

# Solução da equação de Laplace 2D com Repeated Richardson extrapolation

Carlos Henrique Marchi<sup>a</sup>, Leandro Alberto Novak<sup>b</sup>, Cosmo Damião Santiago<sup>c</sup>, Ana Paula da Silveira Vargas<sup>d</sup>

<sup>a</sup>Laboratory of Numerical Experimentation (LENA)  
Mechanical Engineering Department (DEMEC)  
Federal University of Paraná (UFPR)  
Caixa postal 19040, CEP 81531-980, Curitiba, Paraná, Brazil  
phone: (55) 41-3361-3126; fax: (55) 41-3361-3701  
[marchi@ufpr.br](mailto:marchi@ufpr.br)

<sup>b</sup>Programa de Pós-Graduação em Engenharia Mecânica, Universidade Federal do Paraná (UFPR)  
Curitiba, PR, Brazil  
[leandron@sanepar.com.br](mailto:leandron@sanepar.com.br)

<sup>c</sup>Universidade Tecnológica Federal do Paraná (UTFPR)  
Apucarana, Paraná, Brazil  
[cosmo@utfpr.edu.br](mailto:cosmo@utfpr.edu.br)

<sup>d</sup>Faculdades do Brasil (Unibrasil)  
Curitiba, Paraná, Brazil  
[vargas.apaulas@gmail.com](mailto:vargas.apaulas@gmail.com)

**Resumo.** Apresenta-se uma base teórica sobre repeated Richardson extrapolation (RRE) para reduzir e estimar o erro de discretização de soluções numéricas em CHT (Computational Heat Transfer) e CFD (Computational Fluid Dynamics). Um exemplo de aplicação é feito com a equação de Laplace 2D, empregando-se o método de diferenças finitas, domínio de cálculo discretizado com malhas uniformes, aproximações de segunda ordem de acurácia, diversas variáveis de interesse, condições de contorno de Dirichlet, malhas com até 8,193x8,193 nós, multigrid, precisões simples, dupla e quádrupla e até doze extrapolações de Richardson. Verificou-se que: (1) RRE reduz significativamente o erro de discretização (por exemplo, de 2.25E-07 para 3.19E-32 com nove extrapolações e malha 1,025x1,025, obtendo-se ordem de acurácia de 19.1); (2) o estimador de erro de Richardson funciona para resultados numéricos obtidos com RRE; (3) maior redução do erro de discretização com RRE é obtido ao se usar maior precisão nos cálculos, maior número de extrapolações, maior número de malhas e ordens corretas do erro; e (4) para se obter um dado valor de erro, o tempo de CPU e a memória RAM necessários para a solução com RRE são muito menores do que sem RRE.

**Palavras-chave:** erro de discretização, erro numérico, estimador de erro, diferenças finitas, ordem de acurácia, verificação.

## 1. Introdução

Melhorar técnicas que permitam reduzir o erro numérico é importante para diminuir o custo computacional (memória e tempo de CPU) envolvido na obtenção de soluções numéricas. Algumas formas para reduzir o erro numérico causado por erros de discretização são: (a) refinar a malha, cuja desvantagem é aumentar o custo computacional; (b) aumentar a ordem de acurácia das aproximações numéricas, cuja desvantagem é aumentar a complexidade do modelo numérico; e (c) usar técnicas de extrapolação, uma das quais é a extrapolação de Richardson (RE) [1-5]. Para se usar RE, é necessário ter a solução numérica da variável de interesse em duas malhas com número de nós diferentes. RE também é usado como estimador do erro de discretização [6] ou é a base para

outros estimadores, como o GCI (Grid Convergence Index) [7]. Em [8] encontra-se um extenso estudo sobre processos de extrapolação em análise numérica; quase todos são baseados em RE ou variantes dele.

Repeated Richardson extrapolation (RRE) [5] consiste em aplicar sucessivas vezes a extrapolação de Richardson. Portanto, para se usar RRE, é necessário ter a solução numérica da variável de interesse em três ou mais malhas com número de nós diferentes; com isso, é possível se fazer dois ou mais RE. Em outras áreas [4,5], RRE é usado com muitos RE; mas isso não ocorre em CHT e CFD. Os trabalhos [2,9] aplicaram RRE com dois RE em problemas bidimensionais (2D) de condução de calor. RRE foi aplicado com até três RE em [10-13] na solução das equações de Navier-Stokes 2D. Mas em CFD, o mais comum é aplicar RE em vez de RRE; por exemplo: [14] em vários tipos de problemas 1D; [15,16] na equação de Poisson 2D; na equação de advecção-difusão 2D [17] e 3D [18]; nas equações de Navier-Stokes 2D [19]; em convecção natural 2D [20]; e em escoamentos turbulentos 2D [19,21,22]. Portanto, até hoje, RE e principalmente RRE foram muito pouco empregados em CHT e CFD para reduzir o erro de discretização, provavelmente pelas dificuldades relatadas em [2-4,7,12,14,19,21-26].

Com base em [27-30] e no presente trabalho, as principais vantagens do uso de RRE são: (1) reduz muito o erro de discretização; (2) é um pós-processamento simples, ou seja, não interfere diretamente na obtenção da solução numérica em uma dada malha  $h$ ; (3) seu custo computacional é muito baixo em termos de memória e tempo de CPU; (4) pode ser aplicado a códigos computacionais já existentes ou a resultados já obtidos; (5) aplica-se a diversos métodos numéricos, aproximações numéricas e variáveis de interesse; (6) independe de análises a priori e do conhecimento da solução analítica do problema; e (7) mesmo com soluções numéricas sem RRE que são de baixa ordem (um ou dois), consegue-se soluções numéricas com RRE que são de altíssima ordem (maior do que dez). Conforme será mostrado no presente trabalho, RRE pode ser usado de duas formas. A primeira, para se obter o mesmo erro de discretização com uma malha que tem muito menos nós, resultando na redução do custo computacional. Esta forma é indicada especialmente para aplicações práticas e validações. A segunda, para reduzir o erro de discretização em uma malha com o mesmo número de nós, resultando em erros muito menores e maior confiabilidade da solução. Esta forma é indicada especialmente para se gerar benchmarks.

Os objetivos do presente trabalho são: (i) apresentar uma base teórica sobre RRE com a finalidade de reduzir e estimar o erro de discretização em CFD; (ii) mostrar que o uso de RRE é extremamente efetivo na redução do erro; (iii) testar um estimador de erro para RRE; (iv) mostrar os efeitos sobre RRE causados pelo tipo de aproximação numérica, tipo de variável de interesse, precisão dos cálculos, número de extrapolações, número de malhas e ordens do erro; e (v) mostrar o custo computacional de RRE. Para atingir estes objetivos, aplica-se a teoria de RRE na solução da equação de Laplace 2D com até doze RE.

## 2. Modelo matemático

O modelo matemático considerado neste trabalho é a equação de Laplace bidimensional com condições de contorno de Dirichlet, definido por

$$\begin{cases} \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0, & 0 < x, y < 1, \\ T(x,1) = \sin(\pi x), & T(x,0) = T(0, y) = T(1, y) = 0, \end{cases} \quad (1)$$

onde  $x$  e  $y$  são as direções coordenadas e  $T$  representa a temperatura. Fisicamente, esta equação pode modelar um problema de condução de calor em uma placa plana com propriedades constantes, em regime permanente, bem como diversos outros fenômenos físicos. A solução analítica da Eq. (1) é  $T(x,y) = \sin(\pi x) \sinh(\pi y) / \sinh(\pi)$ .

As variáveis de interesse deste trabalho, ou seja, as variáveis para as quais aplica-se a teoria de RRE são: (i) a temperatura no centro do domínio, isto é, em  $x=y=1/2$ , simbolizada por  $T_c$ ; (ii) o perfil de temperaturas em  $x=1/2$ , simbolizado por  $T_y$ ; (iii) a média do campo de temperaturas, simbolizada por  $T_m$ ; (iv) a média da norma  $l_1$  do erro numérico, simbolizada por  $L$ ; e as taxas de transferência de calor nos contornos de  $x=1$  (v) e  $y=1$  (vi), simbolizadas respectivamente por  $Q_e$  e  $Q_n$ . A justificativa da escolha destas variáveis é apresentada na próxima seção.

As variáveis  $T_m$  e  $Q_e$  são definidas matematicamente por

$$T_m = \int_0^1 \int_0^1 T(x, y) dx dy, \quad (2)$$

$$Q_e = -k \int_0^1 \left( \frac{\partial T}{\partial x} \right)_{x=1} dy, \quad (3)$$

onde  $k$  é a condutividade térmica do material, com valor unitário.

### 3. Modelo numérico

#### 3.1 Solução numérica sem extrapolação

A Eq. (1) é discretizada com o método de diferenças finitas [31], malhas uniformes e o esquema CDS (Central Differencing Scheme) de segunda ordem de acurácia, resultando em

$$\frac{(T_{i-1,j} - 2T_{i,j} + T_{i+1,j})}{h^2} + \frac{(T_{i,j-1} - 2T_{i,j} + T_{i,j+1})}{h^2} = 0, \quad (4)$$

onde  $i$  e  $j$  representam cada nó da malha, e  $h$  é a distância entre dois nós consecutivos da malha em cada direção.

Escrevendo-se a Eq. (4) para todos os  $N$  nós da malha, obtém-se um sistema de equações algébricas que é resolvido através do *Modified Strongly Implicit Method* (MSI) [32]. Para acelerar a convergência, utilizou-se o método *multigrid* geométrico [33], com *Full Approximation Scheme* (FAS), ciclo V, restrição por injeção, prolongação por interpolação bilinear e razão de engrossamento dois.

Utilizou-se o valor nulo como estimativa inicial da solução de cada problema. O número de vezes que o ciclo V do método *multigrid* foi repetido é denominado de iterações externas. O processo iterativo foi levado até ser atingido o erro de arredondamento de máquina para a solução numérica das variáveis  $T_c$  e  $T_m$ , visando eliminar a contribuição do erro de iteração sobre o erro numérico.

Foram implementados três programas computacionais em linguagem Fortran 95, versão 9.1 da Intel, um usando precisão simples (Real\*4), outro com precisão dupla (Real\*8) e um terceiro com precisão quádrupla (Real\*16). As simulações foram realizadas em um núcleo de um microcomputador com processador Intel Xeon Quad Core X5355, 2,66 GHz, 16 GB RAM e sistema operacional Windows xp 64 bits.

A solução numérica da variável  $T_c$  foi obtida diretamente do nó central de cada malha após a obtenção da solução numérica da Eq. (4), já que sempre foram usadas malhas com número ímpar de nós. A solução numérica da variável  $T_y$  foi obtida diretamente de 15 nós igualmente espaçados da malha após a obtenção da solução numérica da Eq. (4), para malhas com 17x17 nós ou maiores. A solução numérica da Eq. (2), para  $T_m$ , foi obtida através de integração numérica pela regra do trapézio [34]. A solução numérica da Eq. (3), para  $Q_e$ , foi obtida através de integração numérica pela regra do trapézio antecedida pelo uso do esquema UDS (*Upstream Differencing Scheme*) [31]

de segunda ordem de acurácia em cada nó do contorno. A variável  $Qn$  foi obtida de forma análoga a  $Qe$ .

As variáveis  $Tc$  e  $Ty$  foram escolhidas para verificar o efeito da extrapolação de Richardson em 15 nós específicos das malhas e por serem as variáveis dependentes da equação diferencial do problema, Eq. (1), ou seja, a variável primária do problema, cuja solução envolve duas aproximações numéricas do tipo CDS, conforme mostrado na Eq. (4). A variável  $L$  foi escolhida para monitorar, através de uma única variável, o comportamento do erro numérico do campo inteiro de  $T$  com a diminuição de  $h$ . As variáveis  $Tm$ ,  $Qe$  e  $Qn$  foram usadas para verificar o efeito da extrapolação de Richardson em variáveis secundárias, isto é, obtidas a partir de  $T$ , e envolvendo uma e duas aproximações numéricas adicionais àquelas usadas para obter  $T$ .

A solução numérica ( $\phi$ ) sem extrapolação, de cada variável de interesse descrita acima, foi obtida empregando-se um conjunto de malhas  $g = [1, G]$ , onde  $g = 1$  é a malha mais grossa, que tem o maior  $h$ , e  $g = G$  é a malha mais fina, que tem o menor  $h$ .

### 3.2 Solução numérica com múltiplas extrapolações

Para cada variável de interesse, a solução numérica ( $\phi$ ) na malha  $g$  com  $m$  extrapolações de Richardson é dada por

$$\phi_{g,m} = \phi_{g,m-1} + \frac{\phi_{g,m-1} - \phi_{g-1,m-1}}{r^{p_{m-1}} - 1}, \quad (5)$$

onde  $r = h_{g-1} / h_g$  é a razão de refino de malha, e a variável  $p_m$  representa as ordens verdadeiras [6] do erro de discretização, que podem ser obtidas conforme as explicações dadas na próxima seção (4). A Eq. (5) é válida para  $g = [2, G]$  e  $m = [1, g-1]$ . A Eq. (5) foi obtida estendendo-se ao caso de  $m > 1$  os trabalhos [1,2,6,7] que apresentam equações equivalentes ao caso de  $m = 1$ .

Deve-se perceber que para se obter cada valor de  $\phi_{g,m}$  é necessário ter duas soluções numéricas  $\phi$  em duas malhas ( $g$  e  $g-1$ ) em  $m-1$ . Para qualquer  $g$ ,  $m = 0$  representa a solução numérica de  $\phi$  sem qualquer extrapolação, obtida conforme descrito na subseção anterior (3.1). Para  $m = 1$ , tem-se a extrapolação de Richardson padrão ou simples, comumente usada para se estimar o erro de discretização [6] ou para melhorar a solução de cada malha  $g$  [16]. Para um dado valor de  $g$ , é possível aplicar a Eq. (5) até  $g-1$  vezes, realizando  $m$  extrapolações de Richardson.

### 4. Erro numérico e sua estimativa

Para uma dada variável de interesse, o erro numérico ( $E$ ) da solução numérica ( $\phi$ ) pode ser definido por

$$E(\phi) = \Phi - \phi, \quad (6)$$

onde  $\Phi$  é a solução analítica exata da variável de interesse. Considera-se no presente trabalho que o erro numérico é causado por quatro fontes [6]: erros de discretização, de iteração, de arredondamento, e de programação.

Quando o erro numérico é causado apenas pelo erro de discretização, tem-se [5,8]

$$E(\phi) = C_0 h^{p_0} + C_1 h^{p_1} + C_2 h^{p_2} + \dots = \sum_{m=0}^{\infty} C_m h^{p_m}, \quad (7)$$

onde  $C_0, C_1, C_2, \dots$  são coeficientes que dependem de  $\Phi$  e suas derivadas, bem como das variáveis independentes, mas independentem de  $h$ ; e  $p_0, p_1, p_2, \dots$  são as ordens verdadeiras de  $E(\phi)$ , cujo conjunto é representado por  $p_m$ .

Geralmente os valores de  $p_m$  são números inteiros e positivos [8] com  $0 < p_0 < p_1 < p_2 < \dots$ , e que constituem uma progressão aritmética de razão  $q = p_1 - p_0$ . Além disso,  $p_0$  é denominado de ordem assintótica ou de acurácia de  $E(\phi)$  ou da solução numérica  $\phi$ . Os valores de  $p_m$  podem ser obtidos a priori com um procedimento que emprega a série de Taylor [31], ou a posteriori conforme explicado na próxima subsecção (4.1).

A ordem de acurácia teórica da solução numérica de  $\phi$ , com ordens  $p_m$  constituindo uma progressão aritmética, e  $m$  extrapolações é

$$p_m = p_0 + m(p_1 - p_0), \quad (8)$$

sendo esta equação válida para  $g = [1, G]$  e  $m = [0, g-1]$ .

#### 4.1 Obtenção de ordens do erro

Os valores de  $p_m$ , obtidos a priori, podem ser confirmados a posteriori com o conceito de ordem efetiva ( $p_E$ ) [35] do erro de discretização, que generalizado para repeated Richardson extrapolation é dado por

$$(p_E)_{g,m} = \frac{\log \left[ \frac{E(\phi_{g-1,m})}{E(\phi_{g,m})} \right]}{\log(r)}, \quad (9)$$

onde  $E$  é calculado através da Eq. (6).

A Eq. (9) é válida para  $g = [2, G]$  e  $m = [0, g-2]$ . As demais definições da subsecção 3.2 se aplicam aqui. Deve-se perceber que para se obter cada valor de  $p_E$  é necessário conhecer o erro da solução numérica em duas malhas. Teoricamente, à medida que  $h \rightarrow 0$ , os valores de  $(p_E)_{g,m}$  devem tender à ordem verdadeira ( $p_m$ ) dada pela Eq. (8). A Eq. (9) foi obtida estendendo-se ao caso de  $m \geq 1$  o trabalho [35] que apresenta equação equivalente ao caso de  $m = 0$ .

Conforme a Eq. (9),  $p_E$  é função do erro da variável de interesse. Portanto, esta equação não pode ser aplicada em problemas cuja solução analítica é desconhecida. Além disso, a Eq. (9) não deve ser usada quando se quer confirmar a posteriori os valores de  $p_m$  obtidos a priori e sem utilizar soluções numéricas extrapoladas com os próprios valores de  $p_m$  que foram obtidos a priori. Nestes casos, pode-se usar o conceito de ordem aparente ou observada ( $p_U$ ) [6] da estimativa do erro de discretização, que generalizado para repeated Richardson extrapolation é dado por

$$(p_U)_{g,m} = \frac{\log \left[ \frac{\theta_{g-1,m} - \theta_{g-2,m}}{\theta_{g,m} - \theta_{g-1,m}} \right]}{\log(r)}, \quad (10)$$

onde a variável  $\theta$  é explicada mais abaixo. A Eq. (10) foi obtida estendendo-se ao caso de  $m \geq 1$  o trabalho [6] que apresenta equação equivalente ao caso de  $m = 0$ .

A Eq. (10) é válida para  $g = [3, G]$ ,  $m = [0, \text{Int}((g-3)/2)]$  e  $r$  constante entre as três malhas, isto é,  $r = h_{g-2}/h_{g-1} = h_{g-1}/h_g$ , onde  $\text{Int}(a)$  representa a parte inteira de  $a$ . As demais definições da subsecção 3.2 se aplicam aqui. Também é possível obter-se  $p_U$  para  $r$  variável entre três malhas; este caso é abordado em [3,35]. À medida que  $h \rightarrow 0$ , os valores de  $(p_U)_{g,m}$  devem tender à ordem verdadeira ( $p_m$ ) do respectivo nível de extrapolação ( $m$ ) da Eq. (7), independentemente de qualquer análise a priori. Estes valores de  $p_m$ , obtidos a posteriori, é que devem ser usados na Eq. (5).

Deve-se perceber que para ser obtido cada valor de  $p_U$  na Eq. (10) são necessárias soluções numéricas relacionadas a três malhas. Estas soluções não são obtidas com a Eq. (5) porque ela

admite que os valores de  $p_m$  são conhecidos. Para se obter  $p_m$  a posteriori independente de  $p_m$  a priori, em vez de se usar a variável  $\phi$ , obtida com a Eq. (5), usa-se a nova variável  $\theta$  que é calculada através de

$$\theta_{g,m} = \theta_{g,m-1} + \frac{\theta_{g,m-1} - \theta_{g-1,m-1}}{r^{(p_U)_{g,m-1}} - 1}. \quad (11)$$

A Eq. (11) é válida para  $g = [3, G]$  e  $m = [1, \text{Int}((g-1)/2)]$ . As demais definições da subseção 3.2 se aplicam aqui. Além disso, para  $m = 0$ , a Eq. (11) não se aplica; neste caso, tem-se  $\theta_{g,0} = \phi_{g,0}$  onde  $\phi_{g,0}$  é a solução numérica obtida sem qualquer extrapolação conforme descrito na subseção 3.1.

Para se caracterizar adequadamente a série de valores de  $p_m$ , idealmente deve-se determinar pelo menos as três primeiras ordens:  $p_0$ ,  $p_1$  e  $p_2$ . Se isso não for possível, como normalmente os valores de  $p_m$  constituem uma progressão aritmética, basta determinar  $p_0$  e  $p_1$ , sendo os demais valores função destes dois primeiros da série. Quando apenas  $p_0$  é determinado, pode-se arbitrar os demais valores. Finalmente no caso de nenhum valor ser determinado, pode-se usar a série com os menores valores: 1, 2, 3, ... O impacto da escolha arbitrária dos valores de  $p_m$  será abordado na seção de resultados.

No presente trabalho, os valores de  $p_m$  usados na Eq. (5) são obtidos de  $(p_U)_{g,m}$  da Eq. (10) para  $m = [0, 2]$ . Eles são números inteiros e positivos que são extraídos, para cada nível de extrapolação ( $m$ ), da tendência dos valores de  $(p_U)_{g,m}$  quando  $h \rightarrow 0$ .

## 4.2 Estimador de erro

Para qualquer variável de interesse, uma estimativa ( $U$ ) do erro de discretização da solução numérica ( $\phi$ ) na malha  $g$ , com  $m$  extrapolações de Richardson, é dada por

$$U(\phi_{g,m}) = \frac{\phi_{g,m} - \phi_{g-1,m}}{r^{p_m} - 1}. \quad (12)$$

A Eq. (12) é válida para  $g = [2, G]$  e  $m = [0, g-2]$ . As demais definições da subseção 3.2 se aplicam aqui. A Eq. (12) foi obtida estendendo-se ao caso de  $m > 1$  os trabalhos [1,2,6,7] que apresentam equações equivalentes ao caso de  $m = 1$ .

## 5. Resultados

Com precisão dupla (Real\*8), foram obtidas soluções numéricas para as variáveis de interesse em malhas com 3x3, 5x5, 9x9, ... até 8,193x8,193 nós; portanto  $G = 13$  malhas. Já com precisões simples (Real\*4) e quádrupla (Real\*16), a malha mais fina foi de 4,097x4,097 nós; portanto  $G = 12$  malhas. Embora tenham sido feitas 50 e 20 iterações externas respectivamente com as precisões dupla e quádrupla, o erro de arredondamento de máquina foi atingido respectivamente com apenas seis e doze iterações externas. Para se atingir o erro de arredondamento de máquina, o tempo máximo de CPU foi de 20 min e 1 h 5 min, respectivamente para as precisões dupla e quádrupla. O número de algarismos significativos das soluções numéricas sem extrapolação é no mínimo de 12 e 30, respectivamente para as precisões dupla e quádrupla; isto significa que nestes algarismos as soluções não têm erro de arredondamento. Para se medir o erro numérico com a Eq. (6), a solução analítica ( $\Phi$ ) de cada variável de interesse foi obtida através do software Maple com 30 e 64 algarismos, respectivamente para as soluções numéricas obtidas com as precisões dupla e quádrupla. São apresentados abaixo apenas alguns dos resultados do trabalho. Os resultados omitidos tiveram o mesmo comportamento qualitativo daqueles apresentados abaixo.

## 5.1 Redução e estimativa do erro com RRE

As Figs. 1, 2 e 3 apresentam respectivamente para as variáveis  $Tc$ ,  $Tm$  e  $Qe$ , em função do tamanho da malha ( $h$ ), o módulo dos seguintes resultados obtidos com precisão quádrupla e  $G = 12$  malhas:

- $Eh$ , calculado com a Eq. (6), que é o erro da solução numérica de  $\phi$  sem qualquer extrapolação ( $m=0$ ) e obtida conforme descrito na subseção 3.1 para  $g = [1, G]$ .
- $Uh$ , calculado com a Eq. (12) para  $g = [2, G]$  e  $m = 0$ , e que é a estimativa de  $Eh$ .
- $Em2$ , calculado com a Eq. (6), que é o erro da solução numérica de  $\phi$  com extrapolação, obtida conforme descrito na subseção 3.2 através da Eq. (5) para  $g = [3, G]$  e  $m = g-2$ .
- $Um2$ , calculado com a Eq. (12) para  $g = [3, G]$  e  $m = g-2$ , e que é a estimativa de  $Em2$ .
- $Em1$ , calculado com a Eq. (6), que é o erro da solução numérica de  $\phi$  com extrapolação, obtida conforme descrito na subseção 3.2 através da Eq. (5) para  $g = [2, G]$  e  $m = g-1$ .
- $d\phi = \phi_{g,m} - \phi_{g-1,m}$ , que é o numerador da Eq. (12), onde  $\phi_{g,m}$  é obtido através da Eq. (5), conforme descrito na subseção 3.2, para  $g = [2, G]$  e  $m = g-2$ .

A estimativa do erro de soluções numéricas obtidas com RRE só é possível na segunda malha mais grossa em cada nível de extrapolação  $m$ . Devido a isso, são apresentados os resultados de  $Em2$  e  $Um2$ . Embora não possa ser feita de forma consistente a estimativa do seu erro, o resultado de  $Em1$  é apresentado para mostrar o efeito do nível de extrapolação para uma mesma malha  $h$ .

Pode-se ver nestas três figuras que em geral: (i)  $Uh$  coincide visualmente com  $Eh$  em qualquer  $h$ ; (ii)  $Um2$  está próximo de  $Em2$  em qualquer  $h$ , com valores levemente inferiores a  $Em2$ ; (iii)  $Em1$  é um pouco menor do que  $Em2$ ; (iv)  $d\phi$  é muito maior do que  $Em2$ ; e (v)  $Em1$  e  $Em2$  são muitíssimo menores do que  $Eh$  e se tornam cada vez menores em relação a  $Eh$  quanto menor é  $h$ . Além disso, nas malhas mais finas, com menor  $h$ , devido ao fato do erro de arredondamento ( $\epsilon\pi$ ) ser maior do que o erro de discretização: RRE perde o seu efeito de reduzir o erro;  $Um2$ , ou seja, a Eq. (12) é inútil; e  $d\phi$  pode ser usado como um estimador para  $Em1$  e  $Em2$ .

Com base nestes resultados, conclui-se que: (a) RRE é extremamente eficiente para reduzir o erro de discretização, exceto no caso de malhas muito grossas; (b) em cada malha  $g$  com espaçamento  $h$ , o menor erro com RRE ocorre para  $m = g-1$ , isto é,  $Em1$ ; (c) RRE reduz com a mesma eficiência o erro de variáveis primárias ou secundárias, não importando o número de aproximações numéricas usadas; (d) o estimador de Richardson, dado pela Eq. (12), é acurado para prever o erro de discretização de soluções numéricas obtidas através da Eq. (5) com RRE; (e)  $d\phi$  pode ser usado como um estimador confiável, porém inacurado, do erro de discretização de soluções numéricas obtidas com RRE; e (f) os erros de arredondamento afetam o desempenho de RRE em malhas muito finas quando eles prevalecem sobre os erros de discretização.

As Tabelas 1 e 2 ajudam a mostrar a eficiência de RRE para reduzir o erro da variável  $Tc$ , obtida com precisão quádrupla. Para três malhas específicas, a Tabela 1 mostra o efeito de RRE sobre a redução do erro de discretização, medido pela razão  $Eh/Em1$ , ao se refinar a malha e aumentar o número de extrapolações ( $m$ ). Por exemplo, mesmo em uma malha tão grossa quanto  $17 \times 17$ , com apenas três extrapolações ( $m=3$ ), o erro já é reduzido em mais de três mil vezes. Portanto, RRE pode ser usado para se obter soluções benchmark.

**Tabela 1. Redução do erro de  $Tc$  para malhas específicas (Real\*16).**

Malha	17x17	129x129	1,025x1,025
$h$	6.25E-02	7.81E-03	9.77E-04
$ Eh $	9.20E-04	1.44E-05	2.25E-07
$ Em1 $	2.71E-07	4.56E-17	3.19E-32
$m$ para $Em1$	3	6	9
$ Eh  /  Em1 $	3.39E+03	3.16E+11	7.05E+24

Para três níveis de erro específicos, a Tabela 2 mostra o efeito de RRE sobre a redução do número de nós de uma malha para se obter o mesmo erro de discretização. Por exemplo, para o nível de erro  $5.00E-07$ , sem RRE é necessário usar a malha  $1,025 \times 1,025$  para atingir este nível de erro, e a malha  $17 \times 17$  com RRE. Assim, com RRE é necessário uma malha mais de três mil vezes menor do que sem RRE. Esta razão entre o número de nós das malhas de  $Eh$  e  $Em1$  indica o nível de redução do custo computacional (memória e tempo de CPU) ao seu usar RRE em relação a não usar RRE. Portanto, RRE pode ser empregado em aplicações práticas com grande eficiência computacional.

**Tabela 2. Redução de nós de malha para erros específicos de  $Tc$  (Real\*16).**

Nível do erro	5.00E-03	5.00E-05	5.00E-07
Malha de $Eh$	9x9	65x65	1,025x1,025
$ Eh $	3.65E-03	5.76E-05	2.25E-07
Malha de $Em1$	5x5	9x9	17x17
$ Em1 $	1.92E-03	3.84E-05	2.71E-07
$m$ para $Em1$	1	2	3
Razão entre o número de nós das malhas de $Eh$ e $Em1$	3.24E+00	5.22E+01	3.64E+03

## 5.2 Efeito da precisão dos cálculos

Em função do tamanho da malha ( $h$ ), a Fig. 4 apresenta para a variável  $Tc$  o valor do módulo de  $Eh$  das soluções numéricas sem extrapolação obtidas com as precisões simples (Real\*4), dupla (Real\*8) e quádrupla (Real\*16). Esta figura também apresenta para a variável  $Tc$  o valor do módulo de  $Em1$  das soluções numéricas extrapoladas e obtidas com as mesmas três precisões.  $Eh$  e  $Em1$  seguem as mesmas definições da subseção 5.1. Nesta figura são apresentados resultados para as malhas  $g = [1, G=12]$ .

Pode-se observar na Fig. 4 que: (i) o erro de arredondamento ( $E\pi$ ) passa a ser a principal fonte do erro numérico abaixo de um valor de  $h$ , que depende da precisão usada nos cálculos bem como se a solução numérica foi obtida sem ou com extrapolação; (ii) soluções numéricas com extrapolação são muito mais afetadas por  $E\pi$  do que sem extrapolação, isto é, o valor de  $h$  abaixo do qual as extrapolações são afetadas por  $E\pi$  é muito maior do que o  $h$  das soluções numéricas sem extrapolação; e (iii) quanto maior é a precisão usada nos cálculos, maior é a eficiência de RRE em reduzir o erro de discretização.

## 5.3 Efeito do número de extrapolações ( $m$ )

A Fig. 5 mostra o efeito do número de extrapolações ( $m$ ) sobre o módulo do erro de  $Tc$  versus o tamanho da malha ( $h$ ) para resultados obtidos com precisão quádrupla (Real\*16). Nesta figura são apresentados resultados para as malhas  $g = [1, G=12]$ . Cada curva representa um nível de extrapolação  $m = [0, g-1]$ . Deve-se lembrar que  $m = 0$  refere-se a resultados do erro de  $Tc$  sem extrapolação, obtidos conforme descrito na subseção 3.1; e para  $m \geq 1$  tem-se resultados do erro de  $Tc$  com extrapolação, obtidos conforme descrito na subseção 3.2 com a Eq. (5).

É evidente na Fig. 5 que quanto maior  $m$ , para um mesmo  $h$ , maior é a eficiência de RRE em reduzir o erro de discretização. Além disso, para um mesmo  $m$ , quanto menor é  $h$ , menor é o erro; este resultado é bem conhecido para  $m=0$ . Porém, nas malhas mais finas, isto é, com os menores valores de  $h$ , que neste caso ocorrem para  $m \geq 5$ ,  $E\pi$  já reduz a eficiência de RRE. O menor erro ( $3.19E-32$ ) foi obtido com  $m = 9$  em seu maior  $h$ , ou seja, na malha  $1,025 \times 1,025$  ( $g = 10$ ).

A Fig. 6 apresenta a ordem efetiva ( $p_E$ ) do erro de  $Tc$  versus tamanho da malha ( $h$ ) e número de extrapolações ( $m$ ) referentes aos erros da Fig. 5. Os valores de  $p_E$  foram calculados com a Eq. (9), conforme descrito na subseção 4.1, para  $g = [2, G]$  e  $m = [0, g-2]$ . Cada curva representa um nível de extrapolação  $m$ . O maior valor de  $p_E$  (17.4) foi obtido com  $m = 8$  em seu maior  $h$ , ou seja, na malha



1,025x1,025 ( $g = 10$ ). Nas malhas mais finas, isto é, com os menores valores de  $h$ , que neste caso ocorrem para  $m \geq 4$ ,  $E\pi$  afeta o cálculo de  $p_E$ , resultando em valores que devem ser desconsiderados. Nesta figura estão bem claros os seguintes valores de  $p_m$ : 2, 4, 6, 8, 10 e 12; e há tendência das curvas para os valores 14 e 16.

A Fig. 7 apresenta a ordem efetiva ( $p_E$ ) do erro de  $Tc$  versus tamanho da malha ( $h$ ) para as curvas de  $Em2$  e  $Em1$  da Fig. 1, denominadas respectivamente de  $pE2$  e  $pE1$ . Também é apresentada nesta figura a curva teórica de ordem, dada pela Eq. (8), com  $p_0 = 2$  e  $p_1 = 4$ . Desconsiderando-se os dois valores menores de  $h$  nos quais  $E\pi$  domina o erro, pode-se ver que o desempenho de RRE em geral acompanha a curva teórica, com uma diferença em cada  $h$  entre 0.35 e 1.38. O maior valor de  $p_E$  (19.1) foi obtido entre  $m = 8$  e 9 na malha 1,025x1,025 ( $g = 10$ ).

#### 5.4 Verificação e efeito das ordens do erro

A Fig. 8 apresenta a ordem aparente ( $p_U$ ) da estimativa do erro de  $Tc$  versus tamanho da malha ( $h$ ). Os valores de  $Tc$  foram obtidos com a Eq. (11) e precisão quádrupla, e os de  $p_U$  com a Eq. (10), conforme a teoria da subseção 4.1. São mostrados resultados para  $g = [3,12]$  e os níveis de extrapolação  $m = 0, 1$  e 2. Com esses resultados, fica claro que à medida que  $h \rightarrow 0$ , os valores de  $p_U$  tendem às ordens 2, 4 e 6.

A Fig. 9 mostra o efeito da escolha arbitrária dos valores de  $p_m$  sobre o desempenho de RRE. Visando servirem de referência, nesta figura são rerepresentadas as curvas de  $Eh$  e  $Em1$  (denota por  $p_m=2,4,6...$  right) da Fig. 1, para a variável  $Tc$  obtida com precisão quádrupla. Além disso, são mostradas curvas de  $Em1$  que foram obtidas com três séries de valores arbitrários para  $p_m$ . O uso desses valores arbitrários faz com que a eficiência de RRE seja reduzida consideravelmente, em relação à curva correta, até a quase eliminação do efeito de RRE quando  $Em1$  quase coincide com  $Eh$ .

#### 5.5 Eficiência computacional de RRE

A Fig. 10 apresenta o módulo do erro versus o tempo de CPU (em segundos) necessário para se obter a solução da variável  $Tc$  até ser atingido  $E\pi$ . Nota-se nesta figura que os erros ( $Eh$ ) das soluções sem extrapolação diminuem quase linearmente com o aumento do tempo de CPU. Para um mesmo valor de  $h$ ,  $Eh$  com Real\*16 é igual a  $Eh$  com Real\*8, mas o tempo de CPU de  $Eh$  com Real\*16 é cerca de 20 vezes maior do que com Real\*8. Porém o número de iterações para se atingir  $E\pi$  com Real\*16 é o dobro de Real\*8. Portanto, por iteração, o tempo de CPU da solução com Real\*16 é cerca de 10 vezes maior do que com Real\*8.

Na Fig. 10, o tempo de CPU do erro com RRE ( $Em$ ) em uma dada malha soma os tempos de CPU de todas as malhas mais grossas, pois elas são necessárias para a obtenção de  $Em$ ; por exemplo: o tempo de CPU plotado na figura correspondente a  $Em$  na malha 1,025x1,025 contabiliza também o tempo de CPU das malhas 513x513, 257x257 até a 3x3, ou seja, 10 malhas. Para  $Em1$  com Real\*8, são mostrados os resultados do tempo de CPU apenas para a malha 129x129 e maiores, nas quais  $E\pi$  já supera o erro de discretização; nas malhas menores, não se conseguiu medir o tempo de CPU por ele ser muito pequeno.

Pode-se ver na Fig. 10 que o incremento do tempo de CPU com a redução de  $Em1$  (Real\*16) diminui quanto menor é o erro. Esse é um comportamento melhor do que os das curvas de  $Eh$ . Além disso, para um dado valor de erro, o tempo de CPU de  $Em1$ , mesmo com Real\*16, é muito menor do que o de  $Eh$ , ainda que se use o menor deles (Real\*8); por exemplo: para o nível de erro de  $2E-7$ , a solução com RRE é obtida com tempo de CPU mais de 200 vezes menor do que sem RRE; outro exemplo: para o nível de erro de  $5E-10$  (foi necessário extrapolar a curva de  $Eh$ ), a solução com RRE é obtida com tempo de CPU mais de 26 mil vezes menor do que sem RRE. Portanto, a eficiência de RRE em termos de tempo de CPU aumenta com a redução do valor do erro.

Conforme a Fig. 10, para um dado valor de tempo de CPU, o erro com RRE ( $Em1$ ), mesmo com  $Real*16$ , é muito menor do que sem RRE ( $Eh$ ); por exemplo: para o nível de tempo de CPU de 1 s, a solução com RRE tem erro mais de  $3E+6$  vezes menor do que sem RRE; outro exemplo: para o nível de tempo de CPU de 100 s, a solução com RRE tem erro mais de  $8E+17$  vezes menor do que sem RRE. Portanto, a eficiência de RRE em reduzir o erro aumenta com o aumento do valor do tempo de CPU.

Em termos de memória RAM, o custo computacional do uso de RRE é quase o mesmo que sem RRE para uma dada malha.

## 6. Conclusão

No presente trabalho, apresentou-se uma base teórica para realizar repeated Richardson extrapolation (RRE) em CHT e CFD. RRE foi aplicado para reduzir e estimar o erro de discretização da solução numérica da equação de Laplace 2D. O trabalho foi realizado considerando-se: método de diferenças finitas, domínio de cálculo quadrado e discretizado com malhas uniformes, aproximações de segunda ordem de acurácia, diversas variáveis de interesse, condições de contorno de Dirichlet, malhas com até milhões de nós, multigrid, precisões simples, dupla e quádrupla, número suficiente de iterações para atingir o erro de arredondamento de máquina, e até doze extrapolações de Richardson.

Com a realização deste trabalho, verificou-se que:

- 1) RRE é extremamente eficiente em reduzir o erro de discretização de variáveis primárias ou secundárias, não importando o número de aproximações numéricas usadas.
- 2) O estimador de Richardson é acurado para prever o erro de discretização de soluções numéricas obtidas com RRE.
- 3) Maior redução do erro de discretização com RRE é obtido ao se usar: maior precisão nos cálculos, maior número de extrapolações ( $m$ ), maior número de malhas, e ordens corretas do erro.
- 4) Para se obter um dado valor de erro, o tempo de CPU e a memória RAM necessários para a solução com RRE são muito menores do que sem RRE.
- 5) Quando o erro de arredondamento é maior do que o erro de discretização: RRE perde o seu efeito de reduzir o erro; e o estimador de Richardson não funciona, podendo ser usado  $d\phi$  em sua substituição.

A base teórica de RRE e os trabalhos [10,11,13,28-30] permitem dizer que os resultados do presente trabalho também se aplicam, entre outras, às equações de Laplace, Poisson, Burgers e Navier-Stokes em duas e três dimensões, bem como em problemas transientes.

## ACKNOWLEDGEMENTS

The authors thank The UNIESPAÇO Program of the AEB (Brazilian Space Agency), CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico, Brazil), CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brazil) and Fundação Araucária (Paraná, Brazil) for their financial support. The first author is scholarship of CNPq. **Dedicatória:** o presente trabalho é dedicado à memória de L. F. Richardson e em comemoração aos 100 anos de seu trabalho de 1910.

## References

- [1] Richardson, L. F., 1910. The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. *Philosophical Proceedings of the Royal Society of London Serial A*, v. 210, p. 307-357.
- [2] Richardson, L. F.; Gaunt, J. A., 1927. The deferred approach to the limit. *Philosophical Proceedings of the Royal Society of London Serial A*, v. 226, p. 299-361.

- [3] Roache, P. J., 1998. *Verification and Validation in Computational Science and Engineering*. Hermosa, 446 p.
- [4] Livro Sidi (2003)
- [5] Livro Dahlquist e Bjorck (2008)
- [6] Marchi, C. H. and Silva, A. F. C., 2002. Unidimensional Numerical Solution Error Estimation for Convergent Apparent Order. *Numerical Heat Transfer, Part B*, Vol.42, pp. 167-188.
- [7] Roache, P. J., 1994. Perspective: a Method for Uniform Reporting of Grid Refinement Studies. *ASME Journal of Fluids Engineering*, Vol.116, pp. 405-413.
- [8] Joyce 1971
- [9] Rahul e B. 2006
- [10] Benjamin, A. S. and Denny, V. E., 1979. On the Convergence of Numerical Solutions for 2-D Flows in a Cavity at Large Re. *Journal of Computational Physics*, Vol.33, pp. 340-358.
- [11] Schreiber, R. and Keller, H. B., 1983. Driven Cavity Flows by Efficient Numerical Techniques. *Journal of Computational Physics*, Vol.49, pp. 310-333.
- [12] Demuren e Wilson 1994
- [13] Erturk, E., Corke, T. C. and Gökçöl, C., 2005. Numerical Solutions of 2-D Steady Incompressible Driven Cavity Flow at High Reynolds Numbers. *Int. J. NuRRE. Meth. Fluids*, Vol.48, pp. 747-774.
- [14] Burg e Erwin 2009
- [15] Roache e Knupp 1993
- [16] Wang e Zhang 2009
- [17] Sun e Zhang 2004
- [18] Ma e Ge 2010
- [19] Shyy et al. 2002
- [20] De Vahl Davis, G., 1983. Natural Convection of Air in a Square Cavity: a Bench Mark Numerical Solution. *International Journal for Numerical Methods in Fluids*, Vol.3, pp. 249-264.
- [21] Celik e Zhang 1995
- [22] Celik e K. 1997
- [23] Lee 1986
- [24] Vilenkin e K. 1987
- [25] Gabey e Shyy 2003
- [26] Celik et al 2005
- [27] Makai 1986
- [28] Leandro cilamce2008
- [29] Germer cilamce 2009
- [30] RBCM 2009
- [31] Tannehill, J. C.; Anderson, D. A.; Pletcher, R. H., 1997. *Computational Fluid Mechanics and Heat Transfer*. 2 ed. Taylor & Francis.
- [32] Schneider, G. E. and Zedan, M., 1981. A Modified Strongly Implicit Procedure for the Numerical Solution of Field Problems. *Numerical Heat Transfer*, Vol.4, pp. 1-19.
- [33] Wesseling, P., 1992. *An introduction to multigrid methods*. Wiley.
- [34] Kreyszig, E., 1999. *Advanced Engineering Mathematics*. 8 ed., Wiley.
- [35] Marchi, C. H., 2001. *Verificação de soluções numéricas unidimensionais em dinâmica dos fluidos*. Florianópolis: Universidade Federal de Santa Catarina. Tese de doutorado em Engenharia Mecânica.

## FIGURE CAPTIONS

Figure 1. Erros ( $E$ ) e suas estimativas ( $U$ ) versus tamanho de malha ( $h$ ) para  $Tc$  (Real\*16).

Figure 2. Erros ( $E$ ) e suas estimativas ( $U$ ) versus tamanho de malha ( $h$ ) para  $Tm$  (Real\*16).

Figure 3. Erros ( $E$ ) e suas estimativas ( $U$ ) versus tamanho de malha ( $h$ ) para  $Qe$  (Real\*16).

Figure 4. Efeito da precisão dos cálculos sobre o módulo do erro de  $Tc$  versus o tamanho da malha ( $h$ ).

Figure 5. Efeito do número de extrapolações ( $m$ ) sobre o erro de  $Tc$  versus o tamanho da malha ( $h$ ) (Real\*16).

Figure 6. Ordem efetiva ( $p_E$ ) do erro de  $Tc$  versus tamanho da malha ( $h$ ) e número de extrapolações ( $m$ ) (Real\*16).

Figure 7. Ordem efetiva ( $p_E$ ) das curvas de  $Em1$  e  $Em2$  de  $Tc$  versus tamanho da malha ( $h$ ) (Real\*16).

Figure 8. Ordem aparente ( $p_U$ ) da estimativa do erro de  $Tc$  versus tamanho da malha ( $h$ ) e número de extrapolações ( $m$ ) (Real\*16).

Figure 9. Erros ( $E$ ) de  $Tc$  versus tamanho da malha ( $h$ ) e valores das ordens do erro (Real\*16).

Figure 10. Erros ( $E$ ) de  $Tc$  versus tempo de CPU e precisão dos cálculos.