

Project Mach-2D-5.8.1

Revision report

Guilherme Bertoldo

September 11, 2012

This document presents the revision and subversion of the code Mach-2D-5.8. This code aims to implement parallel computation in the code Mach-2D-5.7, which was a revision of code Mach-2D-5.6 written for Linux OS.

As a need to implement parallel computation, the principal subroutines of Mach-2D-5.7 were rewritten with explicit reference to input and output variables. All subroutines and comments were translated to english to avoid codification mismatch.

Revision 001

In the coefficient module, the subroutine GetMetrics was implemented where some metrics are calculated using means. Maliska (book, 2nd ed., page 221) suggests to avoid this practice. So this revision intend to correct this point. All coordinates are calculated by interpolation, then derivatives are applied to calculate the metrics.

Test01

Tests were done using

x = 10 csi + 20 eta + csi eta

y = 30 csi - 40 eta - csi eta

No differences between analytic and numerical solution were found for

alpha-e

gamma-n

beta-e

beta-n

Je

Jn

Jp

when centroids are calculated by simple mean. Differences appears if weighted mean is used.

Test02

Contains revision 007 of mach2d5.7-linux for further comparisons

Test03

Subroutine raios of Mach2D_5p7_coeficientes.f90 (rv007) was rewritten using the two methods.

The first method is the same as before and applies the central radius of two neighbours volumes to calculate the radius of the fictitious one by extrapolation.

The second method applies the central radius of one neighbour and the radius at the face on the boundary between the real neighbour and the fictitious volume to calculate the radius of the fictitious volume by extrapolation.

No differences was found when the same method was applied, indicating no programming errors.

No differences was found up to the 10th figure in mass flow rate (fme) and drag (Fd) when the second method was applied.

Test04

Subroutine metricas of Mach2D_5p7_coeficientes.f90 (rv007) was rewritten. In the metricas subroutine some

metrics on the boundary are calculated by interpolation. Maliska (book, 2nd ed. page 221) suggests to avoid this practice.

The new subroutine GetMetrics was verified by comparing its results with analytic results.

No differences was caused by the new subroutine in the mass flow rate (fme) and drag (Fd) up to the eighth figure.

OBS.: It is necessary to study the influence of the metrics calculation on the final solution.

Test05

Subroutine x_y_nodal of Mach2D_5p7_coeficientes.f90 (rv007) was rewritten using two methods. The first method calculates the coordinate of the centroid using a simple mean of the coordinates of the corners of the volume. The second one divides the volume in two triangles, calculates the area of each triangle and their centroids, then calculates the centroid of the volume by a weighted mean based on the triangles area.

No differences in the results were found for the first method. This suggests no programming errors.

For the second method, differences appeared in the fifth figure of mass flow rate (fme) and drag (Fd).

Test06

In the Test01 it was found differences in the results of metrics when centroids are calculated by weighted mean. In order to search for errors, the centroids given by weighted mean was compared with some results obtained analytically in TM702 classes. No differences were found.

Revision 002

Subroutines massa_especifica_nos and massa_especifica_faces of Mach2D_5p7_coeficientes.f90 were rewritten. The new subroutines are called GetDensityAtNodes and GetDensityAtFaces, respectively, and are stored in module coefficients of mach2d-coef.f90.

Test01

Contains the files of revision 007 of Mach2D 5.7. Used only for comparison with modified subroutines

Test02

Contains the files of revision 007 of Mach2D 5.7 and mach2d-coef.f90. The new subroutines GetDensityAtNodes and GetDensityAtFaces are used instead of massa_especifica_nos and massa_especifica_faces. No differences were found.

Revision 003

Subroutines coeficientes_velocidades and fonte_velocidade_u of Mach2D_5p7_coeficientes.f90 were rewritten. The new subroutines are called GetUxCoefficients and GetUxSource, respectively, and are stored in module coefficients of mach2d-coef.f90.

Test01

Contains the files of revision 007 of Mach2D 5.7. Used only for comparison with modified subroutines

Test02

Contains the files of revision 007 of Mach2D 5.7 and mach2d-coef.f90. The new subroutines GetUxCoefficients and GetUxSource are used instead of coeficientes_velocidades and fonte_velocidade_u. Differences in results appeared due rounding errors.

Revision 004

Subroutine fonte_velocidade_v of Mach2D_5p7_coeficientes.f90 was rewritten. Two new subroutines were introduced. GetUyCoefficients is new and GetUySource replaces fonte_velocidade_v. Both subroutines are stored in module coefficients of mach2d-coef.f90.

Test01

Contains the files of revision 007 of Mach2D 5.7. Used only for comparison with modified subroutines

Test02

Contains the files of revision 007 of Mach2D 5.7 and mach2d-coef.f90. The new subroutines
GetUyCoefficients and GetUySource

are used. The last one replaces fonte_velocidade_v. Differences in results appeared due rounding errors.

Revision 005

Subroutine coeficientes_fonte_temperatura of Mach2D_5p7_coeficientes.f90 was rewritten. The new subroutine GetTemperatureCoefficientsAndSource is stored in mach2d-coef.f90.

Test01

Contains the files of revision 007 of Mach2D 5.7. Used only for comparison with modified subroutines

Test02

Contains the files of revision 007 of Mach2D 5.7 and mach2d-coef.f90. The new subroutines
GetTemperatureCoefficientsAndSource is used.

The last one replaces coeficientes_fonte_temperatura. Differences in results appeared due rounding errors.

Revision 006

Subroutine coeficientes_fonte_pressao of Mach2D_5p7_coeficientes.f90 was rewritten. The new subroutine GetPressureCoefficientsAndSource is stored in mach2d-coef.f90.

Test01

Contains the files of revision 007 of Mach2D 5.7. Used only for comparison with modified subroutines

Test02

Contains the files of revision 007 of Mach2D 5.7 and mach2d-coef.f90. The new subroutine
GetPressureCoefficientsAndSource is used.

The last one replaces coeficientes_fonte_pressao. Differences in results appeared due rounding errors.

Revision 007

Subroutine velocidades_contravariantes_faces and coeficientes_SIMPLEC of Mach2D_5p7_coeficientes.f90 were rewritten. The new subroutines GetContravariantVelocityAtFaces and GetInternalSimplecCoefficients are stored in mach2d-coef.f90.

Modifications in velocidades_contravariantes_faces: in the subroutine velocidades_contravariantes_faces of Mach2D_5p7_coeficientes.f90, it was not attributed values to variables npsw and npse in the second loop. This was corrected in the revision 008 of Mach2D 5.7.

In subroutine velocidades_contravariantes_faces, the SIMPLEC coefficients due, dve, dun and dvn are calculated. In the new subroutine GetContravariantVelocityAtFaces this task is left to subroutine GetInternalSimplecCoefficients.

Test01

Contains the files of revision 008 of Mach2D 5.7. Used only for comparison with modified subroutines.

Test02

Contains the files of revision 008 of Mach2D 5.7 and mach2d-coef.f90. The new subroutines
GetContravariantVelocityAtFaces

and GetInternalSimplecCoefficients are used instead of velocidades_contravariantes_faces and
coeficientes_SIMPLEC.

Differences in results appeared due rounding errors.

OBS.: GetContravariantVelocityAtFaces needs cartesian velocities at corners.

Revision 008

Subroutine `pressao_e_massa_especifica_com_pl`, `u_v_nos_reais_com_pl` and `Uce_Vcn_faces_internas_com_pl` of `Mach2D_5p7_coeficientes.f90` were rewritten. The new subroutines `get_pressure_density_correction_with_pl`, `get_u_v_at_real_nodes_with_pl` and `get_velocities_at_internal_faces_with_pl` are stored in `mach2d-coef.f90`.

Test01

Contains the files of revision 008 of Mach2D 5.7. Used only for comparison with modified subroutines.

Test02

Contains the files of revision 008 of Mach2D 5.7 and `mach2d-coef.f90`. The new subroutines `get_pressure_density_correction_with_pl`, `get_u_v_at_real_nodes_with_pl` and `get_velocities_at_internal_faces_with_pl` are used instead of `pressao_e_massa_especifica_com_pl`, `u_v_nos_reais_com_pl` and `Uce_Vcn_faces_internas_com_pl`.

There are two ways of update density in `get_pressure_density_correction_with_pl`:

`ro = ro + pl * g`

or

`ro = p * g`

where $g = 1/(Rg * T)$.

If the first way is used in subroutine `get_pressure_density_correction_with_pl`, the differences in final result, relatively to `pressao_e_massa_especifica_com_pl`, arise from the 10th decimal place. On the other hand, if the second way is applied, only rounding errors are found. The second way was used.

When subroutine `get_u_v_at_real_nodes_with_pl` is used instead of `u_v_nos_reais_com_pl` only rounding errors were found.

When `get_velocities_at_internal_faces_with_pl` is applied instead of `Uce_Vcn_faces_internas_com_pl` no differences were found.

Revision 009

In order to clarify the usage of subroutines, some of them were renamed.

MODULE COEFFICIENTS:

OLD NAMES:

- ! 1) `GetRealCentroidsXY`
- ! 2) `GetMetrics`
- ! 3) `GetRadius`
- ! 4) `GetDensityAtNodes`
- ! 5) `GetDensityAtFaces`
- ! 6) `GetUxCoefficients`
- ! 7) `GetUxSource`
- ! 8) `GetUyCoefficients`
- ! 9) `GetUySource`
- ! 10) `GetTemperatureCoefficientsAndSource`
- ! 11) `GetPressureCoefficientsAndSource`
- ! 12) `GetContravariantVelocityAtFaces`
- ! 13) `GetInternalSimplecCoefficients`
- ! 14) `get_pressure_density_correction_with_pl`
- ! 15) `get_u_v_at_real_nodes_with_pl`
- ! 16) `get_velocities_at_internal_faces_with_pl`

NEW NAMES:

- ! 1) `get_real_centroids_xy`
- ! 2) `get_metrics`
- ! 3) `get_radius`
- ! 4) `get_density_at_nodes`
- ! 5) `get_density_at_faces`
- ! 6) `get_u_coefficients`
- ! 7) `get_u_source`

```
! 8) get_v_coefficients
! 9) get_v_source
! 10) get_T_coefficients_and_source
! 11) get_p_coefficients_and_source
! 12) get_velocities_at_faces
! 13) get_internal_simplec_coefficients
! 14) get_pressure_density_correction_with_pl
! 15) get_u_v_at_real_nodes_with_pl
! 16) get_velocities_at_internal_faces_with_pl
```

MODULE COEFFICIENTS HAS BEEN FINISHED!

MODULE DATA:

OLD NAMES:

```
! 1) GetParameters
! 2) AllocateVariables
! 3) InitializeVariables
```

NEW NAMES:

```
! 1) get_parameters
! 2) allocate_variables
! 3) initialize_variables
```

MODULE GRID:

OLD NAMES:

```
! 1) GridGen
! 2) UniformGrid
! 3) BackwardGPGrid
! 4) GetGPRatio
```

NEW NAMES:

```
! 1) set_grid
! 2) get_uniform_grid
! 3) get_backward_GP_grid
! 4) get_GP_ratio
```

Revision 010

Subroutines of module usuario of Mach2D_5p7_usuario.f90 were rewritten.

Test01

Contains the files of revision 008 of Mach2D 5.7. Used only for comparison with modified subroutines.

Test02

temperatura_da_parede replaced by set_wall_temperature

calor_especifico_e_gama replaced by set_cp_and_gamma

viscosidade_laminar_nos replaced by set_laminar_viscosity_at_nodes

viscosidade_laminar_faces replaced by get_laminar_viscosity_at_faces

condutividade_termica_nos replaced by set_thermal_conductivity_at_nodes

condutividade_termica_faces replaced by get_thermal_conductivity_at_faces

No changes in the results was observed due the replacement of subroutines.
No programming errors.

Revision 011

Test01

Contains the files of revision 008 of Mach2D 5.7. Used only for comparison with modified subroutines.

Test02

Subroutines of module usuario of Mach2D_5p7_usuario.f90 were rewritten.

ccu replaced by set_bcu

ccv replaced by set_bcv

ccp replaced by set_bcp

ccT replaced by set_bcT

In order to attend to the observation of revision 007, coefficients of the linear system for u and v are calculated in such a way that u and v are calculated at fictitious corners by extrapolation.

No changes in the results was observed due the replacement of subroutines.

No programming errors.

Revision 012

Test01

Contains the files of revision 008 of Mach2D 5.7. Used only for comparison with modified subroutines.

Test02

Subroutines of module usuario of Mach2D_5p7_usuario.f90 were rewritten.

Tin_pin_entrada	replaced by	get_uin_vin_pin_Tin_Mw
plin_entrada_p_ficticio	replaced by	get_plin_and_p_fictitious
u_v_nos_ficticios_com_pl	replaced by	get_u_v_at_fictitious_nodes_with_pl
Uce_Vcn_de_dn_faces_contornos	replaced by	get_Uce_Vcn_de_dn_at_boundary_faces
Uce_Vcn_faces_contornos_com_pl	replaced by	get_Uce_Vcn_at_boundary_faces_with_pl
EXATO_MACH_fluxo_massa	replaced by	get_isentropic_mass_flow
EXATO_MACH_newton	replaced by	get_mach_area
estimativa_inicial	replaced by	get_initial_guess

Only rounding errors were found.

Revision 013

module user:

get_mach_area - a conditional was inserted to ensure that when the area ratio = 1 then Mach = 1.

get_inital_guess - an error in the index of a vector was corrected.

module solvers:

Inserted subroutines

norm_l1_5d

norm_l1_9d

Comparing these suboutines with

norma_l1_5d

norma_l1_9d

of Mach2D5p6, only rounding errors were found.

At this revision the same results for mass flow rate and dynamic thrust of rv009 of Mach2D 5.7 were obtained provided metrics are calculated with get_metrics (metricas and get_metrics use different methods to calculate metrics).

In other words, rv009 of Mach2D5p7 and present revision produce the same result.

Test01

Contains rv009 of Mach2D5p7 for comparison with other tests.

Test02

Contains rv009 of Mach2D5p7 with metrics replaced by get_metrics subroutine. Comparison between this modification of rv009 of Mach2D5p7 with present revision (rv013), shows that differences arise only by rounding errors.

Revision 014

Parallel computation was implemented using revision 013 and OpenMP. Parallelization is used to calculate thermal and viscous properties, to calculate coefficients of linear systems for u and v and to solve these linear systems.

Test01

Contains rv013 for comparisons with rv014.

Comparing mass flow rate and dynamic thrust, differences appeared due rounding errors only.

Test02

When solver MSI was replaced by a TDMA solver (see revision 015 bellow), it was observed that mass flow rate and dynamic thrust suffered changes when parameters associated to the linear system of pressure correction, such as number of iterations or tolerance, were changed. This test aims to quantify such changes. Seven simulations were run. The differences between simulations is a parameter associated to iterations of the linear systems. The results are presented bellow:

DATA OF SIMULATION 1:

```
S001 ....: Simulation identification (up to 100 characters)
      22 ....: Number of real+fictitious volumes in the csi direction
      10 ....: Number of real+fictitious volumes in the eta direction
      2 ....: Kind of grid
1.0000000E-03 ....: Initial step for the geometric progression grid
      1 ....: Coordinate system ( 1=cylindrical, else cartesian)
2.8700000E+02 ....: Perfect gas constant
1.4000000E+00 ....: gamma = Cpo / Cvo in the chamber (Specific heat ratio)
1.0000000E+06 ....: Stagnation pressure in the chamber
5.0000000E+02 ....: Stagnation temperature in the chamber
1.0000000E+00 ....: Constant of the UDS/CDS mixing scheme
      1 ....: modvis = 0 -> Euler; modvis = 1 -> Navier-Stokes
      1 ....: ccTw = 0 -> adiabatic; ccTw = 1 -> prescribed temperature
1.0000000E-05 ....: Time step
      10000 ....: Maximum number of interactions of the time evolution
      2 ....: Maximum number of interactions for the pressure correction
      5 ....: Maximum number of interactions for the MSI method for u, v and T
1.0000000E-01 ....: Tolerance for the MSI method for u, v and T
      10 ....: Maximum number of interactions for the MSI method for p
1.0000000E-02 ....: Tolerance for the MSI method for p
```

RESULTS OF SIMULATIONS

	it	norm / norm1	fmi	fme	Fd	diff. relative to S001
S001	10000	1.351191530930245E-09	5.678668617572837E+01	5.678668617572841E+01	2.982437587913622E+04	none
S002	10000	1.312730616567316E-09	5.678668617572841E+01	5.678668617572841E+01	2.982437587913619E+04	nitm_u = 10
S003	10000	5.043297097173190E-06	5.678668617572836E+01	5.678668617572841E+01	2.982437587913627E+04	tolu = 1.d-5
S004	10000	1.325729267781418E-09	5.678668618187415E+01	5.678668618187427E+01	2.982437542910004E+04	itmax = 4
S005	did not converge					itmax = 8
S006	10000	1.351191530930245E-09	5.678668617572837E+01	5.678668617572841E+01	2.982437587913622E+04	nitm_p = 20
S007	10000	1.264250970584168E-09	5.678668618681436E+01	5.678668618681429E+01	2.982437551831575E+04	tolp = 1.d-4

Note that when the number of interactions (nitm_u) or tolerance (tolu) of linear systems for u, v and T are changed (S001-S003), the differences in the results are caused only by rounding errors. On the other hand, if the number of external interactions (itmax) and tolerance (tolp) are changed for linear system of pressure correction, then the mass flow rate and thrust change from the 9th figure. No differences appeared in S006 because tol_p finishes interactions before nitm_p is reached.

Intending to evaluate the influence of the grid size on this results, more three simulations were performed (S008-S010):

DATA OF SIMULATION 8:

```

S008 ..... Simulation identification (up to 100 characters)
  42 ..... Number of real+fictitious volumes in the csi direction
  18 ..... Number of real+fictitious volumes in the eta direction
   2 ..... Kind of grid
1.0000000E-03 ..... Initial step for the geometric progression grid
   1 ..... Coordinate system ( 1=cylindrical, else cartesian)
2.8700000E+02 ..... Perfect gas constant
1.4000000E+00 ..... gamma = Cpo / Cvo in the chamber (Specific heat ratio)
1.0000000E+06 ..... Stagnation pressure in the chamber
5.0000000E+02 ..... Stagnation temperature in the chamber
1.0000000E+00 ..... Constant of the UDS/CDS mixing scheme
   1 ..... modvis = 0 -> Euler; modvis = 1 -> Navier-Stokes
   1 ..... ccTw = 0 -> adiabatic; ccTw = 1 -> prescribed temperature
5.0000000E-06 ..... Time step
 10000 ..... Maximum number of interactions of the time evolution
   2 ..... Maximum number of interactions for the pressure correction
   5 ..... Maximum number of interactions for the MSI method for u, v and T
1.0000000E-01 ..... Tolerance for the MSI method for u, v and T
  10 ..... Maximum number of interactions for the MSI method for p
1.0000000E-02 ..... Tolerance for the MSI method for p
    
```

RESULTS OF SIMULATIONS

it	norm / norm1	fmi	fme	Fd	diff. relative to S008
S001 10000	1.351191530930245E-09	5.678668617572837E+01	5.678668617572841E+01	2.982437587913622E+04	nx and ny
S008 10000	2.104156554023244E-09	5.677792729792442E+01	5.677792729792442E+01	2.982387558848702E+04	none
S009 10000	1.953476906984203E-09	5.677792730044253E+01	5.677792730044254E+01	2.982387553809946E+04	tolp = 1.d-4
S010 10000	2.171127112735485E-09	5.677792729710496E+01	5.677792729710502E+01	2.982387547107686E+04	imax = 4

Note that the same problem appears: results change from 9th figure.

Test03 and Test04

In order to study the time reduction of using parallel computation, the time needed to solve a problem with the following parameters

```

S01 ..... Simulation identification (up to 100 characters)
  42 ..... Number of real+fictitious volumes in the csi direction
  18 ..... Number of real+fictitious volumes in the eta direction
   2 ..... Kind of grid
1.0000000E-03 ..... Initial step for the geometric progression grid
   1 ..... Coordinate system ( 1=cylindrical, else cartesian)
2.8700000E+02 ..... Perfect gas constant
1.4000000E+00 ..... gamma = Cpo / Cvo in the chamber (Specific heat ratio)
1.0000000E+06 ..... Stagnation pressure in the chamber
5.0000000E+02 ..... Stagnation temperature in the chamber
1.0000000E+00 ..... Constant of the UDS/CDS mixing scheme
   1 ..... modvis = 0 -> Euler; modvis = 1 -> Navier-Stokes
   1 ..... ccTw = 0 -> adiabatic; ccTw = 1 -> prescribed temperature
1.0000000E-05 ..... Time step
 10000 ..... Maximum number of interactions of the time evolution
   2 ..... Maximum number of interactions for the pressure correction
   5 ..... Maximum number of interactions for the MSI method for u, v and T
1.0000000E-01 ..... Tolerance for the MSI method for u, v and T
  10 ..... Maximum number of interactions for the MSI method for p
1.0000000E-02 ..... Tolerance for the MSI method for p
    
```

was measued (three times) for one processor and two processors. The results are presented bellow

TIME CONSUMPTION USING TWO PROCESSORS

```

2.2621489018027205E+01 ..... Time1 (s)
2.2589501234004274E+01 ..... Time2 (s)
2.2614214453962632E+01 ..... Time3 (s)
-----
Mean time = 22.6 s
    
```

TIME CONSUMPTION USING ONE PROCESSOR

```

2.7378585043014027E+01 ..... Time1 (s)
2.7386815523961559E+01 ..... Time2 (s)
2.7380245146050584E+01 ..... Time3 (s)
-----
Mean time = 27.4 s
    
```


One can see that time reduction was about 17.5%.

These measured time refers to the one needed to solve the linear system iteratively.
(loop of the time evolution of the solving procedure)

Revision 015

Solver MSI of revision rv014 was replaced by a line-by-line TDMA method of 5 and 9 diagonals.

It was observed that:

- 1) For the linear systems of u, v and T: mass flow rate and dynamic thrust did not change when solver was replaced (only rounding errors were found).
- 2) For the linear system of pressure correction pl: mass flow rate and dynamic thrust CHANGED when solver was replaced. Changes occurred from the 10th decimal place.
This change is related to the number of iterations applied to solve the linear system.
- 3) The residual ratio $L1_n/L1_1$ changed significantly when the solver was changed for linear system of T.
This change is related to the number of iterations applied to solve the linear system.

Test02 and Test03

In order to verify time optimization of parallel computation, some measurements were made. The procedure is similar to the one applied in revision 014. Parameters used:

```
"S01" .....: sim_id - Simulation identification (up to 100 characters)
 20 .....: nx-2  - Number of real volumes in the csi direction
  8 .....: ny-2  - Number of real volumes in the eta direction
  2 .....: kg    - Kind of grid (1=eta uniform, 2=geometric progression for eta)
1.d-3 .....: a1   - Initial step for the geometric progression grid
  1 .....: coord - Coordinate system ( 1=cylindrical, else cartesian)
287.d0 .....: Rg  - Perfect gas constant
 1.4d0 .....: gamma - gamma = Cpo / Cvo in the chamber (Specific heat ratio)
10.0d+5 .....: po  - Stagnation pressure in the chamber
500.d0 .....: T0  - Stagnation temperature in the chamber
 1.d0 .....: beta - Constant of the UDS/CDS mixing scheme
  1 .....: modvis - modvis = 0 -> Euler; modvis = 1 -> Navier-Stokes
  1 .....: ccTw  - ccTw = 0 -> adiabatic; ccTw = 1 -> prescribed temperature
1.d-5 .....: dt   - Time step
10000 .....: itmax - Maximum number of interactions of the time evolution
  2 .....: imax  - Maximum number of interactions for the pressure correction
  5 .....: nitm_u - Maximum number of interactions for the TDMA method for u, v and T
 10 .....: nitm_p - Maximum number of interactions for the TDMA method for p
```

Results are presented below:

```
TIME CONSUMPTION USING TWO PROCESSORS
2.8339234838029370E+01 .....: Time1 (s)
2.8226870027021505E+01 .....: Time2 (s)
2.8235274304985069E+01 .....: Time3 (s)
-----
```

Mean time 28.3 s

```
TIME CONSUMPTION USING ONE PROCESSOR
3.8742839808983263E+01 .....: Time1 (s)
3.8341871729004197E+01 .....: Time2 (s)
3.8323719265987165E+01 .....: Time3 (s)
-----
```

Mean time 38.5 s

Time reduction was about 26.5%. This efficiency is higher than that observed for MSI method.

Time measurements of this revision can not be compared to those of revision 014 because the number of interactions are different.

Revision 016

Modification of the rv014 global algorithm: in the iterative solution of the linear system for the pressure correction, only the source term and boundary conditions are updated.

Code::Blocks is used as IDE.

It was found that the results do not depend on dt anymore.

GRID 1

```
"S01" ..... sim_id - Simulation identification (up to 100 characters)
 20 ..... nx-2 - Number of real volumes in the csi direction
  8 ..... ny-2 - Number of real volumes in the eta direction
  2 ..... kg   - Kind of grid (1=eta uniform, 2=geometric progression for eta)
1.d-3 ..... a1 - Initial step for the geometric progression grid
  1 ..... coord - Coordinate system ( 1=cylindrical, else cartesian)
287.d0 ..... Rg - Perfect gas constant
1.4d0 ..... gamma - gamma = Cpo / Cvo in the chamber (Specific heat ratio)
10.0d+5 ..... po - Stagnation pressure in the chamber
500.d0 ..... T0 - Stagnation temperature in the chamber
 1.d0 ..... beta - Constant of the UDS/CDS mixing scheme
  1 ..... modvis - modvis = 0 -> Euler; modvis = 1 -> Navier-Stokes
  1 ..... ccTw - ccTw = 0 -> adiabatic; ccTw = 1 -> prescribed temperature
1.d-7 ..... dt - Time step
500000 ..... itmax - Maximum number of interactions of the time evolution
  2 ..... imax - Maximum number of interactions for the pressure correction
  5 ..... nitm_u - Maximum number of interactions for the MSI method for u, v and T
1.d-1 ..... tolu - Tolerance for the MSI method for u, v and T
 10 ..... nitm_p - Maximum number of interactions for the MSI method for p
1.d-2 ..... tolp - Tolerance for the MSI method for p

  it          L1_N/L1_0      mass flow rate (in)      mass flow rate (out)      Fd      dt
6700  1.200833356147843E-09  5.678668618688039E+01  5.678668618688037E+01  2.982437551838925E+04  E-5
44700 2.060865231062360E-06  5.678668618688006E+01  5.678668618687961E+01  2.982437551838901E+04  E-6
***** 1.283540243530924E-02  5.678668618687591E+01  5.678668618687220E+01  2.982437551838311E+04  E-7
```

GRID 2

```
"S04" ..... sim_id - Simulation identification (up to 100 characters)
 50 ..... nx-2 - Number of real volumes in the csi direction
 40 ..... ny-2 - Number of real volumes in the eta direction
  2 ..... kg   - Kind of grid (1=eta uniform, 2=geometric progression for eta)
1.d-3 ..... a1 - Initial step for the geometric progression grid
  1 ..... coord - Coordinate system ( 1=cylindrical, else cartesian)
287.d0 ..... Rg - Perfect gas constant
1.4d0 ..... gamma - gamma = Cpo / Cvo in the chamber (Specific heat ratio)
10.0d+5 ..... po - Stagnation pressure in the chamber
500.d0 ..... T0 - Stagnation temperature in the chamber
 1.d0 ..... beta - Constant of the UDS/CDS mixing scheme
  1 ..... modvis - modvis = 0 -> Euler; modvis = 1 -> Navier-Stokes
  1 ..... ccTw - ccTw = 0 -> adiabatic; ccTw = 1 -> prescribed temperature
9.d-6 ..... dt - Time step
100000 ..... itmax - Maximum number of interactions of the time evolution
  8 ..... imax - Maximum number of interactions for the pressure correction
  5 ..... nitm_u - Maximum number of interactions for the MSI method for u, v and T
1.d-1 ..... tolu - Tolerance for the MSI method for u, v and T
 10 ..... nitm_p - Maximum number of interactions for the MSI method for p
1.d-2 ..... tolp - Tolerance for the MSI method for p

  it          norm / norm1      fmi          fme          Fd      dt
100000 1.936198862908188E-10  5.677840202523103E+01  5.677840202523105E+01  2.982345306844190E+04  9E-6
100000 3.964047530452071E-07  5.677840202523078E+01  5.677840202523080E+01  2.982345306844167E+04  9E-7
500000 3.601368373405795E-03  5.677840202524728E+01  5.677840202521814E+01  2.982345306839366E+04  9E-8
```

GRID 3

```
"S01" ..... sim_id - Simulation identification (up to 100 characters)
 100 ..... nx-2 - Number of real volumes in the csi direction
  80 ..... ny-2 - Number of real volumes in the eta direction
  2 ..... kg   - Kind of grid (1=eta uniform, 2=geometric progression for eta)
```

```

1.d-3 .....: a1      - Initial step for the geometric progression grid
  1 .....: coord    - Coordinate system ( 1=cylindrical, else cartesian)
287.d0 .....: Rg     - Perfect gas constant
  1.4d0 .....: gamma  - gamma = Cpo / Cvo in the chamber (Specific heat ratio)
10.0d+5 .....: po    - Stagnation pressure in the chamber
  500.d0 .....: T0    - Stagnation temperature in the chamber
  1.d0 .....: beta   - Constant of the UDS/CDS mixing scheme
  1 .....: modvis   - modvis = 0 -> Euler; modvis = 1 -> Navier-Stokes
  1 .....: ccTw    - ccTw = 0 -> adiabatic; ccTw = 1 -> prescribed temperature
  9.d-6 .....: dt    - Time step
1000000 .....: itmax - Maximum number of interactions of the time evolution
  8 .....: imax    - Maximum number of interactions for the pressure correction
  5 .....: nitm_u  - Maximum number of interactions for the MSI method for u, v and T
  1.d-1 .....: tolu - Tolerance for the MSI method for u, v and T
  10 .....: nitm_p - Maximum number of interactions for the MSI method for p
  1.d-2 .....: tolp - Tolerance for the MSI method for p

  it          norm / norm1          fmi          fme          Fd          dt
100000      6.362339528296163E-11  5.677857069917401E+01  5.677857069917400E+01  2.982220075339333E+04  9E-6
500000      9.511188338297630E-08  5.677857069917421E+01  5.677857069917417E+01  2.982220075339388E+04  9E-7
530300      7.296501011656949E-04  5.677857069918527E+01  5.677857069918247E+01  2.982220075340243E+04  9E-8

```

Revision 017

Modification of revision rv016. A new equation for pressure correction was implemented.

The new vectors Ucem, Vcnm were define to store the Uce and Vcn of the previous iteration.

In the module mach2d-user, subroutine get_Uce_Vcn_de_dn_at_boundary_faces was split in get_boundary_simplec_coefficients and get_Uce_Vcn_at_boundary_faces.

In the subroutine get_pressure_density_correction_with_pl of module mach2d-coef.f90, density correction as $ro = ro * + g*pl$ or $ro=g*p$ produce the same results. In rv008, it was observed that the two ways produced different results. Now, the first way is applied.

The order of some subroutines were changed.

NEW ALGORITHM

Initializes all the variables to zero.
 Defines the boundary nodes.
 Generates the grid based on the boundary nodes.
 Calculates the centroids of all real volumes.
 Calculates the metrics.
 Calculates the radius at the volume faces and centre.
 Sets the wall temperature (for non-adiabatic walls only).
 Variables initialization based on the 1D isentropic flow.

Starts the time evolution cycle

```

  Updates all the fields.
  Updates boundary data on the nozzle entrance.
  Calculates Cp and Gamma at the center of each volume.
  Calculates viscosity and thermal conductivity at centroids.
  Calculates viscosity and thermal conductivity at faces of each volume using the Patankar method.
  Calculates the coefficients and source of linear systems for u and v based on the data of the previous iteration.
  Calculates the boundary conditions for u and v.
  Calculates the SIMPLEX coefficients.
  Calculates the coefficients of the pressure equation.
  Solves the equations for u and v. (This solution does not satisfy the mass equation)
  Calculates the cartesian and contravariant velocities at faces.
  Calculates the source term of the pressure equation.

```

Starts the mass correction cycle

```

  Calculates the boundary condition of the pressure equation.
  Solves the equation for the pressure correction.
  Ends the mass correction cycle

```

Corrects the pressure and specific mass with the pressure deviation.
 Corrects the velocities at centroids.
 Corrects the cartesian and contravariant velocities at volume faces.
 Calculates the specific mass at faces based on the corrected specific mass and velocities.
 Calculates the coefficients and source of the temperature equation.
 Calculates the boundary conditions of the temperature.
 Solves the temperature equation.
 Calculates specific mass at centroids using the new values of p and T and the state equation.
 Calculates the specific mass at faces based on the corrected specific mass and velocities.

Error analysis.

Ends of time evolution cycle

Post processing.

Test01

S01: Simulation identification (up to 100 characters)
 22: Number of real+ficitious volumes in the csi direction
 10: Number of real+ficitious volumes in the eta direction
 2: Kind of grid
 1.0000000E-03: Initial step for the geometric progression grid
 1: Coordinate system (1=cylindrical, else cartesian)
 2.8700000E+02: Perfect gas constant
 1.4000000E+00: $\gamma = C_{po} / C_{vo}$ in the chamber (Specific heat ratio)
 1.0000000E+06: Stagnation pressure in the chamber
 5.0000000E+02: Stagnation temperature in the chamber
 1.0000000E+00: Constant of the UDS/CDS mixing scheme
 1: modvis = 0 -> Euler; modvis = 1 -> Navier-Stokes
 1: ccTw = 0 -> adiabatic; ccTw = 1 -> prescribed temperature
 9.0000000E-06: Time step
 5000: Maximum number of interactions of the time evolution
 8: Maximum number of interactions for the pressure correction
 5: Maximum number of interactions for the MSI method for u, v and T
 1.0000000E-01: Tolerance for the MSI method for u, v and T
 10: Maximum number of interactions for the MSI method for p
 1.0000000E-02: Tolerance for the MSI method for p

it	norm / norm1	fmi	fme	Fd	dt
5000	1.711421489285850E-09	5.678668618688051E+01	5.678668618688041E+01	2.982437551838931E+04	9E-6
50000	2.851253845665069E-06	5.678668618687895E+01	5.678668618687980E+01	2.982437551839050E+04	9E-7
500000	2.400199927648853E-02	5.678668618687410E+01	5.678668618687441E+01	2.982437551837889E+04	9E-8

Revision 018

(NOT FINISHED)

Baldwin-Lomax turbulence model is being implemented in rv017.

- initialize module blomax
 - conditional to turbulence calculation

! CAUTION: In blomax module the Baldwin-Lomax model is applied to the line of
 ! constant ksi, instead to the line perpendicular to the wall surface. This
 ! must be improved.

A module blomax (mach2d-blomax.f90) was written for the calculation of the eddy viscosity. When the eddy viscosity is added to the laminar viscosity and the turbulent thermal conductivity is added to the regular one, it is observed that the solution oscilates and does not converge.

Test

Metrics produced by blomax module (mach2d-blomax.f90) was verified using x(ksi,eta) and y(ksi,eta) as

```
xf(ksi,eta) = (-2.d0*(ksi-1.d0)/nx-1.d0)*(eta-1.d0)/ny+4.d0*(ksi-1.d0)/nx+1.d0
yf(ksi,eta) = ( 2.d0*(ksi-1.d0)/nx+1.d0)*(eta-1.d0)/ny
```

No differences between numerical and analytical solution were found.

Vorticity calculation was verified using a prescribed velocity field:

```
u(np) = + (eta-ny) * ksi / nx / ny
v(np) = - (eta-ny) * ksi * 3.d0 / nx / ny
```

No differences were found between numerical and analytical solutions.

Laminar viscosity and density were given by

```
vlp(np) = 2.d0*eta/ny
ro(np) = 3.d0*eta/ny
```

No differences between numerical and analytical results were found for the

- shear stress
- gradient of the tangent velocity in the direction normal to the wall
- Udif
- eddy viscosity of the inner and outer layer

Relatively to the Baldwin-Lomax module, some improvements must be performed

- ymax and Fmax must be calculated using a quadratic interpolation
- ym: a criteria must be used to determine whether or not ym was found correctly

Numerical results produced by module blomax were compared to those produced by Maxima (see maxima folder of this revision)

In order to look for programming errors the module (mach2d-blomax.f90) was rewritten (blomax.f90). In this new module, ymax and Fmax were calculated by a quadratic interpolation. No programming errors were found.

It was observed that the outer part of the eddy viscosity is not used in the calculation, i.e. ym is not found in the calculations. The reason of the oscillation is still unknown.

Revision 019

Line-by-line TDMA method is being implemented. Almost all files were incorporated from rv017. Files mach2d-solvers.f90, tdma2d5.f90 and tdma2d9.f90 were incorporated from rv015.

Test01

Results were compared for single and multiprocessing, no differences were found. Results also were compared with those of rv017. The only differences were caused by rounding errors.

Revision 020

This revision is an extension of rv017.

New features

- variable time step
- periodic backup
- post-processing

Necessary softwares:

gnuplot

evince
emacs

Points that need improvements:

- Check subroutine write_main_results of module postp
- Check planar flow option

Revision 021

This revision changes rv020.
Solver TDMA line-by-line is applied.

Revision 022

Bug fixing of rv020.

Bugs fixed:

- 1 - In the subroutine get_uniform_grid, line

```
dx = (xf-xi) / (nx-2)
```

was replaced by

```
dx = (xf-xi) / (ny-2)
```

(This bug does not change the results, since $xf-xi$ along an eta line is always 0)

- 2 - In subroutines get_uniform_grid and get_backward_GP_grid x and y were changed to intent(inout), instead of intent(out).

- 3 - In the subroutine post_processing_boundaries (postp module), the method of calculation of v in the NE corner of the latest real volume ($i=nx-1$, $j=ny-1$) (the data is stored in the fictitious volume at $i=nx$ and $j=ny$) was changed. Instead of using

```
v(np) = (v(npw)+v(npsw))/2
```

it is applied now

```
v(np) = (v(np)+v(nps)+v(npw)+v(npsw))/4.
```

(In the code some modifications are used in order to avoid pitfalls)

No differences in the results were found due bugs 1 and 2. These bugs did not modify the results because all of them were related to the variation of x with eta, which is zero.

Revision 023

Bug fixing of rv021.

Bugs fixed:

The same of rv022.

Revision 024

Applies the blomax module to rv023.

Subroutine get_eddy_viscosity_fictitious was moved from user module to blomax module.

A variable to choose turbulence or laminar flow (modtur) was introduced.

In the calculation of the eddy viscosity, rho, rho_n, u and v are taken as a mean of the values obtained in the two previous time levels. This is applied to avoid divergence.

Since eddy viscosity is calculated along eta lines, these lines must be normal to the nozzle wall. In the grid used this is not the case. So there is an error associated to the

non-orthogonality of the grid near the wall. This error should increase if the ratio of the chamber to the throttle radius increases.

Stream function and eddy viscosity are plotted now in the post processing module. Extrapolation of the stream function and eddy viscosity to the fictitious are done through a zero order interpolation function. Extrapolation has effects only on the plot, where the function needs to be known in the grid boundary.

A new kind of grid was introduced. In this grid the points along the eta lines are distributed according to a power law, with more points near to the wall.

Revision 025

Parent: rv024

An elliptic grid generator based on the Poisson equation and sliding boundary is applied.

set_grid a 'close' statement was inserted

Revision 026

Parent: rv023

From revision rv020 to rv022 some errors were corrected. One of them, however, was not documented (a mistake):

* In the subroutine get_u_souce, module coefficients, it was missing the viscosity coefficient in the term S3.

This error does not appear in the results because S3 also has a factor x_eta, which is zero in the grid used.

In the revisions rv021, rv023 and rv024, which were derived from rv020, this non-documented error was not corrected. So this revision intends to correct it only for revision rv023.

Another error was found during the compilations:

* In the subroutine write_nozzle_parameters of postp module, it was found

```
integer, parameter :: pi = acos(-1.d0)
```

which was changed to

```
real(8), parameter :: pi = acos(-1.d0)
```

In this revision it was introduced a version file: version.txt. This file contains the version and revision information of the present code.

Revision 027

Parent: rv024

Applies the same corrections done in rv026 to rv024.

Revision 028

Parent: rv022

An error was found during the compilations:

* In the subroutine write_nozzle_parameters of postp module, it was found

```
integer, parameter :: pi = acos(-1.d0)
```

which was changed to

```
real(8), parameter :: pi = acos(-1.d0)
```

In this revision it was introduced a version file: version.txt. This files contains the version and revision information of the present code.