



**Simpósio de Métodos
Numéricos em Engenharia**

25 a 27 de outubro, 2017

Otimização da análise de recorrência utilizando placas de vídeo com tecnologia CUDA

Bruno Rafael Reichert Boaretto, Roberto Cesar Budzinski Neto, Thiago de Lima Prado, Sergio Roberto Lopes
Departamento de Física, Universidade Federal do Paraná.
Curitiba, Brasil
brrb12@fisica.ufpr.br

Resumo—Este trabalho apresenta um método computacional para otimizar quantificadores de recorrência de uma série temporal genérica. Utiliza-se de placas de vídeo da Nvidia (Geforce GTX 960M) constituída por 5 multiprocessadores que ao fazer o uso de técnicas de computação paralela reduz-se o custo computacional em até 10 vezes.

Palavras-chave—gráficos de recorrência, análise de recorrência, otimização.

I. INTRODUÇÃO

A ideia de recorrência surgiu com *Henri Poincaré*, com a publicação do trabalho [1] definindo o *Teorema da Recorrência de Poincaré*, que pode ser entendido da seguinte forma: para um sistema dinâmico, as trajetórias retornam infinitas vezes arbitrariamente próximas a quase todos os pontos iniciais, ou seja, marca-se um ponto de referência \mathbf{x}_i , e após um longo período de tempo a trajetória visita outro ponto \mathbf{x}_j tal que

$$|\mathbf{x}_i - \mathbf{x}_j| < \delta \quad \text{para } \delta \geq 0 \quad (1)$$

Em sistema caóticos, é praticamente impossível ocorrer de $|\mathbf{x}_i| = |\mathbf{x}_j|$, por isso, é completamente compreensível o uso do δ na equação 1 que é uma espécie de limiar (do inglês *threshold*) que determina a precisão a ser levada em consideração.

Um gráfico de recorrência (do inglês *Recurrence plots*) trata-se de uma forma de visualizar sistemas recorrentes [2] o

qual pode ser definido por:

- Se $|\mathbf{x}_i - \mathbf{x}_j| \leq \delta$, diz-se que os pontos são recorrentes, consequentemente marca-se um ponto preto no gráfico (matriz) de recorrência.
- Se $|\mathbf{x}_i - \mathbf{x}_j| > \delta$ os pontos são ditos não-recorrentes, consequentemente marca-se um ponto branco no gráfico.

A Fig. 1 (parte superior) é referente a série temporal para um mapa logístico Eq. 2 onde para $r = 4$ a série apresenta comportamento caótico, e na parte inferior da Fig. 1 trata-se de seu respectivo gráfico de recorrência utilizando $\delta = 0,01$.

$$x_{n+1} = rx_n(1 - x_n) \quad (2)$$

A Fig. 2 trata-se da série temporal de uma função senoidal gráfico de recorrência para uma função senoidal $x_n = \sin(2\pi n/300)$ o qual apresenta comportamento periódico.

Os gráficos de recorrência apresentam estruturas das quais é possível retirar informações importantes da dinâmica da série utilizada, por exemplo, as estruturas diagonais ocorrem quando pontos da trajetória evoluem paralelamente a outros pontos, ou seja, os pontos retornam a uma região do espaço de fase em tempos diferentes. A duração dessa evolução determina comprimento dessas linhas diagonais [2].

II. QUANTIFICADORES DE RECORRÊNCIA

Ao analisar séries temporais longas, em torno de 10 mil pontos, os gráficos de recorrência serão imagens de

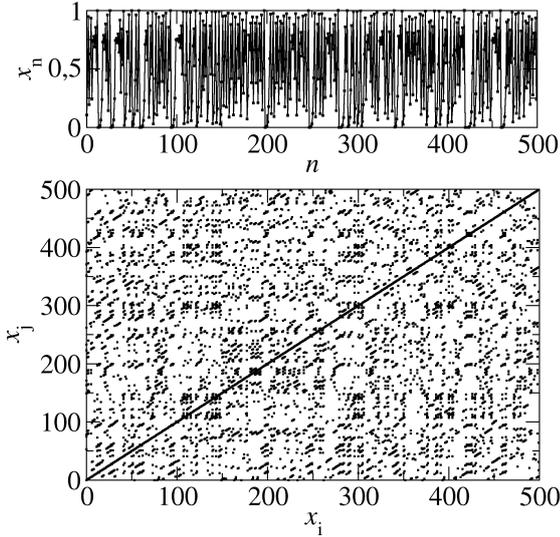


Figura 1: No painel superior a série temporal para um mapa logístico (Eq. 2) com $r = 4$, no painel inferior o seu respectivo gráfico de recorrência para $\delta = 0,01$.

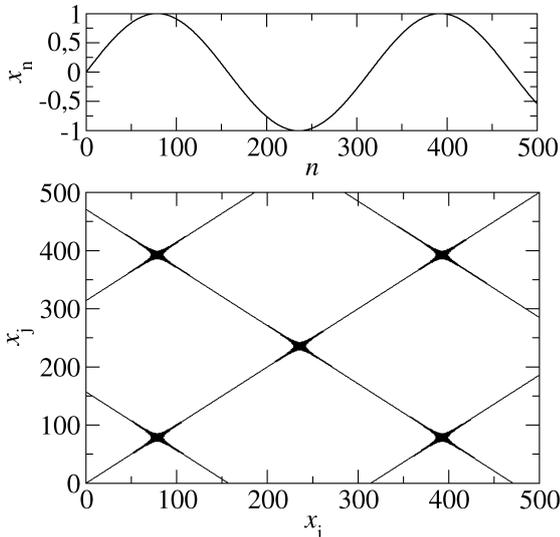


Figura 2: No painel superior a série temporal para uma função senoidal $x_n = \text{sen}(2\pi n/300)$, no painel inferior o seu respectivo gráfico de recorrência para $\delta = 0,01$.

10000×10000 , logo, perde-se total informação qualitativa da dinâmica da série. Para tais sistemas faz-se a necessidade de criar certas ferramentas quantificadoras que carregam informações referentes a essas estruturas. Um quantificador proposto é a *Taxa de recorrência REC*, a qual mede a razão de pontos recorrentes dentre todos os pontos [2]:

$$REC(\delta) = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N \mathbf{R}_{i,j}(\delta). \quad (3)$$

onde:

$$\mathbf{R}_{(i,j)}(\delta) = \begin{cases} 1, & |\mathbf{x}_i - \mathbf{x}_j| \leq \delta, \\ 0, & \text{caso contrário.} \end{cases} \quad (4)$$

III. METODOLOGIA

Para fazer a análise da *REC* é preciso fazer a comparação de cada i -ésimo ponto da série temporal com todos os j -ésimos pontos, ou seja:

- 1) É feito um *loop* variando o i de 1 a N .
- 2) É feito outro *loop* variando j de 1 a N , isso certifica-nos de que cada ponto i é comparado com todos os pontos da série.
- 3) É feita a seguinte pergunta: $|\mathbf{x}_i - \mathbf{x}_j| \leq \delta$? Se verdadeiro soma-se 1 em um acumulador. Se falso nada é feito.
- 4) Por fim de todos os ciclos divide-se o acumulador por todos os pontos do gráfico (N^2).

Portanto é feita em torno de N^2 contas (perguntas), esse grande número de contas faz com que os programas se tornem relativamente lentos conforme o aumento da série temporal em questão.

Para contornar esse problema faz-se o cálculo dessas rotinas em uma placa de vídeo com tecnologia CUDA (GPUs, *Graphics Processing Unit*) [3]. As placas de vídeo apresentam um número grande de *threads*, ou seja, núcleos que podem realizar tarefas distintas um do outro, esses núcleos são menos poderosos que os encontrados nas CPUs (*Central Processing Unit*), porém, quando as contas são relativamente simples como é o caso da recorrência em que é feita apenas uma única pergunta, nesse caso, o número grande de *threads*, compensa sua menor velocidade.

Para esse projeto a placa de vídeo utilizada foi uma **Geforce GTX 960M**, essa placa possui 5 multiprocessadores onde cada multiprocessador roda em paralelo 2048 *threads* [3], ou seja, um total de 10240 tarefas distintas podem ser feitas ao mesmo tempo.

Tabela I: Resultados parciais, onde N é o tamanho da série analisada e t o tempo em segundos da execução do programa.

	N (10^3)	t(s)						
CPU	10	1	20	4	50	26	70	53
GPU	10	1	20	1	50	2	70	3

AGRADECIMENTOS

Este projeto teve apoio financeiro do CNPq.

REFERÊNCIAS

- [1] H. Poincaré, *Sur le probleme des trois corps et les equations de la dynamique*. *Acta mathematica*, 13(1):A3–A270, 1890.
- [2] N. Marwan, M. Romano, M Thiel, e J. Kurths. *Recurrence plots for the analysis of complex systems*. *Physics reports*, 438(5):237–329, 2007.
- [3] URL http://www.nvidia.com.br/object/cuda_home_new_br.html acessado em 10 de outubro de 2017.