)_{cd}(y_d-y_c)+(UV)_{da}(y_a-y_d)-$$
$$V_{ab}^2(x_b-x_a)-V_{bc}^2(x_c-x_b)-V_{cd}^2(x_d-x_c)-V_{da}^2(x_a-x_d))=$$
$$\frac{1}{\rho}(P_{ab}(x_b-x_a)+P_{bc}(x_c-x_b)+P_{cd}(x_d-x_c)+P_{da}(x_a-x_d))+$$
$$\nu\left(\left.\frac{\partial V^*}{\partial x}\right|_{ab}(y_b-y_a)+\left.\frac{\partial V^*}{\partial x}\right|_{bc}(y_c-y_b)+\left.\frac{\partial V^*}{\partial x}\right|_{cd}(y_d-y_c)+\left.\frac{\partial V^*}{\partial x}\right|_{da}(y_a-y_d)-\right.$$
$$\left.\left.\frac{\partial V^*}{\partial y}\right|_{ab}(x_b-x_a)-\left.\frac{\partial V^*}{\partial y}\right|_{bc}(x_c-x_b)-\left.\frac{\partial V^*}{\partial y}\right|_{cd}(x_d-x_c)-\left.\frac{\partial V^*}{\partial y}\right|_{da}(x_a-x_d)\right) \tag{4.9}$$

The iterative scheme used in the present code is explicit. The nonlinear terms are evaluated using the most currently available values of U, V, P as indicated in section#4.3. Average values in the above equations are evaluated by computing a value at the midpoint of an edge using formulas (2.19). Partial derivatives in equations (4.8) and (4.9) are taken to be average values of the derivatives over corresponding edges. These average values are approximated as the average value of the derivative in the cell $1b5a$ (Figure 2.1). For example, the equations for computing the average values of the derivatives along edge $ab$ are

$$\begin{aligned}
\left.\frac{\partial U}{\partial x}\right|_{ab} &= \frac{1}{A_{abcd}} \int_{1b5a} \left(\frac{\partial U}{\partial x}\right) dxdy \\
\left.\frac{\partial U}{\partial y}\right|_{ab} &= \frac{1}{A_{abcd}} \int_{1b5a} \left(\frac{\partial U}{\partial y}\right) dxdy \\
\left.\frac{\partial V}{\partial x}\right|_{ab} &= \frac{1}{A_{abcd}} \int_{1b5a} \left(\frac{\partial V}{\partial x}\right) dxdy \\
\left.\frac{\partial V}{\partial y}\right|_{ab} &= \frac{1}{A_{abcd}} \int_{1b5a} \left(\frac{\partial V}{\partial y}\right) dxdy
\end{aligned} \tag{4.10}$$

Applying Green's Theorem, the above equations can be replaced by

$$\begin{aligned}
\left.\frac{\partial U}{\partial x}\right|_{ab} &= \frac{1}{A_{abcd}} \oint_{1b5a} U dy \\
\left.\frac{\partial U}{\partial y}\right|_{ab} &= -\frac{1}{A_{abcd}} \oint_{1b5a} U dx \\
\left.\frac{\partial V}{\partial x}\right|_{ab} &= \frac{1}{A_{abcd}} \oint_{1b5a} V dy \\
\left.\frac{\partial V}{\partial y}\right|_{ab} &= -\frac{1}{A_{abcd}} \oint_{1b5a} V dx
\end{aligned} \tag{4.11}$$

Evaluation of the partial derivatives can be completed using the above equations (4.11) as follows

$$\begin{aligned}
\left.\frac{\partial U}{\partial x}\right|_{ab} &= \frac{1}{A_{abcd}}(U_{1b}(y_b - y_1) + U_{b5}(y_5 - y_b) + U_{5a}(y_a - y_5) + U_{a1}(y_1 - y_a) \\
\left.\frac{\partial U}{\partial y}\right|_{ab} &= -\frac{1}{A_{abcd}}(U_{1b}(x_b - x_1) + U_{b5}(x_5 - x_b) + U_{5a}(x_a - x_5) + U_{a1}(x_1 - x_a)
\end{aligned}$$

$$\left.\frac{\partial V}{\partial x}\right|_{ab} = \frac{1}{A_{abcd}}(V_{1b}(y_b - y_1) + V_{b5}(y_5 - y_b) + V_{5a}(y_a - y_5) + V_{a1}(y_1 - y_a)$$

$$\left.\frac{\partial V}{\partial y}\right|_{ab} = -\frac{1}{A_{abcd}}(V_{1b}(x_b - x_1) + V_{b5}(x_5 - x_b) + V_{5a}(x_a - x_5) + V_{a1}(x_1 - x_a)$$

The rest of the partial derivatives in equations (4.8) and (4.9) can be discretized in the same way using control cells $b2c5$, $5c3d$ and $a5d4$. It is easy to show that the finite-volume method for this specific discretization leads to a central finite-difference scheme if applied to a uniform rectangular mesh. The term "average" is used several times above, the detailed procedure of finding the "average" value is given in Appendix A.2.

## 4.3 Iterative Procedure

The core of the code consists of an explicit finite-volume Navier-Stokes equations solver. An initial guess is set for pressure and velocity field (zero) at all nodes and cell centers. Pressure and velocity on boundaries are set according to boundary specifications. On each iteration, the code determines a new pressure distribution based on the velocity field from the previous iteration and then uses these values to find a new velocity field. In general, the procedure for one iteration can be described by the relations

$$P^{n+1} = F(P^n, U^n, V^n)$$

$$U^{n+1} = G(P^{n+1}, U^n, V^n)$$

$$V^{n+1} = H(P^{n+1}, U^{n+1}, V^n)$$

where superscript $n$ denotes values of variables on a previous iteration while $n + 1$ denotes values to be computed on the current iteration and $F$, $G$ and $H$ are functions

obtained from (4.3), (4.8) and (4.9), determining dependence of new values on the values from the previous iteration. Each iteration is completed for each of the $Nb$ blocks. On an interface between two blocks an additional cell layer from the adjacent neighbouring block is assigned to carry the computations in the boundary cells. In this case smooth data transfer between the two blocks occurs. The code does not solve for pressure and velocity field simultaneously but rather for pressure in all blocks and then for velocity field. This procedure is illustrated with Figure 4.1, where boundary cells marked with • are interface cells between two adjacent blocks, used to transfer data between the blocks.

The numerical solution of the mass conservation equation could become unstable and can significantly slow down the convergence process or even cause divergence. In order to prevent divergence and accelerate convergence, an extra dissipative term is needed. Thus equation (4.2) is replaced by [5]

$$\frac{1}{\beta^2}\int_{abcd}\left(\frac{\partial P}{\partial t}\right)dxdy + \oint_{abcd}(Udy - Vdx) - D(P_{abcd}) = 0 \tag{4.12}$$

where the artificial dissipation term $D(P_{abcd})$ is given by

$$D(P_{abcd}) = \sum_{k=1}^{4} d_k^{abcd}, \tag{4.13}$$

$$d_k^{abcd} = \frac{\epsilon}{2}\left(\frac{A_{abcd}}{\Delta t} + \frac{A_k}{\Delta t}\right)\left(\nabla^2 P_{abcd} - \nabla^2 P_k\right)$$

In the equation (4.13) $A_{abcd}$ is area of the cell $abcd$ (Figure 2.1), $A_k, k = 1,2,3,4$ are the areas of the adjacent cells and $\nabla^2$ is the Laplacian operator on a two dimensional quadrilateral mesh.

The calculation of the pressure switch $\epsilon$ is described in details in [5].

## 4.4 Convergence Criteria

The convergence criteria to be satisfied can be written as

$$\left|R_{UV}^{n+1} - R_{UV}^{n}\right| < \delta_{UV} \tag{4.14}$$

$$\left|R_{P}^{n+1} - R_{P}^{n}\right| < \delta_{P} \tag{4.15}$$

where $\delta_{UV}$ and $\delta_P$ are some small positive constants, $n$ is the iteration number and the functions $R_{UV}^{n}$ and $R_{UV}^{n}$ are

$$R_{UV}^{n} = \frac{1}{\Delta t}\sum_{i=1}^{Ni}\sum_{j=1}^{Nj}\left(U_{i,j}^{2} + V_{i,j}^{2}\right)^{n} \tag{4.16}$$

$$R_{P}^{n} = \frac{1}{\Delta t}\frac{1}{\beta^2}\sum_{i=1}^{Ni}\sum_{j=1}^{Nj}\left(P_{i,j}^{2}\right)^{n} \tag{4.17}$$

where $n$ as a superscript in the right-hand side of equations (4.16) and (4.17) denotes that all values under the summation signs are to be taken at $n$-th iteration level. It should be noticed that the above convergence criteria is written for one block. In the case of a multiblock mesh, a summation over all blocks has to be done using (4.14) and (4.15) for each of the blocks:

$$\sum_{i=1}^{Nb}\left|R_{UV\,i}^{n+1} - R_{UV\,i}^{n}\right| < \delta_{UV} \tag{4.18}$$

$$\sum_{i=1}^{Nb}\left|R_{P\,i}^{n+1} - R_{P\,i}^{n}\right| < \delta_{P} \tag{4.19}$$

where subscript $i$ denotes number of a block. Special attention should be paid to the pressure change divided by $\beta^2$. This difference must be small enough to deliver conservation of mass throughout the whole physical domain. The method error and approximation error are also important factors. Estimation of these two errors are given in the Appendix A.3.

# Chapter 5

# Validation of the Code

## 5.1 Driven Cavity Problem

The first test problem chosen to validate the code described in Chapter 4 is the laminar incompressible flow in a square cavity with moving top wall. The main reason for choosing this problem as a test problem is that there is a large volume of work available describing results of testing different numerical techniques on this problem. The problem configuration is represented by Figure 5.1.

Boundary conditions for velocity are taken to be no-slip on the vertical walls and on the bottom and

$$U(x) = -16x^2(1-x)^2\,,\ V = 0 \tag{5.1}$$

on the top of the cavity.

Boundary condition (5.1) is selected so that $U(0) = U(1) = U'(0) = U'(1) = 0$, which eliminates singularities in the two upper corners.

Boundary conditions for pressure derived from the momentum equations are

$$Along\ AB\ and\ CD\ lines: \quad \frac{\partial P}{\partial y} = \frac{1}{Re}\frac{\partial^2 V}{\partial y^2}$$

$$Along\ AD\ and\ BC\ lines: \quad \frac{\partial P}{\partial x} = \frac{1}{Re}\frac{\partial^2 U}{\partial x^2}$$

where $Re = \frac{UL}{\nu}$ is Reynolds number. Calculation for Reynolds number 400 shows good agreement with results achieved by [25].

Figure 5.2 shows the velocity field for Re=400 on a uniform 40x40 grid system.

Figure 5.3 presents the pressure distribution in the cavity. It is easy to see non-smooth pressure contours in this picture, which is due to the fact that no smoothing operators are involved in the solution procedure and by the low order of approximation for the mass conservation equation (see Appendix A.3). Despite the fact that the code features a smoothing procedure for pressure, in this calculation it has been switched off due to the high computational cost ivolved.

Although the calculation on a 40x40 mesh for Reynolds number 400 is not accurate enough (due to the coarse grid) to pick up boundary layer effects, the calculation still predicts a secondary recirculation in the left lower corner of the cavity, which can be seen by zooming into the corner, Figure 5.4.

Velocity profiles along the centerlines of the cavity are illustrated in Figures 5.5 and 5.6, showing good agreement with the reference literature [25]. Data for the y-direction velocity profile along the horizontal centerline was unavailable so comparison is made only for the x-component of velocity. It is worth mentioning that results obtained by the code presented here match the best results obtained from a fourth-order Hermitian scheme. These graphs validate the code presented in the thesis against results obtained by different codes written for streamfunction - vorticity formulation of the problem, presented in [25].

## 5.2 Rapidly Expanding Channel

The second test problem is chosen to be a rapidly expanding channel problem. This problem has been used as a test problem for numerous numerical techniques and a large data archive is available for different configurations of the problem [10, 12, 13]. The problem configuration is represented by Figure 5.7, with the following boundary conditions

$$\begin{aligned}
&\textit{Along } AB \textit{ line}: && U = 0,\ V = 0,\ \tfrac{\partial P}{\partial n} = \tfrac{1}{Re}\tfrac{\partial^2 U}{\partial n^2} \\
&\textit{Along } BC \textit{ line}: && \tfrac{\partial^2 P}{\partial x^2} = 0,\ \tfrac{\partial U}{\partial x} = 0,\ \tfrac{\partial V}{\partial x} = 0 \\
&\textit{Along } CD \textit{ line}: && \tfrac{\partial P}{\partial y} = 0,\ \tfrac{\partial U}{\partial y} = 0,\ V = 0 \\
&\textit{Along } DA \textit{ line}: && \tfrac{\partial^2 P}{\partial x^2} = 0,\ U = U(y),\ V = 0
\end{aligned}$$

where $U(y)$ is chosen to be

$$U(y) = 1.5(2y - y^2) \tag{5.2}$$

to avoid singularity in velocity at the point $A$. The coefficient in the above equation is chosen to be 1.5 to get a bulk velocity through the inlet to be equal 1. The curved boundary AB is described by the following equation [12]

$$y(x) = \frac{1}{2}tanh\left(2 - \frac{30x}{Re}\right) - \frac{1}{2}tanh(2)$$
$$and \quad 0 \le x \le \frac{Re}{3}.$$

The nomenclature $A_{i,j}$ below refers to a value of some variable $A$ at the grid point with coordinates $(i, j)$, where number $i$ corresponds to $x$ coordinate and number $j$ corresponds to $y$ coordinate, $0 \le i \le Imax$, $0 \le j \le Jmax$. Finite-difference discretization of the Neumann boundary condition for pressure along boundary $AB$ is done according to the following formula

$$\frac{P_{i,0} - P_{i,1}}{\sqrt{(X_{i,0} - X_{i,1})^2 + (Y_{i,0} - Y_{i,1})^2}} = \frac{1}{Re}\frac{2U_{i,0} - 5U_{i,1} + 4U_{i,2} - U_{i,3}}{(X_{i,0} - X_{i,1})^2 + (Y_{i,0} - Y_{i,1})^2} \tag{5.3}$$

for all $i$, where $j = 0$ corresponds to the line $AB$. The left-hand side of equation (5.3) corresponds to $\frac{\partial P}{\partial n}$ and is true if outgoing mesh lines are perpendicular to the boundary. The right-hand side of the equation is the second order discretization of $\frac{\partial^2 U}{\partial n^2}$. In the case when one does not have a mesh which is orthogonal to a boundary, the same approach used for discretization of $\frac{\partial P}{\partial n} = 0$ can be used, as described in Appendix A.4, letting the right-hand side be equal to $\frac{1}{Re}\frac{\partial^2 U}{\partial n^2}$ rather than zero. Finite-difference discretization of the Neumann boundary conditions for velocity components and the boundary condition for pressure ($\frac{\partial^2 P}{\partial x^2} = 0$) along boundary $BC$ are

$$2P_{Imax,j} - 5P_{Imax-1,j} + 4P_{Imax-2,j} - P_{Imax-3,j} = 0$$

$$U_{Imax,j} - U_{Imax-1,j} = 0$$

$$V_{Imax,j} - V_{Imax-1,j} = 0$$

for all $j$, where $Imax$ corresponds to the $i$ value on the outlet boundary $BC$. These approximations are second order accurate for pressure and first order accurate for velocity. For the symmetry boundary condition, which holds on the line $CD$, we have

$$P_{i,Jmax} - P_{i,Jmax-1} = 0$$

for all $i$, where $Jmax$ corresponds to the line $CD$. For $U_{i,Jmax}$ a parabolic extrapolation is used (Appendix A.5). For the inlet boundary condition, which is applied on the line $DA$, we have

$$2P_{0,j} - 5P_{1,j} + 4P_{2,j} - P_{3,j} = 0$$

for all $j$ in the range $0 \leq j \leq Jmax$.

The program was run for different Reynolds numbers in single and multiblock modes. In both cases the computational mesh is chosen to be packed near the wall

and clustered near the point $x = 1$, to pick up the separation effects which occur near this point. The 80x50 mesh system for the single block run is shown in Figure 5.8. The velocity field and pressure distribution for Reynolds number 10 are given by Figures 5.9 and 5.10 The most important parameter which is used for validating numerical codes on this particular test problem is the flow reattachment position. Comparison of the reattachment coordinate computed by the finite-volume method described in this thesis with that obtained by [12] can be accomplished by comparing values of vorticity along the wall. The comparison is illustrated by Figure 5.11. The present results are in good agreement with those obtained by Cliffe et al, (see [12]), except at the beginning of the channel. The finer 100x50 mesh allows better resolution of the recirculation region, Figure 5.12. Pressure distribution along the wall is represented by Figure 5.13 for the case of 100x50 mesh and results are compared to those obtained by other methods [10]. Two results from [10] were chosen for comparison, one was considered a benchmark result (finite-element approach by Cliffe et al) and the other result belongs to Gosman (finite-volume approach) [12]. Oscillations in pressure are caused by the singularity in the left lower corner of the domain. This singularity also affects vorticity. Nevertheless these mismatches are situated and confined in the region of no interest and they do not propagate along the channel.

The two test problems numerically solved validate the code suggested in this thesis. It is worth mentioning that results for multiblock runs are identically to those for one block when that one block is just split into several. These runs were completed to verify correct implementation of the multiblock code.

# Chapter 6

# Airflow Around a Road Vehicle

## 6.1 Introduction

The primary aerodynamics consideration in the modern design of road vehicles is drag reduction. A history of research on car aerodynamics and drag reduction has been presented in [26-28]. In the early stages, it was believed that good aerodynamic shapes for an automobile would be similar to shapes used in airplane design. Only after models were tested in winds tunnels did it become clear that ground effects were very important, and that the aerodynamic behaviour of road vehicles is quite different from that of space vehicles.

Early studies on the aerodynamic shape of road vehicles in the 1930's were driven by the desire to maximize speed and ensure stability in crosswinds. However, after the worldwide energy crisis in the 1970's, automobile companies began to seriously look at the issue of drag reduction as a means of improving fuel economy.

The aerodynamic performance of the automobile is largely determined by the flow over the rear portion of the vehicle. Hence, the vortex generated behind the vehicle

must be accurately resolved. Other regions which strongly influence the vehicle's drag are the engine compartment, mirrors and the wheel wells. The airflow from the front grill, through the radiator, around the engine and its interaction with the external flow must also be examined carefully.

## 6.2 Problem Description and Boundary Conditions

The problem to be solved using the Navier-Stokes equations solver described in Chapter 4 is the airflow around a car moving with constant speed.

The problem configuration is represented by Figure 6.1. An extremely important role in correct modeling of the problem is played by the boundary conditions on the physical domain. Among possible boundary conditions described in Chapter 2 only those that represent realistic physical phenomenon for the given configuration should be chosen. The boundary conditions are set relative to a coordinate system fixed in the vehicle. On the left side of the physical domain the inlet boundary condition is imposed, which implies constant velocity. No-slip condition is imposed on the car's surface and on the ground. It should be noticed that the lower horizontal side of the domain ("Ground", see Figure 6.1) is assigned to move with the same velocity as the inlet relative to the car. Because the upper horizontal side of the domain is taken far enough from the car, we may apply free stream boundary condition on the upper boundary. This means that any disturbance caused by the moving car does not affect the air far enough from the car in the vertical direction. This statement does not hold for the right boundary of the domain and thus, the outlet boundary condition is

applied there. In this particular case this is implemented as zero gradients of variables along horizontal grid lines far downstream of the vehicle. Now, for pressure boundary conditions, constant pressure is imposed on the upper horizontal boundary of the domain. This is taken as reference pressure ($P_{up} = 0$) because pressure is determined only to within an additive quantity. On the solid walls (i.e vehicle surface and ground) pressure can be calculated from the Neumann-type boundary condition for pressure in the boundary layer (see section 2.1) [20]. On the inlet the same constant pressure as on the upper boundary is taken. In the code itself, inlet velocity and velocity of the ground is taken to be equal 1, and Reynolds number is adjusted in such a way to deliver similarity with the physical conditions described above.

In general, outlet boundary condition for pressure is more difficult to realistically model, because the outlet must be far enough to impose constant pressure or zero gradients along the horizontal grid lines. For the present configuration, the right vertical boundary is chosen sufficiently far away that, the conditions of constant pressure gradient and zero gradient for velocity components were applied.

To summarize, boundary conditions are imposed as follows: for all block sides which correspond to a segment of the left vertical boundary of the physical domain, the inlet boundary condition is applied with constant velocity and zero second derivative of pressure in the horizontal direction. On all block sides which correspond to the car's body, no-slip boundary condition is applied. On all block sides which represent the ground, constant velocity in the horizontal direction is applied (equal to the inlet velocity) and zero normal gradient for pressure is imposed. For all block sides which represent the right vertical boundary of the physical domain the outlet boundary condition is applied. In two cases the upper boundary of the physical domain is taken

far enough away to impose the free stream boundary condition, and in the other case it is taken to be at the distance of one car length from the ground.

## 6.3 Mesh Arrangements

### 6.3.1 Mesh #1

The first mesh arrangement for the physical domain described in the previous section consists of 8 blocks. The complete mesh system is represented by Figure 6.2, and a zoomed view of the mesh near the car is shown by Figure 6.3.

The first block is 40x40 cells packed near the bottom and right boundary to match the mesh arrangement of other adjacent blocks. The second block is 40x20 cells packed near the right boundary to deliver a fine mesh near the car. The third block has 40x20 cells packed near the right boundary for a smooth transition to the fourth block. The fourth block consists of 40x20 uniform cells. The fifth block is 40x20 cells packed near the left boundary. The sixth block is 40x30 cells packed near the tail of the car. The seventh block has 40x40 cells packed near the bottom and left boundary for a smooth transition to the sixth and eighth blocks respectively. The eighth block is 40x40 cells packed near the bottom to get a fine mesh near the car and on the left and right boundary for a smooth transition into the first and seventh blocks.

### 6.3.2 Mesh #2

The second mesh arrangement has the same topology as mesh # 1, but this mesh is tight and uniform in each block (see Figure 6.4 and Figure 6.5). The sixth block is

split into two blocks horizontally. The first block has 90x40 cells. The second block has 90x10 cells. The third block has 90x10 cells. The fourth block has 40x10 cells. The fifth block has 80x10 cells. The sixth block has 80x10 cells. The seventh block has 80x10 cells. The eighth block has 80x40 cells. The ninth block has 40x40 cells.

### 6.3.3 Mesh #3

The third mesh arrangement consists of 4 blocks as shown by Figure 6.6 and in zoomed view near the car by Figure 6.7. The first block is 60x5 cells packed near the right boundary to deliver smooth transition into adjacent block number 2. The second block is uniformly meshed with 40x5 cells. The third block has 60x5 cells packed near the left boundary for a smooth transition to the second block. The fourth block consists of 80x60 cells packed in such a way to match boundaries of the other blocks and to deliver a fine mesh clustering near the car.

## 6.4 Results

### 6.4.1 Results for mesh #1

The first set of results for this mesh arrangement were obtained for Reynolds number 450, based on car length. The most interesting computational result for the problem of a moving road vehicle is the prediction of the separation region behind the car. The velocity vector field behind the car is represented by Figure 6.8. A zoomed view of the same region and with enlarged vector length is shown in Figure 6.9. Velocity magnitude plot for the whole computational domain is shown in Figure 6.10, the zoomed view near the car is illustrated in Figure 6.11. Pressure distribution along the

top and the bottom of the car for this and the other mesh arrangements can be found in Figures 6.12 to 6.15, where they are compared to results obtained by STARCD [29]. The non-smoothness of these plots can be explained by the facts that only first order accuracy approximation was used to define pressure and that adjacent cells on either side of the boundaries have different aspect ratios. Computational errors for pressure propagate along walls. This was also observed in both benchmark problems for driven cavity and rapidly expanding channel. In these cases, the error tends to disappear and does not propagate far downstream. However, for the problem of a moving road vehicle, these errors appear on the surface of the car and cause inaccuracies in the surface pressure profiles.

The second set of results was obtained for the case of Reynolds number 45. Velocity magnitude plot in the zoomed region near the car is shown by Figure 6.16. By comparing to Figures 6.11 and 6.18 it is easy to see the difference between these cases. For example the region of highest velocity magnitude appears much higher above the car than in the case with higher Reynolds number.

For this mesh arrangement, computations for Reynolds number 1800 were also performed. The velocity field behind the car and velocity magnitude contours are presented in Figures 6.17 and 6.18. A longer recirculation region is observed behind the car for Re of 1800.

### 6.4.2 Results for mesh #2

For the second mesh configuration, computations were accomplished for Reynolds number 45 and 450. The mesh system is represented by Figure 6.4 and a close look at the region near the car by Figure 6.5. The main difference between this

mesh configuration, and mesh #1 is that in this case the upper boundary is much closer to the car. The calculations on this mesh illustrate the importance of the far field location for implementation of boundary conditions, since significant changes are observed in the computed flow. Figure 6.19 shows velocity field behind the car for Re=450, and in zoomed view by Figure 6.20. Velocity field behind the car for Reynolds number 45 matches the one from mesh #3 arrangement. Velocity magnitude contours are presented in Figures 6.21 and 6.22 for Re=450 and 45 respectively.

### 6.4.3 Results for mesh #3

For the third mesh configuration, computations were accomplished for Reynolds number 45 and 450. The mesh system is illustrated by Figure 6.6 and zoomed view by 6.7. It is important to note that the rear window has been modified and is much steeper than for meshes 1 and 2. The velocity field behind the car is shown in Figure 6.23 for Reynolds number 45. It is easy to observe that no separation was resolved behind the rear end of the car. There is a small separation behind the rear window (see Figure 6.23). Velocity magnitude is shown at Figure 6.24. Maximum velocity in this case is higher than for the first mesh arrangement and the region of high velocity magnitude has a different shape. This difference can be explained by the different shape of the outer boundary, which confirms that the upper boundary should be moved far enough away from the car to avoid influence of the upper boundary shape on the flow field near the car.

Velocity field for Re=450 is shown in Figure 6.25 and velocity magnitude contours in Figure 6.26.

## 6.5 Comparison with STARCD Results

STARCD calculations were performed on mesh #1 for Reynolds numbers 45 and 450.

Velocity field behind the car for Re=45 is illustrated by Figure 6.27. There is not much difference in the velocity fields between STARCD's calculations and calculations from the code developed in this thesis (see Figure 6.23) for this low Reynolds number. The results indicate that it is very unlikely that a recirculation region develops behind the car and the back window. Figure 6.28 presents the velocity magnitude contours. The maximum velocity from STARCD's calculation is slightly higher than those described above. Results from the first mesh (Figure 6.16) seem close to STARCD's results near the car, but the region of maximum velocity magnitude above the car is smaller. The second mesh (Figure 6.22) also gives good agreement with STARCD near the car, but above the car the profiles are longer than for the first mesh. One can observe that the region with high velocity magnitude is closer to the car than in all other cases. This can be explained by the closeness of the upper boundary. The results from the third mesh (Figure 6.22) are closest to STARCD.

Velocity field behind the car for Re=450 is illustrated by Figure 6.29. Two long recirculation regions are noticeable and a separation zone appears behind the rear window. The recirculation region for mesh #1 is shorter than that obtained by STARCD (Figures 6.8 and 6.9). The second mesh gives longer recirculation region, but not as long as STARCD (Figures 6.19 and 6.20). A longer recirculation region was obtained for the third mesh. The upper vortex behind the vehicle for this calculation is much higher than in other cases, and has upper boundary as high as the height of the car (Figure 6.25). Also, separation behind the rear window was detected.

Velocity magnitude profiles obtained by STARCD are illustrated by Figure 6.30. The first mesh arrangement gives higher velocity profiles which are situated closer to the car and thinner than those obtained by STARCD. The second mesh arrangement is different, and one can observe that contours in the top portion of the channel go down after they pass the car (Figure 6.21), which does not occur in all other cases. In the third case, maximum velocity occurs near the tip of the car.

It should be noticed that, for all Reynolds numbers, maximum velocity obtained by STARCD remains almost the same, while in the other calculations it increases with Reynolds number.

Pressure distribution along the top and bottom of the car for Reynolds number 45 is shown in Figures 6.12 and 6.13. One can observe qualitatively the same behaviour for the pressure on the top of the car for meshes #1 and #3 and STARCD. The second mesh gives a much different picture due to the location of the upper boundary in the mesh setup. Oscillations occur near the front and rear end of the car and, for the third mesh, on the edge of the rear window and roof of the car. Overall, the third mesh gives the best matching with STARCD results. For the pressure on the bottom of the car the total picture remains the same: the first mesh gives reasonably good agreement, while the second one falls farthest from STARCD. The third mesh delivers the best agreement with STARCD again.

The pressure distribution along the top and bottom of the car for Reynolds number 450 is shown in Figures 6.14 and 6.15. Pressure in all cases for the top of the car qualitatively behaves similar except for the first portion of the plot (hood region) for the third mesh. For the pressure on the bottom of the car, the third mesh gives results closest to those obtained by STARCD, except right at the front end and tail

of the car.

# Chapter 7

# Concluding Remarks

## 7.1 Conclusions and Recommendations

It has been shown that the code based on the suggested approach is capable of solving incompressible, viscous, steady, laminar flows for low Reynolds numbers. Results obtained for the model problems with the code are in a good agreement with the results obtained by other researchers. Results for an airflow around a moving road vehicle were obtained and compared to the results obtained by STARCD. These results seem to be in reasonably good agreement. Some limitations of the present code were found. Several multiblock mesh arrangements were investigated in order to build a better mesh. Below are some remarks on limitations of the code, suggested treatments to overcome these limitations and advice on mesh generation for the moving road vehicle problem.

In these calculations, first order approximation was used to determine pressure. It is more desirable to enhance the approximation to second order to increase the accuracy in determining the pressure.

This code is similar to central difference approximation schemes, and therefore is not very stable for high Reynolds numbers. It is possible to develop a finite-volume code with upwinding, which should help to reduce instabilities, but at the cost of increased numerical diffusion. It should be noticed that the present code is explicit and should be modified to become an implicit one in order to reduce run time.

In addition, to more accurately model the airflow over a vehicle, a turbulence model should be added for computations with large Reynolds numbers.

It is obvious that mesh generation plays an important role in successful numerical calculations. It affects the accuracy of results dramatically and can even cause divergence. It is also important to refine the mesh in the regions where such effects as recirculations and/or separations are expected, to successfully resolve them. The last mesh arrangements in the above calculations seems to give reasonably good results though a problem occurs with assigning boundary conditions on the outer boundary. The first mesh arrangement is the easiest to assign boundary conditions to, but this mesh is not of good quality near the front end of the car. Of course it is possible to create a very fine mesh, however in this thesis it was not possible due to lack of computing power for such a fine mesh. Also, it was noticed that cells with not too large aspect ratio give much better results. It is recommended to refine the mesh along the car surface to improve aspect ratio of boundary cells. Also, a better mesh could be a combination of the first mesh and third one, with the new mesh looking like the third from the left end of the domain up to the end of the car and after that it should look like the first mesh. In this case, implementation of boundary conditions will be easy and at the same time the mesh will be of better quality in front of the car.

## 7.2 Car Model with Engine Compartment

In order to build a more accurate model of a travelling road vehicle it is very important to take into consideration the engine compartment and the airflow under the hood. The difference in results of calculations with and without engine compartment is large enough that it becomes obvious that modeling airflow around a moving road vehicle should not be done without taking into account the engine compartment. To illustrate the importance of the engine compartment in the calculations [17, 30], the flow around an automobile (Figure 7.1) has been simulated using the commercial CFD package STARCD[29]. The airflow over a vehicle travelling at 20 m/sec (72km/hr) corresponding to a Reynolds number of $5.2\times10^6$ based on car length has been simulated. Far ahead and above the vehicle the flow is assumed to be undisturbed. Figures 7.2 and 7.3 show the velocity magnitude for both cases. Figure 7.4 shows pressure distribution. It is easy to observe the effects of the underhood flow. More detailed discussion of these results can be found in [30].

## 7.3 Further Notes on Modeling of Road Vehicle

The engine cooling airflow is also important as a means to diffuse the heat generated under the hood. The wheel wells, and the wheels themselves, contribute significantly to the remaining drag (50% for low-drag cars). There is a variety of literature available on design of lateral tapering near the rear end and the spats around the wheels [17, 18] and rear and front spoilers [19] to bring better properties for drag reduction. It appears that small regions in cars can be very important to the whole aerodynamic performance of cars. In recent years engineers have also directed lots of efforts trying

to enhance acoustic properties of cars, i.e. to reduce noise caused by airflow around side mirrors and the antenna [19].

# Chapter 8

# Appendices

## A.1 Control functions for mesh generator

A very important requirement for a multiblock mesh generator is to be able to control the angle of intersection of grid lines emanating from the two sides of the interface between blocks. For all problems solved in this thesis an elliptic mesh generator was used with choice of control functions suggested by Barron [23]. This method is described here since it does not yet appear in the literature. As was mentioned before, the 2-D elliptic grid generation system can be expressed in transformed $(\xi, \eta)$ space as

$$g_{22}\vec{r}_{\xi\xi} - 2g_{12}\vec{r}_{\xi\eta} + g_{11}\vec{r}_{\eta\eta} = -J^2(P\vec{r}_{\xi} + Q\vec{r}_{\eta}) \tag{8.1}$$

where $\vec{r} = (x, y)$, $J = x_\xi y_\eta - x_\eta y_\xi$ is the transformation Jacobian and the metrics are given by

$$g_{11} = \vec{r}_{\xi} \cdot \vec{r}_{\xi}, \quad g_{12} = \vec{r}_{\xi} \cdot \vec{r}_{\eta}, \quad g_{11} = \vec{r}_{\eta} \cdot \vec{r}_{\eta} \tag{8.2}$$

The control functions $P$ and $Q$ can be chosen to enforce orthogonality at the boundaries of the solution domain. In a multiblock grid generator, these "boundaries" could

be interfaces between adjacent blocks. For example, to find $P$ we may impose the orthogonality condition $\vec{r}_\xi \cdot \vec{r}_\eta = 0$ along a boundary curve $\eta = constant = \eta_1$. Taking the dot product of equation (8.1) with $\vec{r}_\xi$ we get

$$P(\xi,\eta)\mid_{\eta=\eta_1} = -\frac{\vec{r}_\xi \cdot \vec{r}_{\xi\xi}}{\mid \vec{r}_\xi \mid^2} - \left.\frac{\vec{r}_\xi \cdot \vec{r}_{\eta\eta}}{\mid \vec{r}_\eta \mid^2}\right|_{\eta\eta_1} \tag{8.3}$$

Equation (8.3) is true for $\eta = \eta_1$. Suppose also that along the boundary $\eta = \eta_1$ we have

$$\mid \vec{r}_\eta \mid^2 = f(\xi) \tag{8.4}$$

That is, we specify the spacing to the grid line $\eta = \eta_1 + \Delta\eta$. We then have the following system of two equations

$$\begin{aligned} x_\xi y_\eta + x_\eta y_\xi &= 0 \\ x_\eta^2 + y_\eta^2 &= f(\xi) \end{aligned} \tag{8.5}$$

The above system of two equations can be solved to find $x_\eta$ and $y_\eta$ as functions of $x_\xi, y_\xi$ and $f(\xi)$. Furthermore, differentiating the first equation in (8.5) with respect to $\eta$, one gets

$$\vec{r}_\xi \cdot \vec{r}_{\eta\eta} + \vec{r}_{\xi\eta} \cdot \vec{r}_\eta = 0 \tag{8.6}$$

which can be easily rearranged using equation (8.4) in the following way

$$\vec{r}_\xi \cdot \vec{r}_{\eta\eta} = -(\vec{r}_{\xi\eta} \cdot \vec{r}_\eta) = \frac{-1}{2}(\vec{r}_\eta \cdot \vec{r}_\eta)_\xi = \frac{-1}{2} f'(\xi) \tag{8.7}$$

which can now be substituted into equation (8.3). The first term in (8.3) can be evaluated from the given boundary data on $\eta = \eta_1$. In the mesh generator, function $f(\xi)$ is chosen to be a linear interpolation between it value at the two end points of $\eta = \eta_1$.

## A.2 Determining an average value

Suppose that we have a nodal mesh system in a domain. In this thesis, location of the cell center is determined by the intersection of two lines connecting opposite sides of the cell (Figure 8.1). Then, a value of a variable in the cell center is determined as one quarter of the sum of values of the variable at all four corners. In the case when one must get an average value at a node using cell values of surrounding cells, such a simple averaging as described above does not work in the general case. For example, it cannot be used when dealing with a highly irregular mesh system (Figure 8.2). In this case, averaging with weights is introduced. Figure 8.2 illustrates notations for the case, where $a, b, c, d$ correspond to the distances between different cell corners and the cell center. If $L$ is introduced as

$$\frac{1}{L} = \frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d} \tag{8.8}$$

then an average value of a variable at a node $(i, j)$ can be written as

$$\overline{F_{ij}} = \frac{L}{a} F^*_{i-1,j-1} + \frac{L}{b} F^*_{i,j-1} + \frac{L}{c} F^*_{ij} + \frac{L}{d} F^*_{i-1,j} \tag{8.9}$$

where $F$ denotes a variable of interest and the asterik corresponds to the values in the cells.

## A.3 Model and approximation errors

The mass conservation equation for compressible flow is given by

$$\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} = 0 \tag{8.10}$$

which can be rewritten as

$$\rho\frac{\partial u}{\partial x}+\rho\frac{\partial v}{\partial y}+u\frac{\partial \rho}{\partial x}+v\frac{\partial \rho}{\partial y}=0 \tag{8.11}$$

It is clear that by assuming incompressibility we introduce the following "error"

$$\varepsilon_m = u\frac{\partial \rho}{\partial x}+v\frac{\partial \rho}{\partial y}=0 \tag{8.12}$$

For the purpose of estimating this error, we consider it to be of the order of

$$\varepsilon_m = \max_D |u| \max_D \frac{\partial \rho}{\partial x} \tag{8.13}$$

Now we have to evaluate $\frac{\partial \rho}{\partial x}$. This can be accomplished by introducing a state equation, say

$$P = p(e,\rho) \tag{8.14}$$

In particular, consider an ideal gas with state equation

$$P = \rho RT \tag{8.15}$$

where R is the universal gas constant divided by molar mass of the media and T is temperature. After differentiation with respect to $x$, neglecting variations in temperature and rearranging, the above equation gives

$$\frac{\partial \rho}{\partial x} = \frac{\partial P}{\partial x}\frac{1}{RT} \tag{8.16}$$

Now one needs to extract $\frac{\partial P}{\partial x}$ from the numerical calculations to evaluate the error of the model ($R = 287\frac{J}{kgK}$(air), $T = 300K$).

Approximation errors can be estimated by exploiting the fact that the finite-volume method implemented here is equivalent to a central-difference scheme on a rectangular mesh where partial derivatives of second order are approximated with

second order accuracy. It is important to mention that the momentum equations are approximated with second order accuracy in space, while the mass conservation equation is approximated with first order accuracy.

## A.4 Discretization of $\frac{\partial P}{\partial n} = 0$

For large Reynolds numbers, the boundary condition for pressure on a wall becomes

$$\frac{\partial P}{\partial n} = 0 \tag{8.17}$$

which can be rewritten as a scalar product of two vectors, the pressure gradient and normal vector to the wall:

$$\vec{\nabla} P \cdot \vec{n} = 0 \tag{8.18}$$

Performing multiplication of these two vectors yields

$$\frac{\partial P}{\partial x} n_x + \frac{\partial P}{\partial y} n_y = 0 \tag{8.19}$$

Let us consider one boundary cell and refer to Figure 8.3 for the nomenclature used for the following derivations. We need to use the following formula for directional derivative

$$\frac{\partial P}{\partial s} = \frac{\partial P}{\partial x}\frac{\partial x}{\partial s} + \frac{\partial P}{\partial y}\frac{\partial y}{\partial s} \tag{8.20}$$

Then, for directions $\vec{s_1}$ ans $\vec{s_2}$, we have the following two discretized forms of equation (8.20):

$$\frac{P_D - P_A}{\Delta s_1} = \frac{\partial P}{\partial x}\frac{\Delta x_1}{\Delta s_1} + \frac{\partial P}{\partial y}\frac{\Delta y_1}{\Delta s_1} \tag{8.21}$$

$$\frac{P_C - P_A}{\Delta s_2} = \frac{\partial P}{\partial x}\frac{\Delta x_2}{\Delta s_2} + \frac{\partial P}{\partial y}\frac{\Delta y_2}{\Delta s_2} \tag{8.22}$$

where $\Delta x_1 = x_D - x_A$, $\Delta y_1 = y_D - y_A$, $\Delta x_2 = x_C - x_A$, $\Delta y_2 = y_C - y_A$, $\Delta s_1 = |\vec{s_1}|$ and $\Delta s_2 = |\vec{s_2}|$. Solving the above two equations together with equation (8.19), where $\frac{\partial P}{\partial x}$, $\frac{\partial P}{\partial y}$ and $P_A$ are treated as independent variables, the following expression for pressure on the wall ($P_A$) is obtained:

$$P_A = \frac{P_C(n_y\Delta x_1 - n_x\Delta y_1) - P_D(n_y\Delta x_2 - n_x\Delta y_2)}{(n_y\Delta x_1 - n_x\Delta y_1) - (n_y\Delta x_2 - n_x\Delta y_2)} \tag{8.23}$$

The following procedure is used to determine components of the normal vector $\vec{n}$. Normal vector at the point A (see Figure 8.3) is taken to be the normal vector to $BE$, where $\vec{BE} = \vec{BO} - \vec{EO}$. In other words, components of $\vec{n}$ are

$$n_x = -\frac{BE_y}{\sqrt{BE_x^2 + BE_y^2}}$$
$$n_y = \frac{BE_x}{\sqrt{BE_x^2 + BE_y^2}}$$

or, in terms of coordinates,

$$n_x = -\frac{y_B - y_E}{\sqrt{(x_B - x_E)^2 + (y_B - y_E)^2}}$$
$$n_y = \frac{x_B - x_E}{\sqrt{(x_B - x_E)^2 + (y_B - y_E)^2}}$$

## A.5 Parabolic extrapolation for x-component of velocity on symmetry boundary

Figure 8.4 represents one mesh line intersecting the symmetry boundary. Moving the origin to the point (0), it is easy to see that $U(y)$ should be in the form

$$U(y) = ay^2 + b \tag{8.24}$$

which yields $\frac{\partial U}{\partial y} = 0$ at $y = 0$. By solving the following system of equations

$$U_1 = a(s_1 + s_2)^2 + b$$

$$U_2 = as_2^2 + b$$

we get

$$a = \frac{U_1 - U_2}{2s_1s_2 + s_1^2} \tag{8.25}$$

$$b = \frac{U_2(s_1 + s_2)^2 - U_1s_2^2}{2s_1s_2 + s_1^2} \tag{8.26}$$

# Bibliography

[1] W.G. Habashi, M.F. Peeters, G. Guevremont and M.M. Hafez

*Finite Element Solutions of the Compressible Navier-Stokes Equations*

International Journal for Numerical Methods in Fluids, Vol. 7. pp. 17-27, 1987

[2] P.W. McDonald

*The Computation of Transonic Flow Through Two-Dimensional Gas Turbine Cascades*

ASME Paper 71-GT-89, 1971

[3] R.W. MacCormack and A.J. Paullay

*Computational Efficiency Achieved by Time Splitting of Finite Difference Operators*

AIAA Paper 72-154, San Diego, 1972

[4] A.W. Rizzi and M. Inouye

*Time Split Finite Volume Method for Three-Dimensional Blunt-Body Flows*

AIAA Journal, Vol. 11, pp. 1478-1485, 1973

[5] A. Jameson and D. Mavriplis

*Finite Volume Solution of the Two-Dimensional Euler Equations on a Regular*

*Triangular Mesh*

AIAA Journal, Vol. 24, No. 4. pp. 611-618, 1986

[6] C. Hirsch

*Numerical Computations of Internal and External Flows*

John Wiley & Sons, New York, Vol. 2, 1988

[7] W.S. Pien, A. Ecer, C. Bowes and P. Ward

*Application of a Block - Structured Grid Generation Scheme for Modeling Three-Dimensional Viscous Flows*

Num. Grid Generation in CFD, Proc. of the Int. Conf., J. Hauser, C. Taylor, eds., Landshut, W. Germany, pp. 247-258, July, 1986

[8] A.J. Chorin

*A Numerical Method for Solving Incompressible Viscous Flow Problems*

Journal of Computational Physics, Vol. 2 , pp. 12-26, 1967

[9] D. Choi and C. L. Merkle

*Application of Time-Iterative Schemes to Incompressible Flow*

AIAA Journal, Vol. 23, No. 10, pp. 1518-1524, 1985

[10] M. Napolitano and P. Orlandi

*Laminar Flow in a Complex Geometry: A Comparison*

International Journal for Numerical Methods in Fluids, Vol. 5, pp. 667-683, 1985

[11] M. Peric, R. Kessler and G. Scheuerer

*Comparison of Finite-Volume Numerical Methods with Staggered and Colocated*

*Grids*

Computers and Fluids, Vol. 16, No. 4, pp. 389-403, 1988

[12] K.A. Cliffe, C.P. Jackson and A.C. Greenfield

*Finite Element Solution for Flow in a Symmetric Channel with a Smooth Expansion*

U.K. Atomic Energy Authority, AERE-R 10608, 1982

[13] M. Napolitano and R.W. Walters

*An Incremental Block-Line-Gauss-Seidel Method for the Navier-Stokes Equations*

AIAA Journal, Vol. 24, No. 5, pp. 770-776, 1986

[14] R. Bessey

*3-D Simulation in Vehicle Aerodynamics by FIDAP*

European FIDAP Users Group Meeting, Paris, France 1990

[15] L. Larsson, L. Broberg and C. - E. Janson

*A Zonal Method for Predicting External Automobile Aerodynamics*

SAE Technical Paper Series, 910595, pp. 133-150, 1991

[16] J. Williams, W.J. Quinlan, J.E. Hackett, S.A. Thompson, T. Mariuccio and A. Robertson

*A Calibration Study of CFD for Automotive Shapes and CD*

SAE Technical Paper Series, 940323, pp. 1-20, 1994

[17] K. Yanagimoto, K. Nakagawa, H. China, T. Kimura, M. Yamamoto, T. Sumi and H. Iwamoto

*The Aerodynamic Development of a Small Specialty Car*

SAE Technical Paper Series, 940325, pp. 21-31, 1994

[18] N. Horinouchi, Y. Kato, S. Shinano, T. Kondoh and Y. Tagayashi

*Numerical Investigation of Vehicle Aerodynamics with Overlaid Grid System*

SAE Technical Paper Series, 950628, pp. 85-92, 1995

[19] K. Okumura and T. Kuriyama

*Practical Aerodynamic Simulations ($C_D$, $C_L$, $C_{YM}$) Using a Turbulence Model and 3rd-Order Upwind Scheme*

SAE Technical Paper Series, 950629, pp. 93-105, 1995

[20] K.A. Hoffmann and T. Chiang

*Computational Fluid Dynamics for Engineers*

EES, Wichita, Vol. 1, Section 8.6.1, pp. 308-311, 1993

[21] M.L. Spradling, S. Nakamura and K. Kuwahara

*Application of Elliptic Grid Generation Equations Blended with Hyperbolic Method to Three-Dimensional Grids for Vehicle Aerodynamic Analysis*

Num. Grid Generation in CFD and Related Fields, A.S.-Arcilla, J. Hauser, P.R. Eiseman, J.F. Thompson, eds., pp. 237-245, 1991

[22] J.F. Thompson, Z.U.A. Warsi and C.W. Mastin

*Numerical Grid Generation*

North-Holland, New York, 1985

[23] R.M. Barron

*Improvement to Grid Quality and Cost of Multiblock Structured Grid Generation*

Proceedings of CFD96, CFD Society of Canada, Ottawa, Ontario, June 1996

[24] R.M. Barron and C.-F. An

*EGG3D - A Software Package For 3D Grid Generation*

Unpublished Report, Fluid Dynamics Research Institute, University of Windsor, 1993

[25] R. Peyret and T. D. Taylor

*Computational Methods for Fluid Flows*

Springer-Verlag, New York, pp. 199-204, 1985

[26] S. Ito and R. Hoshino

*Relationship Between Car Styling and Aerodynamic Performance Report No. 1 - History of Research on Car Aerodynamics*

TOYOTA Technical Review, Vol. 41, No. 1, pp. 123-131, 1991

[27] S. Ito and R. Hoshino

*Relationship Between Car Styling and Aerodynamic Performance Report No. 2 - Technology to Reduce Aerodynamic Drag ($C_D$)*

TOYOTA Technical Review, Vol. 41, No. 2, pp. 115-125, 1992

[28] W.- H. Hucho and G. Sovran

*Aerodynamics of Road Vehicles*

Annual Rev. Fluid Mechanics, Vol. 25, pp. 485-537, 1993

[29] STARCD Manuals, Version 2.30, Computational Dynamics Limited, London

[30] R.M. Barron and A.Y. Boglaev

*Numerical Simulation of Flow Around Road Vehicles*

Proceedings of CFD96, CFD Society of Canada, Ottawa, Ontario, June 1996

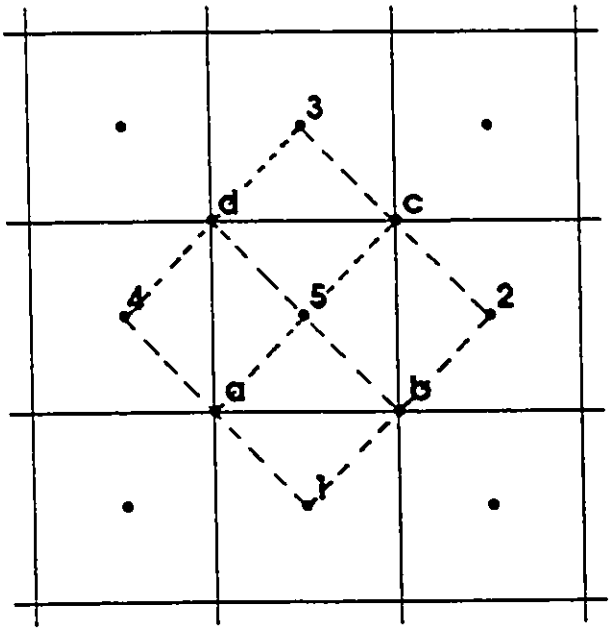


Figure 2.1: Cell abcd

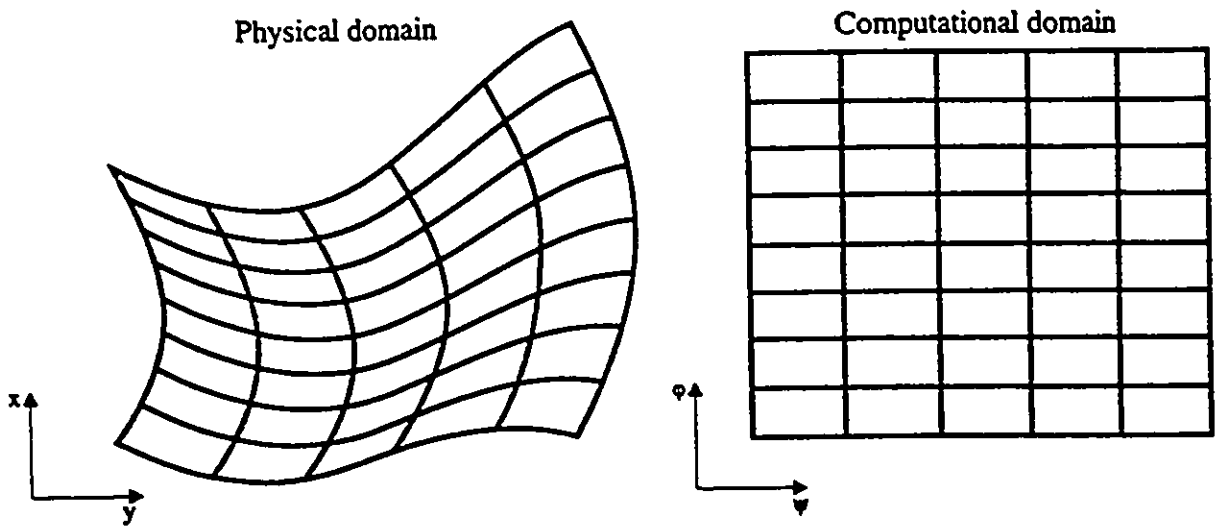


Figure 3.1: Mapping physical domain to computational

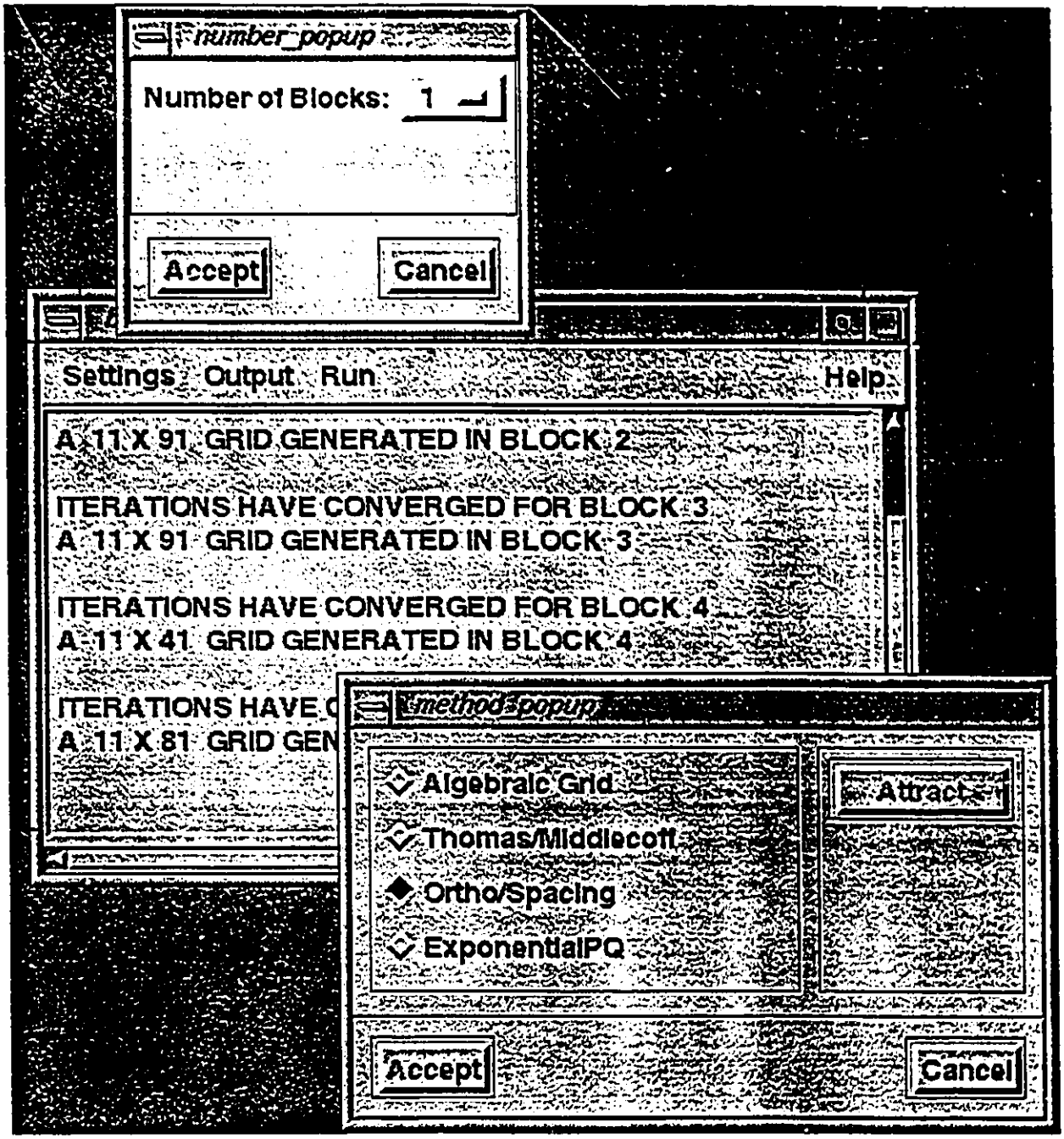


Figure 3.2: GUI window and popup menus

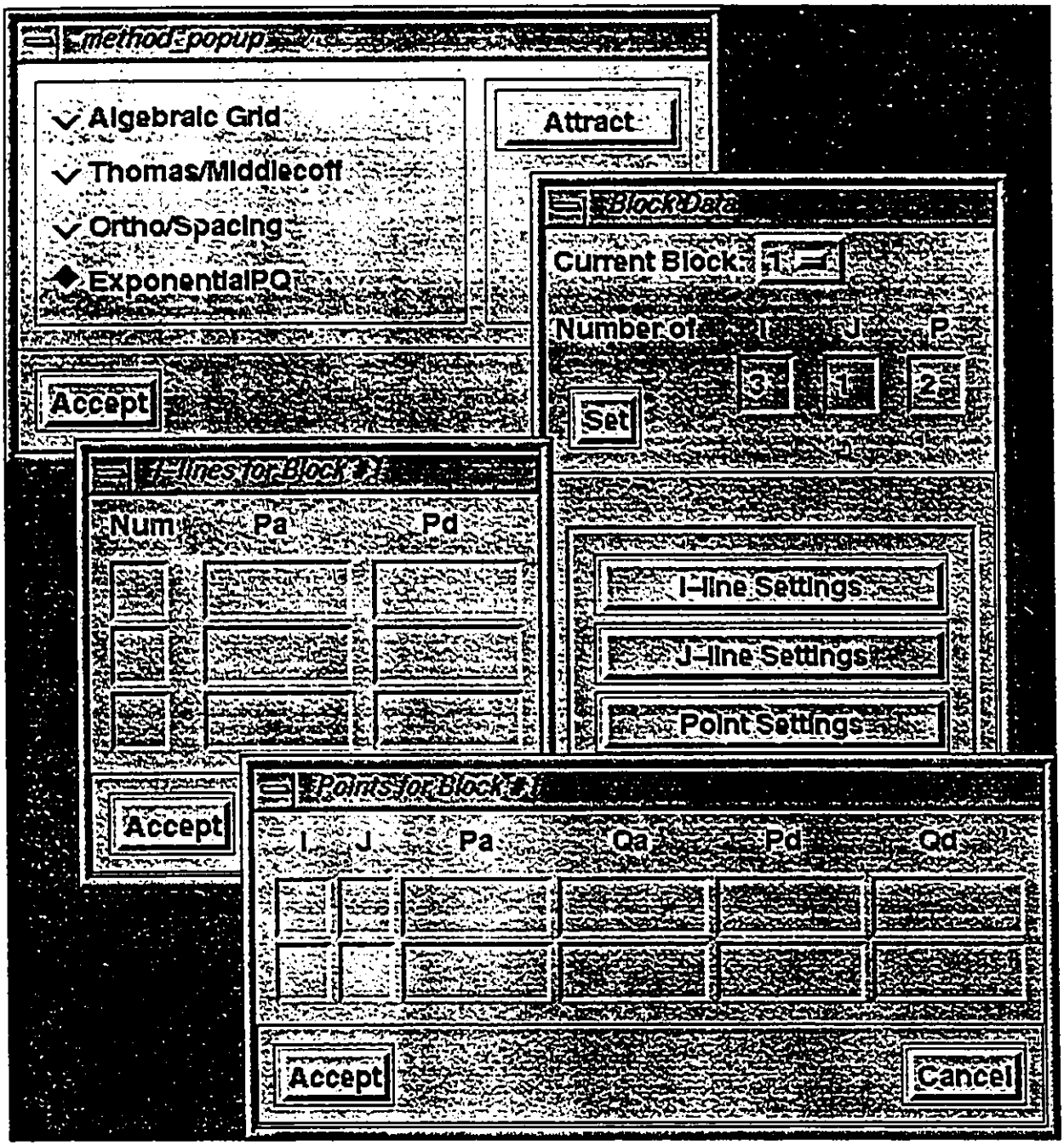


Figure 3.3: GUI popup menus

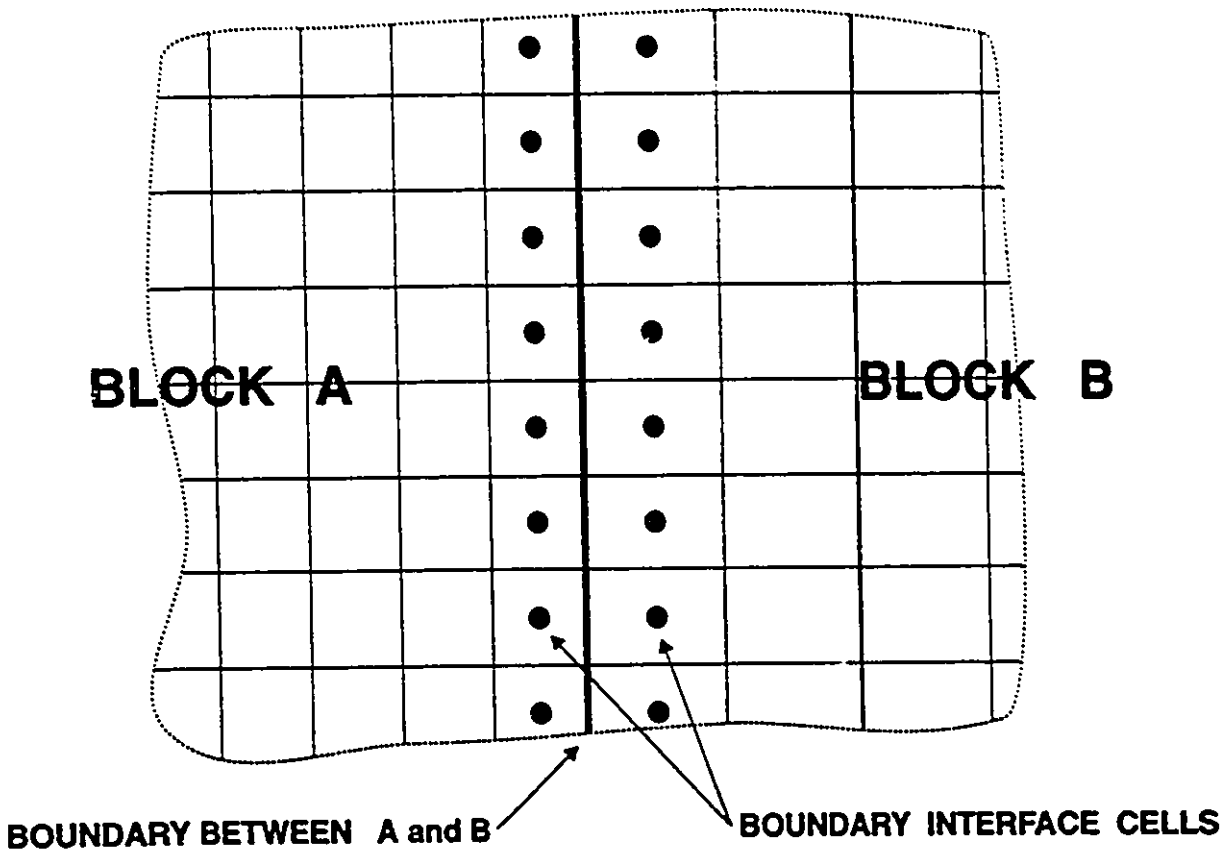


Figure 4.1: Data transfer between two neighbouring blocks

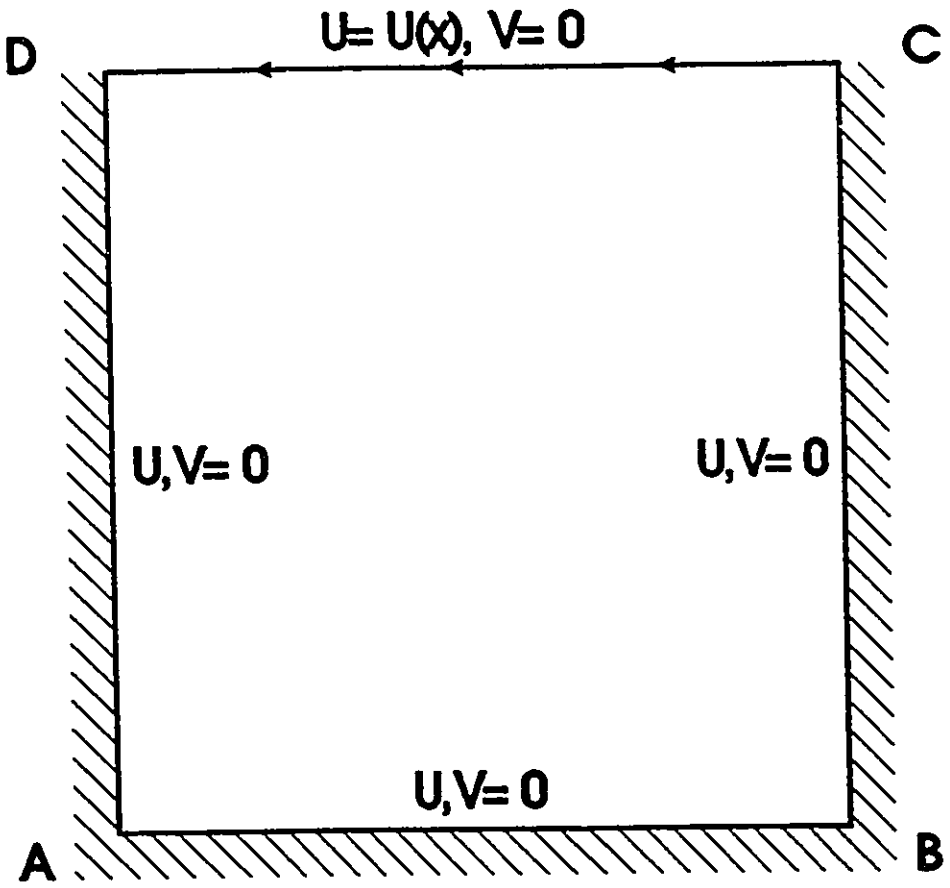


Figure 5.1: Driven cavity problem configuration

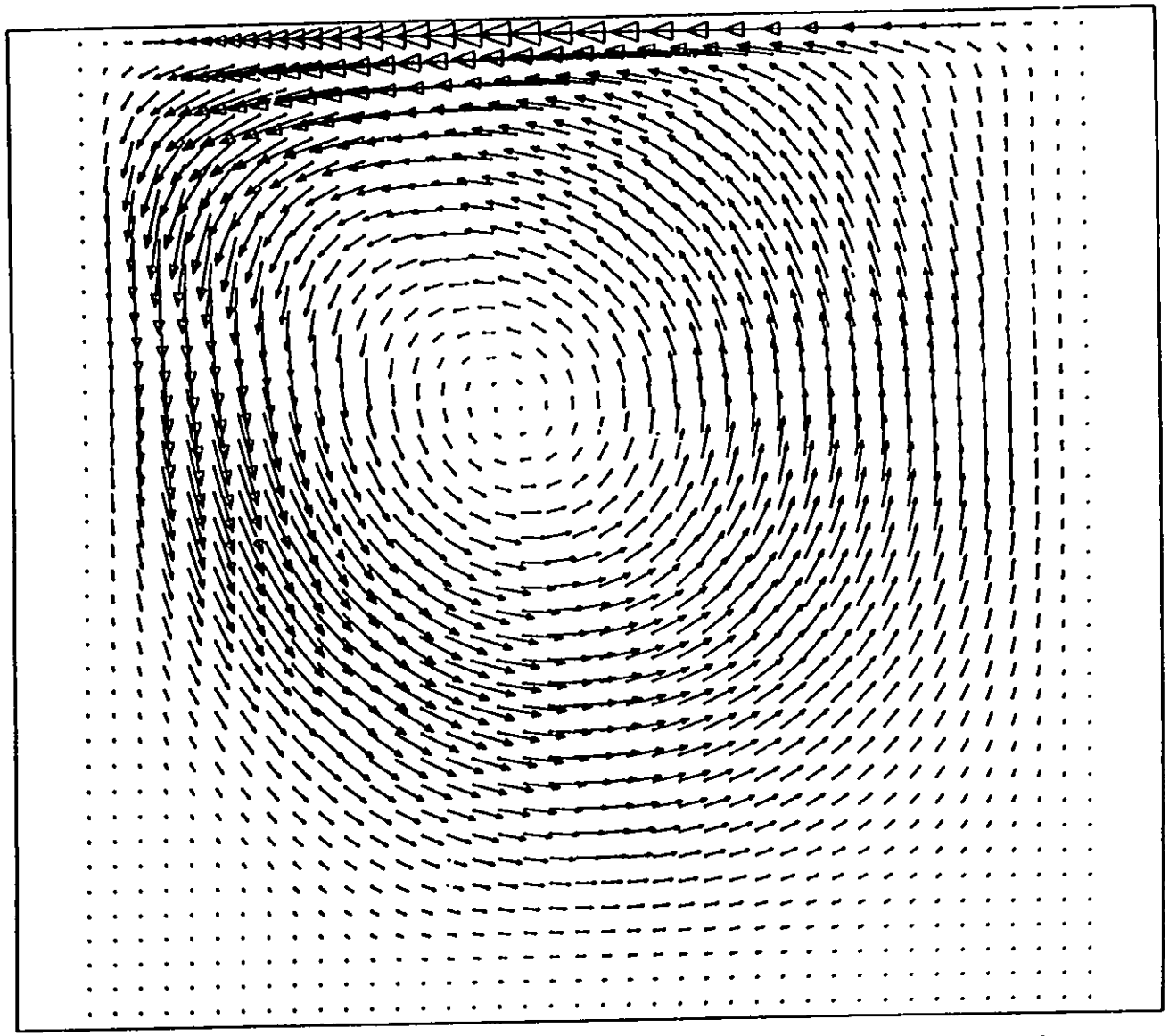

Figure 5.2: Flow in driven cavity, Re=400

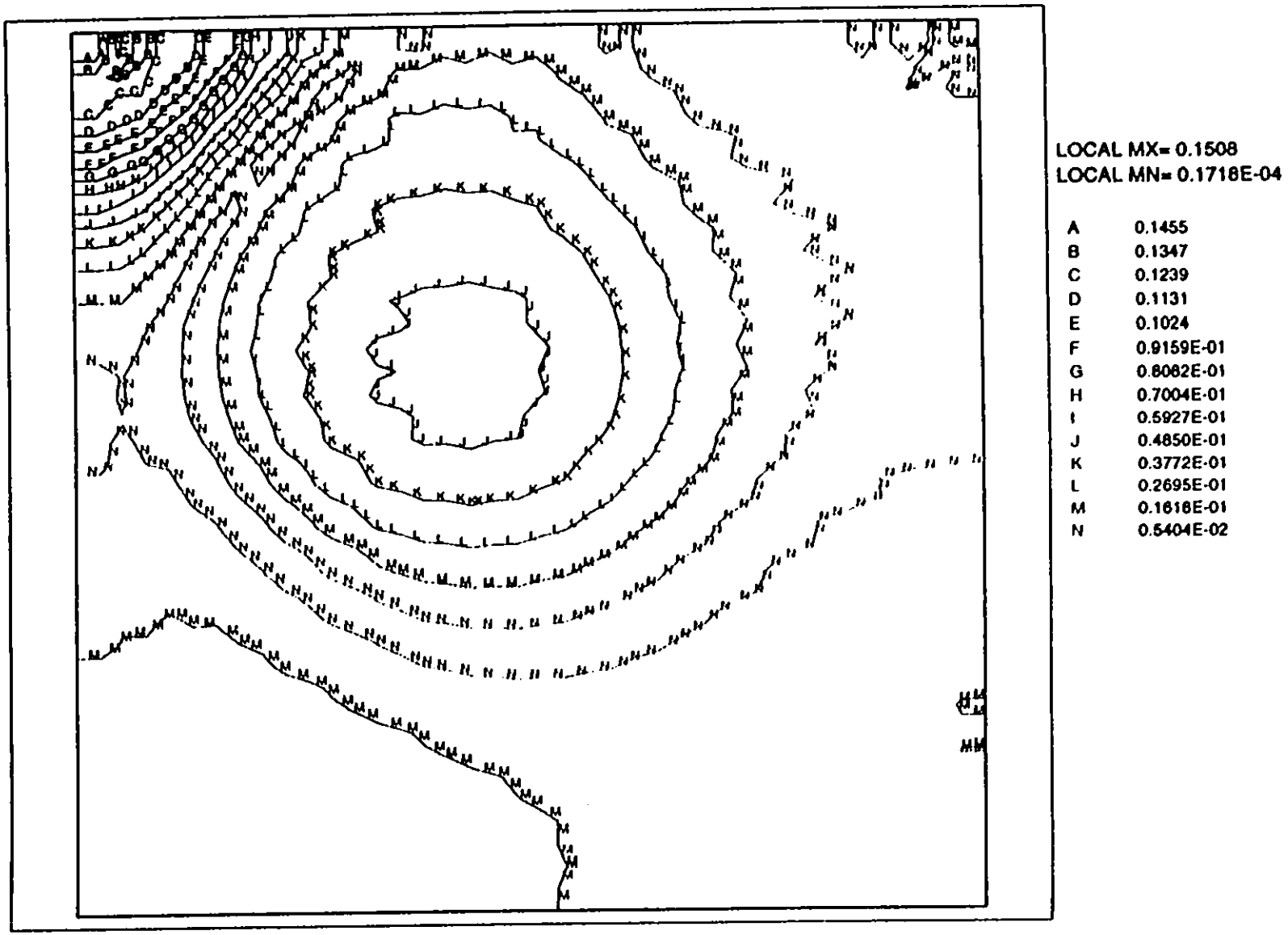


Figure 5.3: Pressure contours for driven cavity, Re=400

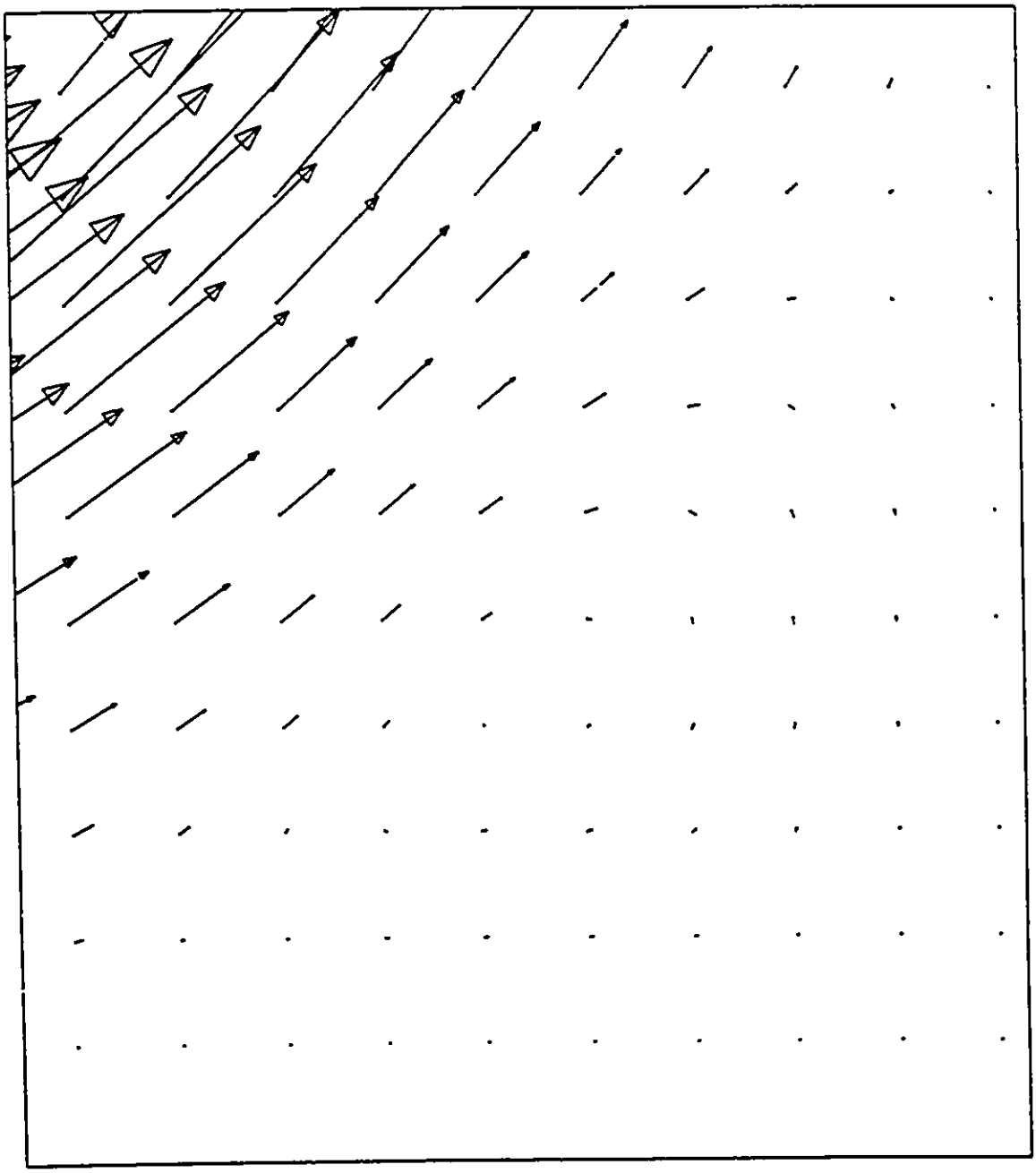

Figure 5.4: Driven cavity secondary vortex in lower left corner, Re=400

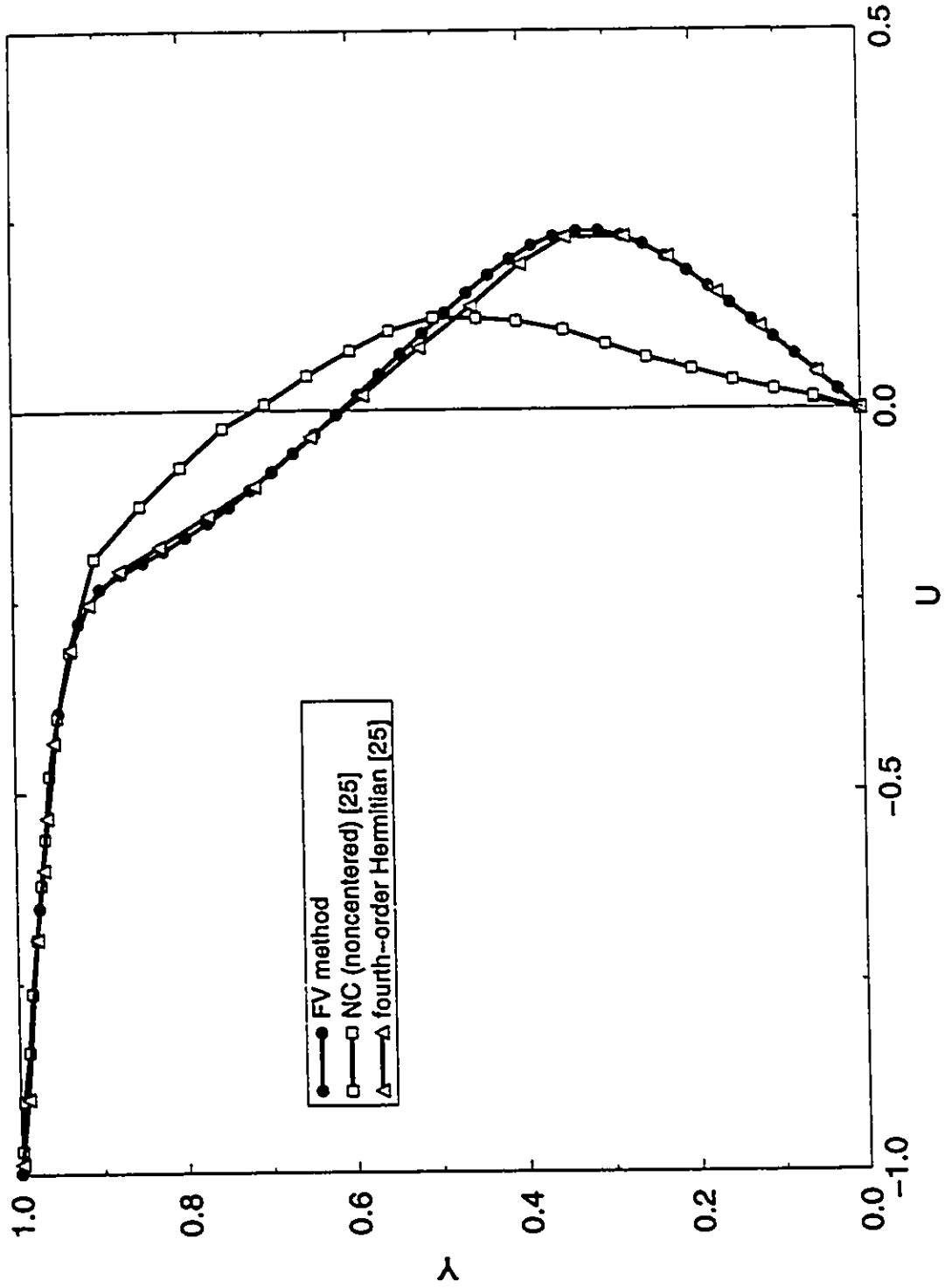


Figure 5.5: Velocity U on line x=0.5, Re=400

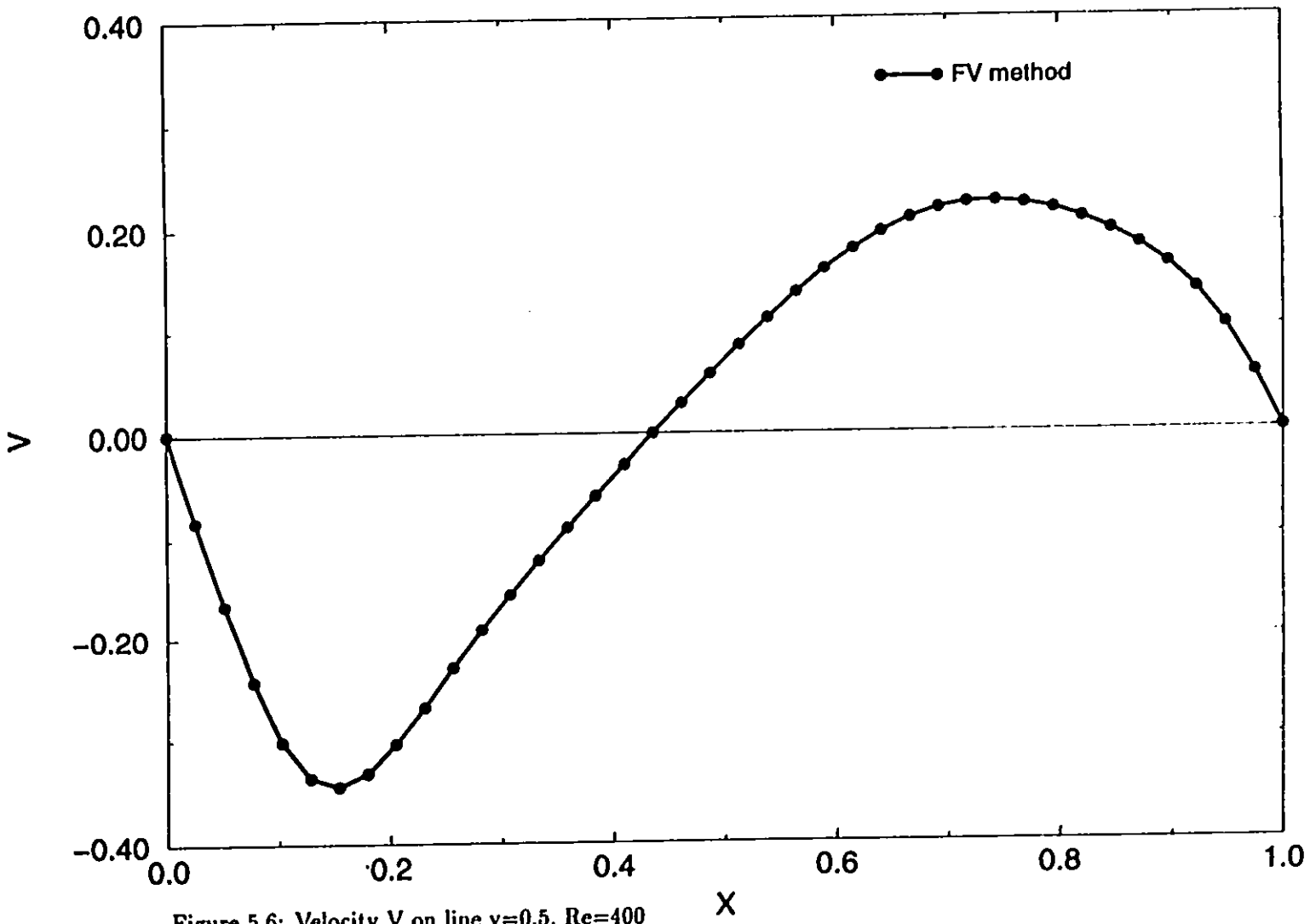


Figure 5.6: Velocity V on line y=0.5, Re=400

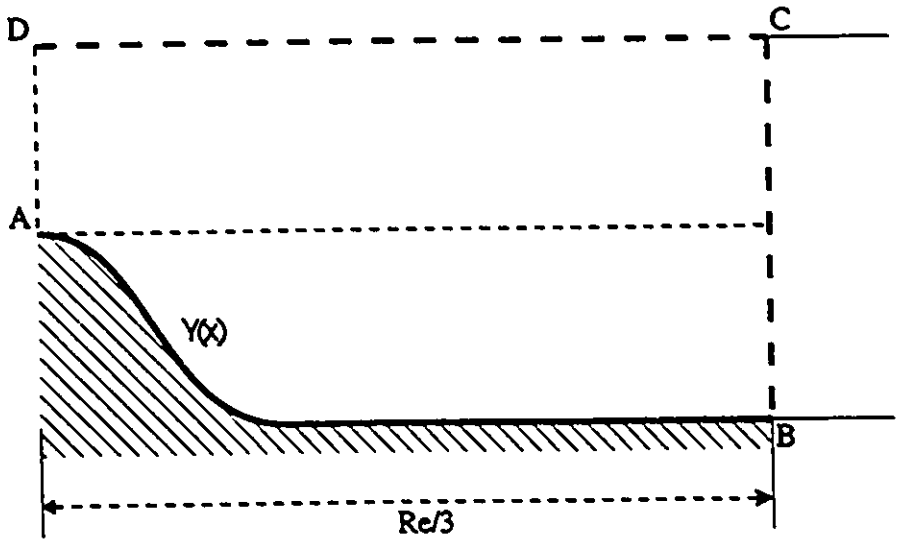


Figure 5.7: Configuration for the rapidly expanding channel problem

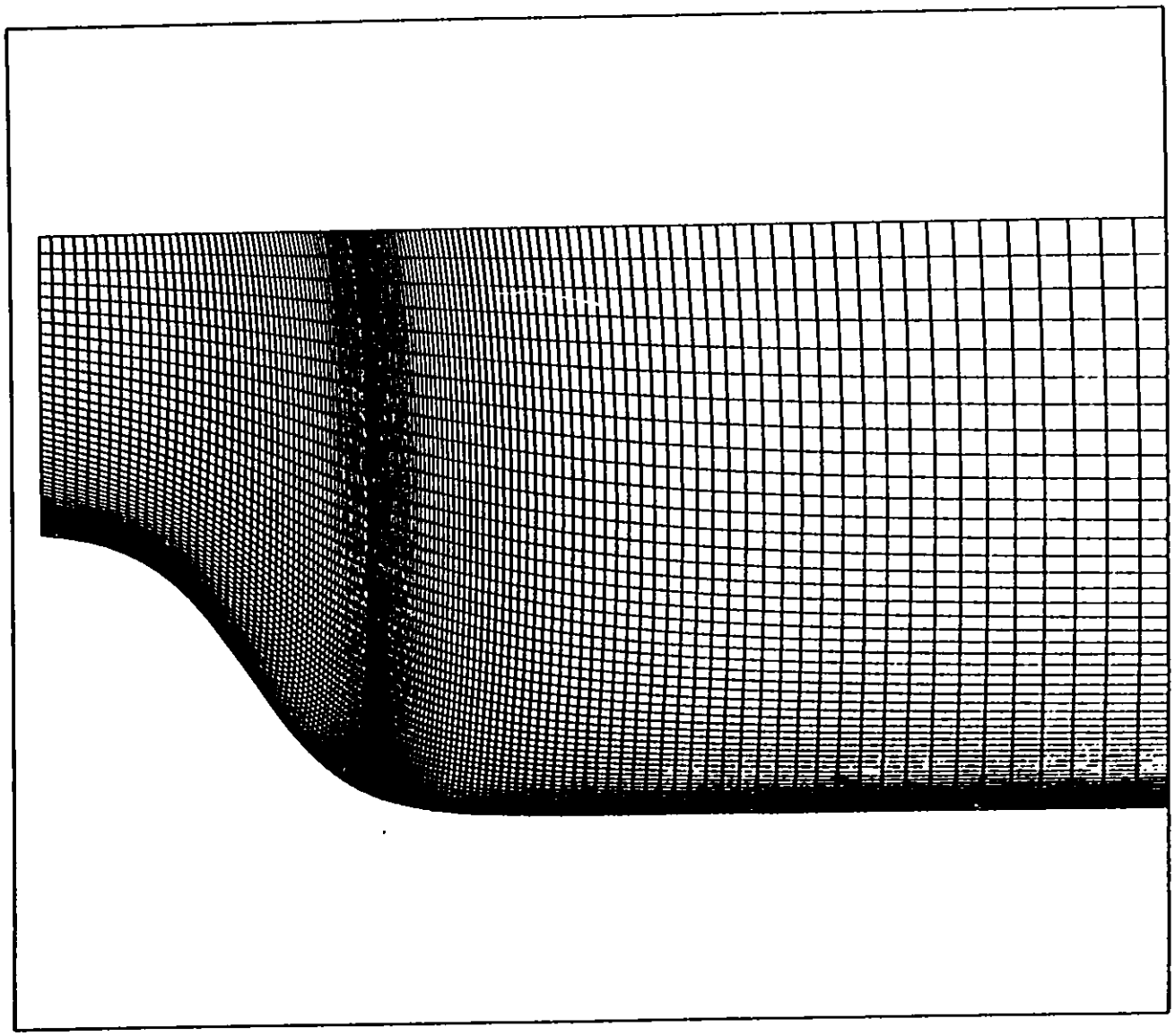

Figure 5.8: 80x50 single block mesh

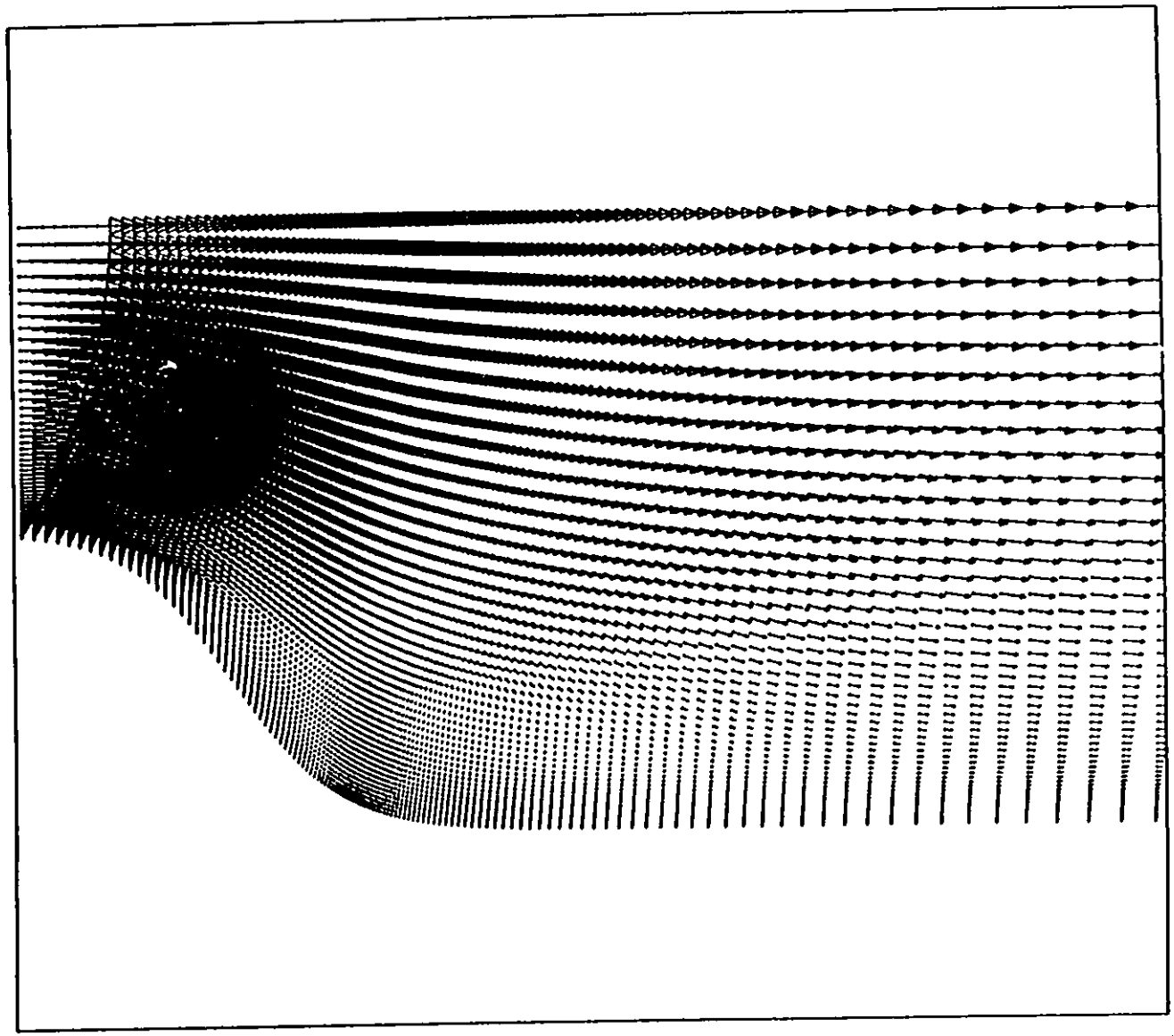

Figure 5.9: Velocity field, Re=10

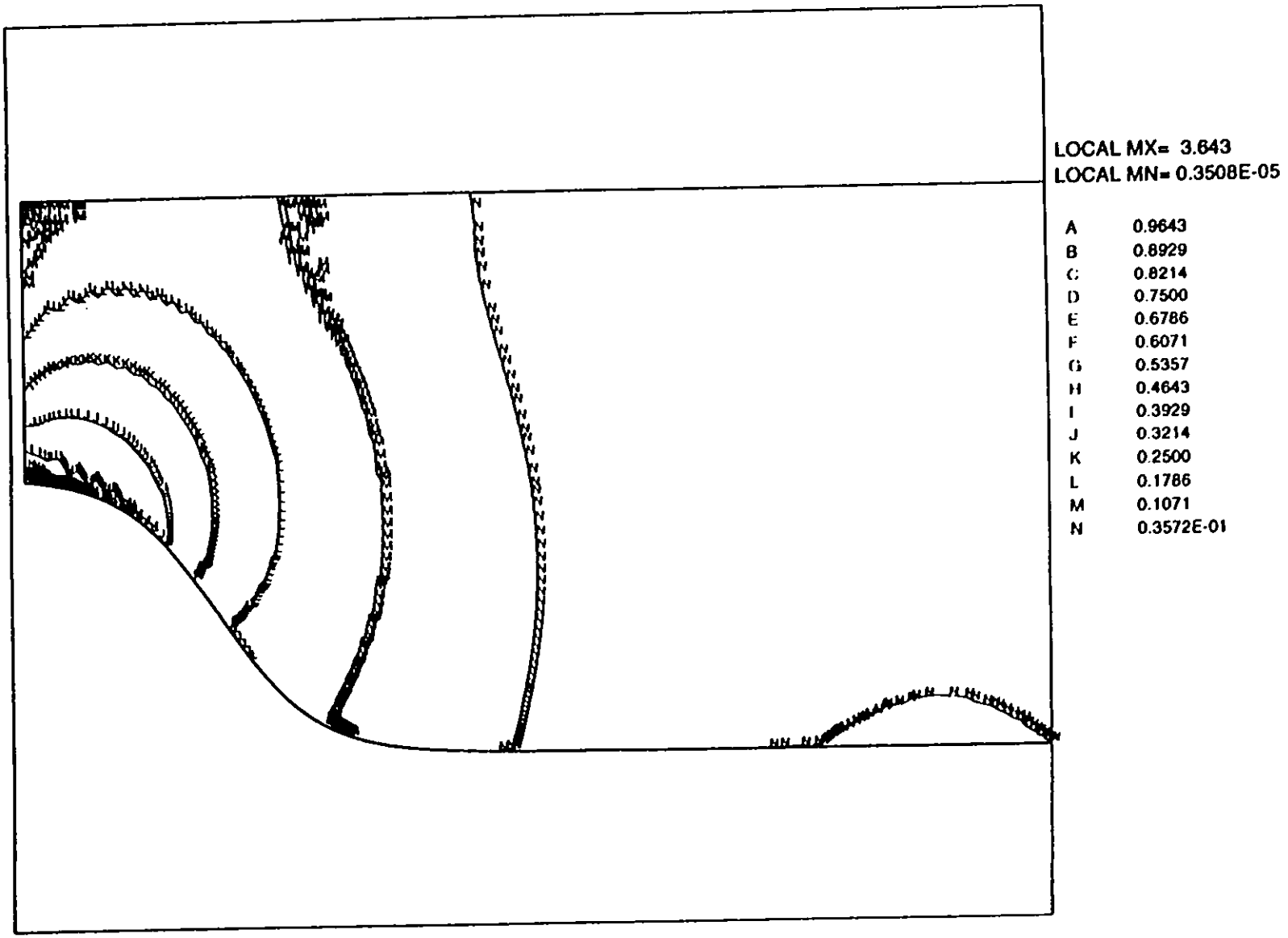


Figure 5.10: Pressure distribution, Re=10

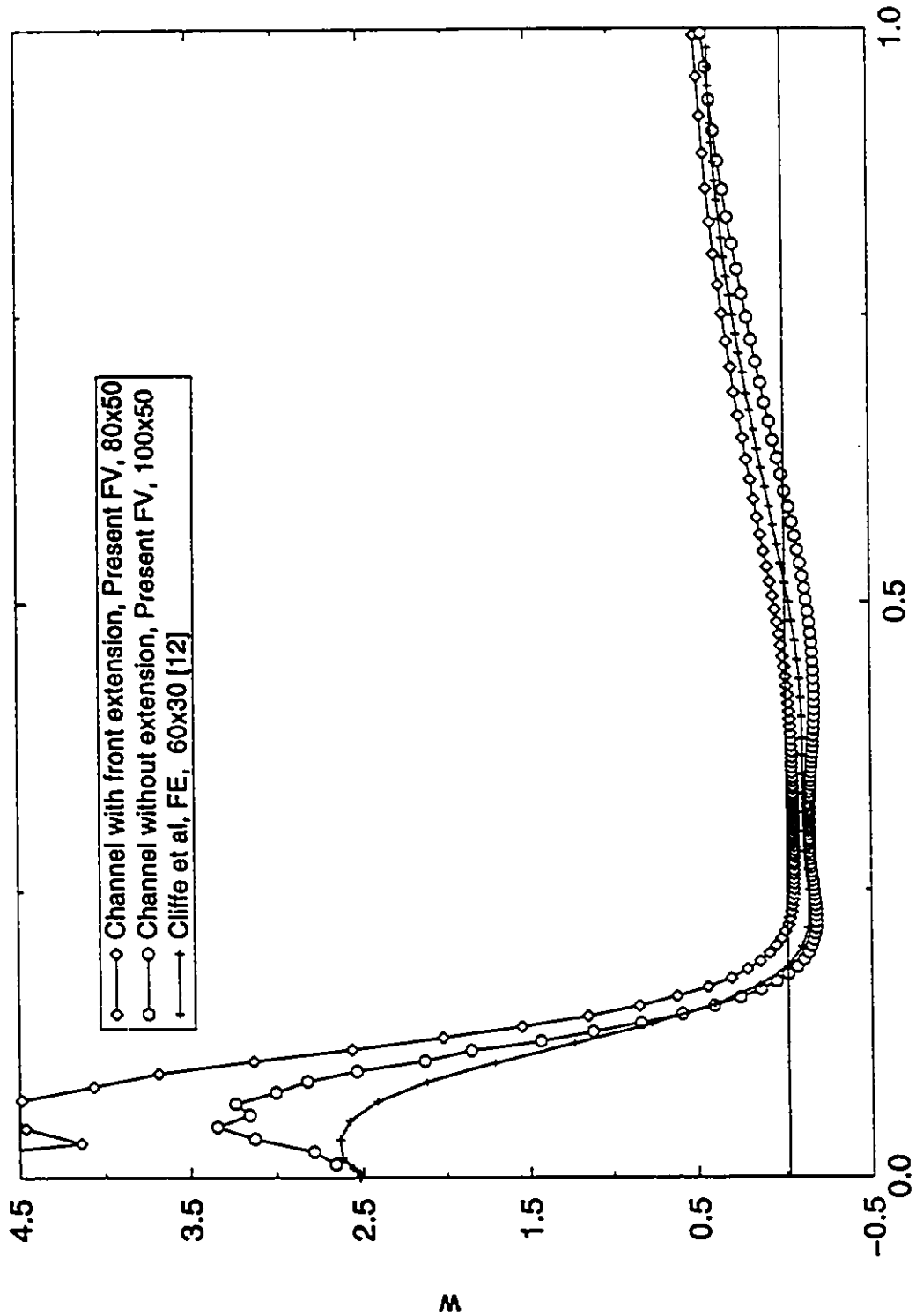


Figure 5.11: Vorticity distribution, Re=10

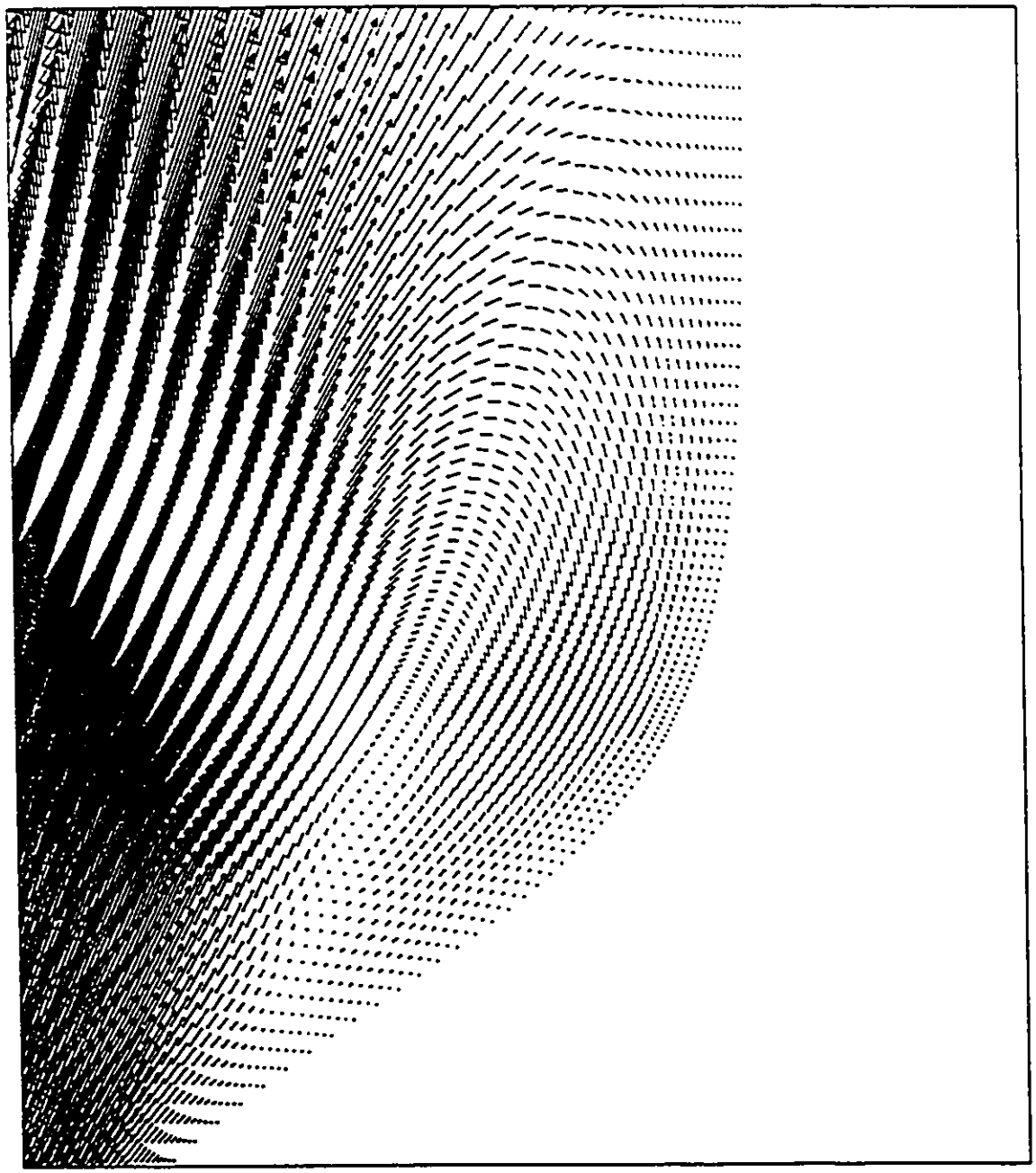

Figure 5.12: Flow recirculation near the wall

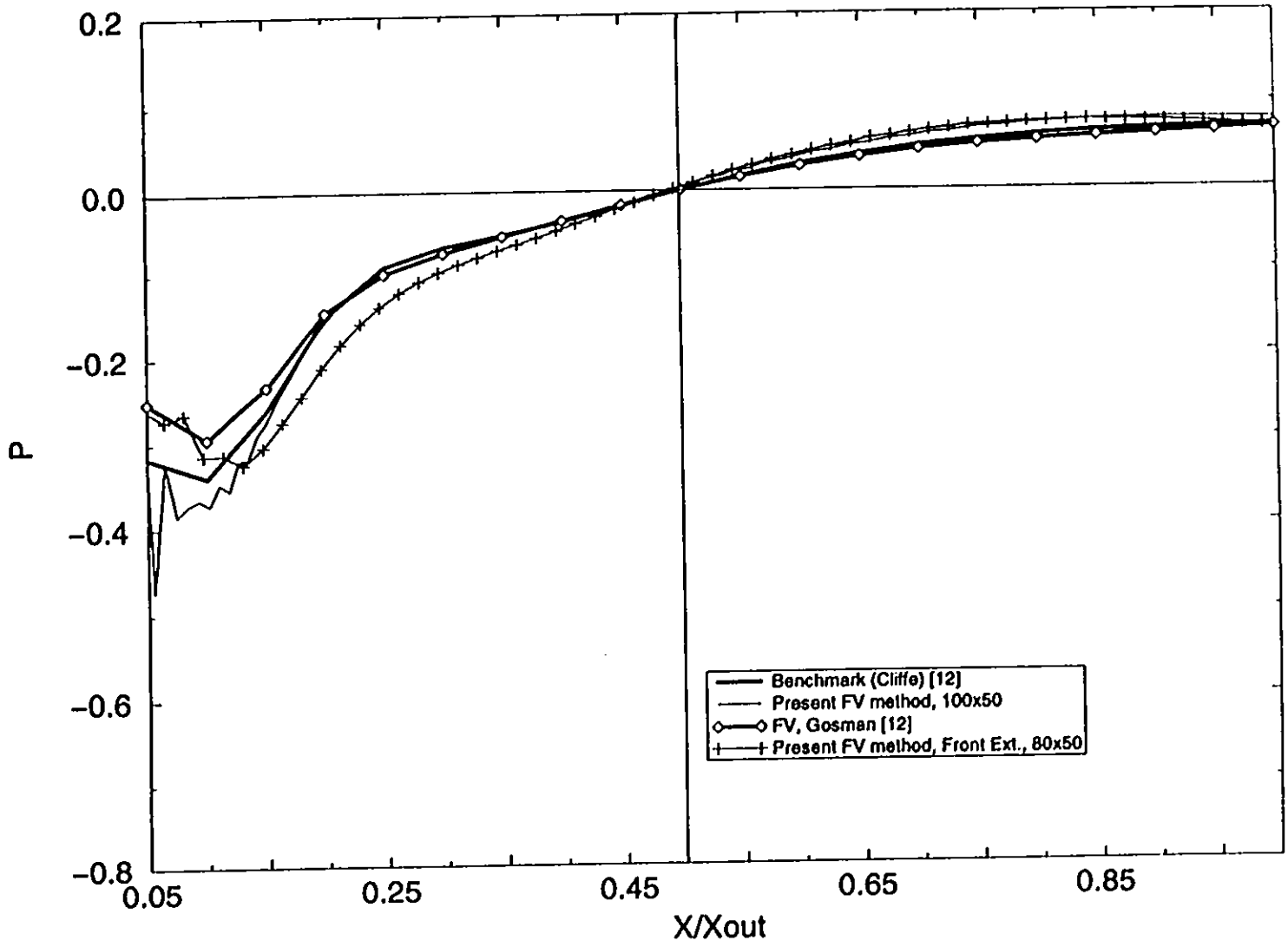


Figure 5.13: Pressure distribution along the wall, Re=10

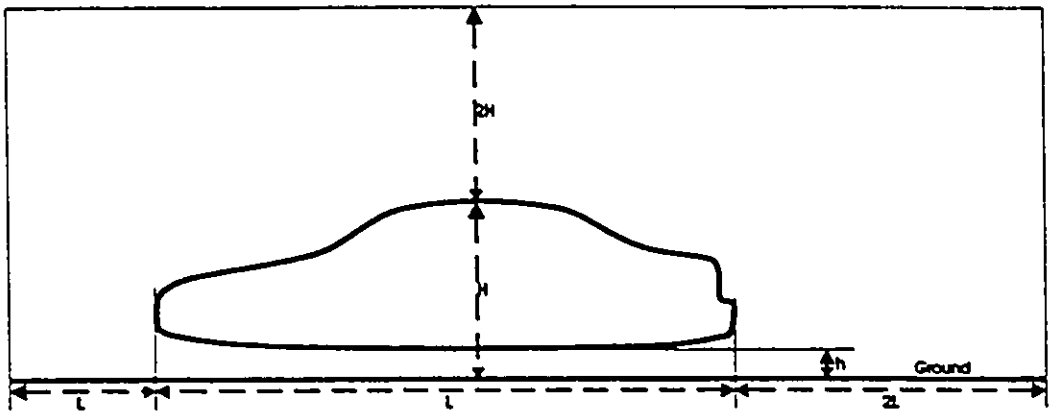


Figure 6.1: Physical domain

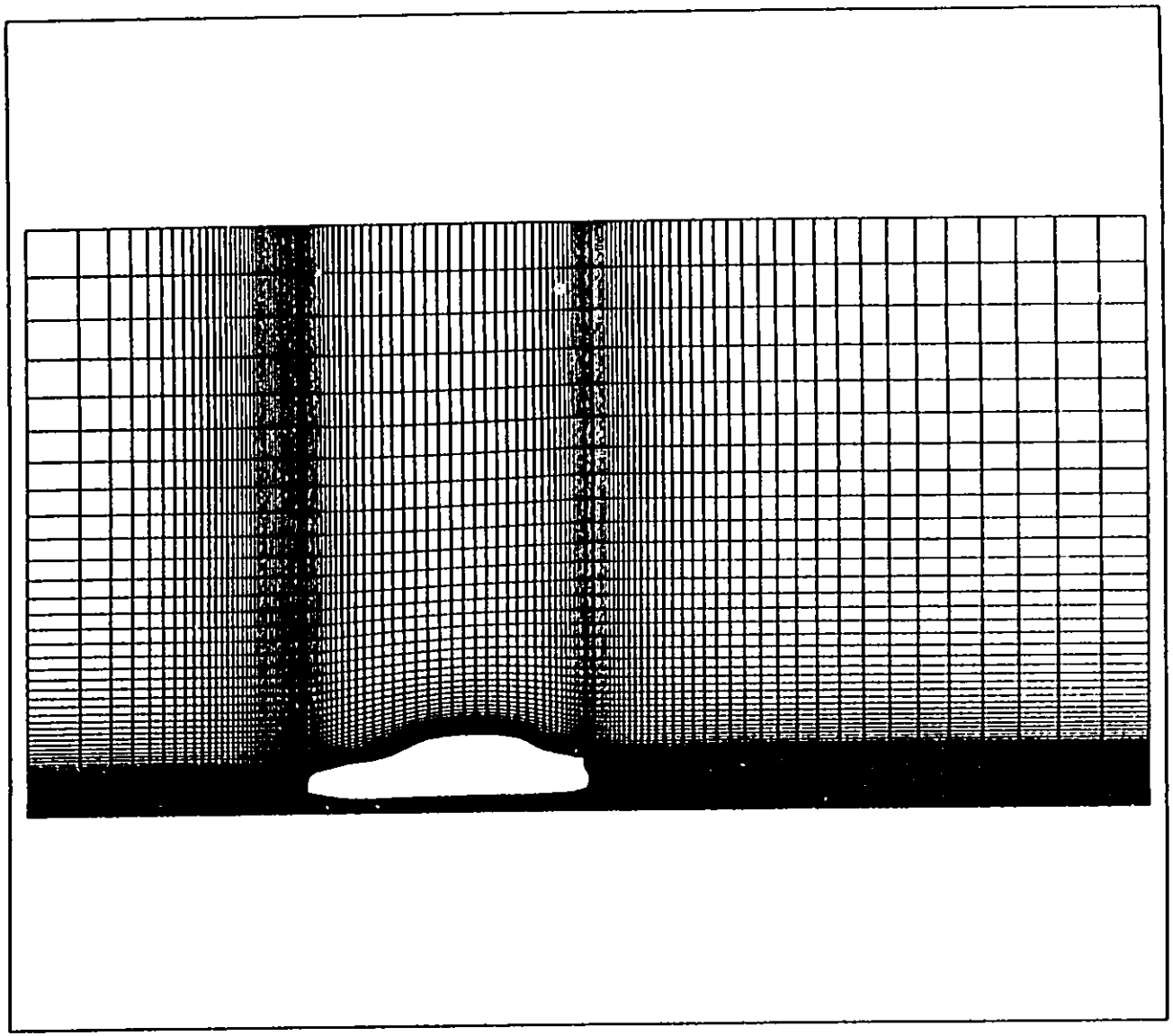

Figure 6.2: Mesh arrangement #1; 8 blocks

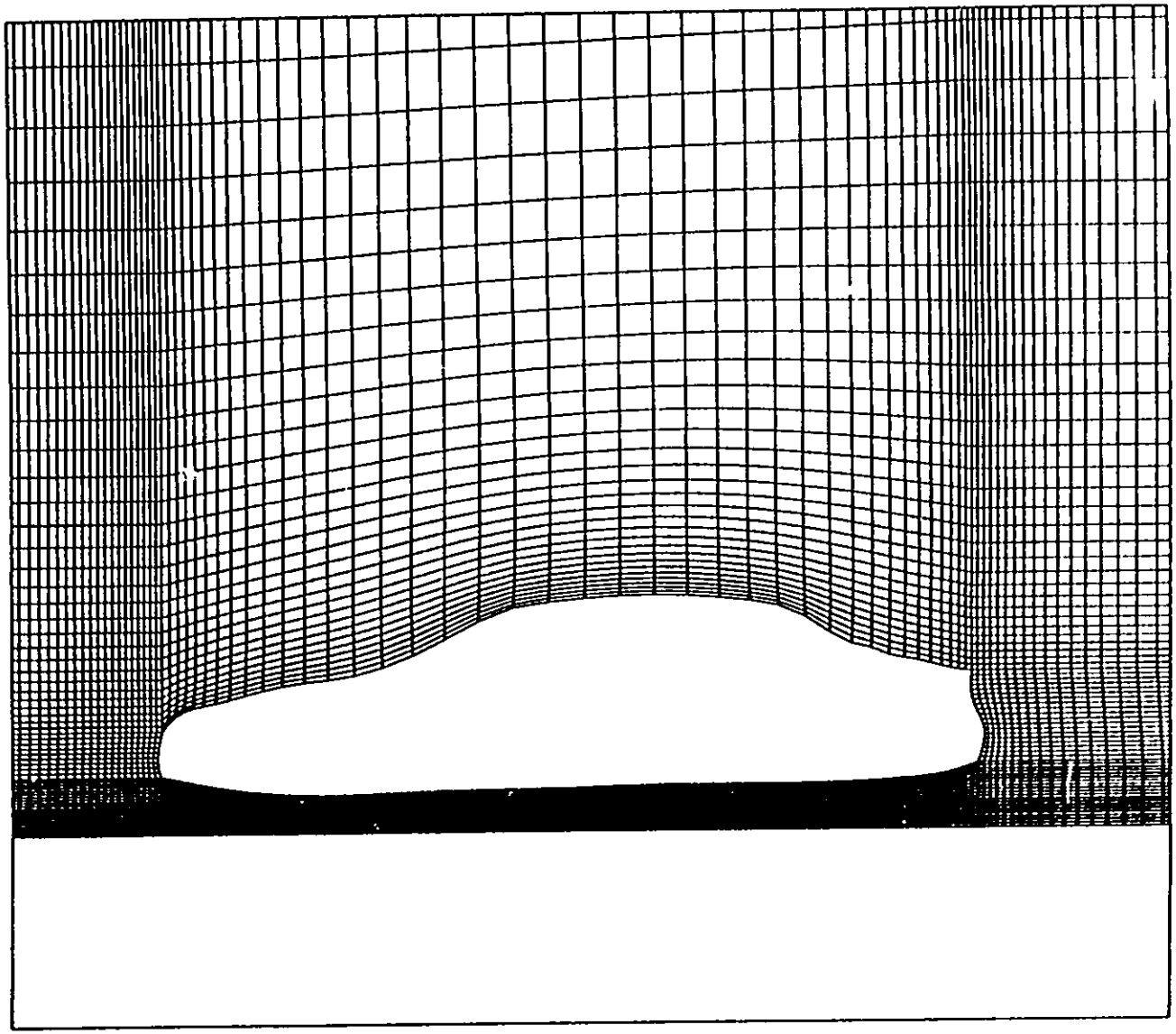

Figure 6.3: View of mesh #1 near vehicle

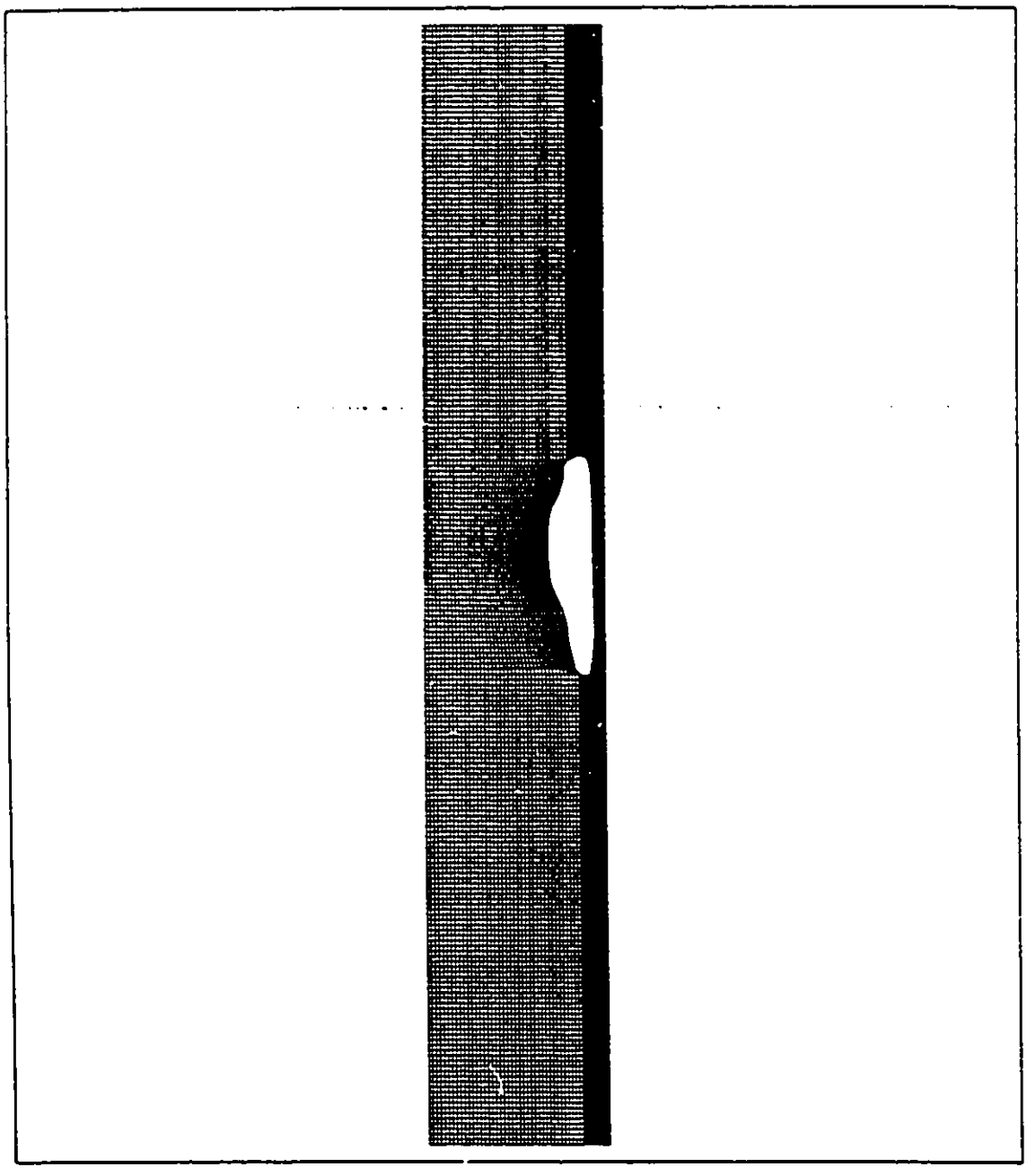

Figure 6.4: Mesh arrangement #2; 9 blocks

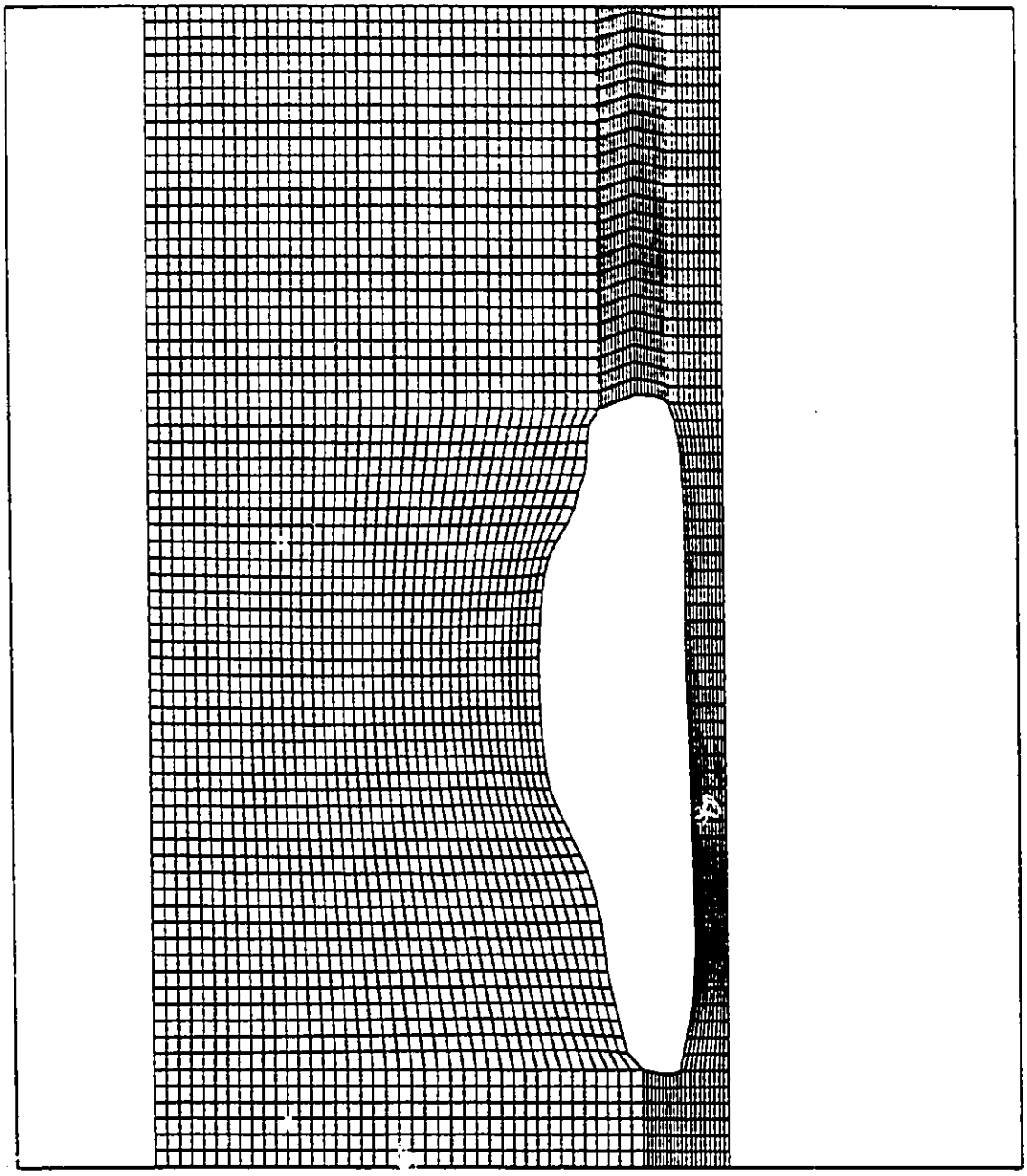

Figure 6.5: View of mesh #2 near vehicle

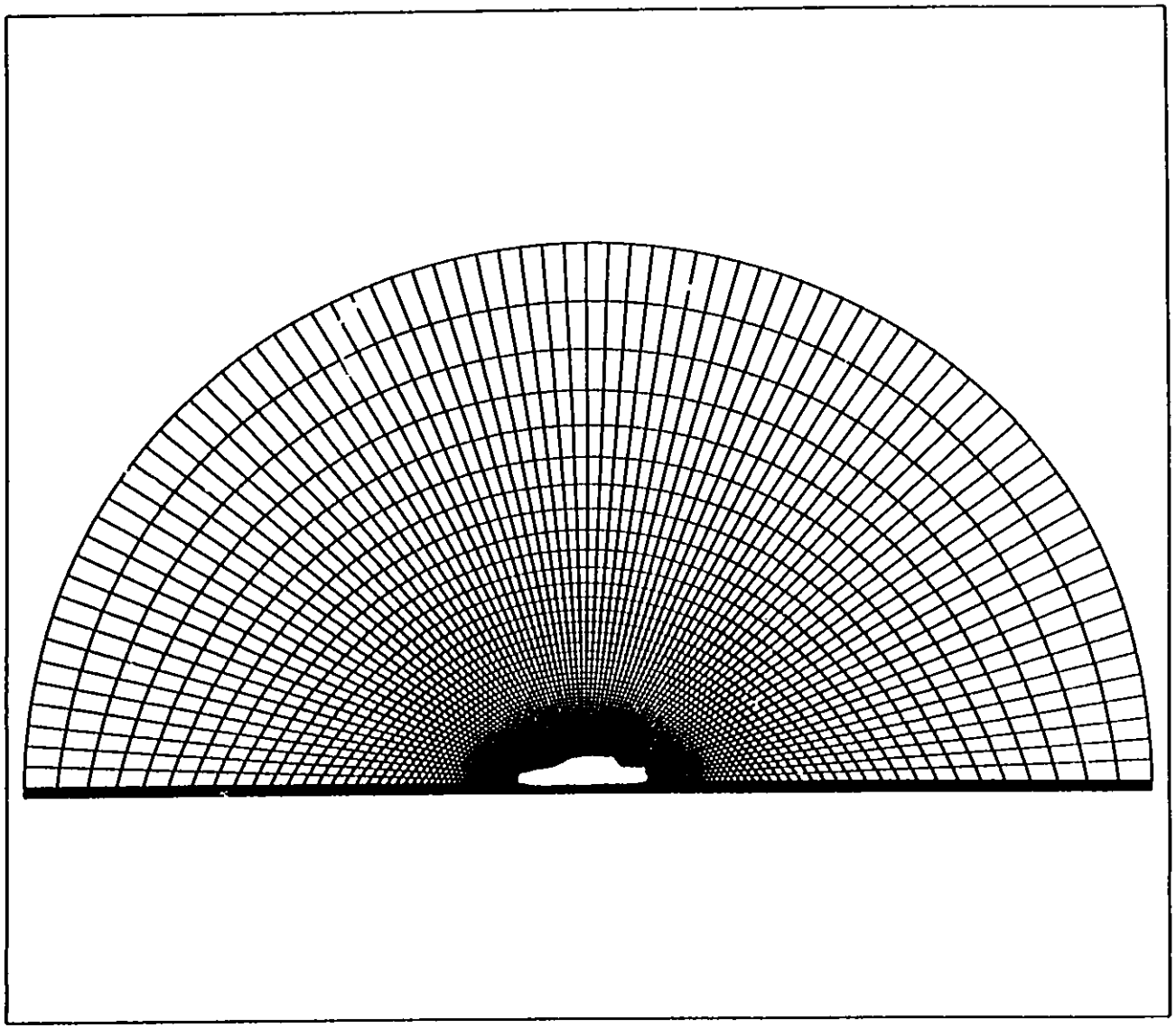

Figure 6.6: Mesh arrangement #3; 4 blocks

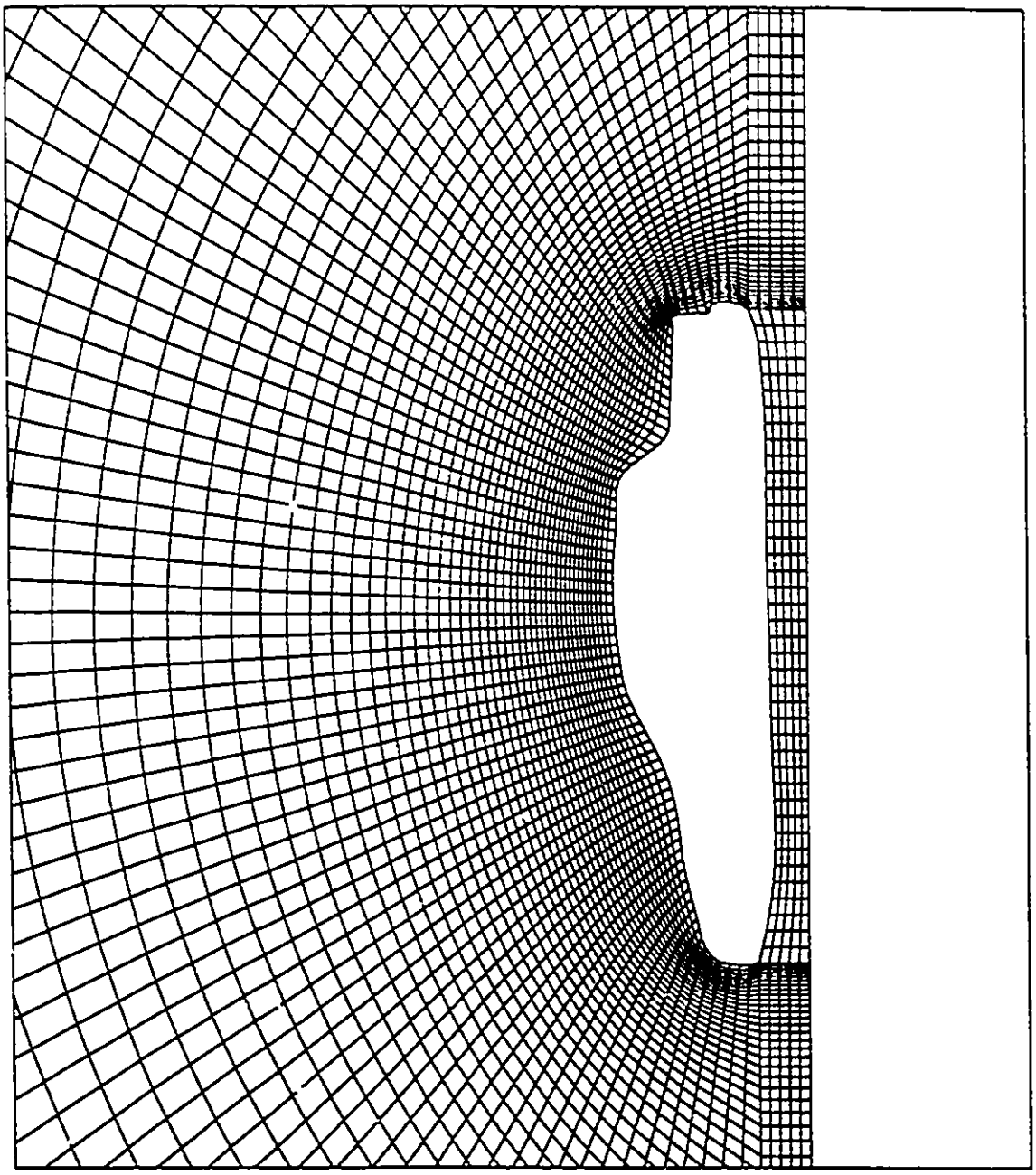

Figure 6.7: View of mesh #3 near vehicle

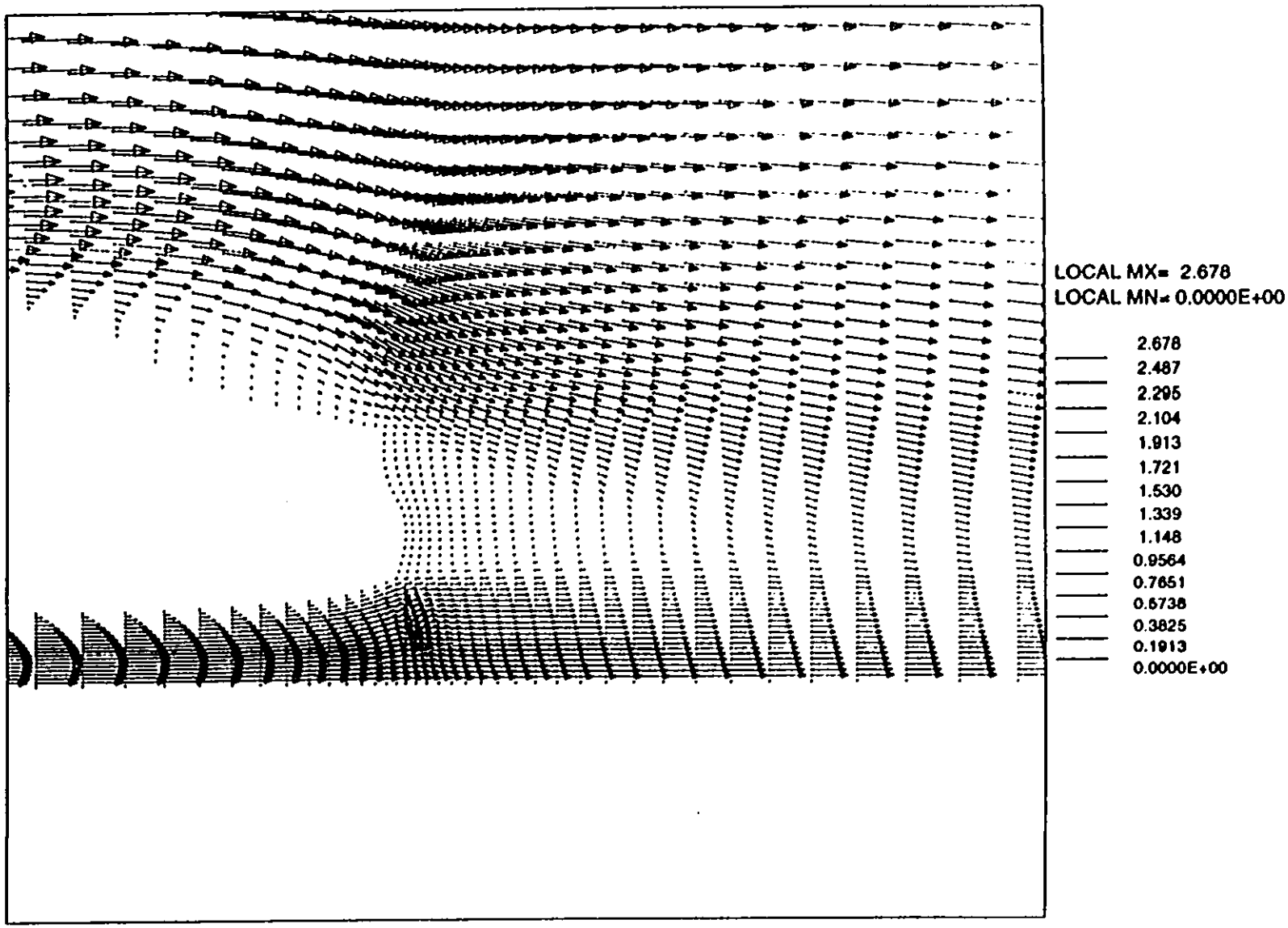


Figure 6.8: Velocity vector field behind the vehicle, mesh #1, Re=450

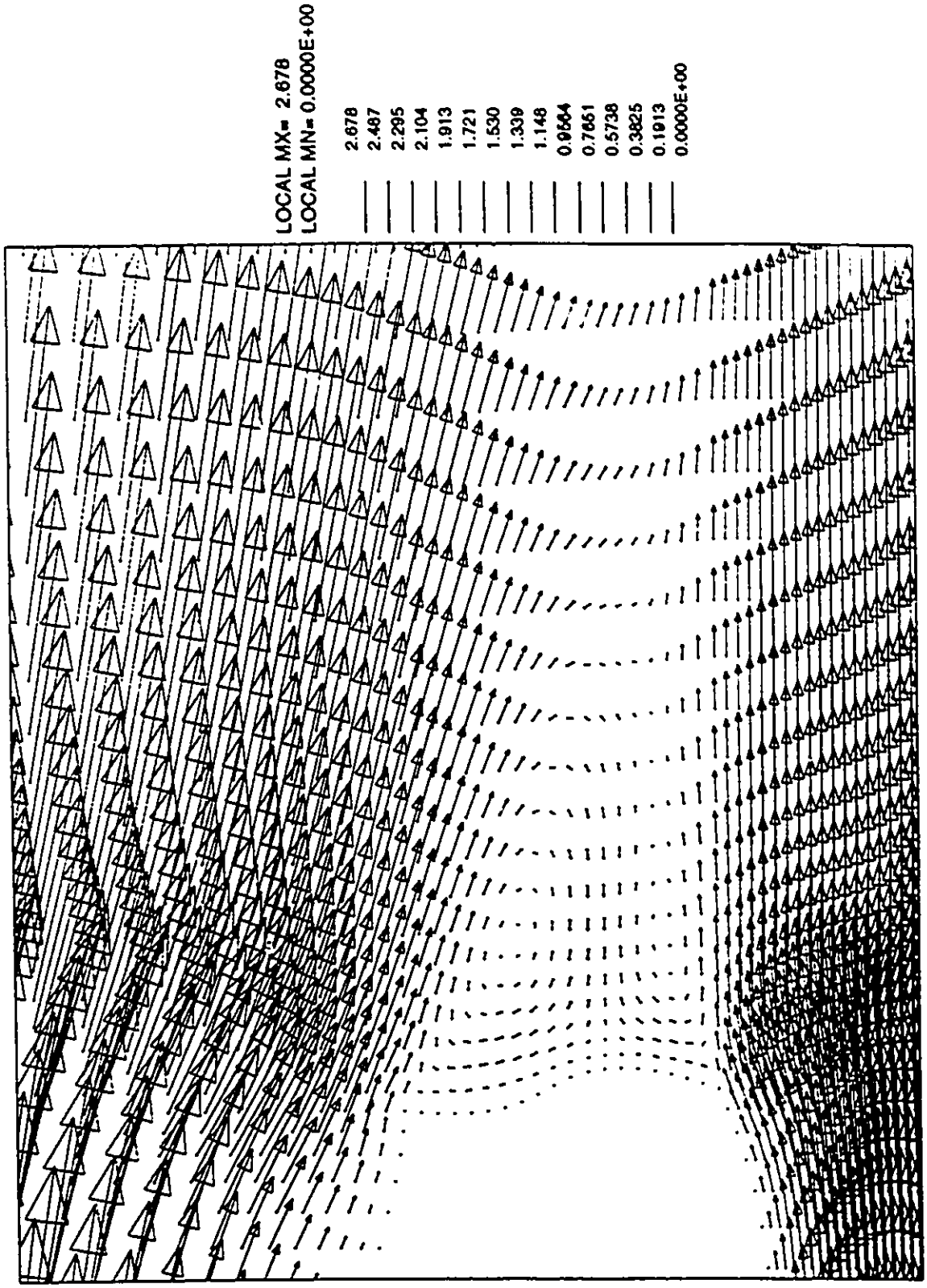


Figure 6.9: Close-up of velocity at rear of vehicle, mesh #1, Re=450

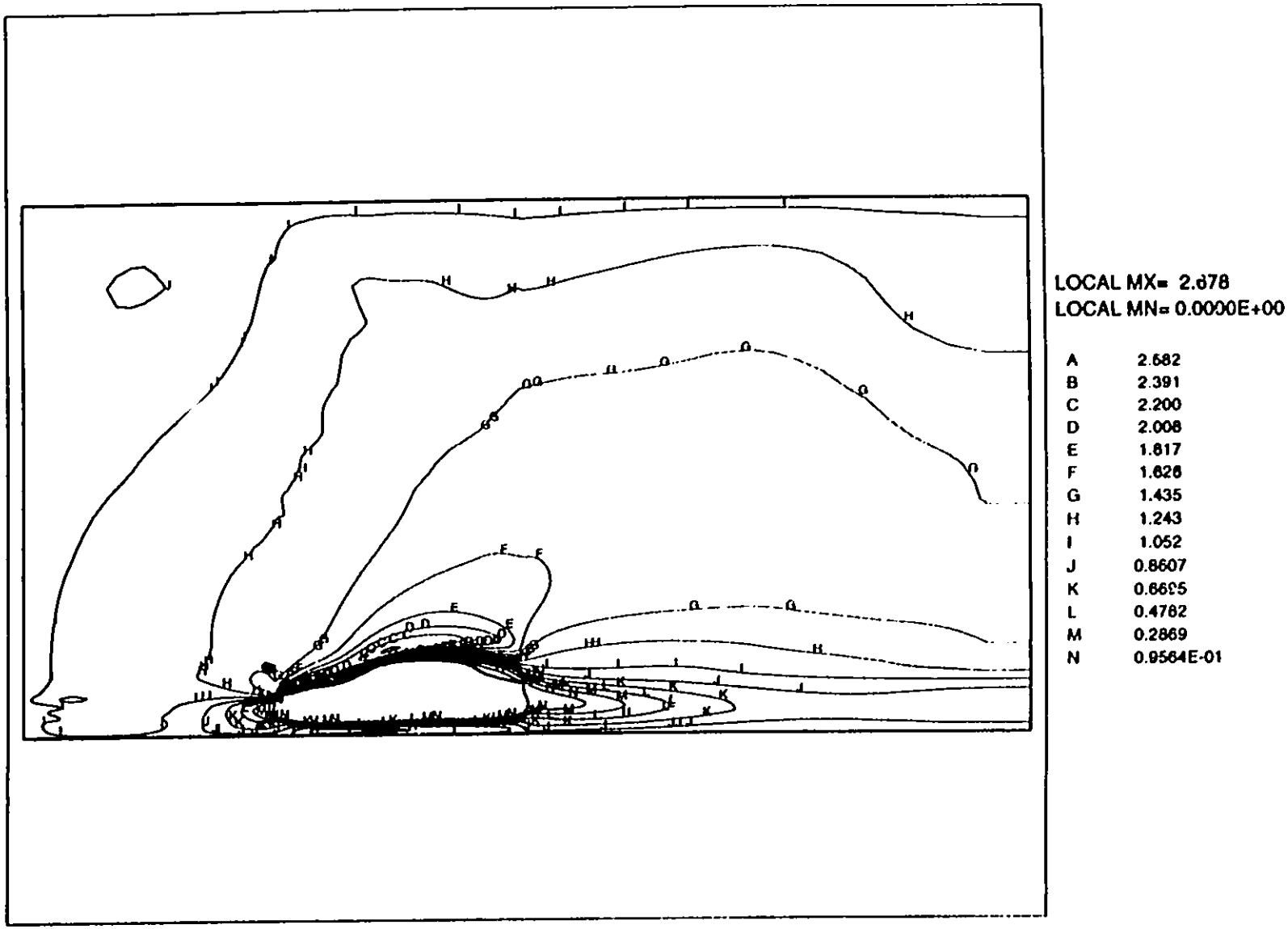


Figure 6.10: Velocity magnitude contours, mesh #1, Re=450

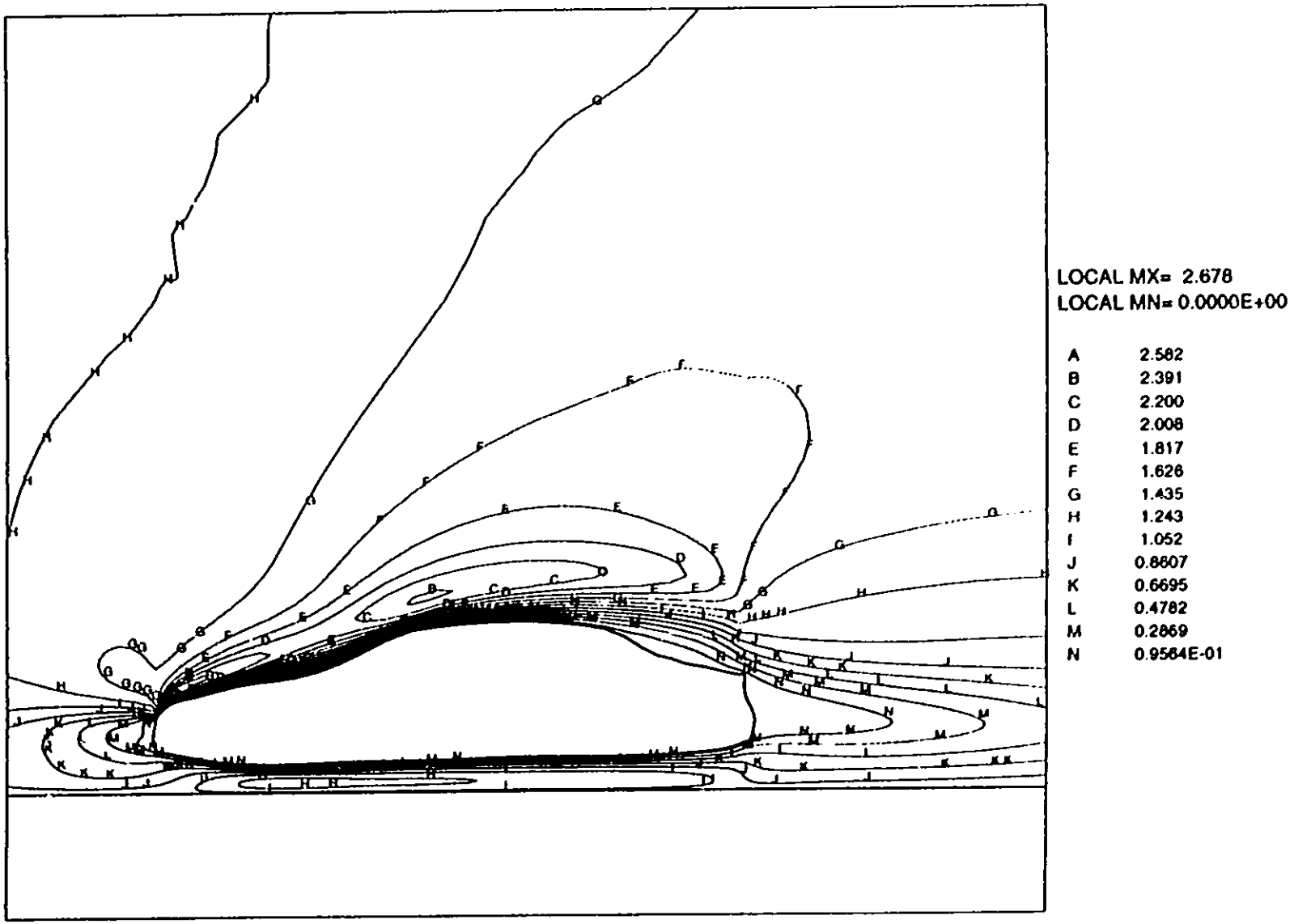


Figure 6.11: Velocity magnitude contours near the vehicle, Re=450

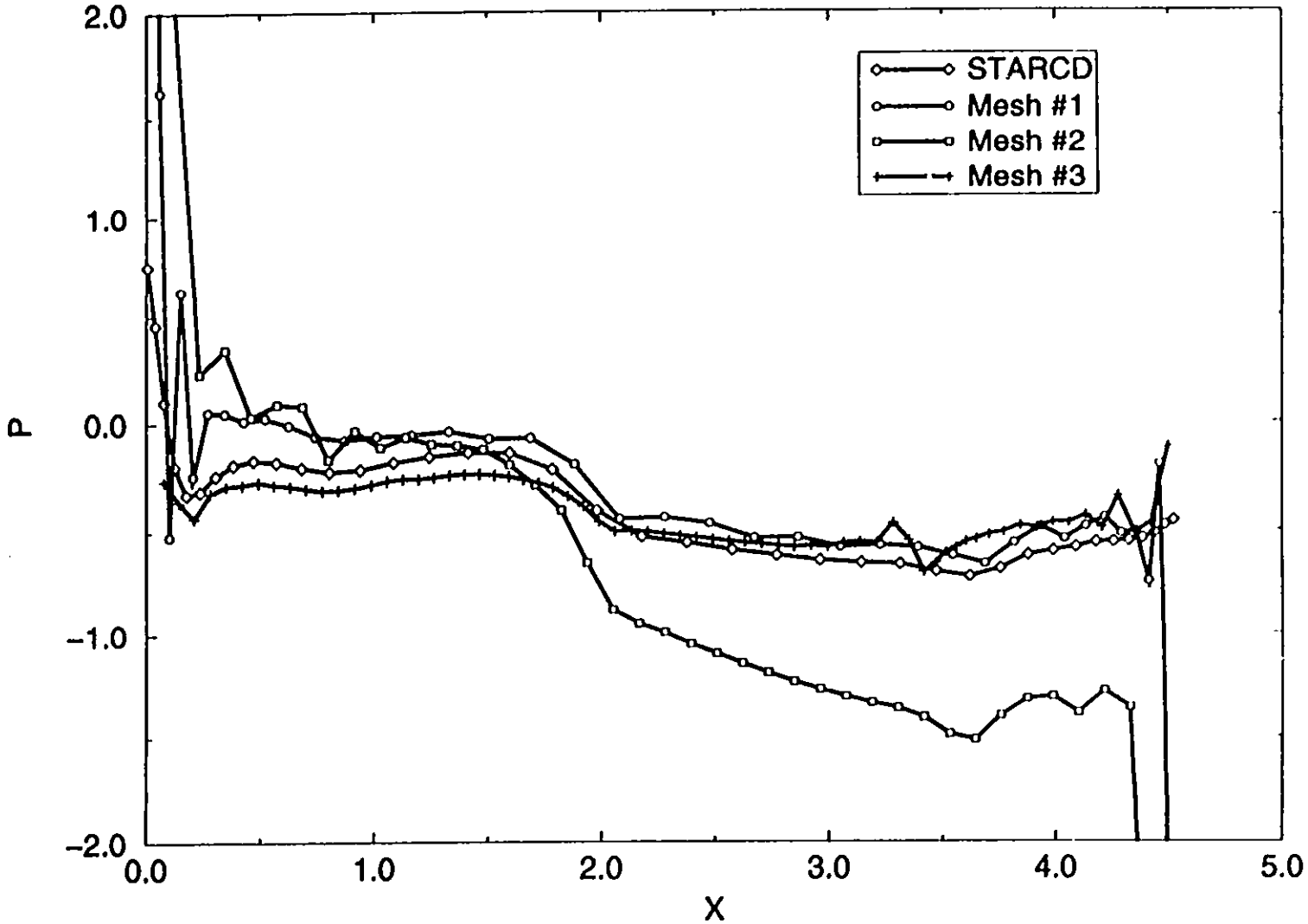


Figure 6.12: Pressure along top of the car, Re=450

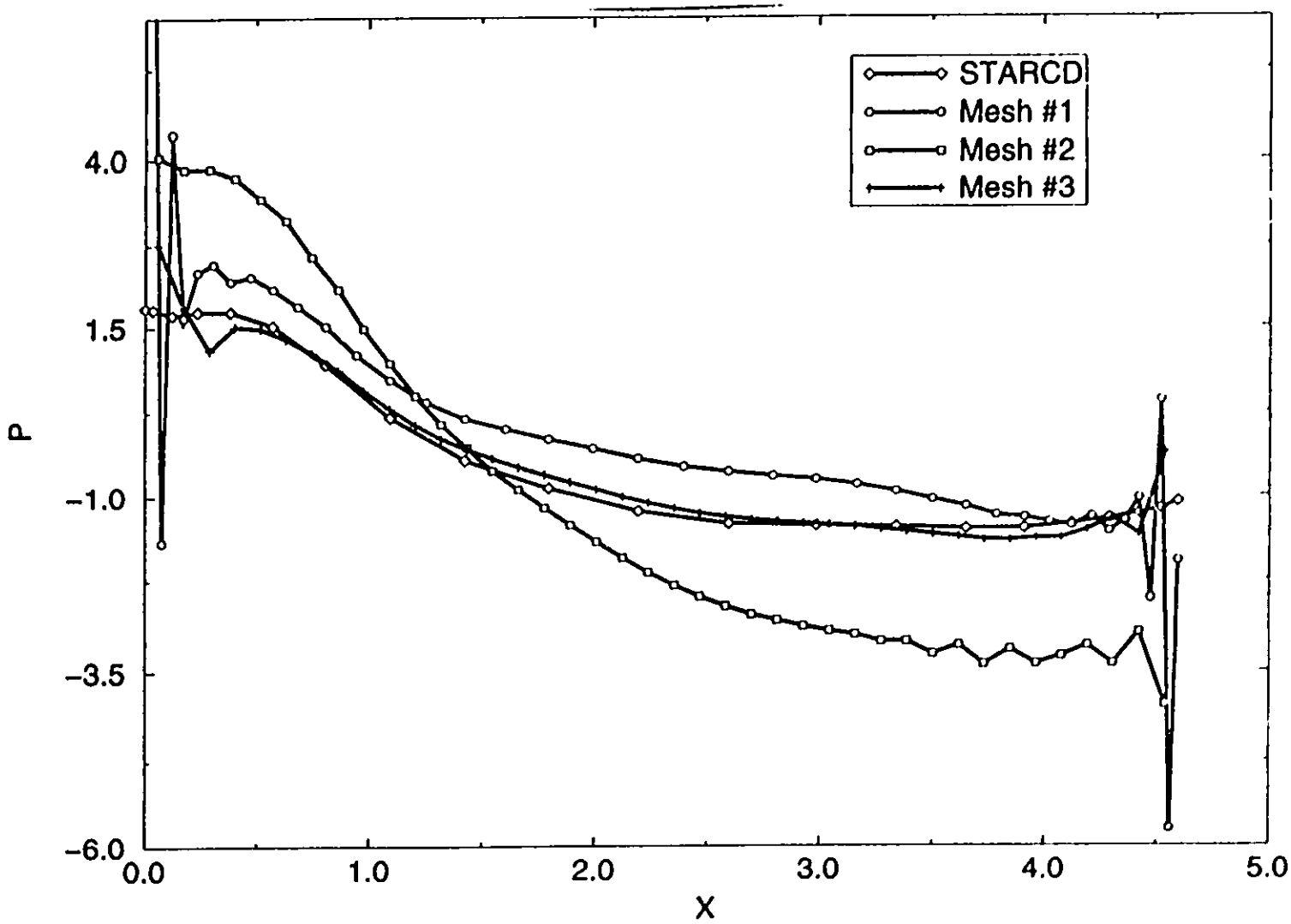


Figure 6.13: Pressure along bottom of the car, Re=450

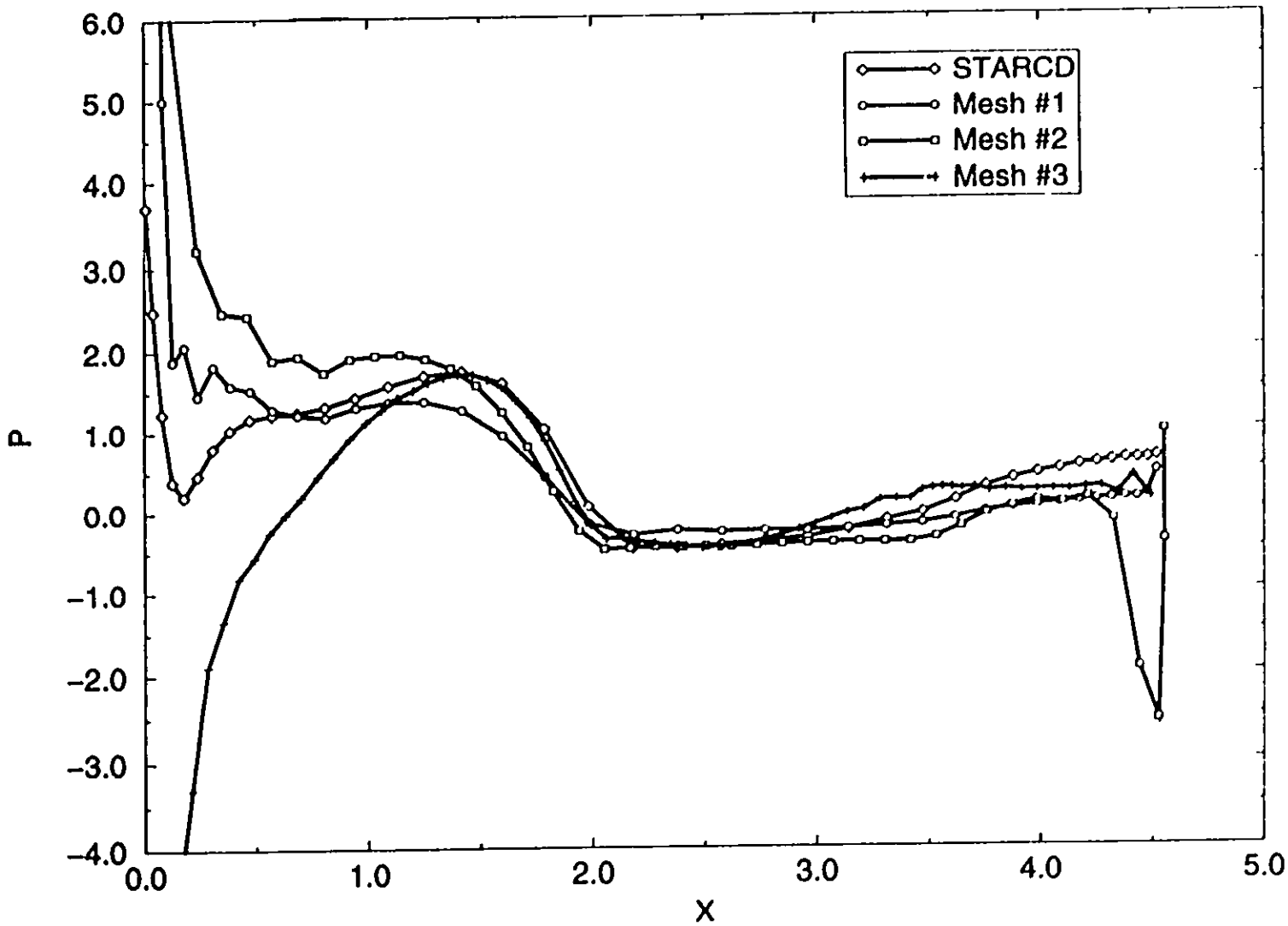


Figure 6.14: Pressure along top of the car, Re=45

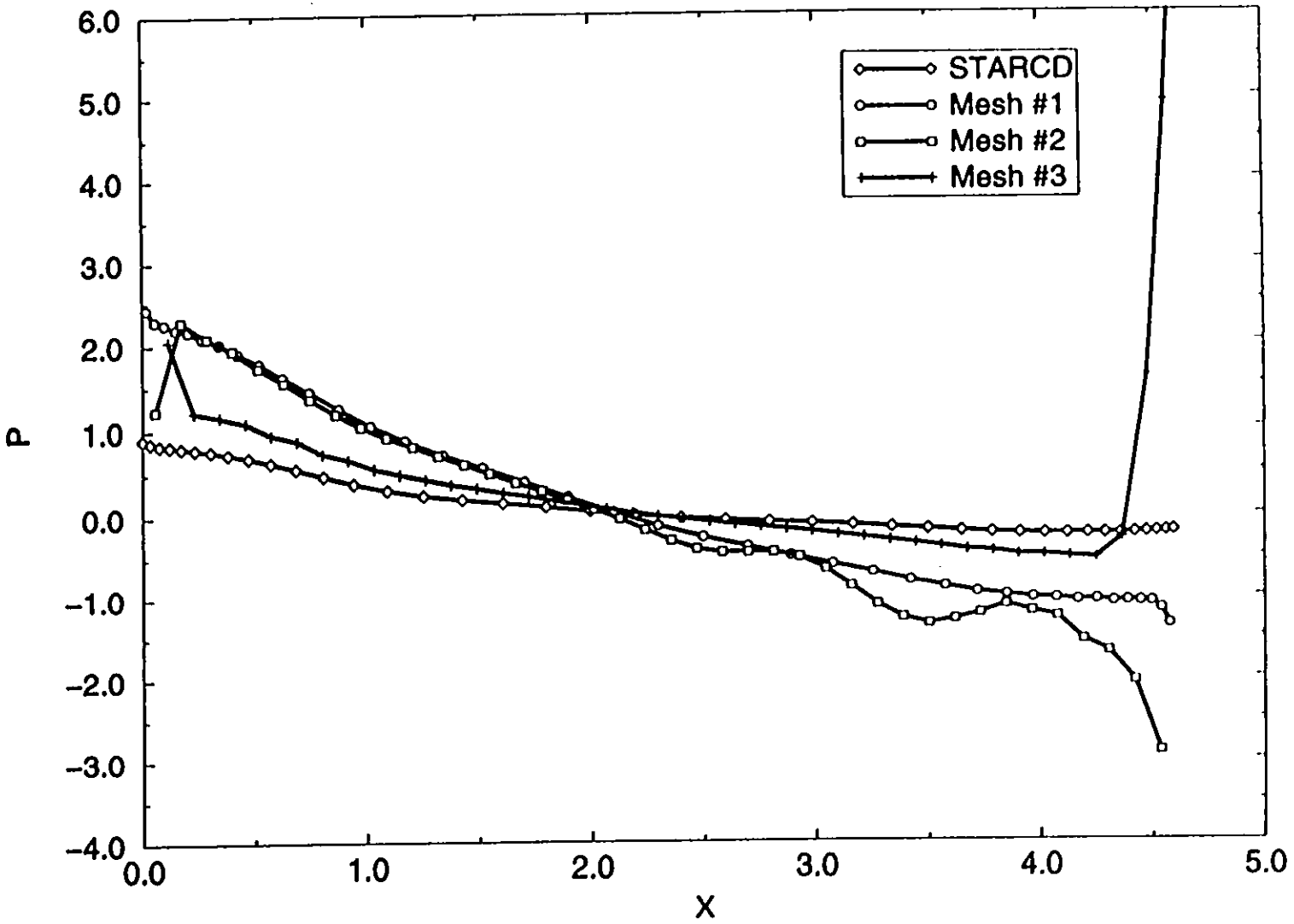


Figure 6.15: Pressure along bottom of the car, Re=45

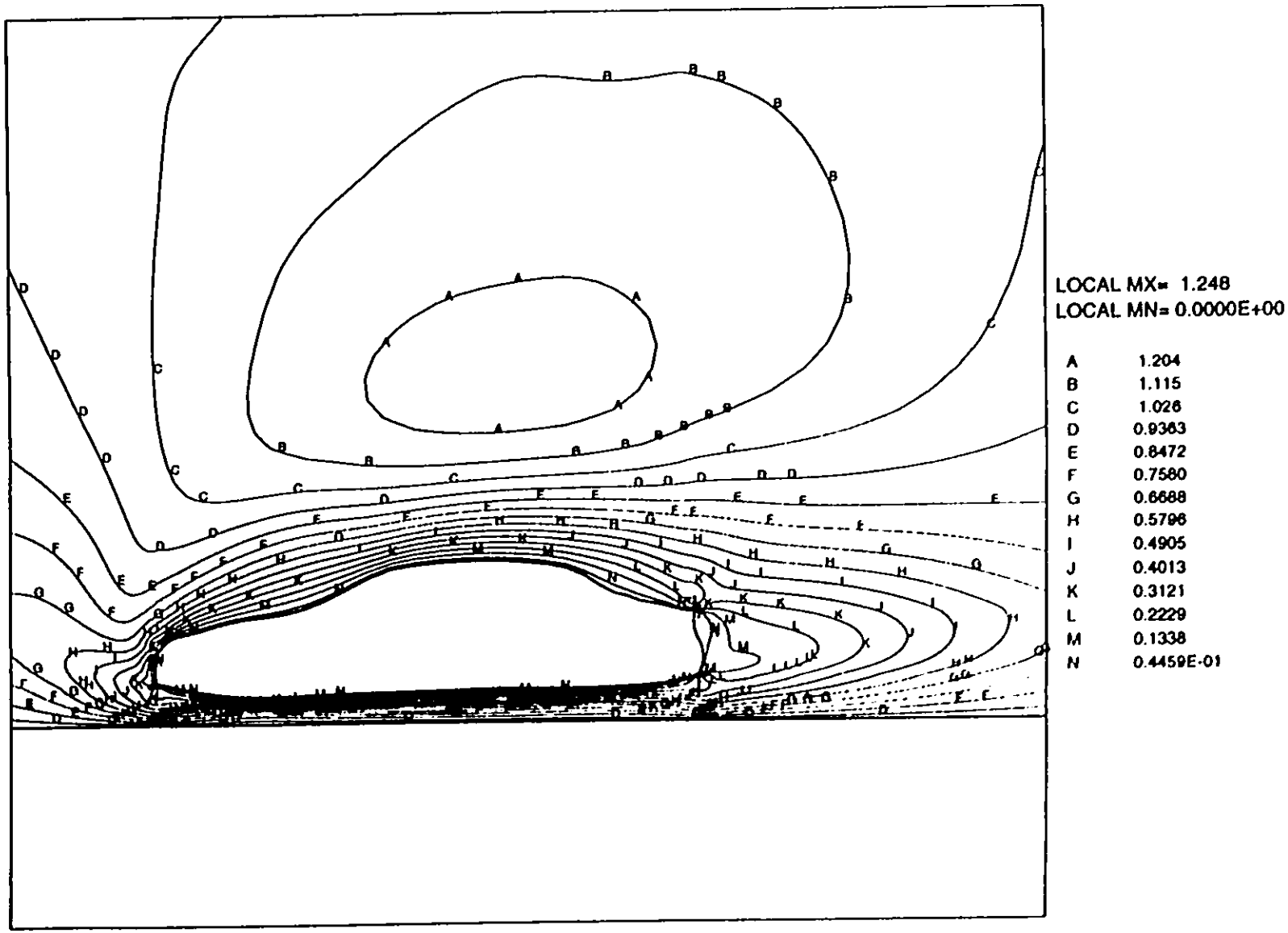


Figure 6.16: Velocity magnitude contours, mesh #1, Re=45

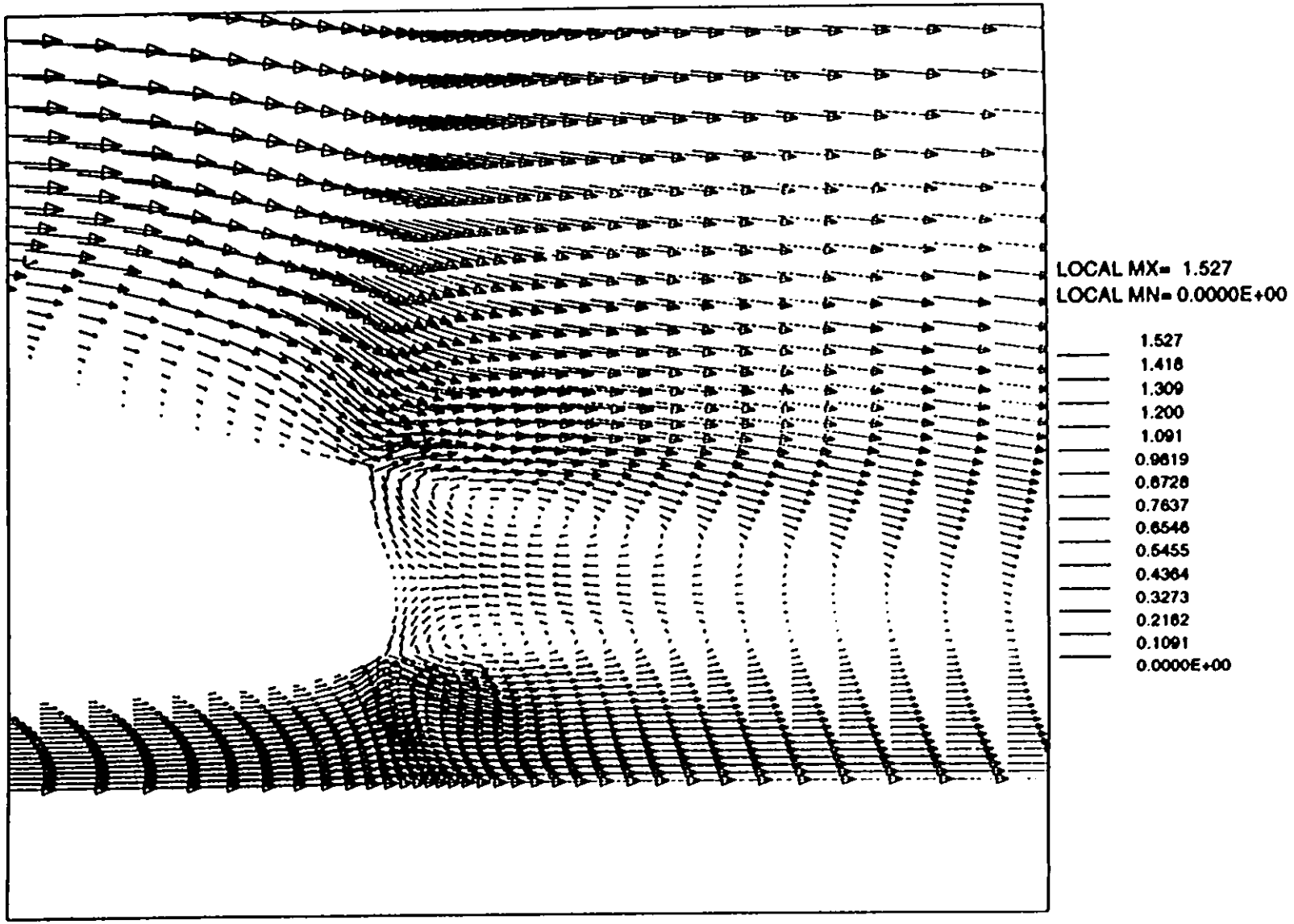


Figure 6.17: Velocity vector field behind the vehicle, mesh #1, Re=1800

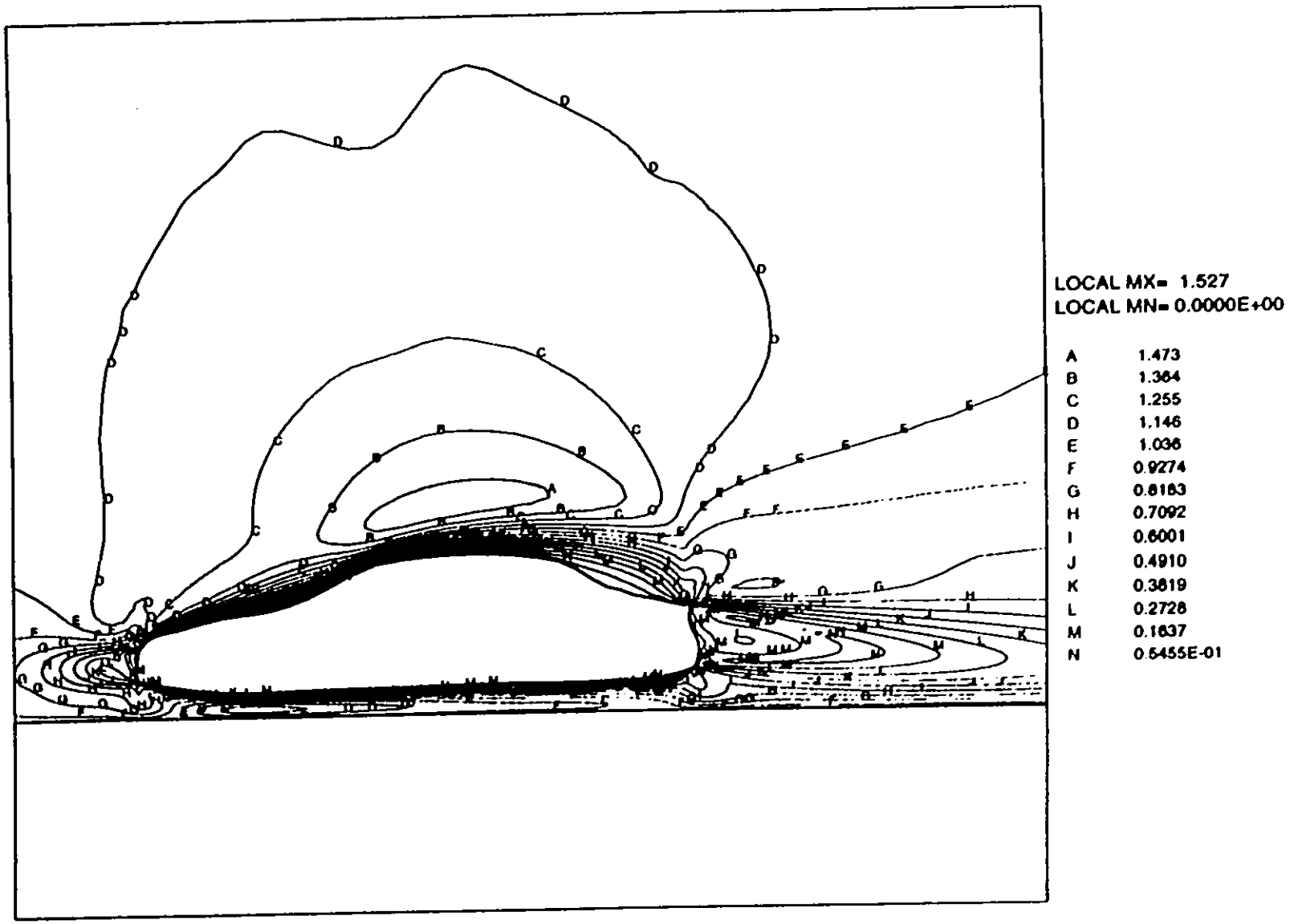


Figure 6.18: Velocity magnitude contours, mesh #1, Re=1800

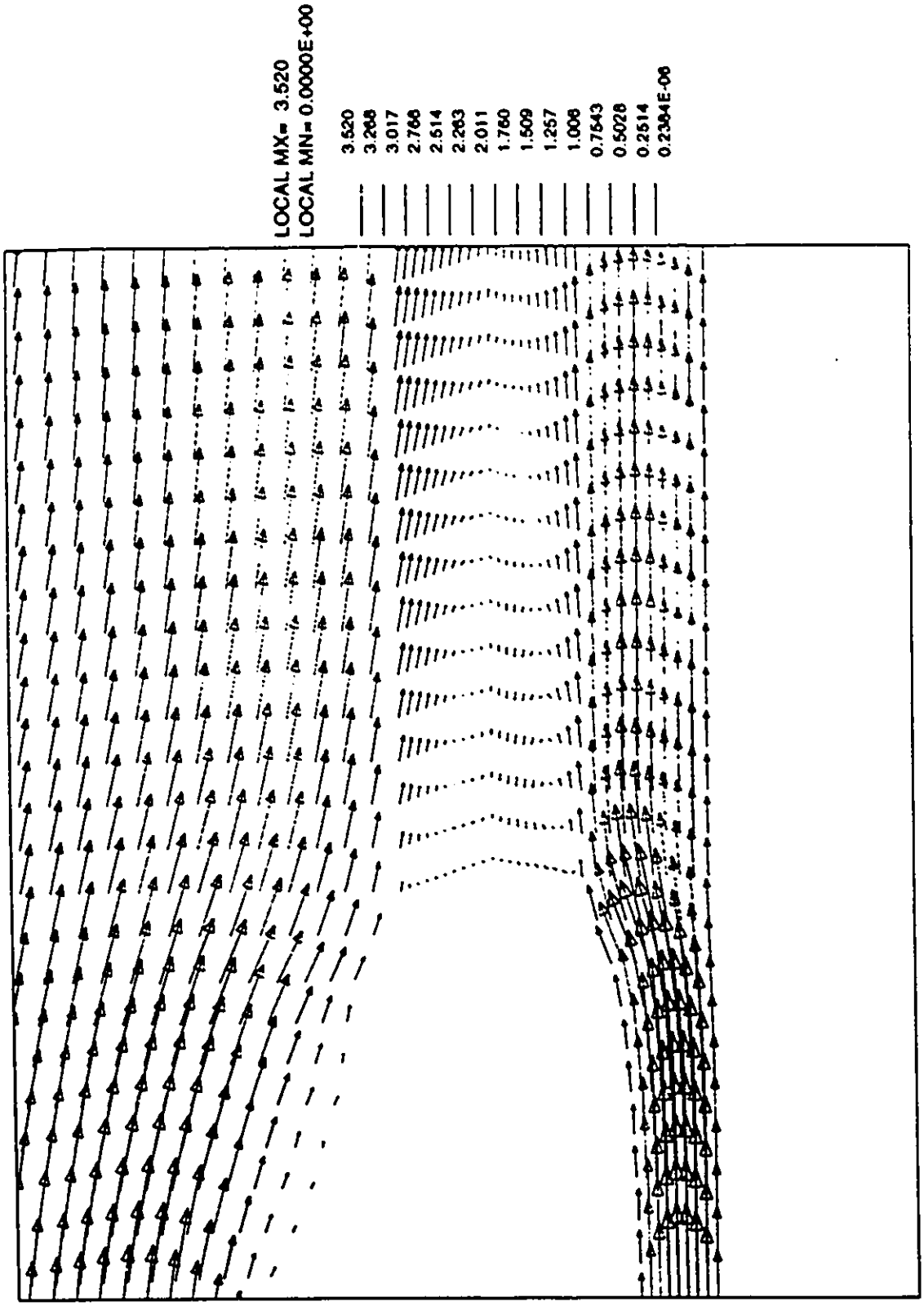


Figure 6.19: Velocity vector field behind the vehicle, mesh #2, Re=450

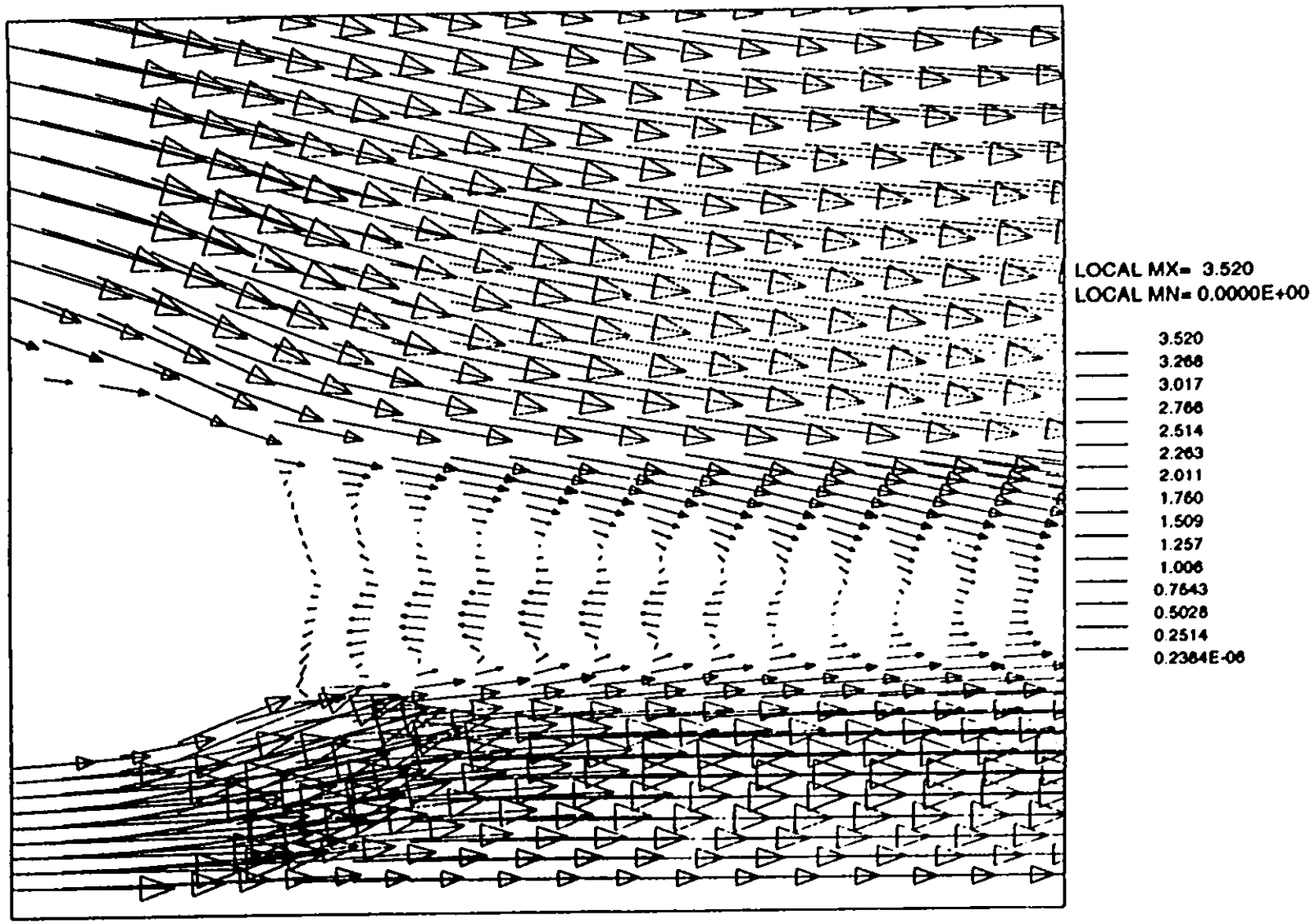


Figure 6.20: Close-up of velocity at rear of vehicle, mesh #2, Re=450

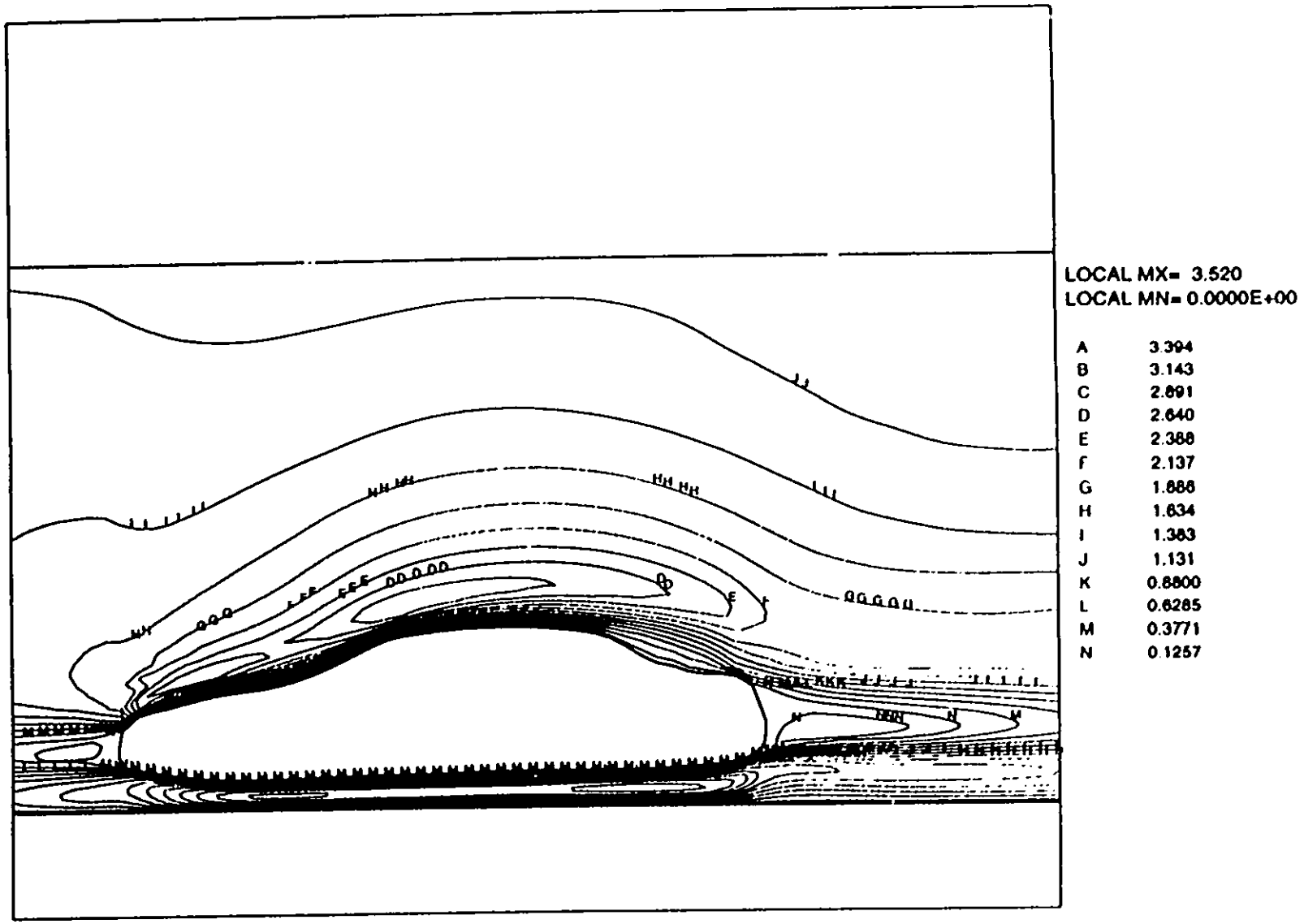


Figure 6.21: Velocity magnitude contours, mesh #2, Re=450

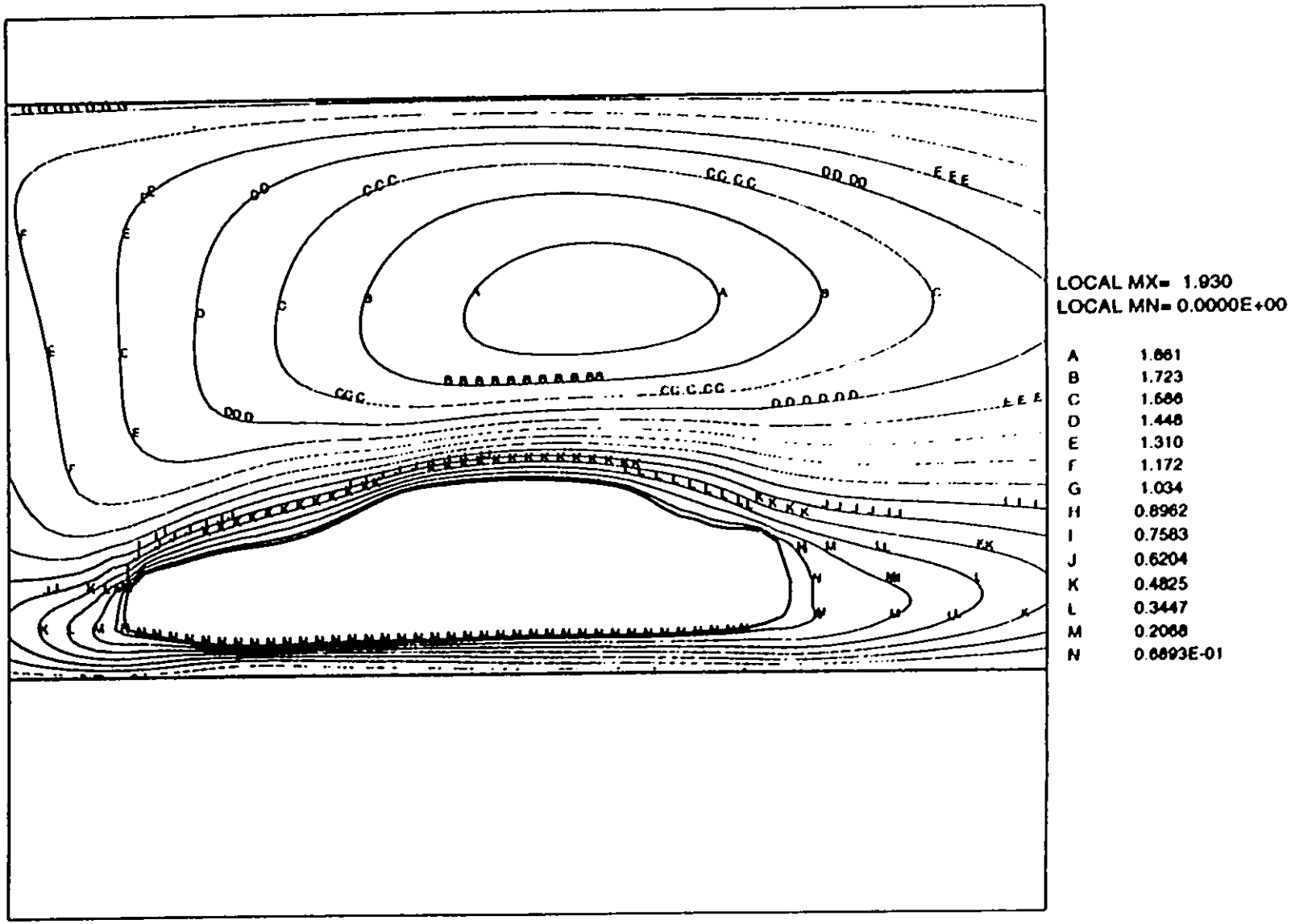


Figure 6.22: Velocity magnitude contours mesh #2, Re=45

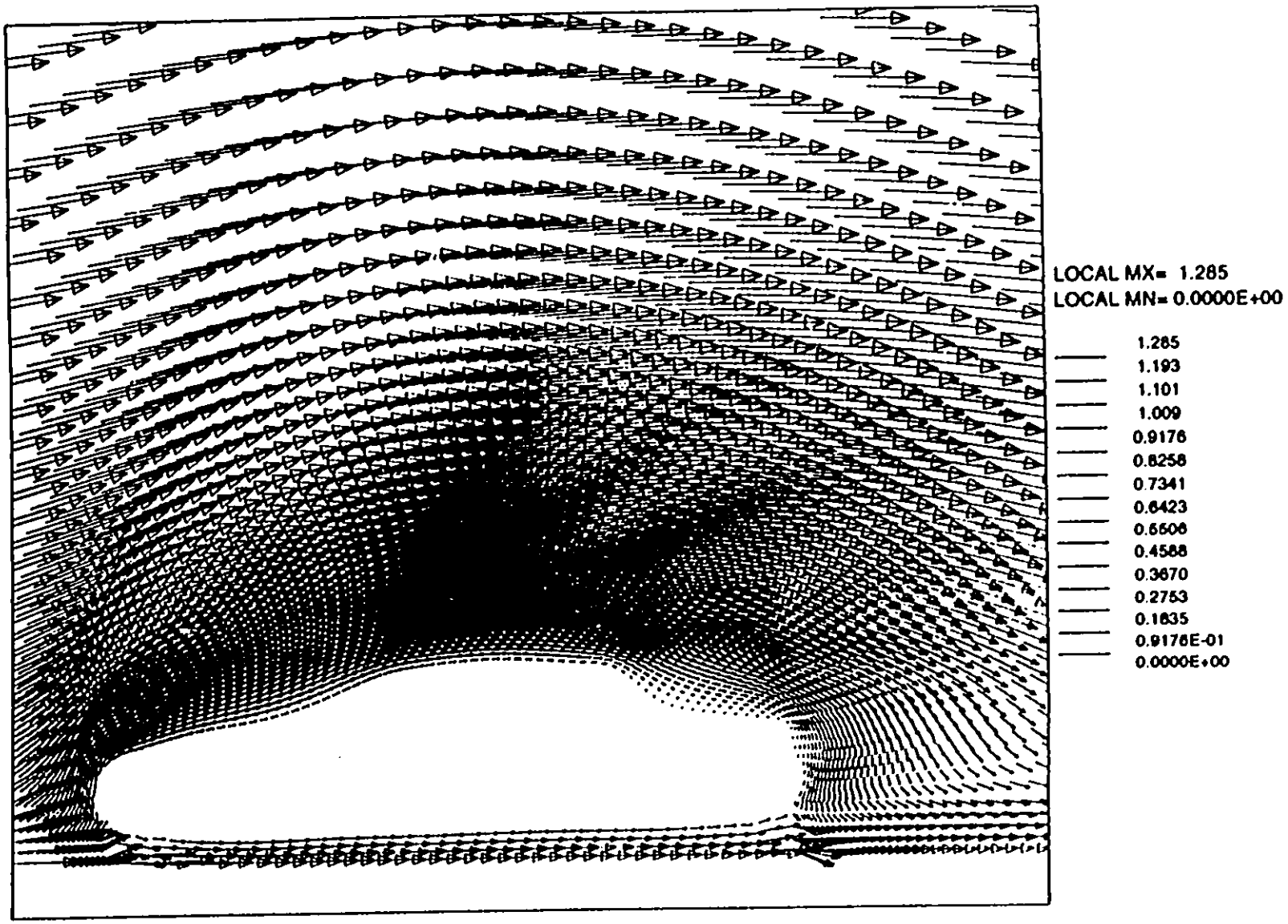


Figure 6.23: Velocity vector field near the vehicle, mesh #3, Re=45

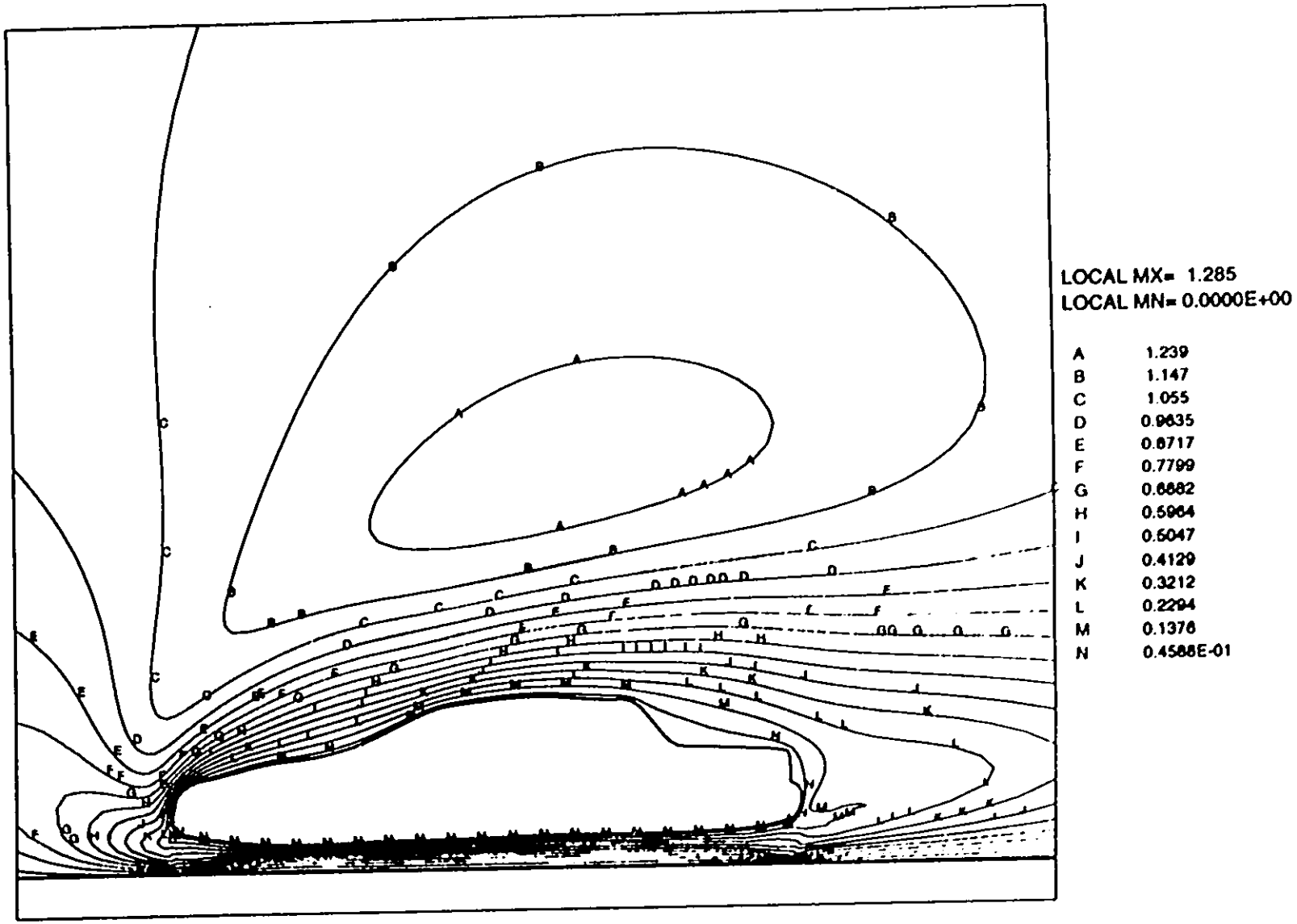


Figure 6.24: Velocity magnitude contours, mesh #3, Re=45

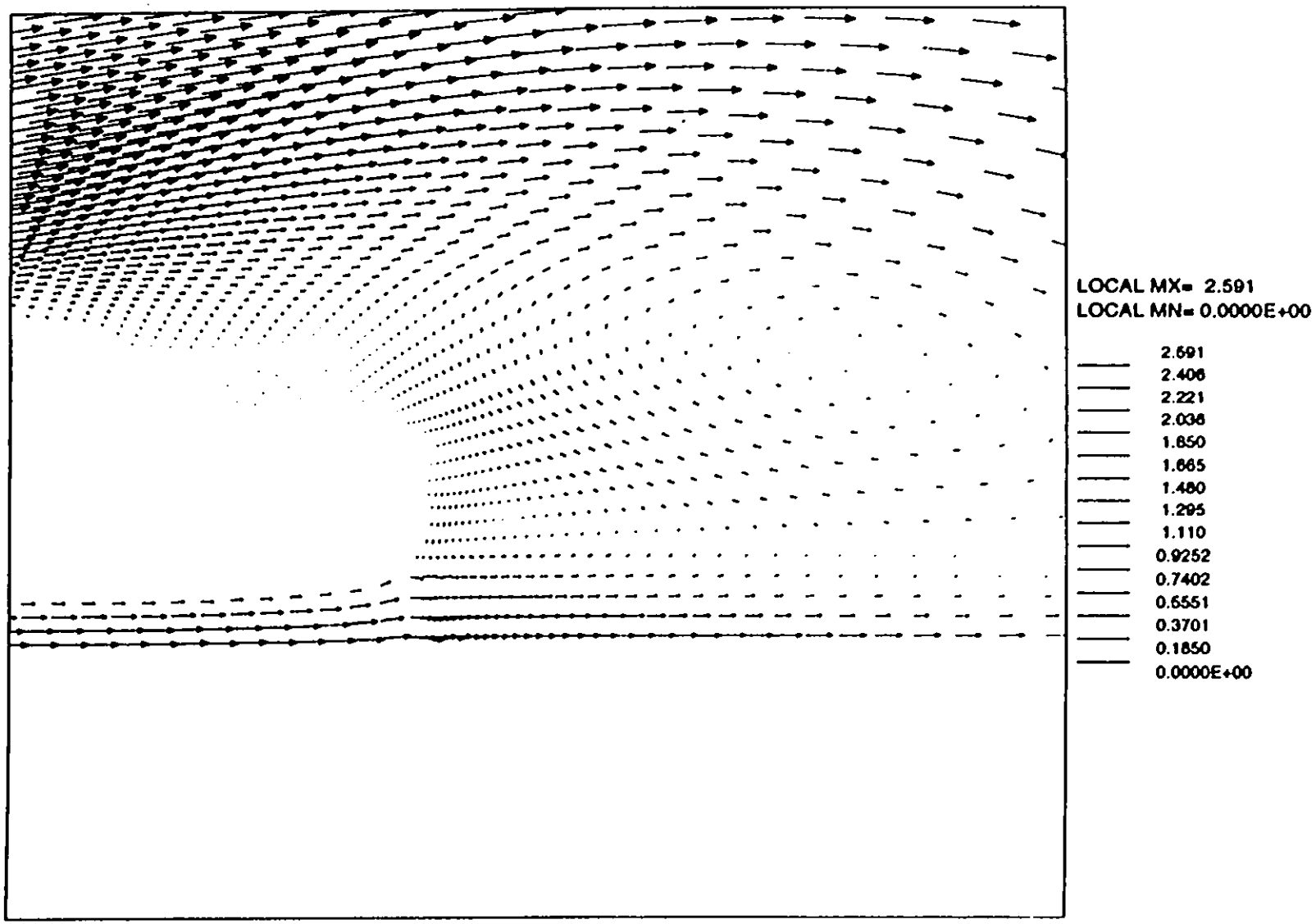


Figure 6.25: Velocity vector field behind the vehicle, mesh #3, Re=450

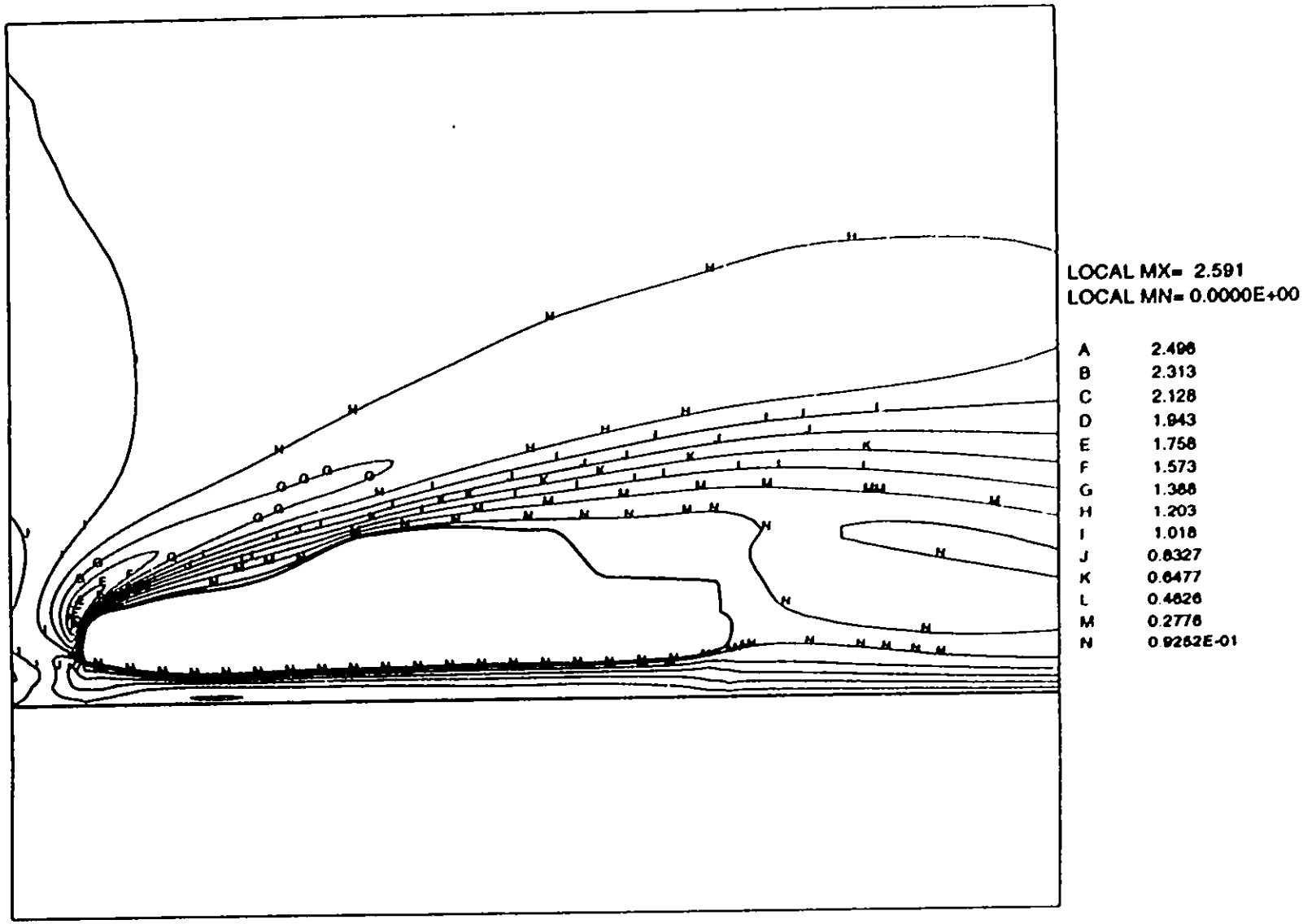


Figure 6.26: Velocity magnitude contours, mesh #3, Re=450

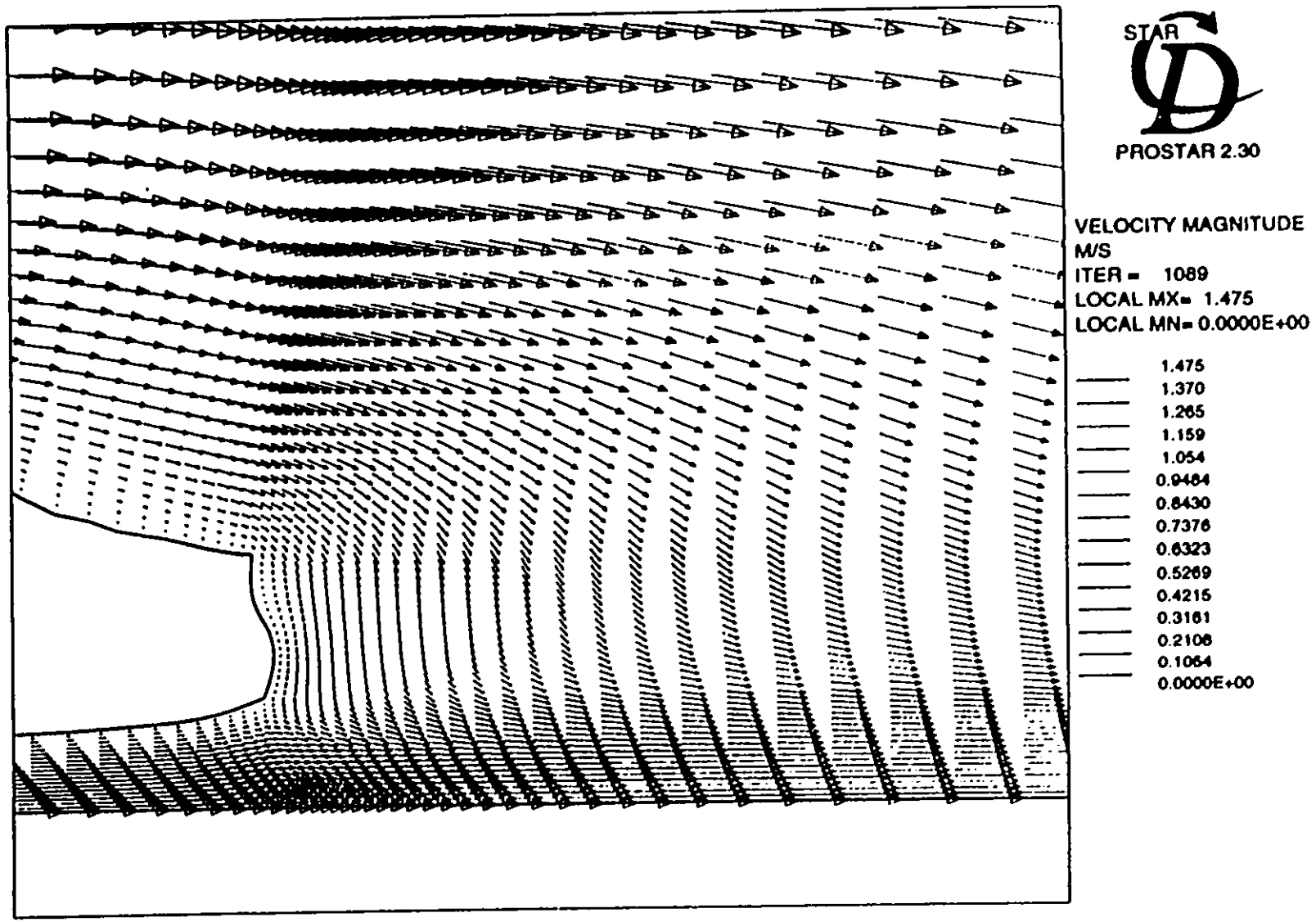


Figure 6.27: Velocity vector field behind the vehicle, STARCD, Re=45

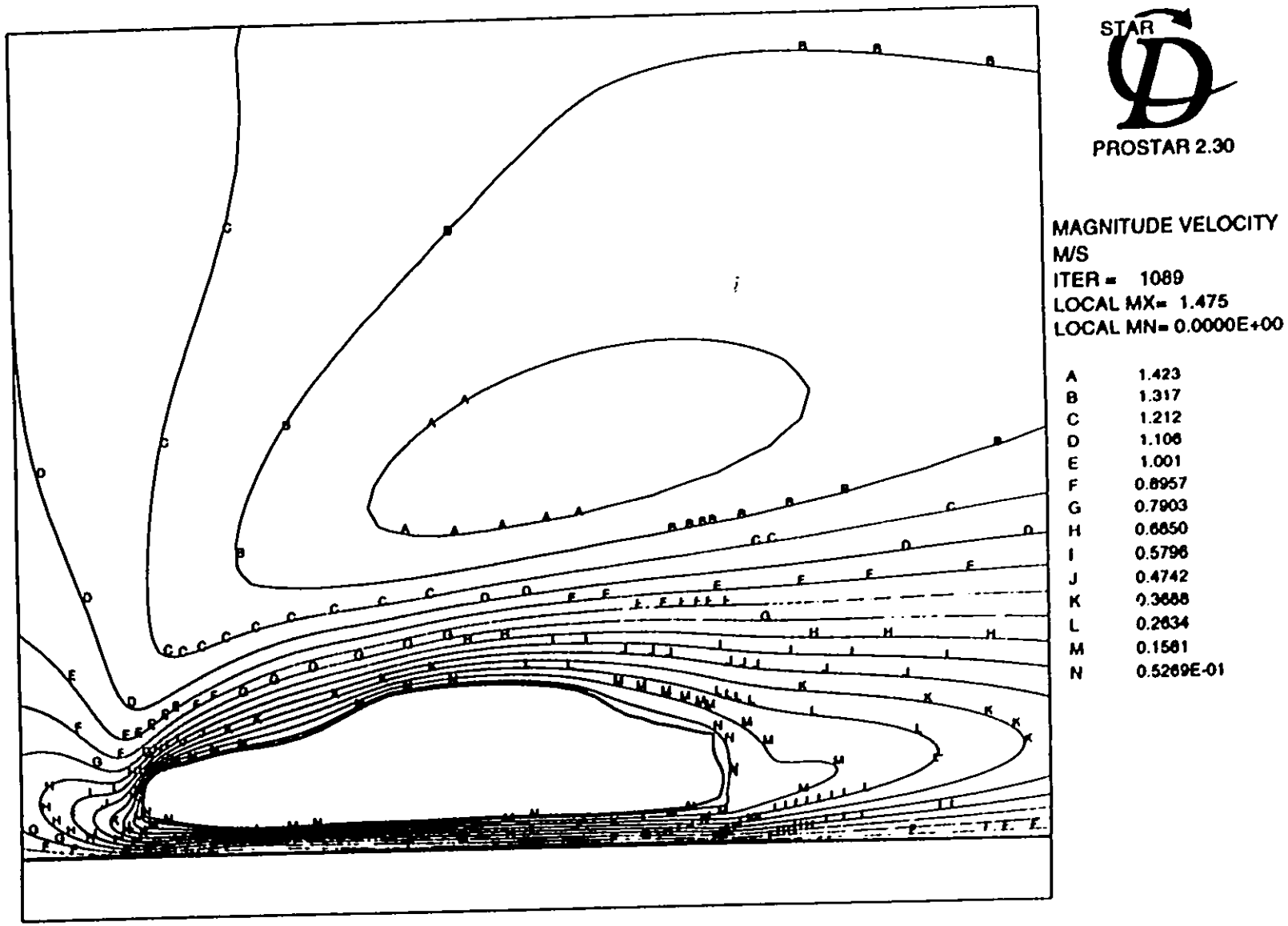


Figure 6.28: Velocity magnitude contours, STARCD, Re=45

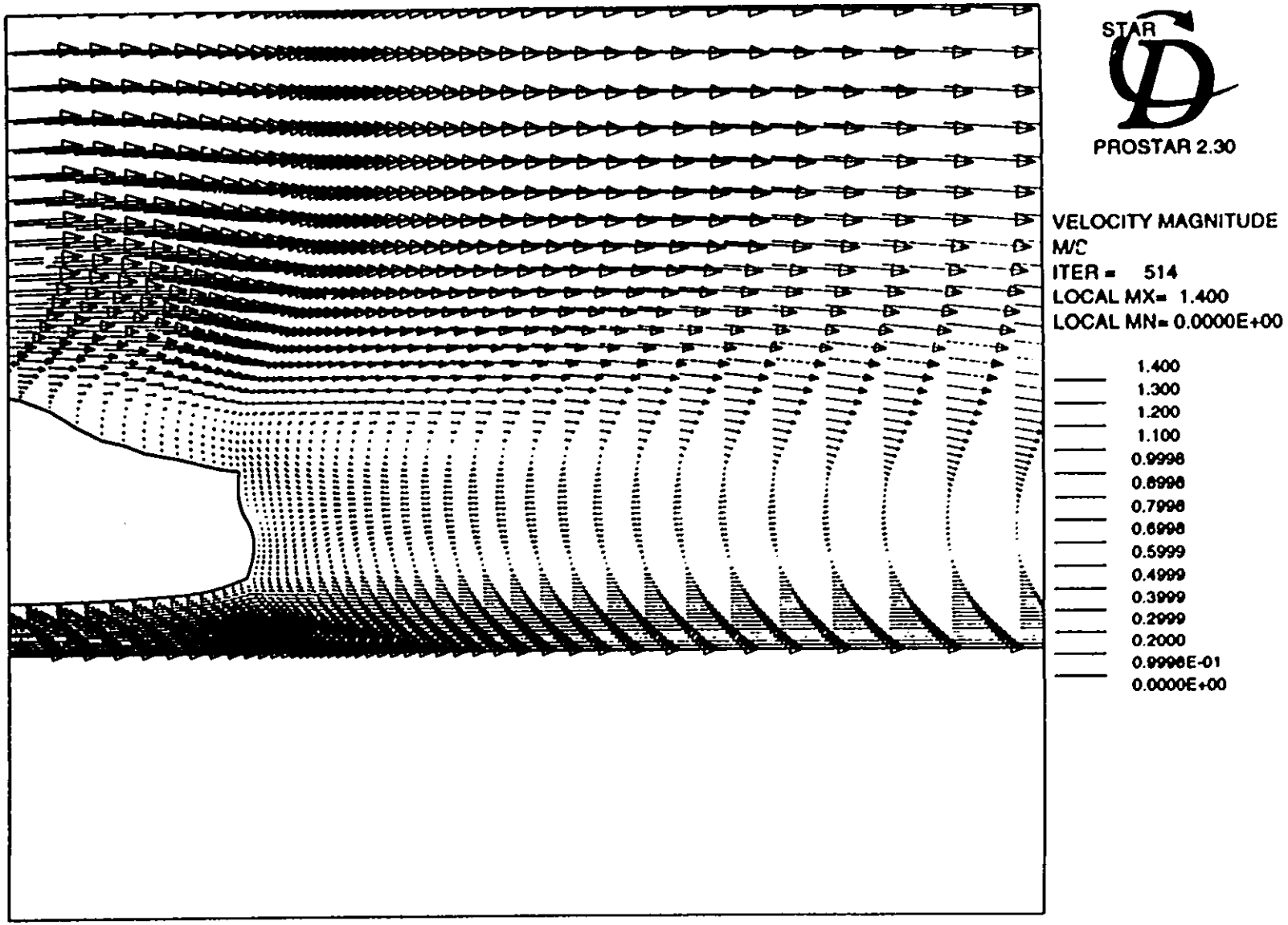


Figure 6.29: Velocity vector field behind the vehicle, STARCD, Re=450

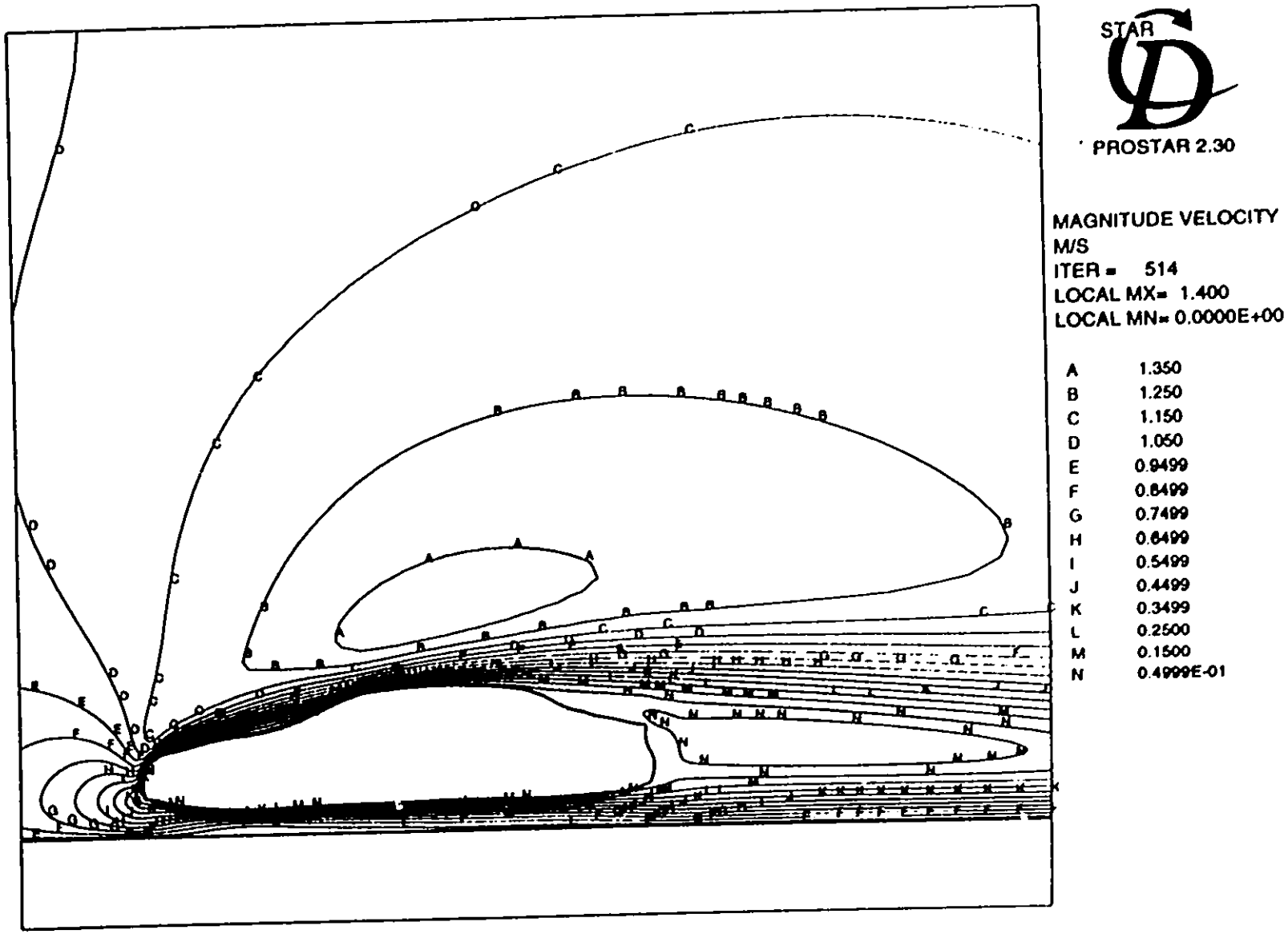


Figure 6.30: Velocity magnitude contours, STARCD, Re=450

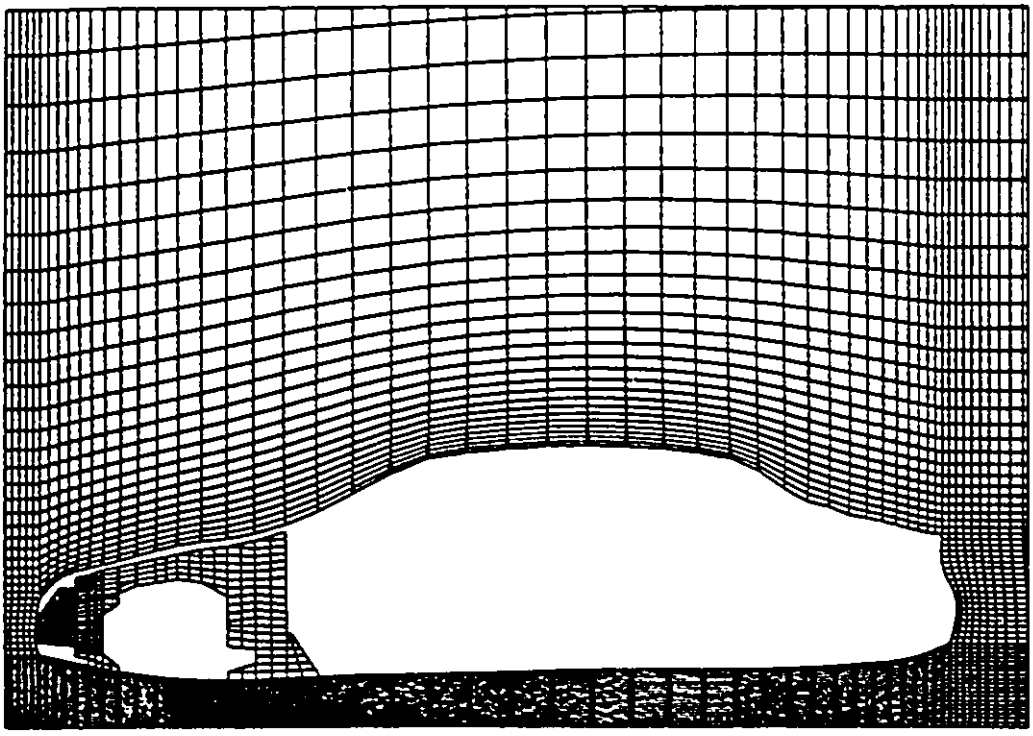

Figure 7.1: Mesh system near vehicle with engine compartment

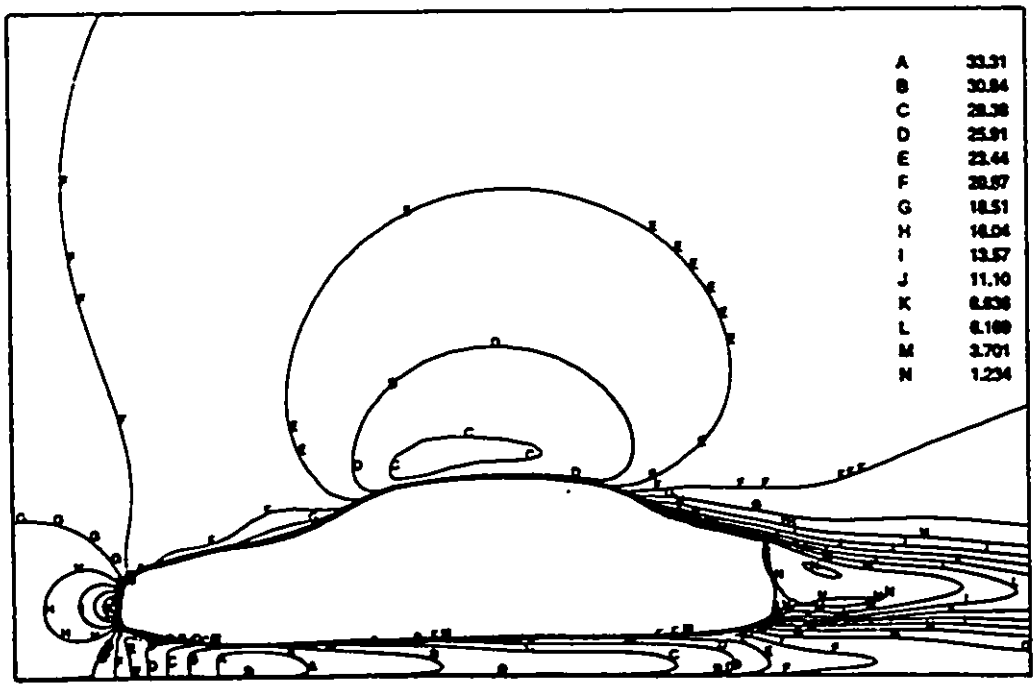


Figure 7.2: Velocity magnitude contours for vehicle without engine compartment

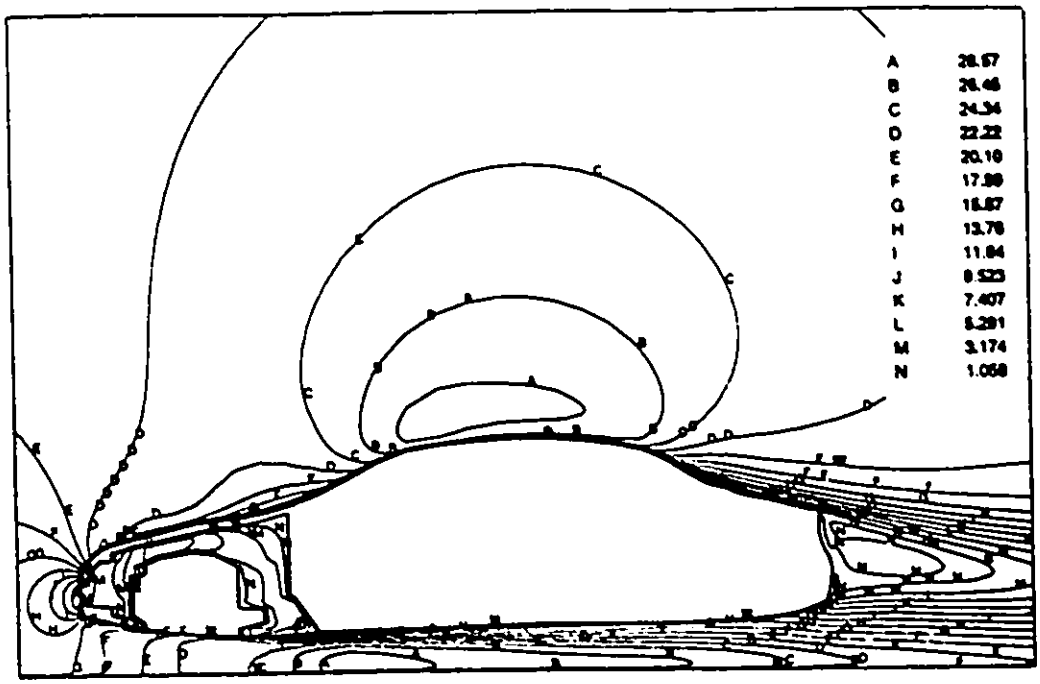


Figure 7.3: Velocity magnitude contours for vehicle with engine compartment with radiator

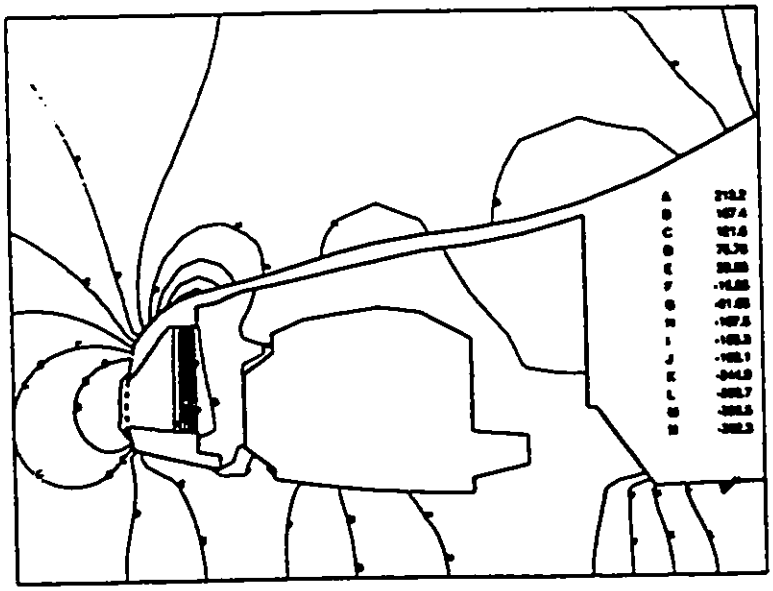

Figure 7.4: Pressure field at vehicle front end and underhood with radiator

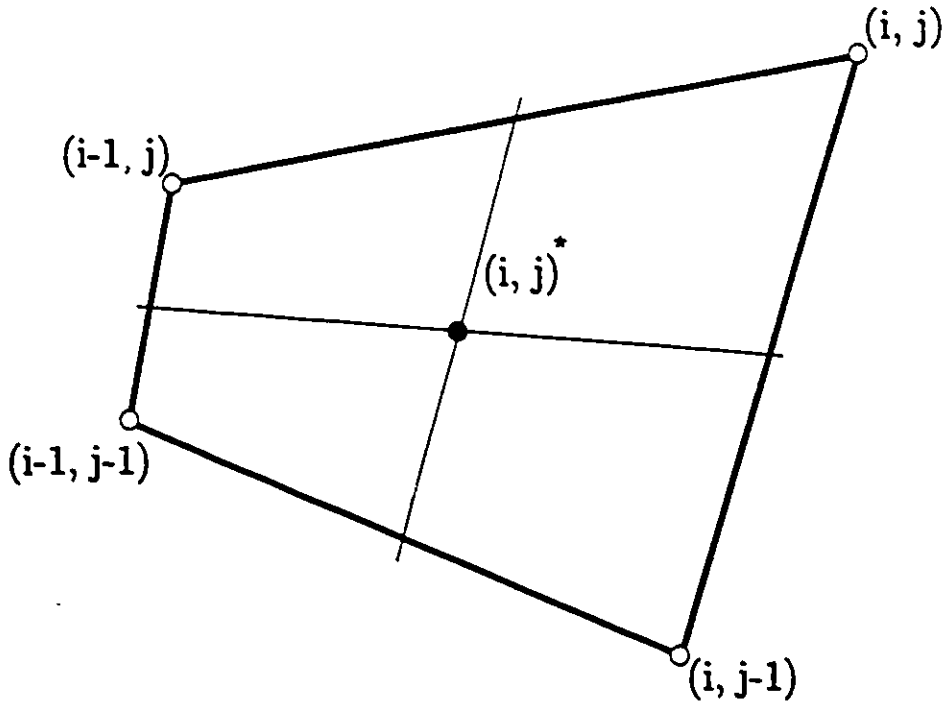


Figure 8.1: Location of cell center

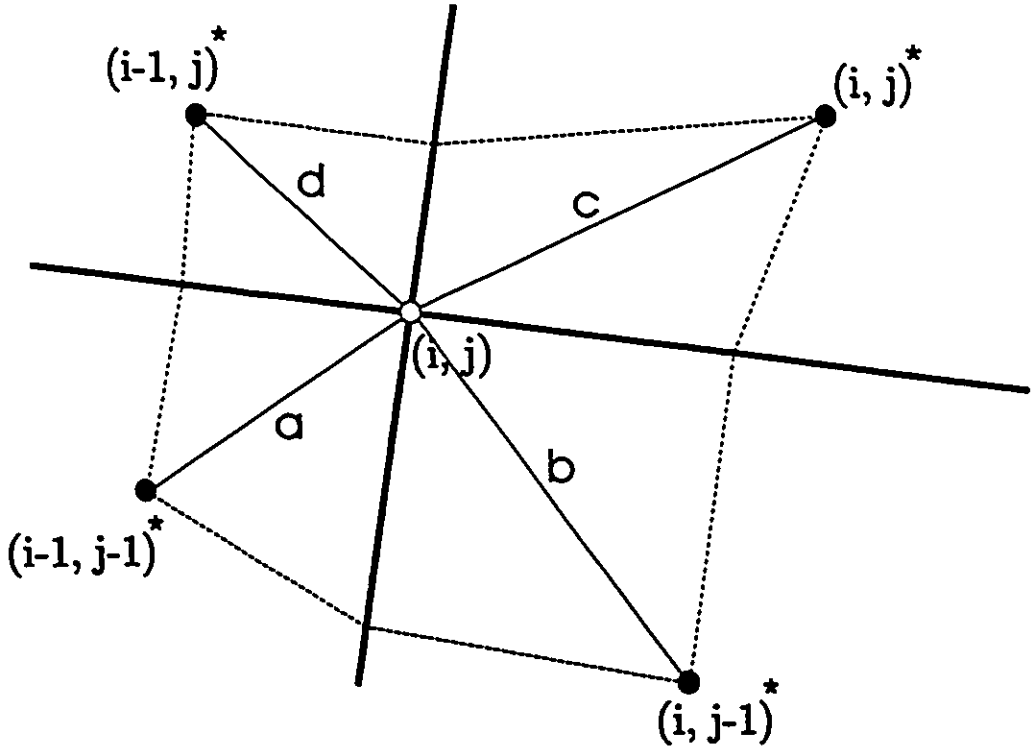


Figure 8.2: Arrangements for node averaging

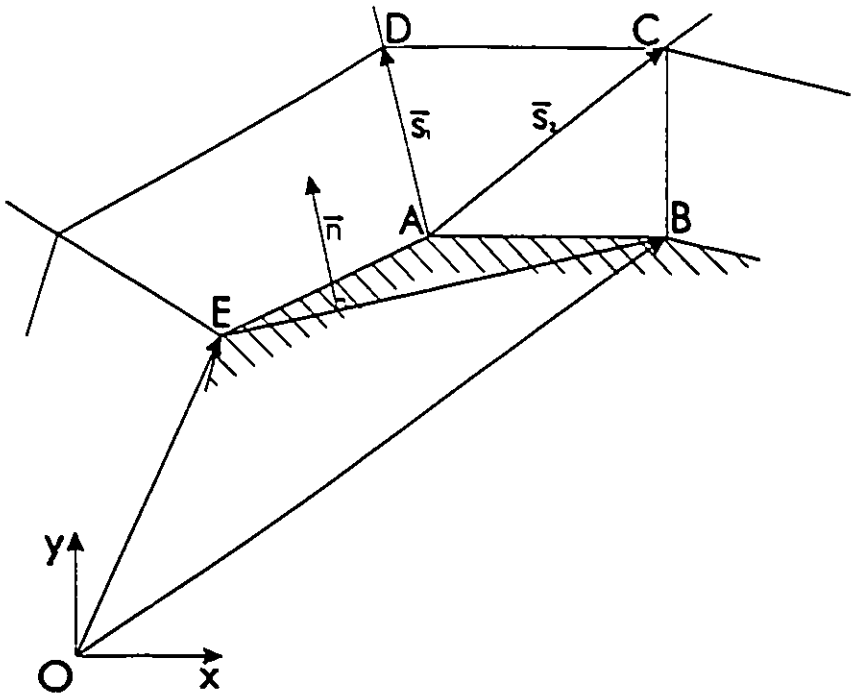


Figure 8.3: Nomenclature for determining normal pressure gradient

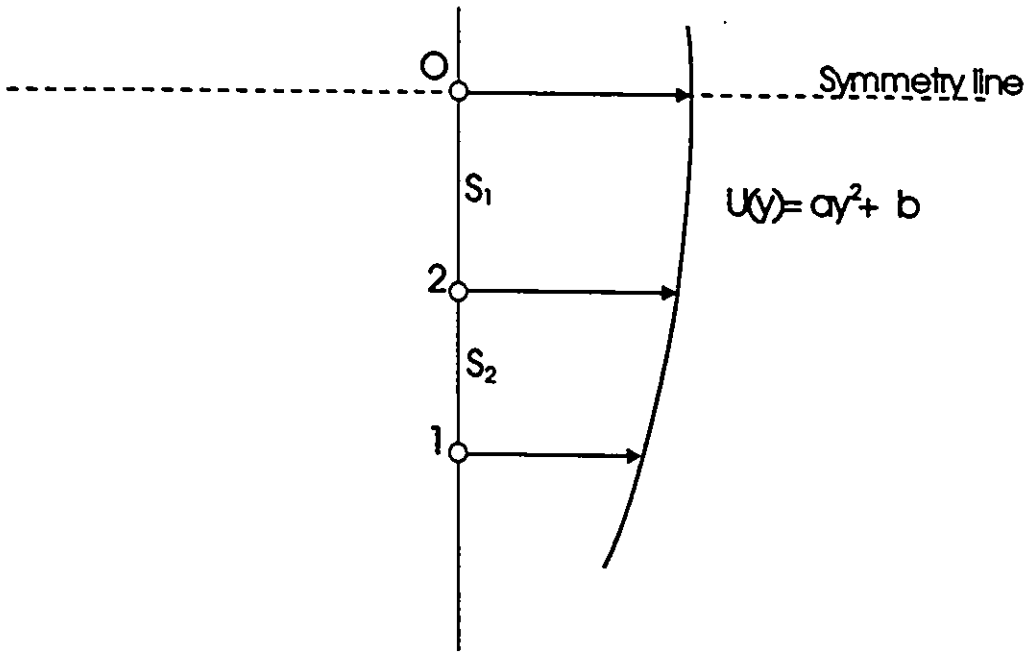


Figure 8.4: Extrapolation of U to the line of symmetry

# VITA AUCTORIS

Alexandre Y. Boglaev was born in 1972 in Moscow, Russia. He graduated from high school in June 1989. In September 1989, he went to Moscow Institute of Physics and Technology where he obtained a B.Sc. in Applied Math in 1993. He is now a student of the University of Windsor and hopes to obtain a Master's degree in Mathematics in Spring 1996.