

Parallel Performance Analysis of Cyclic Correntropy for Energy-Efficient Wireless Communications

Sávio R. M. Melo, Herlan A. P. Silva, Samuel Xavier-de-Souza, Luiz Felipe Q. Silveira and Aluisio I. R. Fontes

Abstract—Cognitive radio (CR) based systems appear as an efficient alternative for providing higher throughput in wireless systems, mainly through dynamic spectrum access (DSA), and automatic adaptation of the transmitter in terms of modulation format, coding rate, among other parameters, as a function of the communication channel conditions accessed by the system. In this context, processing techniques for spectrum sensing and automatic modulation classifications (AMC) has been widely developed in recent years. Among these techniques, cyclostationary correntropy can be applied to both problems the spectrum sensing and automatic modulation classification. However, this technique has a high computational cost that may increase significantly the system energy consumption. This work proposes a strategy to calculate the cyclic correntropy by using parallel processing on multi-core processors in order to decrease energy consumption in spectrum sensing tasks. A mathematical model is used to relate the power consumption of the processors with their parallel performance. The results demonstrate a decrease in the energy consumption of the proposed architecture.

Index Terms—Cognitive Radio, Cyclic Correntropy, Cyclostationary Process, Low Power, Spectrum Sensing, Parallel Scalability

I. INTRODUÇÃO

Constantes crescimentos no uso de sistemas de comunicação sem fio vem contribuindo para a escassez do espectro de frequência. De fato, trabalhos recentes mostram que o modelo atual de gerenciamento espectral que usa alocação fixa é ineficaz [1], [2]. Isso motiva o desenvolvimento de técnicas inteligentes de processamento que permitem a identificação dinâmica de bandas de frequência subutilizadas, bem como a otimização da configuração dos parâmetros de transmissão. Nesse contexto, os rádios cognitivos aparecem como uma opção apropriada, capaz de oferecer um uso eficiente do canal de comunicação e possibilitando o acesso dinâmico do espectro, garantindo uma maior largura de banda aos usuários [3]. Uma das principais características dos sistemas de radio cognitivo

é a capacidade de identificar a modulação e utilizada no transmissor a partir do sinal recebido.

A utilização das técnicas de processamento de sinais no cenário de rádio cognitivo, juntamente com o surgimento do mercado da Internet das Coisas são os principais fatores para o aumento no consumo de energia nos dispositivos móveis [4]. A análise cicloestacionária foi apresentada na literatura como um método robusto para permitir a classificação automática de modulações e sensoriamento espectral em rádio cognitivo [5], [6], [2]. Trabalhos recentes demonstram sua eficiência em vários contextos de sensoriamento espectral [7], [8], classificação de modulação automática [9], detecção de sinal [10], [11], sensoriamento compressivo [12], processamento biológico de sinais [13] e monitoramento mecânico de vibrações [14], [15].

Recentemente, nosso grupo de pesquisa propôs uma nova abordagem para análise cicloestacionária, baseada na teoria da informação, denominada função de correntropia cíclica (CCF) [16], [17]. Foi demonstrado que a CCF obtém um melhor desempenho em ambientes com ruído impulsivo, quando comparada com outros métodos convencionais [12], [18], [19], [20]. A CCF se destaca no desenvolvimento de rádio cognitivo com aplicações de sensoriamento espectral [16], estimação de parâmetros [21] e classificação de modulações digitais em canais impulsivos de comunicação sem-fio [17].

No entanto, as funções de análise cicloestacionárias têm alto custo computacional e podem comprometer a vida útil da bateria dos dispositivos móveis. De fato, o uso de técnicas de processamento de sinais no cenário do rádio cognitivo é um dos principais fatores para o aumento do consumo de energia em dispositivos móveis [4].

A redução do consumo de energia é um desafio na era do processamento digital moderno. Para mitigar esse problema, os processadores *single-core* passaram a ser substituídos pelos processadores *multi-core*, oferecendo uma alternativa viável para aumentar o desempenho de processamento sem um aumento proporcional no consumo de energia [22]. Além disso, é possível mostrar que o uso de processamento paralelo pode permitir a redução do consumo de energia na execução de tarefas usando processadores *multi-core*. Em [23], os autores fornecem condições para economia de energia ideal em processadores *multi-core* usando programação paralela, com base na fração paralela das cargas de trabalho.

Devido a essa situação, torna-se importante realizar o processamento digital de sinais sob o aspecto de economia de energia utilizando programação paralela [23]. Os sistemas

Sávio R. M. Melo, Instituto Federal do Rio Grande do Norte, Pau dos Ferros, Brasil, saviorenna@gmail.com.

Herlan A. P. Silva, Instituto Federal do Rio Grande do Norte, Pau dos Ferros, Brasil, herlanassis@gmail.com.

Samuel Xavier-de-Souza, Universidade Federal do Rio Grande do Norte, Natal, Brasil, samuel@dca.ufrn.br.

Luiz Felipe Q. Silveira, Universidade Federal do Rio Grande do Norte, Natal, Brasil, lfelipeqs@gmail.com.

Aluisio I. R. Fontes, Instituto Federal do Rio Grande do Norte, Pau dos Ferros, Brasil, aluisio.rego@ifrn.edu.br.

Manuscript received June 10, 2020; revised September 15, 2020.

computacionais agora têm eficiência energética e desempenho como um dos principais requisitos. O conjunto de hardware e software geralmente ignorava problemas de energia, ou não tinham um gerenciamento eficiente em nível de software do ponto de vista de energia [23]. No entanto, não há estudo sobre a eficiência energética da função de correntropia cíclica em um cenário de rádio cognitivo.

Este artigo propõe o desenvolvimento de um algoritmo paralelo para a função de correntropia cíclica, a fim de obter uma redução do consumo de energia gasto na tarefa de sensoriamento espectral realizado em processadores *multi-core*. Como forma de analisar o consumo de energia, um modelo matemático foi usado para representar a energia elétrica dissipada nos processadores. Os resultados obtidos foram promissores, indicando uma redução significativa da energia quando utilizado o processamento paralelo.

O artigo está organizado da seguinte forma. Na Seção II é apresentada a correntropia, seguido da correntropia cíclica na Seção III. Na Seção IV discute-se sobre as métricas de análise de código paralelo. Na Seção V é apresentado um modelo matemático para a economia de energia com múltiplos processadores. Os detalhes sobre a proposta de implementação da correntropia cíclica com múltiplos processadores são apresentados na Seção VI. Os experimentos e resultados das simulações são discutidos na Seção VII, e por fim, as conclusões sobre o artigo são feitas na Seção VIII.

II. FUNÇÃO DE CORRENTROPIA

A extração de informações de uma fonte de dados é um problema frequente e relevante em muitas aplicações de processamento de sinais. Nesse contexto, medidas estatísticas de similaridade, como a correntropia, podem ser utilizadas com sucesso para a obtenção de informações e caracterização de dados.

A correntropia entre sinais aleatórios $\{x_t : t \in T\}$, em que t é o tempo e T é o conjunto de índices de interesse que pode ser definida como [24]:

$$V_x(t, \tau) = E[K_\sigma(x(t), x(t + \tau))], \quad (1)$$

em que $E[\cdot]$ é o operador esperança e $K_\sigma(\cdot, \cdot)$ é qualquer função simétrica definida positiva. Aplicações que utilizam a função Gaussiana como *kernel* são frequentes, pois ela possui propriedades que facilitam sua manipulação matemática. Originalmente, a correntropia definida por [24], $K_\sigma(\cdot, \cdot)$ é um *kernel* Gaussiano, dado por:

$$K_\sigma(x_{t_1}, x_{t_2}) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left\{-\frac{(x_{t_1} - x_{t_2})^2}{2\sigma^2}\right\}, \quad (2)$$

em que σ é denominada de largura do *kernel* ou tamanho do *kernel*. O ajuste desse parâmetro possibilita melhorar o desempenho da função na extração de características do sinal de interesse, além de minimizar os efeitos do ruído não-Gaussiano presentes no processo. Quando apenas um número finito de dados $(X_n, X_n)_{n=1}^N$ está disponível, pode-se definir um estimador para a correntropia por [24]:

$$\widehat{V}(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} k_\sigma(x(n), x(n + \tau)). \quad (3)$$

O estimador definido por (3), não garante média zero mesmo quando os dados de entrada estão centralizados, devido as transformações não lineares produzidas pelo *kernel*. Porém, foi definido em [24] um estimador centralizado para a correntropia, dado por:

$$\widehat{V}(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} k_\sigma(x_n, x_{n+\tau}) - \frac{1}{N^2} \sum_{n=0}^{N-1} \sum_{\tau=0}^{N-1} k_\sigma(x_n, x_{n+\tau}). \quad (4)$$

Nos últimos anos, a correntropia tem sido aplicada com sucesso na solução de diversos problemas em comunicações sem fio, tais como sensoriamento espectral [16], classificação automática de modulações [22], equalização de canal [25], [26], classificação de doenças vocais [27], entre outros.

III. FUNÇÃO DE CORRENTROPIA CÍCLICA

Em [16] considera-se que os processos que têm função de correntropia periódica são cicloestacionários de segunda ordem ou de ordem superior.

Dado um sinal modulado cicloestacionário $x(t)$, a função de correntropia $V_x(t, \tau)$ pode ser indicada pela série *Fourier* como:

$$V_x(t, \tau) = \sum_{\epsilon} V_x^\alpha(\tau) e^{j2\pi\alpha t}, \quad (5)$$

em que $\alpha = 1/T_0$ é a frequência cíclica e T_0 é o período de $V_x(t, \tau)$. É dado o nome Função de Correntropia Cíclica para os coeficientes da série de *Fourier*, $V_x^\alpha(\tau)$ que são definidos por [16]:

$$V_x^\alpha(\tau) = \frac{1}{T} \int_{-T/2}^{T/2} V_x(t, \tau) e^{-j2\pi\alpha t} dt, \quad (6)$$

em que $V_x(t, \tau)$ é a função de correntropia definida em (1). De acordo com essas formulações, são funções de intervalo de tempo e são indexadas pela frequência cíclica α .

Para explicar a eficácia da CCF na extração de características, a Fig. 1 mostra as assinaturas cicloestacionárias BPSK (*Binary Phase Shift Keying*). A assinatura contém componentes cíclicos como $\pm f_b/f_s$, $\pm 2f_b/f_s$, $(2f_c \pm 2f_b)/f_s$, $(4f_c \pm f_b)/f_s$ and $(4f_c \pm 2f_b)/f_s$, em que f_b é a frequência de bit e f_c a frequência da portadora. A CCF é uma função bidimensional da frequência f e de frequência cíclica α .

IV. MÉTRICAS DE ANÁLISE DE PARALELISMO

Atualmente, os processadores *multi-core* alcançaram a maior parte do espaço de mercado anteriormente ocupado pelos processadores *single-core*. Os *multi-core* são formados por p núcleos ($p \geq 2$; $p \in N$), interligados de modo a formar um dispositivo único. Para que se tenha uma ideia do ganho ou perda que se obtém com a execução paralela de um algoritmo é necessário utilizar métricas paralelas, dentre as quais as mais utilizadas são o *speedup* e a eficiência [28].

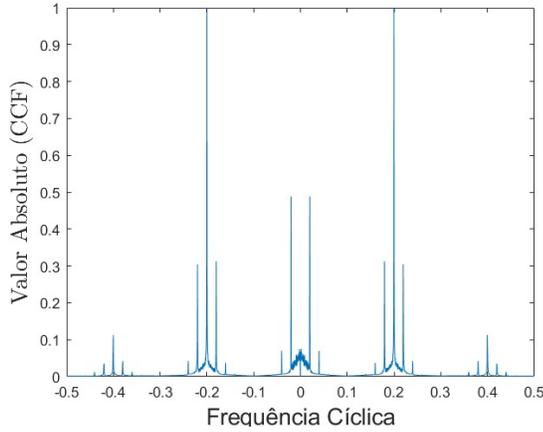


Fig. 1. Assinaturas ciclostacionárias BPSK obtidas pela CCF. Frequência de amostragem (f_s) 10 kHz, frequência portadora (f_c) 1 kHz e taxa de símbolos (f_b) 200 baud.

Em computação paralela, o *speedup* é definido como a divisão do tempo de processamento serial T_s pelo tempo de processamento paralelo T_p . Ele fornece o valor de quantas vezes o algoritmo paralelo é mais rápido que o sequencial e é definido como:

$$S = \frac{T_s}{T_p}. \quad (7)$$

Um *speedup* linear ou ideal é obtido quando a razão entre T_s e T_p é igual ao número de núcleos p da arquitetura *multi-core*. Assim, a meta para os algoritmos paralelos é tentar alcançar o *speedup* linear [29].

A eficiência é um valor que expressa o percentual de *speedup* alcançado pelo algoritmo em relação ao *speedup* linear. Ela demonstra o poder de processamento que realmente está sendo utilizado para realizar o cálculo e quanto dessa energia está sendo desperdiçada devido ao *overhead* do paralelismo, como [29]:

$$E = \frac{S}{p} = \frac{T_s}{pT_p}. \quad (8)$$

A escalabilidade em paralelismo é comumente associada com a eficiência do algoritmo paralelo, e indica a capacidade que um programa paralelo tem em manter uniforme a sua eficiência, a medida que se aumenta o número de unidades de processamento e o tamanho do problema.

Neste trabalho, foi feito o estudo do *speedup*, eficiência e escalabilidade de um novo algoritmo paralelo de análise cicloestacionária utilizando a correntropia. Inicialmente, analisamos a fração sequencial e paralela do algoritmo para entendermos o tempo gasto com comunicação, sincronização e balanceamento de carga. Através do *speedup* foi utilizado um modelo matemático para a representação da potência elétrica dissipada nos processadores.

V. ECONOMIA DE ENERGIA COM MÚLTIPLOS PROCESSADORES

A redução do consumo de energia nos processadores modernos é um desafio, particularmente em plataformas portáteis,

móveis e em supercomputadores. Em particular, a eficiência energética é um dos temas mais importantes para o projeto de um processador moderno [30].

Uma das formas de se compreender o gasto energético de um processador é através da formulação de modelos analíticos que representem bem seus comportamentos. Considerando a tecnologia CMOS, pode-se assumir duas fontes principais de dissipação de potência em um processador, a potência dissipada devido ao chaveamento dos transistores durante as transições lógicas e a potência dissipada por correntes parasitas nos transistores. Admitindo-se o uso de técnicas de redução das correntes parasitas nos transistores, pode-se aproximar a potência total P dissipada em um processador por sua potência de chaveamento, expressa por [31], [32]:

$$P \approx P_{sw} = kV^2F, \quad (9)$$

em que k é uma constante de proporcionalidade, V é a tensão de alimentação e F é a frequência utilizada no circuito.

Por outro lado, a tensão de alimentação V restringe a frequência a um valor máximo F_{max} , dado por:

$$F_{max} \propto \frac{(V - V_{th})^h}{V}, \quad (10)$$

em que V_{th} denominado tensão *threshold*, corresponde ao limiar de tensão que permite colocar o transistor em ação e h é uma constante geralmente assumida igual a 2. Neste caso, pode-se considerar $F \propto V$, e conseqüentemente [23]:

$$P \propto F^3. \quad (11)$$

A equação (11) representa a influência da frequência de operação dos processadores sobre a sua potência dissipada, de modo que uma pequena elevação no seu valor aumenta a potência numa taxa bem maior, com conseqüência imediata para os dispositivos alimentados por bateria, que possuem fortes limitações de disponibilidade de energia para o seu funcionamento. Entretanto, com o advento do processador *multi-core* é possível reduzir a frequência de operação do processador, e conseqüentemente reduzir o seu consumo de energia, mantendo-se, entretanto, o mesmo tempo final de execução que uma determinada tarefa levaria sobre um processador *single-core*, operando a uma frequência mais elevada [33].

Em [31], os autores relacionaram a energia consumida em uma aplicação executada por um processador *single-core* (sc) com aquela consumida por um processador *multi-core* (mc), propondo o fator de redução de energia, definido por:

$$E_r = \frac{E_{mc}}{E_{sc}} = \frac{P_{mc}T_p(F_{mc})}{P_{sc}T_1(F_{sc})}, \quad (12)$$

em que $T_p(F_{mc})$ é o tempo de execução paralela de uma tarefa, por um processador *multi-core* operando a uma frequência F_{mc} , e $T_1(F_{sc})$, o tempo de execução sequencial, por um processador *single-core* operando a uma frequência F_{sc} . P_{mc} é a potência consumida por um processador com p núcleos e P_{sc} , com um único núcleo. Desse modo, só haverá economia de energia se $E_r < 1$. A partir de (12), também é possível relacionar o fator de redução de energia E_r em

função da quantidade de processadores e do *speedup* obtido com o algoritmo paralelo, obtendo-se

$$E_r = \frac{p}{S_p^3}. \quad (13)$$

Dessa expressão pode-se inferir, portanto, que quanto menor o valor de E_r , menor energia o processador *multi-core* consumiria quando comparado com o consumo do *single-core*. Assim, quanto mais escalável for o algoritmo paralelo utilizado, menor será a energia consumida em seu processamento.

VI. PROPOSTA DE ALGORITMO

A computação paralela tem sido bastante utilizada como um poderoso recurso para permitir o desenvolvimento de soluções que exigem alto desempenho e baixo consumo energético [34]. Neste trabalho, foram utilizadas técnicas de paralelismo para permitir a redução da energia consumida e o tempo de processamento nas estimativas das características cicloestacionárias para o cálculo da função de correntropia cíclica. Esse método possui um parâmetro livre que regula os momentos estatísticos na extração de características da fonte de dados. Baseando-se em [8], a largura do *kernel* utilizada no cálculo da correntropia cíclica foi igual a 0.7. É importante destacar que o valor da largura do *kernel* não influencia no custo computacional do método.

A implementação paralela proposta foi desenvolvida na linguagem de programação C com o uso da *Application Programming Interface (API) Open Multi-Processing (OpenMP)* [34]. *OpenMP* é um padrão mantido pela *Architecture Review Board (ARB)*, cujas características para programação paralela inclui eficiência, portabilidade e uso amigável através de plataformas diferentes. As *threads* recebem tarefas que são divididas entre elas e então executados simultaneamente, isso acontece através de um ambiente de execução que as distribui para diferentes processadores.

A estratégia de paralelização adotada no algoritmo proposto, ilustrada nas Fig. 2 e Fig. 3, ocorre em quatro etapas. Nos diagramas, os blocos sobrepostos representam a carga computacional dividida entre os processadores e as setas indicam as *threads* em execução. O primeiro passo é dividir o sinal em L blocos de N amostras utilizando o janelamento de *Hamming*, em seguida computar a correntropia centralizada para cada bloco, e então realizar uma média dos blocos. O T_w ilustrado na Fig. 3 demonstra a quantidade de amostras enquanto N e r demonstra o atraso do janelamento de *Hamming*. Por fim, para o cálculo da ciclorrentropia é utilizada a transformada discreta de *Fourier* (DFT) através da *Fast Fourier Transform* (FFT) [35].

VII. RESULTADOS

O método da correntropia cíclica paralela foi implementado com o compilador GCC 6.3.0, que suporta a API *OpenMP*. A plataforma de testes consiste em uma máquina com 4 processadores AMD *Opteron(tm) Processor 6376*, cada um com 16 núcleos, totalizando 64 núcleos. Compostos por uma memória *cache* L1 de tamanho 64 KB, L2 de tamanho 2048

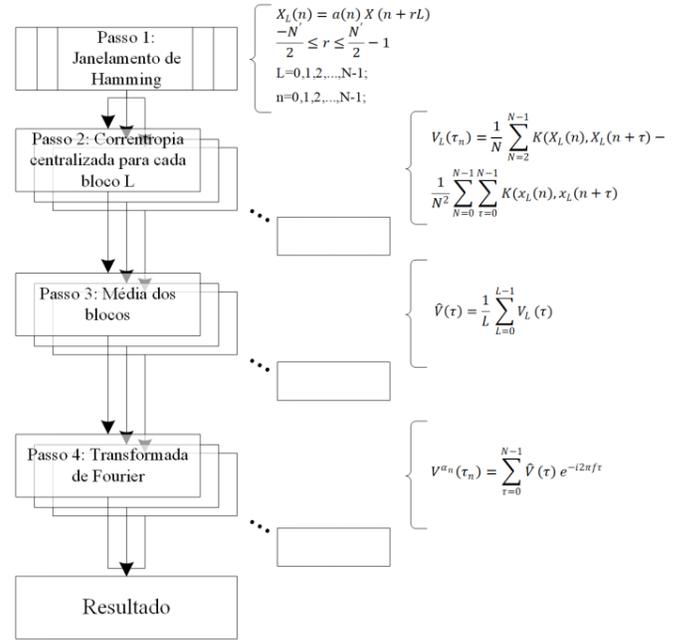


Fig. 2. Diagrama de blocos com o pseudocódigo do algoritmo paralelo proposto para o cálculo da Correntropia cíclica.

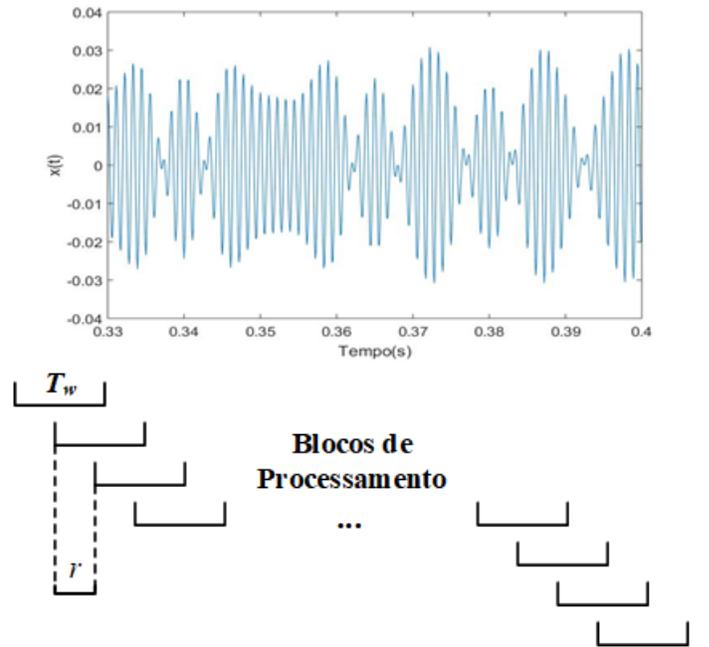


Fig. 3. Ilustração do processamento do sinal dividido em blocos com o janelamento de *Hamming*.

KB e uma L3 de 6144 KB, operando em uma frequência de 2.3 GHz. Esta máquina tem 264 GB de memória e usa o sistema operacional GNU / *Linux Ubuntu 16.04.2* de 64 bits.

Os testes foram realizados em um sinal digital caracterizado pela modulação *Binary Phase Shift Keying* (BPSK), podendo ser representado pela equação:

$$s(t) = A(t)\cos(2\pi f_i + \theta(t)), \quad (14)$$

em que $A(t)$ é a amplitude do sinal, f_i representa a frequência intermediária e θ a fase do sinal. Todos esses parâmetros foram definidos de acordo com o formato de modulação, com frequência de amostragem (f_s) de 10 KHz, f_i de 1 KHz e taxa de símbolo (f_b) de 200 *baud*, com filtro de formatação de pulso do tipo raiz quadrada do cosseno elevado, com fator de rolamento (*roll-off*) $r = 0,5$. Esse mesmo sinal foi utilizado no artigo [17].

A avaliação do fator de energia E_r ocorre de forma relativamente direta, a partir de (12). Assim, é possível observar que p contribui de forma direto no consumo de energia. Através do modelo proposto por [31], para haver garantia de economia de energia é suficiente que o S_p seja superior a $p^{\frac{1}{3}}$.

Foram analisados o *speedup* para os blocos de processamento de N amostras, indicados por T_w na Fig. 3, que representam o seu tamanho, variando-se os números de processadores de 1 a 64. Os resultados mostram uma redução no tempo de execução de aproximadamente 25 vezes quando todos os 64 núcleos são usados, considerando uma comparação em relação a simulação que utiliza um único núcleo, como pode ser visto na Fig. 4.

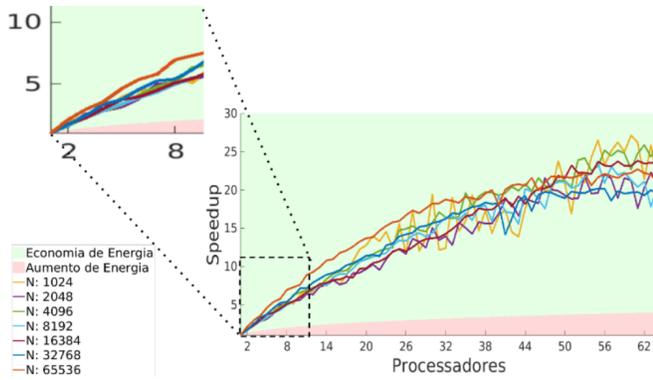


Fig. 4. *Speedups* para diferentes entradas N para o número de processadores $P = 1, \dots, 64$.

Observa-se na Fig. 4 que utilizando 10 processadores o *speedup* fica próximo ao ideal, isso significa que o algoritmo paralelo nessas condições alcança a sua maior redução no tempo de execução e consumo de energia. A área do gráfico em verde corresponde ao conjunto de *speedups*, que de acordo com (13), representa uma economia de energia, enquanto valores abaixo dessa faixa representam aumento no consumo de energia.

A Fig. 5 evidencia a relação entre o grau de aproveitamento dos recursos computacionais disponíveis. Observa-se que para um número de até 10 processadores, as três maiores entradas de N , obtivemos uma eficiência próxima de 80%. Isso significa que o grau de aproveitamento dos recursos computacionais utilizados por este algoritmo alcança valores próximos do ótimo. Entretanto, a eficiência cai para valores próximos a 60% quando utilizados todos os processadores, essa queda é explicada pelo *overhead* de comunicação entre as *threads*. É importante destacar que, o tempo gasto no *overhead* é minimizado quando o tamanho da entrada a ser processada é grande.

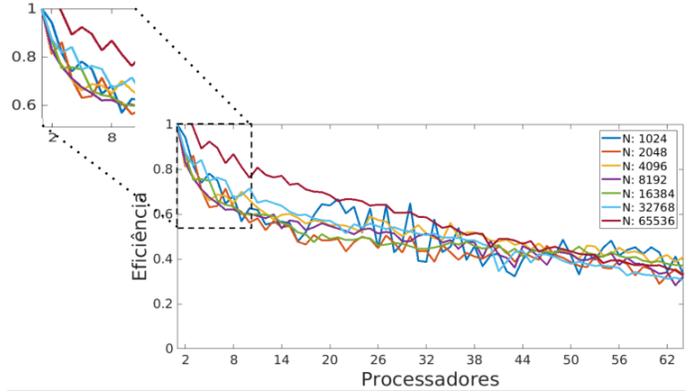


Fig. 5. Eficiência para diferentes entradas N para o número de processadores $P = 1, \dots, 64$.

Observa-se na Fig. 6, o comportamento do fator de energia frente a um conjunto de valores de P dados por 1, 2, 3, ..., 64. A medida que o valor de P aumenta, cresce a economia de energia, de modo que o valor ótimo referente a quantidade de núcleos a serem empregados no processamento também corresponde à quantidade máxima de processadores disponíveis ($P=64$). Portanto, a paralelização do algoritmo demonstra uma economia de energia para diferentes tamanhos de N .

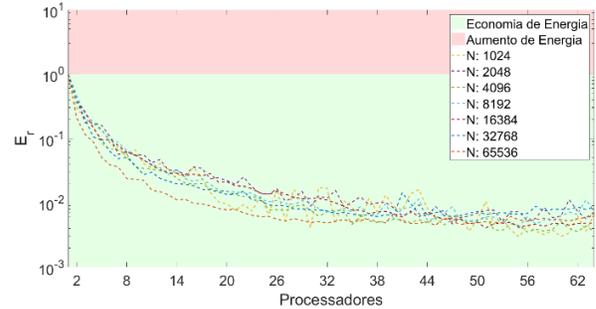


Fig. 6. Fator de escala de energia para diferentes entradas N para o número de processadores $P = 1, \dots, 64$.

No cenário de rádio cognitivo e rádio definido por software, a economia de energia e a minimização do tempo de execução das tarefas cognitivas são muito importantes. Em [16], [17], ficou comprovado que o uso da cicloestacionariedade com correntropia funciona muito bem para as aplicações de sensoriamento espectral e classificação automática de modulações, duas funcionalidades básicas do rádio cognitivo. E agora, foi comprovado que essa função de correntropia cíclica é escalável, diminuindo seu tempo a medida que aumenta a quantidade de processadores.

VIII. CONCLUSÃO

Neste trabalho foi proposta uma implementação paralela da correntropia cíclica com o objetivo de avaliar o seu *speedup*, eficiência e consumo de energia. Os resultados demonstraram que a correntropia cíclica é escalável com características de sistemas paralelos. Foram alcançadas reduções no tempo de

execução de aproximadamente 8 vezes e uma eficiência de 80% quando utilizados 10 processadores. Considerando o modelo matemático utilizado para a representação da potência elétrica dissipada nos processadores, constatou-se uma redução no consumo de energia com a utilização de processadores *multi-core*. Observou-se ainda que essa redução de consumo é tanto maior quanto maior for o número de núcleos usados no processamento do algoritmo paralelo proposto. Para trabalhos futuros, pretende-se medir o consumo de energia do algoritmo realizando experimentos em outras plataformas de computação paralela como CUDA NVIDIA e Intel Xeon Phi, com o objetivo de aferir o desempenho computacional do algoritmo proposto quando submetido a essas arquiteturas.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio do Instituto Federal do Rio Grande do Norte e da Universidade Federal do Rio Grande do Norte.

REFERÊNCIAS

- [1] F. S. P. T. Force, "Report of the spectrum efficiency working group," http://www.fcc.gov/spiff/files/SEWGFfinalReport_1.pdf, 2002.
- [2] F. Zhou, N. C. Beaulieu, Z. Li, J. Si, and P. Qi, "Energy-efficient optimal power allocation for fading cognitive radio channels: Ergodic capacity, outage capacity, and minimum-rate capacity," *IEEE Transactions on Wireless Communications*, vol. 15, no. 4, pp. 2741–2755, 2016.
- [3] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE communications surveys & tutorials*, vol. 11, no. 1, pp. 116–130, 2009.
- [4] M. R. Mili, L. Musavian, K. A. Hamdi, and F. Marvasti, "How to increase energy efficiency in cognitive radio networks," *IEEE Transactions on Communications*, vol. 64, no. 5, pp. 1829–1843, 2016.
- [5] S. Haykin, D. J. Thomson, and J. H. Reed, "Spectrum sensing for cognitive radio," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 849–877, 2009.
- [6] W. M. Jang, "Blind cyclostationary spectrum sensing in cognitive radios," *IEEE Communications Letters*, vol. 18, no. 3, pp. 393–396, 2014.
- [7] M. Yang, Y. Li, X. Liu, and W. Tang, "Cyclostationary feature detection based spectrum sensing algorithm under complicated electromagnetic environment in cognitive radio networks," *China Communications*, vol. 12, no. 9, pp. 35–44, 2015.
- [8] G. R. George and S. C. Prema, "Cyclostationary feature detection based blind approach for spectrum sensing and classification," *Radioengineering*, vol. 28, no. 1, pp. 298–303, 2019.
- [9] D. Zeng, X. Zeng, G. Lu, and B. Tang, "Automatic modulation classification of radar signals using the generalised time-frequency representation of zhao, atlas and marks," *IET radar, sonar & navigation*, vol. 5, no. 4, pp. 507–516, 2011.
- [10] W. A. Gardner and C. M. Spooner, "Signal interception: performance advantages of cyclic-feature detectors," *IEEE Transactions on Communications*, vol. 40, no. 1, pp. 149–159, 1992.
- [11] A. Pries, D. Ramirez, and P. J. Schreier, "Lmpit-inspired tests for detecting a cyclostationary signal in noise with spatio-temporal structure," *IEEE Transactions on Wireless Communications*, vol. 17, no. 9, pp. 6321–6334, 2018.
- [12] T. Liu, T. Qiu, and S. Luan, "Cyclic frequency estimation by compressed cyclic correntropy spectrum in impulsive noise," *IEEE Signal Processing Letters*, vol. 26, no. 6, pp. 888–892, 2019.
- [13] L. Yu, L. Yan, M. Chen, and L. Dong, "Cyclostationary modeling of surface electromyography signal during gait cycles and its application for cerebral palsy diagnosis," *Journal of Shanghai Jiaotong University (Science)*, vol. 23, no. 1, pp. 56–61, 2018.
- [14] A. Barrau and M. El Badaoui, "About the cumulants of periodic signals," *Mechanical Systems and Signal Processing*, vol. 99, pp. 684–690, 2018.
- [15] X. Zhao, Y. Qin, C. He, L. Jia, and L. Kou, "Rolling element bearing fault diagnosis under impulsive noise environment based on cyclic correntropy spectrum," *Entropy*, vol. 21, no. 1, p. 50, 2019.
- [16] A. I. R. Fontes, J. B. Rego, A. d. M. Martins, L. F. Silveira, and J. C. Príncipe, "Cyclostationary correntropy: Definition and applications," *Expert Systems with Applications*, vol. 69, pp. 110–117, 2017.
- [17] T. V. R. O. Câmara, A. D. L. Lima, B. M. M. Lima, A. I. R. Fontes, A. D. M. Martins, and L. F. Q. Silveira, "Automatic modulation classification architectures based on cyclostationary features in impulsive environments," *IEEE Access*, vol. 7, pp. 138 512–138 527, 2019.
- [18] J. Ma and T. Qiu, "Automatic modulation classification using cyclic correntropy spectrum in impulsive noise," *IEEE Wireless Communications Letters*, 2018.
- [19] S. Luan, T. Qiu, Y. Zhu, and L. Yu, "Cyclic correntropy and its spectrum in frequency estimation in the presence of impulsive noise," *Signal Processing*, vol. 120, pp. 503–508, 2016.
- [20] T. Liu, T. Qiu, and S. Luan, "Cyclic correntropy: Foundations and theories," *IEEE Access*, vol. 6, pp. 34 659–34 669, 2018.
- [21] F. Jin, T. Qiu, S. Luan, and W. Cui, "Joint estimation of the doa and the number of sources for wideband signals using cyclic correntropy," *IEEE Access*, vol. 7, pp. 42 482–42 494, 2019.
- [22] C. Isci, A. Buyuktosunoglu, C.-Y. Cher, P. Bose, and M. Martonosi, "An analysis of efficient multi-core global power management policies: Maximizing performance for a given power budget," in *Proceedings of the 39th annual IEEE/ACM international symposium on microarchitecture*. IEEE Computer Society, 2006, pp. 347–358.
- [23] C. A. Barros, L. F. Q. Silveira, C. A. Valderrama, and S. Xavier-de Souza, "Optimal processor dynamic-energy reduction for parallel workloads on heterogeneous multi-core architectures," *Microprocessors and Microsystems*, vol. 39, no. 6, pp. 418–425, 2015.
- [24] W. Liu, P. P. Pokharel, and J. C. Príncipe, "Correntropy: Properties and applications in non-gaussian signal processing," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5286–5298, 2007.
- [25] J. P. Guimaraes, A. I. Fontes, J. B. Rego, A. d. M. Martins, and J. C. Príncipe, "Complex correntropy: Probabilistic interpretation and application to complex-valued data," *IEEE Signal Processing Letters*, vol. 24, no. 1, pp. 42–45, 2017.
- [26] W. Ma, H. Qu, G. Gui, L. Xu, J. Zhao, and B. Chen, "Maximum correntropy criterion based sparse adaptive filtering algorithms for robust channel estimation under non-gaussian environments," *Journal of the Franklin Institute*, vol. 352, no. 7, pp. 2708–2727, 2015.
- [27] A. I. Fontes, P. T. Souza, A. D. Neto, A. d. M. Martins, and L. F. Silveira, "Classification system of pathological voices using correntropy," *Mathematical Problems in Engineering*, vol. 2014, 2014.
- [28] G. M. Amdahl, "Validity of the single processor approach to achieving large scale computing capabilities," in *Proceedings of the April 18-20, 1967, spring joint computer conference*. ACM, 1967, pp. 483–485.
- [29] X.-J. Yang, Y. Dou, and Q.-F. Hu, "Progress and challenges in high performance computer technology," *Journal of Computer Science and Technology*, vol. 21, no. 5, pp. 674–681, 2006.
- [30] B. Keller, B. Nikolic, and K. Asanović, "Opportunities for fine-grained adaptive voltage scaling to improve system-level energy efficiency," Ph.D. dissertation, Master's thesis, EECS Department, University of California, Berkeley, 2015.
- [31] A. D. Lima, C. A. Barros, L. F. Q. Silveira, S. Xavier-De-Souza, and C. A. Valderrama, "Parallel cyclostationarity-exploiting algorithm for energy-efficient spectrum sensing," *IEICE Transactions on Communications*, vol. 97, no. 2, pp. 326–333, 2014.
- [32] S. Usman, S. U. Khan, and S. Khan, "A comparative study of voltage/frequency scaling in noc," in *Electro/Information Technology (EIT), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1–5.
- [33] K. Banerjee, M. Pedram, and A. H. Ajami, "Analysis and optimization of thermal issues in high-performance vlsi," in *Proceedings of the 2001 international symposium on Physical design*. ACM, 2001, pp. 230–237.
- [34] B. Chapman, G. Jost, and R. Van Der Pas, *Using OpenMP: portable shared memory parallel programming*. MIT press, 2008, vol. 10.
- [35] M. Frigo and S. G. Johnson, "The design and implementation of fftw3," *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, 2005.



Sávio Rennan Menêzes Melo possui graduação em Análise e Desenvolvimento de Sistemas pelo Instituto Federal do Rio Grande do Norte (IFRN). Seus interesses de pesquisa estão voltados para o desenvolvimento de sistemas móveis colaborativos, sistemas distribuídos, tópicos de algoritmos e programação paralela. Em geral, suas maiores experiências estão concentradas na área de Sistemas da Computação, com ênfase em Ciência da Computação. <http://lattes.cnpq.br/8548403866737249>



Aluisio Igor Rêgo Fontes possui graduação em Engenharia da Computação pela Universidade Federal do Rio Grande do Norte (2010), mestrado e doutorado em Engenharia Elétrica e Computação pela Universidade Federal do Rio Grande do Norte 2012 e 2015 respectivamente. Atualmente é professor do Instituto Federal do Rio Grande do Norte. Tem experiência na área de Engenharia de Computação, com ênfase em Sistemas de Computação e Sistemas de Telecomunicações, atuando principalmente nos seguintes temas: Programação paralela, Computação de Alto Desempenho, Arquiteturas Embarcadas, Teoria da Informação, Correntropia e Rádio Cognitivo. <http://lattes.cnpq.br/7848819859172650>

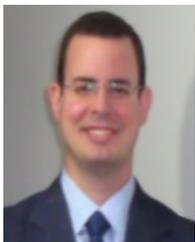


Herlan Assis Pereira da Silva possui graduação em Análise e Desenvolvimento de Sistemas pelo Instituto Federal do Rio Grande do Norte (IFRN). Seus interesses de pesquisa estão voltados para o desenvolvimento de sistemas colaborativos, sistemas distribuídos e programação paralela. Em geral, suas maiores experiências estão concentradas na área de Sistemas de Informação, com ênfase em Ciência da Computação. <http://lattes.cnpq.br/1364066585618049>



Samuel Xavier de Souza possui graduação em Engenharia da Computação pela Universidade Federal do Rio Grande do Norte - UFRN (2000) e doutorado em Engenharia Elétrica pela Katholieke Universiteit Leuven, Bélgica (2007). Trabalhou como Engenheiro de Software/Hardware no Centro Interuniversitário de Micro- e Nanoeletrônica - IMEC, Bélgica, e como consultor de pesquisa no Centro de Supercomputação de Flanders - VSC, Bélgica. Desde 2009 é professor do Departamento de Engenharia de Computação e Automação da UFRN.

Fundou em 2016 o Núcleo de Processamento de Alto Desempenho-NPAD que hoje coordena com o objetivo de facilitar o uso da supercomputação como ferramenta de pesquisa na UFRN. Em 2016, recebeu o prêmio Royal Society-Newton Advanced Fellowships por suas pesquisas em Software Paralelo Energeticamente Eficiente. Seus interesses de pesquisa estão em eficiência energética de software, eficiência e escalabilidade de sistemas paralelos, algoritmos e arquiteturas paralelas e suas aplicações. <http://lattes.cnpq.br/9892239670106361>



Luiz Felipe de Queiroz Silveira possui graduação em Engenharia Elétrica pela Universidade Federal da Paraíba [C. Grande] (2000), mestrado em Engenharia Elétrica pela Universidade Federal de Campina Grande (2002) e doutorado em Engenharia Elétrica pela Universidade Federal de Campina Grande (2006). Atualmente é professor adjunto da Universidade Federal do Rio Grande do Norte. Tem experiência na área de Engenharia Elétrica, com ênfase em Sistemas de Telecomunicações, atuando principalmente nos seguintes temas: Codificação de Canal, Wavelets, Rádio Cognitivo e Modulação Adaptativa. <http://lattes.cnpq.br/4139452169580807>