

Aplicação da extrapolação de Richardson para estimativa da condição inicial em malhas mais finas

Guilherme Bertoldo

23 de Outubro de 2012

Conteúdo

1	Introdução	1
2	Algoritmo	1
3	Malha e parâmetros numéricos	2
4	Resultados	3
5	Conclusão	3

1 Introdução

Ao se resolver o escoamento com o código Mach2D, percebe-se que o número de iterações necessárias para se atingir uma tolerância prescrita aumenta com o refino da malha. Por este motivo deseja-se investigar se o uso da solução numérica obtida em malhas mais grossas pode ser utilizada para reduzir o tempo computacional em malhas mais finas. A ideia básica é utilizar interpolações conjugadas à extrapolação de Richardson.

2 Algoritmo

Os resultados apresentados neste relatório foram obtidos através de um galho (mach2d-5.8.2.1-SVN-r110, Branch: IC) gerado a partir do código mach2d-5.8.2.1-SVN-r104, Branch: cone_wedge. Esse novo código, salva a solução convergida em um arquivo e pode ou não reutilizá-la como condição inicial. As variáveis salvas são apenas u , v , T e p nos centroides dos volumes reais. Como o código também necessita das velocidades e densidades nas faces, estas são estimadas a partir dos respectivos valores nodais.

O gerenciamento do código mach2d é realizado pelo código IC, que tem o seguinte algoritmo:

Algoritmo do código IC

1. Gerar N_g malhas uniformes com razão de refino 2.
2. Alocar memória para as variáveis u , v , T e p nas malhas supergrossa (ϕ_1), grossa (ϕ_2) e fina (ϕ_3), bem como para a interpolação ($\bar{\phi}_2$) da solução da malha grossa sobre a malha supergrossa e a estimativa da solução com base na extrapolação de Richardson (ϕ_∞).
3. Executar o código mach2d na malha 1 (supergrossa).
4. Executar o código mach2d na malha 2 (grossa).
5. Ler as soluções ϕ_1 e ϕ_2 dos arquivos de saída do mach2d.
6. Ler as coordenadas dos centroides das malhas supergrossa, grossa e fina.
7. Para $n = 3$ até N_g fazer:
 - (a) Interpolar a solução da malha grossa nos centroides da malha supergrossa para obter $\bar{\phi}_2$.
 - (b) Gerar a extrapolação de Richardson com a fórmula:

$$\phi_\infty = \bar{\phi}_2 + \frac{\bar{\phi}_2 - \phi_1}{r^{p_L}} \quad (1)$$

onde $r = 2$ e $p_L = 1$.

- (c) Estender a extrapolação de Richardson da malha supergrossa ϕ_∞ para a fina ϕ_3 com base na interpolação bilinear.
- (d) Guardar ϕ_3 em um arquivo para ser utilizada como condição inicial na malha fina.
- (e) Executar o mach2d na malha n (fina) com a estimativa inicial ϕ_3 .
- (f) Ler de um arquivo a solução convergida na malha fina ϕ_3 .
- (g) Se $n < N_g$, fazer
 - i. $\phi_1 = \phi_2$ (desalocar e realocar memória dinamicamente)
 - ii. $\phi_2 = \phi_3$ (desalocar e realocar memória dinamicamente)
 - iii. Ler as coordenadas dos centroides da malha $n + 1$.

3 Malha e parâmetros numéricos

A malha base utilizada nas simulações é apresentada na Fig. 1.

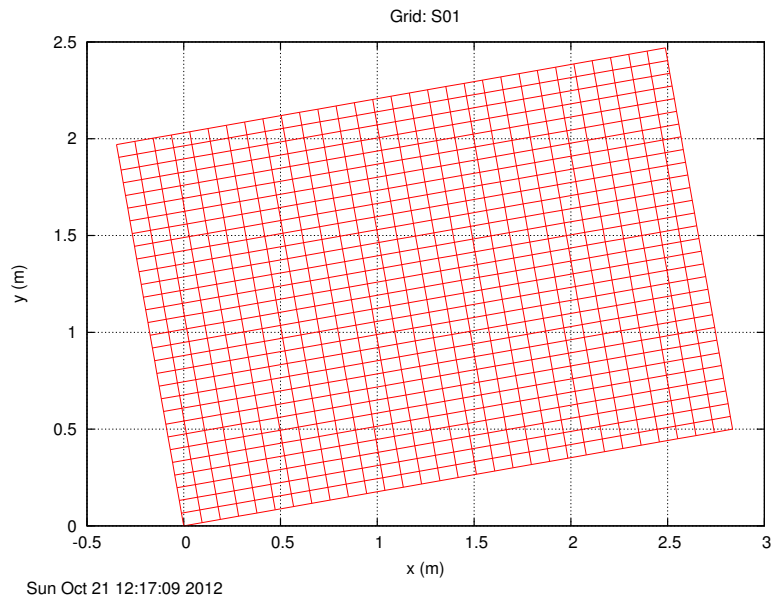


Figura 1: Malha base empregada nas simulações.

Os parâmetros numéricos de entrada do código mach2d-5.8.2.1-SVN-r110, Branch: IC são dados na Tab. 2. Na terceira coluna desta tabela são indicados os parâmetros comuns a todas as simulações.

Tabela 1: Parâmetros numéricos de entrada

Simulation identification (up to 100 characters)	sim_id	
Number of real volumes in the csi direction	nx2	
Number of real volumes in the eta direction	ny2	
Upload saved initial conditions (0=no, 1=yes)	ic	
Width of the domain of calculation (m)	lb	2.000000000000E+00
length of the body (m)	lr	2.8356409098089E+00
frontal radius/semi-height of the body (m)	rf	0.000000000000E+00
base radius/semi-height of the body (m)	rb	0.500000000000E+00
kind of boundary (1=orthogonal grid, 2=non-orthogonal)	kb	1
Kind of grid (1=uniform, 2=geometric progression, 3=power law)	kg	1
Kind of centroid mean (1=simple mean, 2=weighted mean)	kcm	1
Kind of coord. system (1=cylindrical, 0 = cartesian)	coord	1
width of the volume closer to the wall (m)	a1	1.000000000000E-03
Exponent of the power law for the south boundary	aks	1.000000000000E+00
Maximum number of iterations for time cycle	itmax	50000
"Maximum number of iterations for the correction cycle"	itimax	1
Maximum number of iterations for mass cycle	imax	1
Number of iterations up to which dt = dt1	it1	100
Number of iterations from which dt = dt2	it2	300
Initial time step (s)	dt1	
Final time step (s)	dt2	
Maximum number of iterations for solving the linear systems for u, v and T	nitm_u	5
Maximum number of iterations for solving the linear system for p	nitm_p	5
Number of iterations to calculate the mean of the residuals	nit_res	1
Tolerance in the MSI for solving the linear systems for u, v and T	tol_u	1.000000000000E-02
Tolerance in the MSI for solving the linear system for p	tol_p	1.000000000000E-02
Tolerance for the sum of residuals	tol_res	5.000000000000E-12
Frequency of printing in the listing file	wlf	1
1 = do not open result files, 0 = open	sem_a	1
0 = visualize the plot, 1 = do not visualize	sem_g	1
Frequency of writing data for graphics	w_g	1
1 = write the fields, 0 = do not	w_cam	0
Number of iterations up to which beta = beta1	itb1	300
Number of iterations from which beta = beta2	itb2	1000
Initial beta (UDS/CDS mixing constant (0=UDS, 1=CDS))	beta1	0.000000000000E+00
Final beta (UDS/CDS mixing constant (0=UDS, 1=CDS))	beta2	0.000000000000E+00
GF = gamma = Cp / Cv (for the free stream)	GF	1.400000000000E+00
Perfect gas constant (J/kg.K)	Rg	2.870000000000E+02
Free stream pressure (Pa)	PF	1.000000000000E+05
Free stream temperature (K)	TF	3.000000000000E+02
Free stream Mach number	MF	4.000000000000E+00

4 Resultados

Dois conjuntos de simulações foram realizados. Em um deles (S0104) utilizou-se as soluções das malhas mais grossas para se estimar a condição inicial para a malha mais fina, enquanto no outro não (S0105). Os resultados são apresentados na Tab. 2. Nesta tabela, Cdfi é o coeficiente de arrasto frontal invíscido, res é o resíduo dos sistemas lineares na última iteração, $d(it)/it$ é a diferença relativa entre o número de iterações do conjunto de simulações S0104 e S0105, $d(tcpu)/tcpu$ é a diferença relativa do tempo de CPU do conjunto de simulações S0104 e S0105.

Tabela 2: Resultados principais dos conjuntos de simulações S0104 e S0105.

Simulation	nx	ny	ic	it	tcpu	dt1	dt2	RAM	Cdfi	res	$d(it)/it$	$d(tcpu)/tcpu$
S01	30	30	0	172	0.196	5.0E-04	5.0E-04	15.4	7.813236211793244E-02	1.26E-14	0%	0%
S02	60	60	0	206	0.836	1.0E-04	1.0E-04	17.8	7.858293449326295E-02	4.69E-15	0%	1%
S03	120	120	1	204	3.668	1.0E-04	1.0E-04	27.5	7.882454496741947E-02	4.69E-14	6%	12%
S04	240	240	1	292	22.961	5.0E-05	5.0E-05	65.3	7.901964539396703E-02	1.08E-13	10%	50%
S05	480	480	1	952	291.03	1.0E-05	1.0E-05	211.4	7.916181627421115E-02	1.71E-14	8%	24%
S06	960	960	1	1004	1442.238	1.0E-05	1.0E-05	798.5	7.925289660918854E-02	2.17E-13	6%	13%

5 Conclusão

A utilização da solução de malhas mais grossas para estimar a condição inicial em malhas mais finas reduziu o tempo computacional de 12 a 50%, dependendo da malha. Talvez este desempenho possa ser melhorado se as variáveis sobre as faces forem estimadas com melhor precisão a partir dos seus valores nodais. Em um teste não registrado, a solução (u, v, T, p) convergida em uma malha foi salva e em seguida carregada para ser utilizada como condição inicial para a mesma malha. Como a solução inicial era a solução numérica convergida, esperava-se que o processo iterativo fosse curto, entretanto, observou-se uma redução de apenas 27% do número de iterações.