

Método ANN-MoC *versus* Linear-MoC para Problemas de Transporte de Partículas Neutras em Geometria 1D

Augusto Tchantchalam^{*1}, Pedro Henrique de Almeida Konzen¹

¹PPGMAp, IME, UFRGS

Resumo

O fenômeno de transporte de partículas neutras aparece em muitas importantes aplicações na engenharia e na medicina. A modelagem fundamental baseia-se na equação linear de Boltzmann, uma equação integro-diferencial de primeira ordem. O método MoC (do inglês, *method of characteristics*) fornece uma forma explícita da solução da equação de transporte, forma que depende da integral da densidade de partículas. Neste trabalho, assumimos uma geometria unidimensional e apresentamos uma comparação entre as variantes Linear-MoC e ANN-MoC. A primeira é uma clássica variante do método, em que interpolação linear é utilizada na estimativa da densidade. A ANN-MoC é uma nova alternativa, as estimativas são obtidas a partir de uma ANN (do inglês, *artificial neural network*). A comparação entre as variantes é apresentada para um caso teste com solução manufaturada. Os resultados obtidos indicam a potencialidade de nova variante.

1 Introdução

O fenômeno de transporte de partículas neutras aparece em muitas importantes aplicações na engenharia e na medicina. Por exemplo, a modelagem do transporte de calor por radiação (Modest, 2013) envolve o transporte de fótons e tem aplicações industriais na manufaturada em altas temperaturas, na medicina ótica entre outras (Abdoulaev & Hielscher, 2003; Frank et. al, 2004; Larsen et. al, 2002; Wang, 2007). Outro exemplo é o transporte de nêutrons com aplicações na geração de energia nuclear e também na medicina (Lewis, 1984; Stacey, 2007).

Neste trabalho, assumimos o transporte de partículas modelado em uma geometria unidimensional $\bar{\mathcal{D}} = [a, b]$, como segue

$$\forall \mu \in (-1, 1) \setminus \{0\} : \mu \cdot \frac{\partial}{\partial x} I(x, \mu) + \sigma_t I = \sigma_s \Psi(x) + q(x, \mu), \quad \forall x \in \mathcal{D}, \quad (1a)$$

$$\forall \mu > 0 : I(a, \mu) = I_a; \quad \forall \mu < 0 : I(b, \mu) = I_b. \quad (1b)$$

^{*}augustotchantchalam11@gmail.com

Por exemplo, em uma aplicação de transporte radiativo, $I(x, \mu)$ (W/sr) é a intensidade de radiação no ponto $x \in \mathcal{D}$ e na direção $-1 < \mu < 1$, $\mu \neq 0$. O coeficiente de absorção total é denotado por $\sigma_t = \kappa + \sigma_s$ (1/m), onde κ (1/m) e σ_s (1/m) são os coeficientes de absorção e de espalhamento, respectivamente. As fontes são denotadas por $q(x, \mu)$ (W/(m · sr)) no domínio e por I_a e I_b (W/sr) nas fronteiras. A densidade radiativa (W/sr) é denotada por

$$\Psi(x) := \frac{1}{2} \int_{-1}^1 I(x, \mu) d\mu. \quad (2)$$

O método *MoC* (do inglês, *method of characteristics*) (Evans, 2010) é comumente usado na computação numérica da solução de (1). Sua aplicação fornece uma forma explícita da solução $I(x, \mu)$ que depende da integral da densidade radiativa na linha de direção μ . Usualmente, considera-se uma malha computacional e essa forma da solução é aplicada em cada célula computacional. Por aplicação de uma quadratura numérica, a computação da integral requer a estimativa de $\Psi(x)$ em pontos internos da célula. Neste trabalho, apresentamos uma comparação entre as variantes Linear-MoC e ANN-MoC. Enquanto que na clássica variante Linear-MoC, as estimativas são obtidas por interpolação linear, no recentemente proposto método ANN-MoC (Konzen, 2023), as estimativas de $\Psi(x)$ são fornecidas por uma ANN (do inglês, *artificial neural network*) (Haykin, 2009). O treinamento supervisionado da ANN é feito com as aproximações da solução nos nodos computacionais.

Na sequência, apresentados detalhes sobre o método ANN-MoC. A comparação com a variante Linear-MoC é realizada no contexto de uma aplicação com solução manufaturada. Ao final, apresentados nossas conclusões.

2 Método ANN-MoC

O método ANN-MoC é uma recente variante do método MoC para a solução numérica de problemas de transporte de partículas neutras. Sua aplicação é feita a partir da aproximação DOM (do inglês, *discrete ordinates method* e do esquema SI (do inglês, *source iteration*) (Modest, 2013).

Assumindo a quadratura gaussiana $\{(\mu_i, \omega_i)\}_{i=1}^N$, a formulação DOM & SI de (1a) é dada como segue

$$1 \leq i \leq N : \mu_i \cdot \frac{\partial}{\partial x} I_i^{(j)}(x) + \sigma_t I_i^{(j)} = \sigma_s \Psi^{(j-1)}(x) + q(x, \mu_i), \quad \forall x \in \mathcal{D}, \quad (3a)$$

$$\forall \mu_i > 0 : I_i^{(j)}(a) = I_a; \quad \forall \mu_i < 0 : I_i^{(j)}(b) = I_b, \quad (3b)$$

onde $I_i^{(j)} \approx I_i^{(j)}(x, \mu_i)$, $j = 1, 2, \dots, L$, com dada aproximação inicial $\Psi^{(0)}(x)$. A j -ésima aproximação da densidade de partículas é dada por

$$\Psi^{(j)}(x) = \frac{1}{2} \sum_{i=1}^N \omega_i I_i^{(j)}(x). \quad (4)$$

O problema (3) consiste em um sistema de equações diferenciais parciais de primeira ordem com condições de contorno. O método MoC fornece a seguinte forma da solução nas características

$$x(s) = x_0 + s\mu_i, \quad s \in \mathbb{R},$$

$$I_i^{(j)}(s) = I_i^{(j)}(0)e^{-\int_0^s \sigma_t ds'} + \int_0^s \left[\Psi^{(j)}(s') + q(s', \mu_i) \right] e^{-\int_{s'}^s \sigma_t ds''} ds', \quad (5)$$

onde $I_i^{(j)}(s) = I_i^{(j)}(x(s))$. Usualmente, aplica-se (5) nas células de uma malha computacional (consulte a Figura 1). Vamos assumir uma malha $x_m = a + (m - 1)h_x$, $m = 1, 2, \dots, n_s + 1$, de tamanho $h_x = (b - a)/n_s$. Sem perda de generalidade, para $s > 0$ e em uma célula $[x_{m-1}, x_m]$, a computação da solução no nodo x_m é feita com base na solução no nodo x_{m-1} . A forma (5) requer a computação da integral de $\Psi^{(j-1)}(s')$ na célula computacional, o que, em geral, demanda estimativas de $\Psi^{(j-1)}(s')$ em pontos internos da célula.

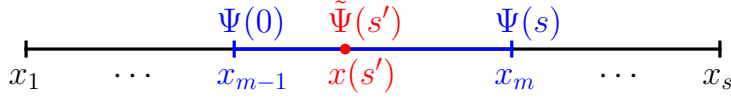


Figura 1: Esquema da malha computacional.

Na clássica variante Linear-MoC as estimativas $\tilde{\Psi}^{(j-1)}(s')$ são obtidas por interpolação linear de seus valores conhecidos nos nodos x_{m-1} e x_m . Na recentemente proposta variante ANN-MoC, uma ANN é treinada para fornecer as estimativas da densidade de partículas. Assumindo-se uma ANN do tipo MLP (do inglês, *multilayer perceptron*) (Modest, 2013), o valor estimado $\tilde{\Psi}(x) \approx \Psi(x)$ é dado por

$$\tilde{\Psi}(x) = \mathcal{N} \left(x; \left\{ \left(W^{(l)}, \mathbf{b}^{(l)}, \mathbf{f}^{(l)} \right) \right\}_{l=1}^{n_h+1} \right), \quad (6)$$

onde $(W^{(l)}, \mathbf{b}^{(l)}, \mathbf{f}^{(l)})$ denota a tripla de pesos $W^{(l)} = [w_{i,j}^{(l)}]_{i,j=1}^{n^{(l-1)}, n^{(l)}}$, de bias $\mathbf{b}^{(l)} = (b_i^{(l)})_{i=1}^{n^{(l)}}$ e de função de ativação $\mathbf{f}^{(l)} : \mathbb{R}^{n^l} \rightarrow \mathbb{R}^{n^l}$ na l -ésima camada da rede. O número de neurônios (unidades de processamento) em cada camada é denotado por $n^{(l)}$, $l = 1, 2, \dots, n_h + 1$, onde n_h é o número de camadas escondidas da rede.

Fixada a arquitetura da MLP (número de camadas escondidas, número de unidades de processamento em cada camada e as respectivas funções de ativação), o treinamento da MLP é feito resolvendo-se o seguinte problema de minimização

$$\min_{\{(W^{(l)}, \mathbf{b}^{(l)})\}_{l=1}^{n_h+1}} \underbrace{\frac{1}{n_s} \sum_{m=1}^{n_s} \left(\tilde{\Psi}^{(m)} - \Psi^{(m)} \right)^2}_{=:\varepsilon}. \quad (7)$$

O conjunto de treinamento $\left\{ x^{(m)}, \tilde{\Psi}(x^{(m)}) \right\}_{m=1}^{n_s}$, consiste dos n_s conhecidos valores da densidade de partículas $\Psi^{(j-1)}$ nos nodos computacionais.

3 Resultados

Nesta seção, vamos apresentar uma discussão de comparação das variantes Linear-MoC com a ANN-MoC aplicadas ao problema de transporte com solução manufaturada

$$\hat{I}(x, \mu) = e^{-\alpha\sigma_t x}. \quad (8)$$

Por substituição em (1a), calculamos a fonte

$$q(x, \mu) = (\kappa - \alpha\sigma_t \mu)e^{-\alpha\sigma_t x}. \quad (9)$$

As condições de contorno são $I_a = \hat{I}(a)$ e $I_b = \hat{I}(b)$ e fixamos $\alpha = 5$. Por cálculo direto, observa-se que $\hat{\Psi}(x) = e^{-\alpha\sigma_t x}$ é a densidade de partículas exata.

O método ANN-MoC requer a escolha *a priori* da arquitetura da ANN. Em todos os testes numéricos, usamos MLPs com tangente hiperbólica e função identidade como funções de ativação nas camadas escondidas e na camada de saída, respectivamente. Arquiteturas $1 - n_n \times n_h - 1$ foram empregadas, i.e. uma entrada, n_h camadas escondidas, cada uma com n_n neurônios, e uma saída. A Tabela 1 apresenta dados sobre o número de épocas n_e demandas e o tempo computacional t_c (s) para o treinamento da rede na primeira iteração de fonte (critério de parada, $\varepsilon < 10^{-6}$). Tendo em vista a estocasticidade do método, cada treinamento foi repetido três vezes e os valores médios foram tabelados. Dos resultados, pode-se observar que uma arquitetura $1 - 40 \times 3 - 1$ forneceu uma boa relação n_e/t_c . Esta arquitetura de rede é utilizada nos demais resultados apresentados na sequência.

Tabela 1: Testes de treinamento de redes com arquitetura $1 - n_n \times n_l - 1$, fixados $n_s = 10$, $N = 10$, $\kappa = 0.75$ e $\sigma_s = 0.25$. Os valores tabelados são n_e/t_c , número de épocas pelo tempo de execução na primeira iteração de fonte.

$n_n \setminus n_l$	1	2	3	4
5	15537/4.07	10066/3.59	10090/4.41	6117/3.26
10	13158/3.51	5424/2.09	3419/1.69	4723/2.67
20	12013/3.20	4428/1.73	3133/1.63	3736/2.21
40	10699/2.91	3115/1.33	2875/1.55	2248/1.46

A primeira iteração de fonte demanda o treinamento mais custoso da rede neural. Em iterações de fonte subsequentes, o treinamento da rede pode ser inicializado com a técnica *warm initialization*, em que seus parâmetros são mantidos do treinamento anterior. Considerando o caso teste $\kappa = 0.75$, $\sigma_s = 0.25$, $n_s = 10$ e $N = 10$, a Figura 2 mostra a demanda do número de épocas em cada iteração de fonte. A linha contínua representa o número médio de épocas necessárias no treinamento em cada iteração. Foram realizadas cinco repetições e a região sombreada corresponde os números máximos e mínimos de épocas.

Agora, assumindo $\kappa = 0.75$ e $\sigma_s = 0.25$, a Tabela 2 apresenta uma comparação entre resultados obtidos com as variantes Linear-MoC e ANN-MoC para diferentes parâmetros de n_s e N . Enquanto a variante Linear-MoC tem resultados mais precisos ao que a malha espacial é refinada, o método ANN-MoC apresenta melhores resultados para malhas menos refinadas, em especial no ponto $x = 0.3$ que não é um nodo das malhas empregadas nesses testes.

Tabela 2: Comparação de resultados com as variantes Linear-MoC e ANN-MoC, fixados $\kappa = 0.75$ e $\sigma_s = 0.25$. Rede de arquitetura $1 - 40 \times 3 - 1$.

n_s	Tipo	$\Psi(0.0)$	$\Psi(0.3)$	$\Psi(0.5)$	$\Psi(0.9)$	L	Tempo (s)
$N = 10$							
2	ANN	1.0056e+0	2.9379e-01	8.5699e-02	2.2911e-03	5	0.49
	Linear	1.0166e+0	4.6575e-01	9.8490e-02	2.9747e-02	8	0.37
4	ANN	1.0006e+00	2.2047e-01	8.1452e-02	1.1564e-02	6	1.11
	Linear	1.0055e+00	2.5149e-01	8.6104e-02	1.5270e-02	8	0.63
8	ANN	9.9975e-01	2.2162e-01	8.3094e-02	1.0715e-02	7	3.68
	Linear	1.0015e+00	2.3472e-01	8.3089e-02	1.1861e-02	8	1.20
16	ANN	9.9878e-01	2.2277e-01	8.2902e-02	1.0745e-02	7	4.57
	Linear	1.0004e+00	2.2536e-01	8.2336e-02	1.1347e-02	9	2.58
32	ANN	9.9670e-01	2.2238e-01	8.2711e-02	1.0801e-02	7	7.64
	Linear	1.0001e+00	2.2388e-01	8.2148e-02	1.1159e-02	9	4.98
$N = 20$							
4	ANN	1.0004e+00	2.2165e-01	8.1187e-02	1.1845e-02	5	1.49
	Linear	1.0054e+00	2.5139e-01	8.5995e-02	1.5262e-02	8	1.20
8	ANN	9.9975e-01	2.2160e-01	8.3096e-02	1.0690e-02	7	5.36
	Linear	1.0015e+00	2.3470e-01	8.3069e-02	1.1863e-02	8	2.28
16	ANN	9.9851e-01	2.2244e-01	8.2823e-02	1.0799e-02	7	8.25
	Linear	1.0004e+00	2.2536e-01	8.2332e-02	1.1348e-02	9	4.98
$N = 40$							
4	ANN	9.9989e-01	2.2213e-01	8.1240e-02	1.1670e-02	5	2.32
	Linear	1.0053e+00	2.5124e-01	8.5954e-02	1.5253e-02	8	2.29
8	ANN	9.9982e-01	2.2174e-01	8.3238e-02	1.0616e-02	7	8.62
	Linear	1.0015e+00	2.3468e-01	8.3064e-02	1.1862e-02	8	4.41
exata	-x-	1.0000E+0	2.2313e-1	8.2085E-2	1.1109e-2	-x-	-x-

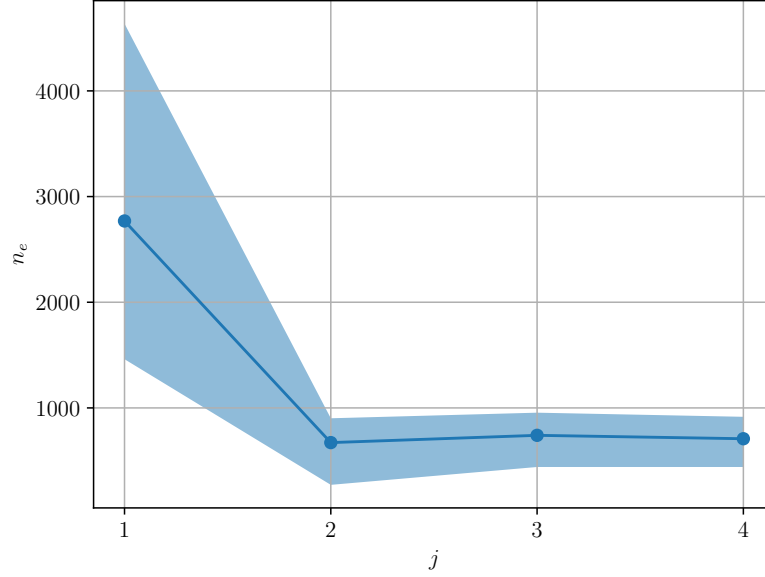


Figura 2: Número de épocas por iteração de fonte. Arquitetura $1 - 40 \times 3 - 1$. Parâmetros $n_s = 10$, $N = 10$, $\kappa = 0.75$ e $\sigma_s = 0.25$.

A ideia principal do método ANN-MoC consiste em explorar o fato de MLPs serem aproximadoras universais. A Figura 3 mostra que a ANN consegue capturar a forma da solução exata $\hat{\Psi}(x) = e^{-\alpha\sigma_t x}$ mesmo para uma malha com apenas $n_s = 8$ células computacionais. Em geral, aproximação de uma função exponencial por uma interpoladora linear requer uma malha fina. O treinamento simultâneo da ANN com base nos dados obtidos nos nodos computacionais faz com que a solução ANN-MoC capture a forma da solução de forma eficiente.

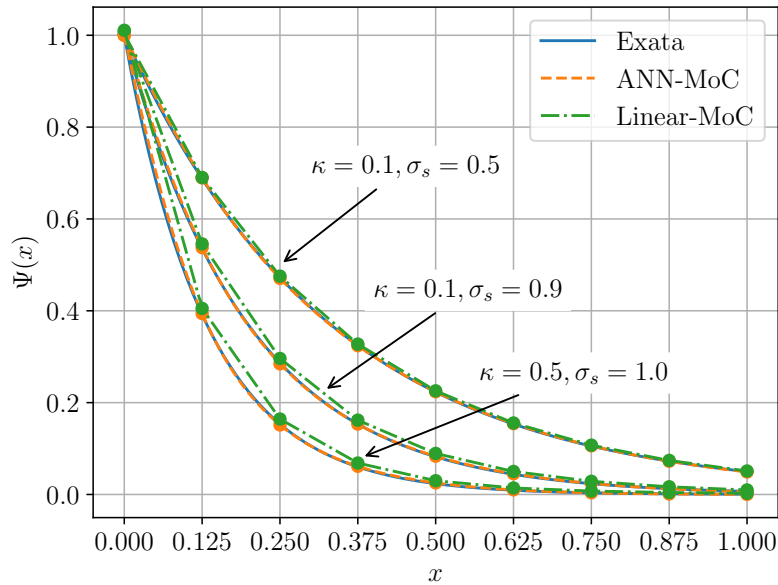


Figura 3: Comparação entre as soluções ANN-MoC e Linear-MoC, fixados $n_s = 8$ e $N = 40$.

4 Considerações Finais

Apresentados um estudo de comparação entre as variantes Linear-MoC e ANN-MoC do método das características para a solução numérica de problemas de transporte de partículas neutras em geometria unidimensional. Enquanto a variante Linear-MoC emprega interpolação linear para estimar os valores da densidade de partículas em pontos internos de células computacionais, a recentemente proposta variante ANN-MoC usa de uma rede neural. A rede é treinada simultaneamente a cada iteração de fonte com base nos dados da densidade de partículas nos nós computacionais. A variante Linear-MoC tem a vantagem de gerar resultados mais precisos, caso uma malha suficientemente refinada seja utilizada. Em contrapartida, a ANN-MoC consegue capturar de forma mais eficiente a forma da solução em malhas pouco refinadas. Os resultados apresentados mostram a potencialidade de nova variante. Ainda a ser verificado, o fato de não demandar malhas muito refinadas indica uma vantagem frente a outras variantes em problemas com geometrias multidimensionais.

5 Agradecimentos

Este estudo foi financiado em parte pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES).

Referências Bibliográficas

- Abdoulaev, G.S., Hielscher, A.H. (2003). Three-dimensional optical tomography with the equation of radiative transfer. *Journal of Electronic Imaging*, 12, 594-601.
- Evans, L.C. (2010). *Partial Differential Equations*, 1ed., AMS.
- Frank, M., Seaïd, M., Klar, A., Pinnau, R., Thömmes, G., Janicka, J. (2004). A comparison of approximate models for radiation in gas turbines. *Progress in Computational Fluid Dynamics*, 4, 191-197.
- Haykin, S. (2009). *Neural Networks and Learning Machines*, 3ed., Pearson.
- Konzen, P.H.A. (2023). ANN-MoC Method for Solving Unidimensional Neutral Particle Transport Problems. A aparecer em: *XLII CNMAC 2023*, Bonito, Brasil, arXiv:2307.03114.
- Larsen, E.W., Thömmes, G., Klar, A., Seaïd, M., Götz, T. (2002). Simplified P_N approximations to the equations of radiative heat transfer and applications. *Journal of Computational Physics*, 183, 652-675.
- Lewis, E.E., Miller, W.F. (1984). *Computational Methods of Neutron Transport*, 1ed., Wiley.
- Modest, M.F. (2013). *Radiative heat transfer*, 3ed., Elsevier.
- Stacey, W.M. (2007). *Nuclear Reactor Physics*, 2ed., Wiley.
- Wang, L.V., Wu, H. (2007). *Biomedical Optics: Principles and Imaging*, 1ed., Wiley.