

Efeito de hardware e software sobre o erro de arredondamento em CFD

Diego Fernando Moro¹

Programa de Pós Graduação em Engenharia Mecânica (PGMec), Centro Politécnico, UFPR, Curitiba, PR

Carlos Henrique Marchi²

Departamento de Engenharia Mecânica, Universidade Federal do Paraná (UFPR), Curitiba, PR

Resumo. Neste trabalho verifica-se o efeito do erro de arredondamento sobre soluções numéricas, pois este efeito não é bem conhecido em transferência de calor e dinâmica dos fluidos computacional. Para análise, foram utilizados dois problemas de condução de calor e um de escoamento de fluidos compressíveis para analisá-los, ambos códigos computacionais foram escritos em Fortran 90. A condução de calor é resolvida utilizando-se o método de diferenças finitas com esquema de segunda ordem de acurácia e considerando: (1) uma e duas dimensões espaciais; (2) nove configurações diferentes de computadores (Intel e AMD); (3) precisões simples ou dupla; (4) sistemas operacionais Windows XP 32 bits, 64 bits e Linux 64 bits (Ubuntu); (5) três tipos de variáveis (globais e pontuais); (6) quatro tipos de compiladores: Microsoft 4.0, Compaq 6.6 e Intel 11.1 no Windows e gfortran no Linux; (7) diversas opções de compilação; (8) solver direto (1D), iterativo com Multigrid (2D) e (9) número de incógnitas de 2 a 67 milhões. Já o escoamento de fluidos compressíveis é resolvido com volumes finitos e esquema de primeira ordem de acurácia em dois sistemas operacionais (Windows e Linux) e dois tipos de solvers (ADI e MSI) com o compilador Intel 14.0. De todos os aspectos citados anteriormente apenas o compilador afetou o erro de arredondamento.

Palavras-chave. Erro de arredondamento, transferência de calor computacional, dinâmica dos fluidos computacional, erro numérico, verificação.

1 Introdução

Para resolver problemas de engenharia, podemos optar por três tipos de métodos: (1) experimentais, (2) analíticos ou (3) numéricos. Nos métodos experimentais (1) avalia-se o fenômeno físico diretamente, são empregados sistemas de medição para obter o valor das grandezas de interesse diretamente e os erros existentes nestes resultados devem-se às condições do experimento, aferição dos sistemas de medição, entre outras.

¹difmoro@ufpr.br, difmoro@gmail.com

²marchi@ufpr.br, chmefd@gmail.com

Nos métodos analíticos (2) utiliza-se uma representação matemática do fenômeno físico chamada modelo matemático, este modelo é então simplificado e resolvido. Apresentam erros de modelagem, ou seja, erros entre o modelo matemático e o fenômeno real.

Nos métodos numéricos (3) também utiliza-se o modelo matemático, no entanto sem as simplificações do método analítico, o que permite resolver problemas com equações, geometrias e condições de contorno mais gerais. No entanto é necessário o uso de computadores para obter as soluções numéricas e estas soluções além de apresentarem os erros de modelagem citados nos métodos analíticos, também apresentam erros numéricos [1].

Os erros numéricos podem ser causados por quatro fontes: (A) erros de truncamento, (B) erros de iteração, (C) erros de programação e (D) erros de arredondamento.

Os erros de truncamento (A) resultam das aproximações numéricas feitas na discretização do modelo matemático. Eles causam os erros de discretização (E) sobre as variáveis de interesse (ϕ), que são modelados por

$$E(\phi) = C_1 h^{P_L} + C_2 h^{P_2} + C_3 h^{P_3} + \dots \quad (1)$$

onde:

p_L, p_2, p_3, \dots = ordens verdadeiras de $E(\phi)$; geralmente números inteiros e positivos

ϕ = solução numérica sem erros de discretização, de iteração e de arredondamento

p_L = ordem assintótica de $E(\phi)$; $p_L \geq 1$; inclinação da curva de erro num gráfico $\log(|E|)$ versus $\log(h)$ para $h \rightarrow 0$

h = tamanho dos elementos da malha

Os erros de iteração (B) resultam de vários fatores: emprego de métodos iterativos para resolver um sistema de equações; problemas não lineares, onde os coeficientes do sistema de equações dependem da própria solução; modelos matemáticos constituídos por mais de uma equação, onde cada uma é resolvida separadamente, entre outros.

Os erros de programação (C) podem ser causados por: uso incorreto do modelo numérico na aproximação do modelo matemático; implementação incorreta do modelo numérico no programa computacional; pelo uso incorreto do programa computacional durante a obtenção da solução numérica; e por qualquer outra eventual fonte de erro.

Os erros de arredondamento (D) são causados pela representação finita das variáveis nas computações numéricas, que está ligada à precisão dos números apresentados pelo computador. O objetivo deste trabalho foi avaliar se este erro depende do software usado para gerar o código computacional e/ou do computador (hardware) empregado em sua execução. Os erros de arredondamento provocam perda de precisão dos números, que ocorre basicamente por dois motivos: um grande número de operações nos cálculos provoca perda de precisão no lado direito dos números e o cancelamento subtrativo nos cálculos que provoca perda de precisão no lado esquerdo dos números.

No entanto não há um trabalho específico que quantifique este efeito na solução de problemas de dinâmica dos fluidos computacional (CFD). Outra motivação para este trabalho foi o constante questionamento em bancas de mestrado/doutorado: se a precisão da

solução numérica depende do computador utilizado? Pois acredita-se que em diferentes computadores os programas computacionais se comportam de maneira diferente, ao se usar linguagens de programação científica como C++ ou Fortran.

2 Metodologia

Foram utilizados dois problemas de condução de calor (1D e 2D) para realizar a análise do erro de arredondamento verdadeiro e um problema de escoamento de fluidos compressíveis (Mach2D) para analisar a variação da solução numérica. O procedimento adotado para analisar o erro de arredondamento verdadeiro foi: discretizar a equação diferencial com diferenças finitas e aproximações de segunda ordem de acurácia, em seguida, aplicar condições de contorno tal que a solução gerada seja linear, desta forma, o modelo numérico resulta na solução analítica nos nós analisados, a qual é conhecida. No caso do Mach2D foi analisada a solução numérica obtida variando o sistema operacional e o solver utilizado para resolver o sistema de equações.

Todos os códigos computacionais foram compilados e executados no mesmo computador. Quando este procedimento não era adotado, resultados significativamente diferentes eram obtidos nos diferentes computadores testados, principalmente em problemas mais complexos. Sugere-se portanto que na execução de códigos computacionais siga esta mesma metodologia.

Em relação ao hardware, a Tab. 1 apresenta os computadores utilizados neste trabalho. Os resultados obtidos com cada um deles serão mencionados pelos seus respectivos nomes. Em relação ao software, foram usados cinco compiladores diferentes da linguagem computacional utilizada nos códigos (Fortran), são eles: (1) Microsoft Fortran PowerStation 4.0, (2) Compaq Visual Fortran 6.6, (3) Intel Fortran Compiler 11.1, (4) GNU Fortran e (5) Intel Fortran Composer XE 2013 14.0.

Tabela 1: Hardware utilizado nas simulações.

Nome	Modelo	Processador	Threads	Clock	RAM	OS
CFD-1	Intel	Pentium III	1	0,75	0,77	Win. XP
CFD-2	Intel	Pentium IV	1	2,40	1,00	Win. XP
CFD-5	Intel	Pentium IV	1	3,41	3,00	Win. XP
CFD-7	Intel	Core 2 6700	2	2,66	3,50	Win. XP
CFD-10	Intel	Xeon X5355	8	2,66	2,49	Win. XP
CFD-13	Intel	Core Q6600	4	2,40	3,25	Win. XP
CFD-14	AMD	Athlon 4200+	2	2,20	1,75	Win. XP
CFD-19	AMD	Athlon 5200B	2	2,69	3,76	Win. XP
CFD-20	Intel	Core E7500	2	2,93	2,00	Ubuntu
CFD-21	Intel	Xeon X5690	24	3,46	192	Ubuntu e Win. Xp

Clock é dado em GHz e a memória RAM em GB

2.1 Equação de Laplace, condução de calor unidimensional e bidimensional em regime permanente sem geração de calor

Para as simplificações utilizadas teremos a Eq. (2), que é a equação diferencial que modela o problema 1D e a Eq. (3) que modela o problema 2D.

$$\frac{d^2T}{dx^2} = 0 \quad (2)$$

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad (3)$$

O domínio 1D é uma barra com uma unidade de medida de comprimento e o 2D é uma placa quadrada com uma unidade de medida de aresta. As condições de contorno no problema 1D são as seguintes: $T(0) = 0$ e $T(1) = 1$ e no problema 2D: $T(0, y) = T(x, 0) = 0$; $T(1, y) = y$ e $T(x, 1) = x$. Isto permite obter soluções analíticas lineares para a incógnita dos problemas: no 1D $T(x) = x$ e no 2D $T(x, y) = xy$.

Os dados numéricos além dos citados no início do capítulo são: as condições de contorno são aplicadas nos nós dos contornos; no problema 1D, foi utilizado o solver TDMA (TriDiagonal Matrix Algorithm) [5], teremos portanto a solução direta do sistema de equações [5]. Já no problema 2D a parada do processo iterativo se dá com base num número fixo de iterações externas igual a 100, de forma a garantir que o processo iterativo termine no dobro de iterações necessárias para chegar ao erro de máquina. Isto ocorre pois utiliza-se o solver Gauss-Seidel com multigrid [4] para resolver o problema 2D. No problema 1D, foi avaliado o erro de arredondamento verdadeiro em 9 malhas distintas: 2, 10^1 , 10^2 , 10^3 , 10^4 , 10^5 , 10^6 , 10^7 e $2,5 \cdot 10^7$ nós. Já no 2D foi avaliado em 5 malhas: 32×32 , 128×128 , 512×512 , 2048×2048 e 8192×8192 nós.

As variáveis analisadas nos problemas 1D e 2D: Temperatura Central (T_c), variável primária local. Solução analítica: $T_{c,1D}(x = 1/2) = 1/2$ e $T_{c,2D}(x = 1/2, y = 1/2) = 1/4$; Temperatura Média (\bar{T}), variável secundária global. Solução analítica: $\bar{T}_{1D} = 1/2$ e $\bar{T}_{2D} = 1/4$. E Inclinação em $x=1$, variável secundária local. Solução analítica: $I_{1D}(x = 1) = 1$.

2.2 Avaliação da variação da solução obtida com o código Mach2D

O código computacional Mach2D utiliza o método dos volumes finitos para a resolução das equações diferenciais de conservação da massa, quantidade de movimento linear axial e transversal, a conservação da energia térmica e a equação de estado dos gases perfeitos. No caso deste trabalho foi simulado uma tubeira de motor-foguete e a variável analisada foi o coeficiente de descarga (C_d), definido pela razão entre o fluxo de massa numérico 2D pelo fluxo de massa teórico 1D. As equações citadas estão apresentadas abaixo:

$$c_p \frac{\partial(\rho u_k T)}{\partial x_k} = u_i \frac{\partial p}{\partial x_i} \quad \frac{\partial(\rho u_k u_i)}{\partial x_k} = - \frac{\partial p}{\partial x_i} \quad (4)$$

$$\frac{\partial(\rho u_k)}{\partial x_k} = 0 \quad p = \rho R T \quad (5)$$

onde:

ρ : Massa específica do gás; u_k : Componente da velocidade na direção k ; x_k : Direção k ; u_i : Componente da velocidade relativa a quantidade de movimento linear na direção i ; p : pressão do gás; c_p : Calor específico a pressão constante do gás; T : Temperatura do gás e R : Constante do gás. A Eq. (5) à esquerda representa a conservação da massa, a Eq. (4) à direita representa a conservação da quantidade de movimento linear em qualquer direção x_k , a Eq. (4) à direita representa a conservação da energia e a Eq. (5) à direita representa a equação de estado.

O código Mach2D foi utilizado neste trabalho para a simulação de escoamento invíscido de fluido compressível, contínuo, não reativo, termicamente perfeito, bidimensional axisimétrico e o esquema numérico é de primeira ordem de acurácia. Os solvers utilizados para comparação da variação do erro de arredondamento foram o ADI [2], que consiste em utilizar alternadamente o método TDMA em uma direção com a outra implícita e vice-versa e o MSI [3] que utiliza uma decomposição LU parcial do sistema de equações para resolver o sistema linear para a correção da variável dependente.

3 Resultados e discussão

Em relação ao hardware, foram realizadas simulações com o problema 1D e o problema 2D em todos os computadores mostrados na Tab. 1, com precisão simples e dupla. Não houve diferenças no erro de arredondamento obtido para o mesmo compilador utilizado (Microsoft Powerstation 4.0).

Em relação ao software, foram realizadas simulações no computador CFD-19 com o problema 1D, modificando as opções de compilação (Debug, Release, otimização ou não). Estas opções não influenciaram no erro de arredondamento obtido. Ainda no software, foram realizadas simulações com o problema 1D e 2D alterando-se os compiladores utilizados e a precisão dos cálculos empregada.

No problema 1D, utilizando-se precisão simples, em dois computadores testados (CFD-19 e CFD-13) houve diferenças no erro de arredondamento obtido para os diferentes compiladores utilizados; como exemplo as diferenças obtidas para a variável T_m estão mostradas na Fig. 1(a). O erro mostrado na figura é qualitativamente análogo para as outras três variáveis. Já na precisão dupla, não houve diferenças no erro de arredondamento obtido.

No problema 2D, utilizando-se precisão simples, foram obtidos erros de arredondamentos diferentes para os compiladores testados. A Fig. 1(b) mostra o comportamento da variável global temperatura média e a Fig. 1(c) mostra o comportamento da variável pontual temperatura central. Já na precisão dupla, das duas variáveis avaliadas, apenas a temperatura central apresentou diferença no erro de arredondamento e apenas no compilador Microsoft Fortran Powerstation 4.0, com mostrado na Fig. 1(d).

O último caso testado neste trabalho foi do Mach2D com precisão dupla: foi avaliado o resultado de C_d obtido com o compilador Intel Fortran Composer XE 2013 14.0 versão Windows e versão Linux na solução numérica e dois tipos de solvers aplicados na resolução do sistema de equações gerado: o método MSI e o método ADI. Foram simuladas seis malhas: 24x10, 48x20, 96x40, 192x80, 384x160 e 768x320 volumes no CFD-21, com diferentes

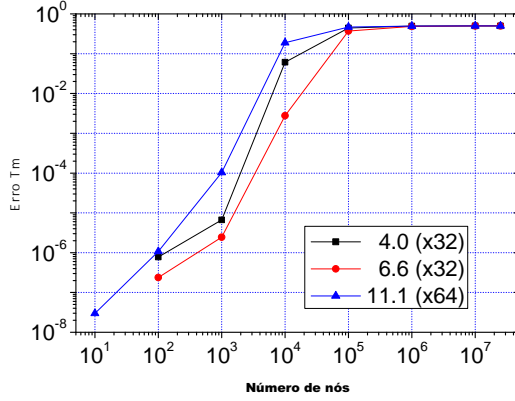
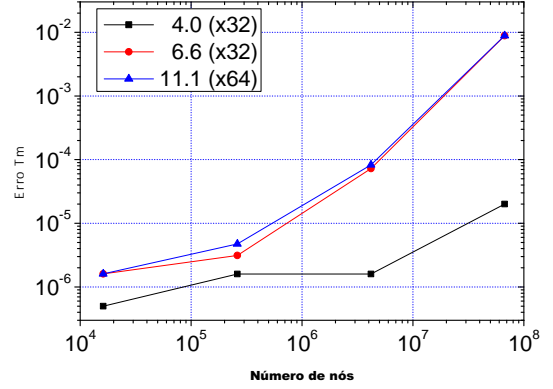
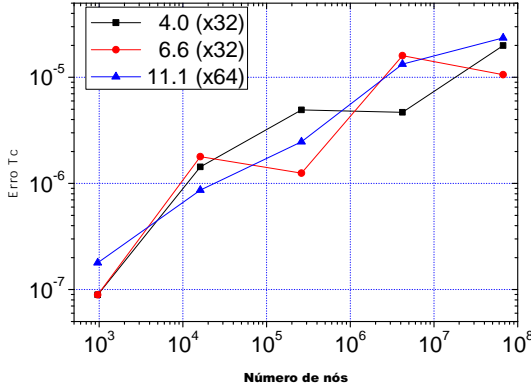
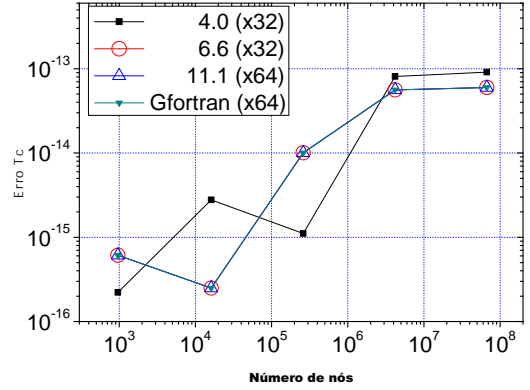
(a) \bar{T} 1D - Precisão Simples(b) \bar{T} 2D - Precisão Simples(c) T_c 2D - Precisão Simples(d) T_c 2D - Precisão Dupla

Figura 1: Comparação do erro de arredondamento verdadeiro das variáveis citadas para os diferentes compiladores

parâmetros de relaxação para conduzir a convergência. Não houve diferenças na solução numérica obtida em cada uma das malhas simuladas.

4 Conclusão

Neste trabalho foram estudados diferentes problemas: condução de calor 1D, 2D e escoamento de fluidos compressíveis. Os erros de arredondamento verdadeiros foram obtidos para os problemas de condução de calor e para o escoamento de fluidos foi analisada a variação da solução obtida. Com base nos testes realizados conclui-se que:

1 - Independente da precisão (dupla, simples) e dimensão (1D e 2D), o hardware e as opções de compilação não influenciam no erro de arredondamento verdadeiro, desde que a compilação e a execução do código sejam no mesmo computador.

2 - Utilizando precisão simples, independente da dimensão (1D e 2D), o erro de arredondamento verdadeiro apresenta diferenças não sistemáticas utilizando os diferentes

compiladores testados: (1) Microsoft Fortran Powerstation 4.0, (2) Compaq Visual Fortran 6.6, (3) Intel Fortran Compiler 11.1.

3 - Não houve variações na solução numérica obtida para a variável C_d no Mach2D utilizando-se dois sistemas operacionais (Windows e Linux) e dois solvers diferentes (ADI e MSI).

4 - Salvo o uso do compilador mais antigo (Microsoft Fortran Powerstation 4.0), cálculos com precisão dupla apresentaram o mesmo erro de arredondamento verdadeiro para todos os fatores analisados (Hardware e Software).

5 - A medida que o número de nós da malha cresce, o erro de arredondamento também cresce, pelo fato do método utilizado sofrer de cancelamento entre numerador e denominador.

Foi avaliado a repetitividade do erro de arredondamento para a variável T_c e as outras variáveis citadas neste trabalho. Observou-se que ao repetir as simulações, o erro de arredondamento obtido é sempre o mesmo, ou seja, neste caso o erro de arredondamento não é aleatório.

Em trabalhos futuros serão feitos estudos sistemáticos com ênfase nas opções de otimização dos compiladores utilizados e uso de outros problemas como equações elípticas e parabólicas.

Agradecimentos

Os autores agradecem ao programa UNIESPAÇO da AEB (Agência Espacial Brasileira), o CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) e a CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pelo apoio financeiro. O primeiro autor é bolsista da CAPES e o segundo é bolsista do CNPq.

Referências

- [1] C. H. Marchi, F. A. Schneider. Introdução à Mecânica Computacional. Disponível online: <http://ftp.demec.ufpr.br/disciplinas/TM797>, acesso em 11/11/2015, Curitiba, PR, Brasil, 2002.
- [2] D. W. Peaceman, H. H. Rachford. *The numerical solution of parabolic and elliptic differential equations*, J. Soc. Ind. Appl. Math, vol. 3, pp. 28-41, 1955.
- [3] G. Schneider, M. Zedan. *A modified strongly implicit procedure for the numerical solution of field problems*. Numerical Heat Transfer, vol. 4, pp. 1-19, 1981.
- [4] R. Suero, M. A. V. Pinto, C. H. Marchi, L. K. Araki e A. C. Alves. Otimização do método multigrid algébrico para as equações bidimensionais de Laplace e Poisson, VI Congresso Nacional de Engenharia Mecânica, Paraíba, Brasil, 2010.
- [5] H. K. Versteeg, W. Malalasekera. *An Introduction to Computational Fluid Dynamics - The Finite Volume Method*, Pearson, Prentice Hall, 2. ed., 2007.