

Esquema de Multi-resolução Adaptativo para Leis de Conservação

M.K. KAIBARA¹, Departamento de Matemática, UNESP, Av. Eng. Luiz Edmundo C. Coube s/n, 17033-360 Bauru, SP, Brasil

S.M. GOMES², Departamento de Matemática Aplicada, IMECC-UNICAMP, Cx.P. 6065, 13084-040 Campinas, SP, Brasil.

Resumo. Neste trabalho desenvolvemos um esquema numérico para leis de conservação em malhas adaptativas com o objetivo de acelerar os cálculos. Estas malhas são esparsas em regiões de suavidade e refinadas próximo de descontinuidades. Na construção de tais malhas as principais ferramentas são provenientes de análise wavelet. Para o fluxo numérico usamos como esquema básico um operador de diferenças finitas adaptativo, mas alternamos para um esquema ENO na parte mais fina das malhas. Apresentamos resultados comparando o tempo de CPU do esquema proposto com o do esquema ENO puro numa malha uniforme. Os resultados demonstram que a eficiência do nosso algoritmo cresce à medida que o porte do problema aumenta, com escalas cada vez mais finas.

1. Introdução

O método proposto neste trabalho se aplica a leis de conservação hiperbólicas em uma dimensão do tipo

$$\begin{cases} u_t(x, t) + (f(u(x, t)))_x = 0 \\ u(x, 0) = u_0(x) \\ u(0, t) = g(t). \end{cases} \quad (1.1)$$

No espírito dos métodos de volumes finitos, trabalhamos com aproximações v_μ^n para as médias celulares da solução $u(x, t_n)$ em células $\mathcal{I}_\mu = [\mu^-, \mu]$ definidas por uma malha irregular Γ , do intervalo $\mathcal{I} = [0, 1]$. Isto é,

$$v_\mu^n \approx \frac{1}{h_\mu} \int_{\mu^-}^{\mu} u(x, t_n) dx,$$

onde μ^- é o vizinho à esquerda mais próximo de μ em Γ e $h_\mu = \mu - \mu^-$.

¹kaibara@fc.unesp.br

²soniag@ime.unicamp.br

Consideramos soluções numéricas de (1.1) obtidas pelo esquema explícito

$$v_{\mu}^{n+1} = [\mathcal{D}v^n]_{\mu} = \mu^n - \frac{\Delta t}{h_{\mu}}(\bar{f}_{\mu} - \bar{f}_{\mu-}), \quad (1.2)$$

onde \bar{f}_{μ} é o fluxo numérico e \mathcal{D} é o operador de evolução numérico definido pelo esquema. Dependendo de como o fluxo numérico \bar{f}_{μ} é definido, vários esquemas do tipo (1.2) tem sido propostos e aplicados extensivamente [4]. Entre eles, destacamos o esquema de Godunov, o de Lax-Wendroff e os esquemas essencialmente não oscilatórios, mais conhecidos como esquemas ENO. O esquema de Godunov resolve bem os choques, mas por ser de primeira ordem, não fornece uma boa precisão na parte suave da solução. Um esquema de segunda ordem é o de Lax-Wendroff. Mas apesar de apresentar uma boa resolução na parte suave, infelizmente pode apresentar sérias oscilações em regiões de descontinuidade. Os esquemas ENO são boas alternativas pois fornecem alta resolução tanto em regiões de suavidade quanto em regiões de singularidade. Mas exigem um alto custo computacional. Diante disso, Ami Harten propôs em [2] e [3] estratégias simples para diminuir o número de fluxos ENO calculados. A ferramenta básica é uma análise de multi-resolução para médias celulares em malhas uniformes, que fornece um indicador dos pontos onde os fluxos são calculados exatamente. Nos demais pontos, os fluxos numéricos são interpolados, o que permite uma economia computacional. Tipicamente, tais pontos são esparsos em regiões de regularidade e mais concentrados próximo das descontinuidades.

O esquema apresentado aqui é baseado nas idéias de Ami Harten mencionadas acima. Mas ao invés de evoluir as médias celulares da malha uniforme, propomos evoluir as médias celulares das malhas adaptativas definidas pelo esquema de multi-resolução. Ou seja, no total teremos um número pequeno de células, com tamanho grande em regiões de suavidade e mais finas próximo das irregularidades. Para isto precisamos usar esquemas de multi-resolução para médias celulares em malhas irregulares do intervalo. Para a definição do fluxo numérico adotamos Lax-Wendroff como esquema básico, mas alternamos para o esquema ENO de segunda ordem na parte mais fina das malhas. Além disso, no cálculo dos fluxos, usamos a idéia de um esquema adaptativo para diferenças finitas em malhas esparsas introduzido por Mats Holmström [5], modificado para o contexto de médias celulares.

Na Seção 2, definimos os principais ingredientes de nosso algoritmo: o tipo de malha irregular sobre a qual construímos as análises de multi-resolução de valores pontuais e de médias celulares, como estender ou reduzir as malhas, e como calcular os fluxos numéricos. Na Seção 3 indicamos os passos principais do algoritmo de evolução. Na Seção 4, apresentamos exemplos numéricos para comparar os tempos de CPU do esquema adaptativo proposto, do ENO puro em malha uniforme, e do esquema de multi-resolução de A. Harten. Estes testes indicam que nosso algoritmo é uma alternativa viável e mais econômica, dependendo do porte do problema e da complexidade da função fluxo. Na Seção 5, apresentamos as conclusões de nosso trabalho.

2. Ingredientes Básicos

2.1. Sobre as Malhas

Trabalhamos com malhas encaixadas

$$\Gamma^l \subset \Gamma^{l+1}, \quad 0 \leq l \leq L-1.$$

No caso uniforme, $\Gamma^l = X^l$ são malhas diádicas do intervalo \mathcal{I} que possuem $N_l = 2^{l_0+l}$ subintervalos de tamanho $h_l = 2^{-l_0-l}$, isto é,

$$X^l = \{\mu \in \mathcal{I} : \mu = jh_l, j = 0, \dots, N_l\}.$$

No caso irregular, assumimos que $\Gamma^0 = X^0$ no nível mais grosso, e para $l > 0$, $\Gamma^l \subset X^l$ é construída a partir de Γ^{l-1} pelo acréscimo de alguns pontos de $X^l \setminus X^{l-1}$. Vamos denotar por $\Lambda^{l-1} = \Gamma^l \setminus \Gamma^{l-1}$ o conjunto que contém estes pontos novos.

Em análise wavelet as malhas que formam um cone são de particular interesse pois elas estão associadas aos coeficientes wavelet significativos de funções que possuem singularidades locais. Por isso, consideramos malhas que satisfaçam essa condição, chamada condição do cone.

Condição do cone: Seja ρ um número inteiro positivo. Uma malha Γ^L satisfaz a ρ -condição do cone se para cada $\mu \in \Lambda^l$, $0 \leq l \leq L-1$ os 2ρ pontos de X^l mais próximos de μ estão em Γ^l .

2.2. Análises de Multi-resolução

Nosso interesse é estabelecer relações entre informações discretas $\{f^{l+1}\}$ associadas a uma malha Γ^{l+1} e as informações do nível inferior $\{f^l\}$, bem como obter a diferença de informação $\{d^l\}$ entre um nível e outro. Os valores $\{d_\mu^l\}$ são conhecidos como coeficientes wavelet. De modo geral, queremos saber como relacionar

$$f^L \xrightarrow{T^L} f_{MR}^L = (f^0, d^0, \dots, d^{L-1}). \quad (2.1)$$

Este processo é chamado de *análise* e a operação inversa é chamada de *síntese*. Descrevemos a seguir estas operações para discretizações por valores pontuais e por médias celulares.

Análise de Multi-resolução de Valores Pontuais

A principal ferramenta utilizada na definição de análises de multi-resolução interpolatórias são esquemas de subdivisão interpolatórios, em particular aqueles definidos por interpolação polinomial [1]. Eles são de nosso particular interesse pois não requerem uma malha estruturada e podem ser facilmente adaptados à fronteira de uma região. Partindo de um nível l , eles produzem operadores de interpolação $\mathcal{I}^l(x; f^l)$ para valores pontuais de f na malha Γ^l . Isto é,

$$\mathcal{I}^l(\mu; f^l) = f_\mu^l, \quad \mu \in \Gamma^l.$$

A diferença de informação entre dois níveis consecutivos é fornecida pelo erro de interpolação. Os algoritmos de análise e síntese associados são os seguintes:

- **Análise** - $f^L \rightarrow (f^0, d^0, \dots, d^{L-1})$
 Para $l = L - 1, \dots, 0$ fazemos
 - Para $\mu \in \Gamma^l$

$$f_\mu^l = f_\mu^{l+1}$$

- Para $\mu \in \Lambda^l$, calculamos

$$d_\mu^l = f_\mu^{l+1} - \mathcal{I}^l(\mu; f^l).$$

- **Síntese** - $(f^0, d^0, \dots, d^{L-1}) \rightarrow f^L$
 Para $l = 0, \dots, L - 1$ fazemos
 - Para $\mu \in \Gamma^l$

$$f_\mu^{l+1} = f_\mu^l$$

- Para $\mu \in \Lambda^l$ calculamos

$$f_\mu^{l+1} = \mathcal{I}^l(\mu; f^l) + d_\mu^l.$$

Análise de Multi-resolução de Médias Celulares

Seja $\bar{\Gamma}^l = \Gamma^l \setminus \{0\}$. Para cada $\mu \in \bar{\Gamma}^l$ associamos a célula \mathcal{I}_μ , de tamanho $h_{l,\mu} = \mu - \mu^-$. A média celular de f na célula \mathcal{I}_μ é definida por

$$\bar{f}_\mu^l = \frac{1}{h_{l,\mu}} \int_{\mu^-}^{\mu} f(x) dx.$$

O conhecimento das médias celulares \bar{f}_μ^l para $\mu \in \bar{\Gamma}^l$ é equivalente ao conhecimento dos valores pontuais $F(\mu)$ em Γ^l da função primitiva $F(x) = \int_0^x f(s) ds$. Mais precisamente

$$\bar{f}_\mu^l = \frac{F(\mu) - F(\mu^-)}{h_{l,\mu}} \quad F(\mu) = \sum_{\nu \leq \mu, \nu \in \Gamma^l} h_{l,\nu} \bar{f}_\nu^l.$$

Portanto, a reconstrução de médias celulares pode ser feita derivando o operador de interpolação da função primitiva

$$\mathcal{R}^l(x; \bar{f}^l) = \frac{d}{dx} \mathcal{I}^l(x; F^l).$$

Dadas as médias celulares no nível l , o operador de reconstrução $\mathcal{R}^l(x; \bar{f}^l)$ pode ser usado para aproximar médias celulares na malha mais fina. Por exemplo, para $\mu \in \Lambda^l$, definimos \tilde{f}_μ^{l+1} , uma aproximação de \bar{f}_μ^{l+1} , por

$$\tilde{f}_\mu^{l+1} = \frac{1}{h_{l+1,\mu}} \int_{\mu^-}^{\mu} \mathcal{R}^l(x; \bar{f}^l) dx = \frac{I^l(\mu; F^l) - I^l(\mu^-; F^l)}{h_{l+1,\mu}}. \quad (2.2)$$

A diferença de informação entre o nível l e $l+1$ é definido pelo erro nesta aproximação

$$\bar{d}_\mu^l = \bar{f}_\mu^{l+1} - \bar{f}_\mu^{l+1}. \quad (2.3)$$

Desta maneira, os algoritmos de análise e síntese para médias celulares ficam bem definidos como segue.

- **Análise** – $\bar{f}^L \rightarrow (\bar{f}^0, \bar{d}^0, \dots, \bar{d}^{L-1})$

Para $l = L - 1, \dots, 0$ fazemos

- Para $\mu \in \bar{\Gamma}^l$ calculamos

$$\begin{aligned} \bar{f}_\mu^l &= \frac{h_{l+1,\mu}}{h_{l,\mu}} \bar{f}_\mu^{l+1}, \quad \text{se } \mu^- \in \Gamma^l \\ \bar{f}_\mu^l &= \frac{h_{l+1,\mu}}{h_{l,\mu}} \bar{f}_\mu^{l+1} + \frac{h_{l+1,\mu^-}}{h_{l,\mu}} \bar{f}_{\mu^-}^{l+1}, \quad \text{se } \mu^- \in \Lambda^l \end{aligned}$$

- Para $\mu \in \Lambda^l$ calculamos

$$\bar{d}_\mu^l = \bar{f}_\mu^{l+1} - \bar{f}_\mu^{l+1}$$

- **Síntese** – $(\bar{f}^0, \bar{d}^0, \dots, \bar{d}^{L-1}) \rightarrow \bar{f}^L$

Para $l = 0, \dots, L - 1$ fazemos

- Para $\mu \in \Lambda^l$, calculamos

$$\bar{f}_\mu^{l+1} = \bar{d}_\mu^l + \bar{f}_\mu^{l+1}$$

- Para $\mu \in \bar{\Gamma}^l$ calculamos

$$\begin{aligned} \bar{f}_\mu^{l+1} &= \frac{h_{l,\mu}}{h_{l+1,\mu}} \bar{f}_\mu^l \quad \text{se } \mu^- \in \Gamma^l \\ \bar{f}_\mu^{l+1} &= \frac{1}{h_{l+1,\mu}} \left[h_{l,\mu} \bar{f}_\mu^l - h_{l+1,\mu^-} \bar{f}_{\mu^-}^{l+1} \right], \quad \text{se } \mu^- \in \Lambda^l \end{aligned}$$

2.3. Redução de Malhas

Na discussão acima consideramos análises de multi-resolução para valores pontuais e médias celulares em uma malha dada. Mas o que temos em mente é ter a malha adaptada à função em questão. Em análise wavelet, esta operação é feita naturalmente pelo processo de truncamento dos coeficientes wavelet d_μ^l . Por exemplo, suponhamos que uma certa função é representada por valores pontuais sobre uma malha Γ . Obtemos uma representação mais econômica usando o operador de truncamento \mathcal{T}_ϵ , que atua sobre os coeficientes wavelet, removendo de Γ aqueles pontos associados a coeficientes com módulo menor do que ϵ . Pela interpretação dos coeficientes como erros de interpolação polinomial, eles podem ser usados como indicadores locais de regularidade da solução. Isto significa que os coeficientes wavelet desprezados correspondem a regiões de suavidade. Como resultado, teremos uma malha mais esparsa nestas regiões e mais refinada próximo das irregularidades. A mesma interpretação é aplicada em caso de médias celulares.

2.4. Extensão das Malhas

Uma operação reversa, de extensão de malhas, também nos será útil. Por exemplo, a partir da representação de médias celulares \bar{f} de uma função em uma certa malha, poderá ser necessário ter este tipo de representação em uma malha mais refinada. Se este for o caso, nas novas células as médias são aproximadas usando o operador de reconstrução $\mathcal{R}(x; \bar{f})$.

Para as nossas aplicações, esta extensão é feita antes de cada passo de tempo, tanto no domínio espacial como no de frequência. Isto significa que, para cada coeficiente wavelet existente, adicionamos alguns coeficientes wavelet vizinhos de mesma escala e outros na próxima escala mais fina. Esta precaução é necessária para captar possíveis translações ou aparecimento de frequências mais altas na solução numérica durante o intervalo de tempo seguinte.

2.5. Fluxo Numérico Adaptativo

Um ingrediente fundamental na solução numérica de leis de conservação é o cálculo dos fluxos numéricos. Em nossos testes adotamos o esquema de Lax-Wendroff de segunda ordem como algoritmo básico. Mas próximo de discontinuidades, isto é, onde existirem coeficientes wavelet significativos na escala mais fina, mudamos para o esquema ENO de segunda ordem. Portanto, o fluxo Lax-Wendroff é calculado em regiões de suavidade, onde a malha é esparsa. Neste caso, a diferença finita é feita com o passo do tamanho da célula. Eventualmente as médias celulares vizinhas de mesmo tamanho podem não estar presentes. Neste caso, elas são aproximadas usando o operador de reconstrução de médias celulares. Esta idéia foi sugerida em [5] no contexto de análises de multi-resolução de valores pontuais.

3. Esquema de Evolução Automaticamente Adaptativo

Durante a evolução temporal, haverá várias malhas envolvidas, com as correspondentes médias celulares, operadores de reconstrução, e os algoritmos de análise e síntese associados. É útil manter uma notação uniforme para todos os casos, inclusive para os operadores de truncamento \mathcal{T}_ϵ , de extensão \mathcal{E} , e de evolução \mathcal{D} , independentemente da malha em questão.

Suponhamos que a solução numérica está representada no nível de tempo t_n pelas médias celulares v^n em uma malha esparsa Γ^n . Os seguintes passos nos conduzem à representação v^{n+1} em Γ^{n+1} do próximo nível de tempo.

1. Extensão

$$\tilde{v}^n \leftarrow \mathcal{E} v^n$$

2. Evolução temporal

$$\tilde{v}^{n+1} \leftarrow \mathcal{D} \tilde{v}^n$$

3. Truncamento

$$v^{n+1} \leftarrow \mathcal{T}_\epsilon \tilde{v}^{n+1}.$$

4. Resultados Numéricos

Apresentamos os resultados obtidos usando as equações de Burgers e de Buckley-Leverett.

4.1. Equação de Burgers

Vamos considerar a equação

$$u_t + \left(\frac{u^2}{2} \right)_x = 0, \quad x \in [0, 1],$$

com condições de fronteira periódicas. Para testar a habilidade do esquema de multi-resolução Adaptativo em seguir as singularidades da solução automaticamente, primeiro escolhemos a condição inicial

$$u(x, 0) = \begin{cases} 1, & x \in [0.25, 0.75] \\ 0, & \text{caso contrário.} \end{cases} \quad (4.1)$$

Adotamos um esquema de multi-resolução onde a malha possui 2^3 pontos no nível mais grosso e 6 níveis de escala. Isto é, $l_0 = 3$, $L = 5$, e $h_L = 2^{-8}$. Para $CFL = 0.8$, o passo temporal Δt utilizado satisfaz a condição

$$\frac{\Delta t}{h_L} \max_{x \in [0, 1]} |f'(u(x, t))| = 0.8.$$

No topo das Figuras 1 – 3 plotamos o gráfico da solução numérica nos tempos $t = 0.0625$, $t = 0.1875$ e $t = 0.25$. Abaixo destas figuras marcamos no plano posição–escala os pontos da malha estendida. Estes pontos, juntamente com a malha mais grossa X^0 , formam a malha sobre a qual é feita a evolução temporal.

Inicialmente a solução possui duas descontinuidades: em $x = 0.25$ e em $x = 0.75$. A primeira gera uma onda de rarefação e a segunda se propaga como um choque. No plano posição–escala das Figuras 1 – 3 observamos que os pontos das malhas possuem formato de cones com vértices apontando para estas singularidades.

Para testar a habilidade do método em refinar perto de singularidades que surgem no decorrer do tempo, escolhemos a condição inicial

$$u(x, 0) = 0.5 + \sin 2\pi x. \quad (4.2)$$

No topo das Figuras 4 – 6 mostramos o gráfico da solução numérica nos tempos $t = 0.015625$, $t = 0.121875$ e $t = 0.3984375$. Abaixo destas figuras marcamos no plano posição–escala os pontos da malha estendida.

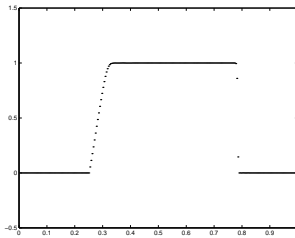


Figura 1: $t = 0.0625$

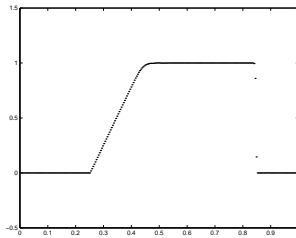


Figura 2: $t = 0.1875$

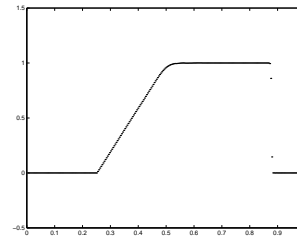


Figura 3: $t = 0.25$

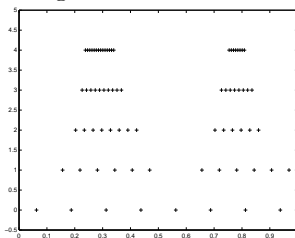


Figura 4: $t = 0.015625$

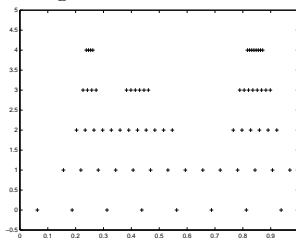


Figura 5: $t = 0.121875$

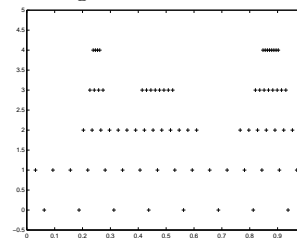
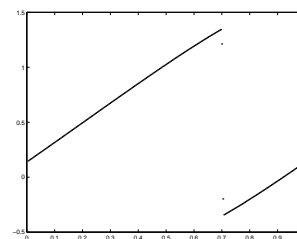
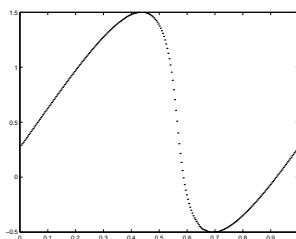
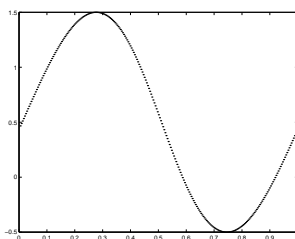


Figura 6: $t = 0.3984375$



A condição inicial é periódica e suave. Em $t = 0.015625$ e $t = 0.121875$ a solução ainda é suave e ainda não foi necessário usar o fluxo ENO. Em $t = 0.3984375$ o choque já está formado e bem resolvido, com um refinamento local próximo dele.

Definimos o erro $ER(t_n) = \bar{w}^n - \tilde{v}^n$ como a diferença entre a solução \bar{w}^n do esquema ENO puro, definido na malha uniforme X^L , e \tilde{v}^n , a solução do esquema Adaptativo estendida a X^L . Calculamos a norma do erro

$$\|ER(t)\|_1 = \frac{1}{N_L} \sum_{\mu \in X^L \setminus \{0\}} |\tilde{v}_\mu^n - \bar{w}_\mu^n|$$

para os dois casos (4.1) e (4.2) indicados acima. Como mostra a Tabela 1, $\max_{0 \leq t \leq 1} \|ER(t)\|_1$ permanece da mesma ordem do parâmetro de truncamento. Verifica-se portanto o mesmo comportamento previsto no esquema adaptativo de A. Harten [2].

ϵ	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
(4.1)	$1.29 \cdot 10^{-2}$	$2 \cdot 10^{-3}$	$1.615 \cdot 10^{-4}$	$1.2028 \cdot 10^{-5}$	$1.5995 \cdot 10^{-6}$
(4.2)	$6.4 \cdot 10^{-3}$	$1.1 \cdot 10^{-3}$	$2.6924 \cdot 10^{-4}$	$4.7212 \cdot 10^{-5}$	$6.944 \cdot 10^{-6}$

Tabela 1: $\max_{0 \leq t \leq 1} \|ER(t)\|_1$.

Para a condição inicial (4.2), medimos o tempo de CPU em segundos para evoluir a solução até $t = 0.109375$, para diferentes parâmetros de truncamento $\epsilon = 10^{-3}$, 10^{-4} e 10^{-5} , e para diferentes níveis da malha mais fina $L = 5, 6, \dots, 11$. Incluímos também os dados referentes ao esquema de multi-resolução de A. Harten e ao esquema ENO puro na malha uniforme X^L . Os resultados integram a Tabela 2. Observamos que a utilização do esquema Adaptativo começa a ser vantajosa a partir do nível $L = 8$. Essa vantagem aumenta conforme L e ϵ aumentam.

ϵ	esquema	$L = 5$	$L = 6$	$L = 7$	$L = 8$	$L = 9$	$L = 10$	$L = 11$
	ENO	0.19	0.74	2.94	11.65	48.51	186.46	744.06
10^{-3}	Harten	0.21	0.78	2.97	12.24	46.28	183.42	729.64
	Adaptativo	0.73	1.66	3.64	10.26	25.98	67.42	187.36
10^{-4}	Harten	0.25	0.85	3.12	11.94	46.89	184.86	733.96
	Adaptativo	1.45	3.35	7.84	16.54	45.95	111.52	284.75
10^{-5}	Harten	0.31	0.95	3.38	12.56	48.18	188.05	741.42
	Adaptativo	2.29	6.57	15.29	36.65	86.85	210.82	501.71

Tabela 2: Tempo de CPU

Na Figura 7 mostramos o tempo de CPU para evoluir a solução até o tempo t para os esquemas ENO puro (linha tracejada) e o Adaptativo (linha contínua). O nível mais fino é $L = 11$ e $\epsilon = 10^{-3}$. Ocorreu aproximadamente 55% de economia computacional usando o esquema Adaptativo, ou seja, nosso método gasta apenas

45% do tempo do outro. Na Figura 8 plotamos o número de células para evoluir a solução em cada nível de tempo. Em média utilizamos 165 células. Na Figura 9 mostramos o gráfico de $\|ER(t)\|_1$ em função do tempo t .

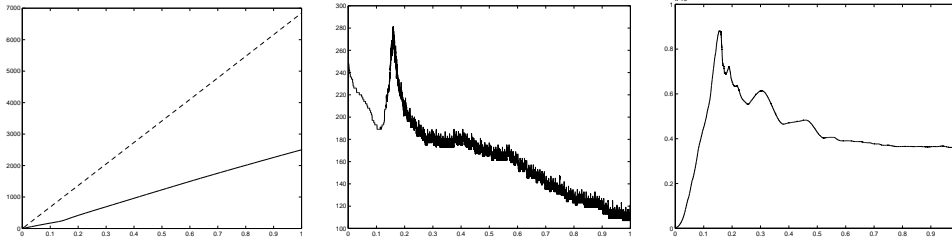


Figura 7: Tempo de CPU

Figura 8: Número de células

Figura 9: $\|ER(t)\|_1$

4.2. Equação de Buckley-Leverett

Neste exemplo, vamos considerar a equação

$$\frac{\partial s}{\partial t} + \frac{\partial f(s)}{\partial x} = 0, \quad (4.3)$$

com condição inicial

$$s(x, 0) = 0, \quad 0 < x \leq 1,$$

e condição de fronteira

$$s(0, t) = 1, \quad t > 0,$$

onde $0 \leq s(x, t) \leq 1$, com $f(s)$ da forma

$$f(s) = \frac{s^2}{s^2 + a(1-s)^2}$$

onde $0 \leq a \leq 1$.

Em nossos experimentos escolhemos $a = 0.7$. Para $\epsilon = 10^{-3}$ e $L = 8$, mostramos na Figura 10 o gráfico da solução numérica no tempo $t = 0.109375$, e na Figura 11 marcamos os coeficientes significativos no plano posição-escala.

Na Tabela 3 fornecemos o erro entre o esquema ENO e o Adaptativo para diferentes valores de truncamento. Na Tabela 4 estão tabelados o tempo de CPU em função de L , para diferentes valores de truncamento ϵ . Observemos que neste caso, o esquema Adaptativo já passa a ser vantajoso sobre o esquema ENO puro e o esquema de multi-resolução de Harten a partir de $L = 5$ e $L = 7$, respectivamente. Notemos que para $\epsilon = 10^{-3}$ e $L = 11$, há um ganho de 87% do esquema Adaptativo sobre o esquema ENO e de 72% sobre o esquema de multi-resolução de Harten. Na Figura 12 mostramos o gráfico do tempo de CPU em função do tempo t entre os

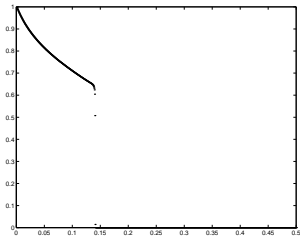


Figura 10: Solução de (4.8) para $t=0.109375$

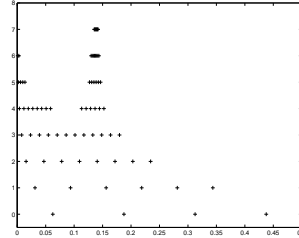


Figura 11: Coeficientes significativos

esquemas ENO (linha tracejada) e Adaptativo (linha contínua). Notamos um ganho de 70% deste sobre o outro. Nas Figuras 13 e 14 mostramos, respectivamente, os gráficos de do número de células em função do tempo t e de $\|ER(t)\|_1$. Na Figura 13 utilizamos uma média de 71 células para evoluir a solução em cada nível de tempo.

ϵ	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
$\max_{0 \leq t \leq 1} \ ER(t)\ _1$	$2.8 \cdot 10^{-3}$	$5.73 \cdot 10^{-4}$	$5.4 \cdot 10^{-5}$	$1.29 \cdot 10^{-5}$	$3.5 \cdot 10^{-6}$

Tabela 3: Erro entre os esquemas ENO e Adaptativo

ϵ	método	$L = 5$	$L = 6$	$L = 7$	$L = 8$	$L = 9$	$L = 10$	$L = 11$
	ENO	0.4	1.57	6.22	24.82	98.98	396.03	1587.20
10^{-3}	Harten	0.22	0.82	3.09	12.03	46.66	184.25	731.46
	Adaptativo	0.35	1.01	2.78	7.57	21.36	63.66	204.07
10^{-4}	Harten	0.24	0.84	3.16	12.1	47.45	187.15	743.08
	Adaptativo	0.38	1.15	3.31	9.32	26.2	75.80	230.93
10^{-5}	Harten	0.25	0.91	3.30	12.46	48.33	189.54	747.93
	Adaptativo	0.39	1.25	4.01	11.80	33.79	96.30	286.35

Tabela 4: Tempo de CPU

5. Conclusões

Neste trabalho, propomos um esquema de multi-resolução adaptativo em malhas irregulares para a resolução de leis de conservação hiperbólicas com o objetivo principal de acelerar os cálculos numéricos. A adaptividade é obtida pelo monitoramento dos coeficientes wavelet fornecidos pela análise de multi-resolução de médias celulares da solução numérica. A análise de multi-resolução automaticamente se

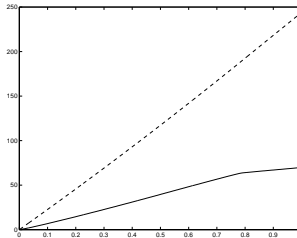


Figura 12: Tempo de CPU

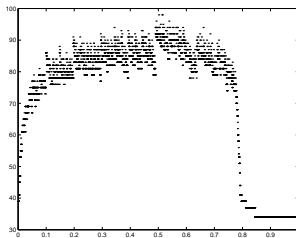
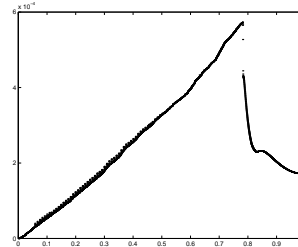


Figura 13: Número de células

Figura 14: $\|ER(t)\|_1$

adapta à evolução temporal da solução. Ela acompanha a solução que se move, bem como detecta o surgimento de choques que se desenvolvem com a evolução temporal. A representação esparsa conduz a uma redução significativa no tempo de CPU quando comparada com o esquema ENO em uma malha uniforme. O método é mais eficiente quando aplicado a problemas com soluções suaves em quase todo o domínio, com pequenas áreas de variação. A eficácia do método aumenta à medida que aumentamos a resolução do nível mais fino, e também à medida que a função de fluxo fica mais complicada. Além dos ganhos em tempo de computação, ocorre também uma economia de memória pelo uso compacto de armazenamento.

Referências

- [1] S. Dubuc, Interpolation through an Iterative Scheme, *J. Math. Anal. Appl.* **114** (1986), 185-204.
- [2] A. Harten, Multiresolution algorithms for the numerical solution of hyperbolic conservation laws, *Comm. Pure Appl. Math.* **XLVIII** (1995), 1305-1342.
- [3] A. Harten, Adaptive Multiresolution Schemes for Shock Computations, *Journal of Computational Physics* **115**(2) (1994), 319-338.
- [4] R. J. LeVeque, *Numerical Methods for Conservation Laws*, Birkhäuser Verlag, 1990.
- [5] M. Holmström, Wavelet based methods for time dependent PDEs. Tese de Doutorado, Universidade de Uppsala, Suécia, 1997.