SMNE UFPR
Simpósio de Métodos Numéricos em Engenharia
25 a 27 de outubro, 2017

Correlação de Pearson e agrupamento por medida de similaridade: um aumento na eficiência energética de sensores

Rayon Lindraz Nunes
Engenharia da Computação
Universidade Federal do Ceará
Sobral, Ceará
rayonnunes@hotmail.com

Fernando Almeida Jr
Engenharia da Computação
Universidade Federal do Ceará
Sobral, Ceará
fernandorodrigues@sobral.ufc.br

***Resumo*—Reduzir o consumo de energia em uma Rede de Sensores sem-fio é um desafio que pode ser alcançado ao agrupar sensores e reduzir o tamanho dos pacotes de dados enviados na rede. Este trabalho apresenta a implementação de duas abordagens: A correlação de Pearson que busca, em um conjunto de dados que serão enviados pelo sensor, leituras que são estatisticamente iguais e com isso evitar mensagens redundantes; E a medida de similaridade, que baseia-se no princípio do agrupamento comportamental, onde são avaliadas leituras em dado período de tempo e agrupadas à sensores com o mesmo padrão na rede. As abordagens foram utilizadas sobre dados reais de sensores sem fio em laboratório, utilizando o simulador SinalGo. Os resultados comparam a Correlação de Pearson com uma abordagem padrão e mostram redução de 40% no tamanho do pacote de leituras enviadas à fonte. Já a medida de similaridade é capaz de reduzir em até 89,4% o número de mensagens na rede se comparadas à mesma abordagem padrão.**

***Palavras-chave*—Redes de sensores; Consumo energético; Simulação.**

I. Introdução

Redes de Sensores sem Fio (RSSF) consistem em um conjunto de nós sensores que possuem recursos limitados. Alguns destes sensores coletam dados de ambientes externos e enviam informações, como temperatura, umidade e luminosidade, para uma estação principal denominada de nó *sink*. A informação é enviada atráves de sensores intermediários até chegar ao sink, sendo esta forma de transmissão denominada *multi-hop* [1], como ilustrado na Figura 1.

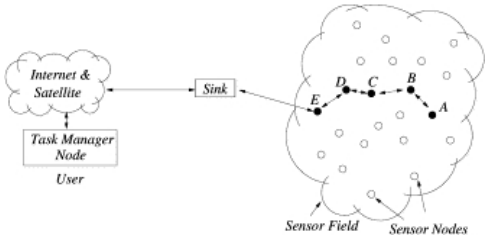

Figura 1: Topologia de uma rede multi-hop. [2]

No âmbito das RSSF, métodos estatísticos são amplamente empregados para modelagem de dados, inferência sobre comportamento ambiental e previsão de fenômenos futuros. Entretanto esse processo de sensoriamento e, principalmente, o envio de mensagens pela rede, demanda custos energéticos consideráveis. Existem diversos padrões de comunicação entre esses sensores como o IEEE 802.11 (wi-fi), IEEE 802.15 (bluetooth) que possui um maior foco em economia de energia. Ademais, há uma necessidade de auto-organização das RSSF, propostos por protocolos como SMACS, EAR, SAR e ASCENT [3]. Suas funcionalidades residem em, respectivamente, diminuir o tamanho das mensagens suprimindo conteúdos repetidos ou irrelevantes e agrupar os sensores que possuem uma mesma correlação comportamental em *clusters*. Uma primeira abordagem para reduzir o tamanho dos pacotes enviados na rede, impactando em uma menor potência necessária para o dispositivo enviar sua mensagem, seria a correlação de Pearson que procura calcular a

intensidade de associação linear de uma variável independente em relação à outra dependente, o coeficiente resultante *r* retorna um valor $-1 \leq r \leq 1$, onde o valor mais baixo negativo indica uma correlação inversamente proporcional, o valor mais alto positivo indica correlação direta (e.g: a medida que a luminosidade aumenta, isto pode influenciar em um aumento da temperatura) e o valor 0 (zero) indica inexistência de correlação entre duas variáveis, como mostra a Figura 2.

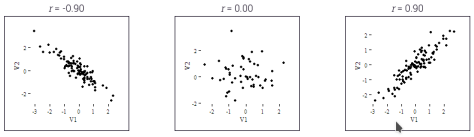

Figura 2: Correlação de Pearson. - [4]

É possivel notar que a medida que *r* aproxima-se de $\pm$ 1 a distribuição das leituras torna-se semelhante a uma equação linear $y = a+bx$. Portanto, dado que *r* é superior a um determinado limiar, é possível modelar esta equação através do cálculo dos coeficientes de regressão linear , como na Equação (2) e (3). É possível modelar a mesma equação de regressão linear para que seus coeficientes possam inferir futuras leituras onde, a partir de então, os sensores apenas comunicarão-se com o *sink* caso haja uma leitura fora da curva de predição que será denominada "novidade".

A segunda abordagem trata do agrupamento de sensores através da medida de similaridade [5], cujo princípio se baseia na correlação comportamental. Diferente da correlação espacial que procura agrupar sensores próximos um do outro, a correlação comportamental se baseia no fato de que mesmo entre sensores distantes, há a possibilidade de valores de leitura semelhantes, e.g.: em um mapeamento de umidade [6] como mostra a Figura 3, é possível notar curvas de nível que indicam um mesmo valor nos pontos contornados.

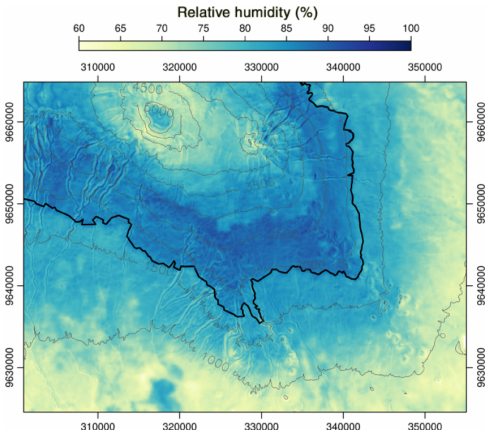

Figura 3: Curvas de nível de umidade [6]

A medida de similaridade então define dois parâmetros que afirmam se dados dois sensores, eles são similares por comportamento ou não: A magnitude apresentada na Equação (4) compara o desvio dos valores absolutos das leituras, quanto menor o desvio, maior a similaridade por magnitude. A tendência que de acordo com a Equação (5) mede as taxas de crescimento ou decaimento das leituras a partir de suas séries temporais.

Sensores que possuem correlação comportamental podem ser agrupados em *clusters*, sendo este representado por um *cluster head*, que consiste no sensor do agrupamento que possuir maior nível de bateria ou menor distância para o *sink*, caso haja sensores com a mesma carga energética. O *cluster head* desempenhará o papel de informar ao *sink* leituras fora da curva de predição até que haja uma reconfiguração do sistema de clusters ou atinja o fim de seu ciclo.

II. Materiais e Métodos

A motivação deste trabalho parte da necessidade de redução no consumo de energia de uma RSSF, estendendo assim tempo de coleta de dados. A realização deste objetivo permite uma mudança significativa na realidade de muitas pesquisas que dependem do tempo de vida útil destas redes para adquirir o maiores e melhores dados para exprimentos. Este trabalho propõe duas abordagens de implementação de um algoritmo de sensoriamento utilizando Correlação de Pearson e Medida de Similaridade [5], [7]. Para realizá-lo, foi necessário o uso da plataforma de simulação de redes de sensores *SinalGO* que utiliza a linguagem de programação *JAVA*, por sua praticidade ao realizar experimentos com centenas de sensores com razoável tempo de execução. Para analisar e comparar os métodos de sensoriamento usados pela simulação, a base de dados utilizada provém do Laboratório Intel Berkeley Research que possui dados de 54 sensores fisicamente distribuídos em laboratório, analisando como parâmetros: temperatura, umidade, luminosidade e voltagem, registrando o tempo de leitura e sendo transmitidas ao nó *sink* a cada 31s entre 28/02/2004 e 05/04/2004, que consistiu em 1000 leituras de cada um dos nós sensores.

Foram coletados dados de 53 em 54 sensores, tendo em vista que o quinto sensor apresentava uma alta taxa de dados corrompidos. O algoritmo então pode ser dividido em 9 passos mostrados abaixo:

1) *Flooding*: o nó *sink* envia uma mensagem via *broadcast* para todos os nós sensores da rede, solicitando uma quantidade *n* de leituras. foram tomadas 70 leituras iniciais na simulação. Como a abordagem trata do encaminhamento de mensagens via *multi-hop*, cada sensor repassa a requisição para os seus nós vizinhos, sendo possível traçar uma rota entre cada sensor e o nó *sink*.
2) Leituras Iniciais: Ao realizar as leituras solicitadas, o nó então deverá decidir como enviará esse pacote para o sink. Para isso, o sensor se utiliza da correlação de Pearson. será feita uma comparação entre todos os parâmetros que o sensor é capaz de analisar tomados dois a dois, como descrito na equação abaixo.

$$r = \frac{\sum_{i=1}^{N} (x_{\mathrm{i}} - \bar{x})(y_{\mathrm{i}} - \bar{y})}{\sqrt{(x_{\mathrm{i}} - \bar{x})^2 (y_{\mathrm{i}} - \bar{y})^2}} \quad (1)$$

Onde r é coeficiente de correlação de Pearson, *N* o número de leituras, x_i e y_i são dois parâmetros comparados sendo $\bar{x}$ e $\bar{y}$ suas respectivas médias. É gerado um coeficiente *r* para cada comparação e esse valor é atribuído como uma pontuação para cada parâmetro. Aquele que possuir maior pontuação, será tratado como variável independente. E.g: dadas temperatura, umidade e luminosidade como parâmetros, o *r* gerado para a correlação entre temperatura e umidade, temperatura e luminosidade, umidade e luminosidade foram de, respectivamente: 0,8; 0,4; e 0,6; logo a umidade seria tratada como variável independente por possuir 1,4 pontos (0,8 pontos comparado com a temperatura mais 0,6 pontos comparado com a luminosidade).

Caso a correlação seja superior a um limiar determinado na configuração da rede de sensores ($r \leq R$), então os dados do parâmetro dependente não serão enviados e serão gerados os coeficientes de regressão linear. Em todo caso, a série de leituras da variável independente deverá ser sempre enviada.

3) Gerando os coeficientes de regressão linear: A ideia para reduzir o tamanho dos pacotes enviados na rede é a de substituir toda a série de leituras por apenas dois coeficientes β e α calculados. Isso torna possível, para o *sink*, reconstruir as leituras dos sensores com uma precisão satisfatória, dado que *r* é suficiente para obter estes valores, tomado as Equações (2) e (3) para modelar a equação da reta que deverá se assemelhar à curva da correlação de pearson.

$$\beta = \frac{\sum_{i=1}^{N}(T_\mathrm{i} - \bar{T})(S_\mathrm{i} - \bar{S})}{\sum_{i=1}^{N}(T_\mathrm{i} - \bar{T})^2} \tag{2}$$

$$\alpha = \bar{S} - b\bar{T} \tag{3}$$

Onde $\mathrm{T_i}$ corresponde ao termo da série temporal de leituras, $\mathrm{S_i}$ aos valores de cada observação, e $\bar{T}$ e $\bar{S}$ aos seus valores médios, respectivamente.

4) Reconstrução dos dados: O nó *sink* recebe a resposta dos sensores que carregam as *n* leituras iniciais, sendo este capaz de perceber se houveram leituras que tiveram parâmetros colapsados através da presença ou ausência dos coeficientes de regressão de cada parâmetro das variáveis dependentes. Caso existam, o *sink*, a partir das leituras da variável independente, é capaz de restaurar as leituras omitidas através dos coeficientes α e β. Então, o *sink* não mais diferenciará as leituras que foram colapsadas das leituras que foram enviadas completamente. Todos os procedimentos serão aplicados sem nenhuma exceção.
5) Agrupando sensores: Através da abordagem de agrupamento comportamental, como mencionada anteriormente, será utilizada a medida de similaridade para formação dos *clusters* baseados na similaridade entre os parâmetros:
Magnitude ($\bar{m}$)

$$\bar{m} = \frac{\sum_{i=1}^{N}|s_1 - {s'}_1|}{n} \leq M \tag{4}$$

Tendência ($\bar{t}$)

$$\bar{t} = \frac{P}{n} \leq T \tag{5}$$

Onde *'n'* consiste no número total de dados sensoreados e *'P'* corresponde ao número de pares de leituras(s_i, s'_i) que satisfazem $\Delta \mathrm{s_i} \times \Delta \mathrm{s'_i} \geq 0$, com $1 \leq i \leq n$.
Então o nó *sink* agrupará os demais nós sensores em *Clusters* iguais, caso a média das diferenças entre as leituras de sensores seja menor ou igual ao parâmetro *M* e a porcentagem de crescimento ou decaimento seja maior ou igual ao parâmetro *T*.

6) Definindo *Cluster Heads*: Dado que os sensores estão agrupados, o *sink* então definirá o nó sensor que possuir o maior nível de bateria como o *Cluster Head*. Para o caso de equivalência entre o nível de bateria de dois sensores, será escolhido aquele que possuir uma menor distância para o nó *sink*. A configuração se mantém até que o *Cluster Head* comunique uma novidade ao *sink*.
7) Cálculo dos coeficientes de regressão linear para predição das leituras: este passo utiliza novamente o conceito de regressão linear, porém, neste momento o *sink* fará o cálculo dos coeficientes de regressão *a* e *b* para cada *cluster*, de modo que estes sejam repassados para cada nó.
8) Predição das leituras: O sensor, de posse dos coeficientes de regressão *a* e *b*, será capaz de detectar se os as leituras seguintes estão dentro da curva de regressão linear, verificando se um valor de leitura *v* possui uma divergência *t* do valor *p* predito pelo *sink*, ou seja, $v \notin [p - t, p + t], t \geq 0$. Caso positivo, o sensor apenas continuará realizando as leituras sem a necessidade de injetar mensagens na rede, impactando diretamente no consumo de energia. Caso contrário ou caso o *Cluster Head* atinja um nível crítico de bateria, o sensor detectará uma novidade e informará ao *Cluster Head* de seu grupo, que analisará a frequência dessas novidades e, caso ultrapasse uma margem especificada pelo utilizador (e.g. 5%), a novidade será então concentrada no *Cluster Head* e encaminhada ao *sink*, levando, além das leituras, os níveis atualizados de bateria. É importante ressaltar que as mensagens serão enviadas somente do *cluster* detector de novidades.
9) Reconstrução dos *Clusters*: Sempre que o nó *sink* receber uma novidade ou não conseguir detectar resposta do *Cluster Head*, um gatilho é disparado recalculando os coeficientes de regressão, e caso haja uma divergência entre os novos coeficientes recebidos e os atuais, os *clusters* podem sofrer um *splitting* (divisão), que detectará o valor de leitura *v* fora do intervalo $[p - t, p + t], t \geq 0$. Com isso, serão divididos em dois ou mais grupos, definindo novos *cluster heads* para cada um. Entretanto, isso causaria em pouco tempo um estado de fragmentação a tal ponto que os *clusters* seriam compostos apenas por um único sensor. Procurando solucionar este problema, ocorrerá uma processo de *merging* (mesclagem) onde sensores com coeficientes dentro do mesmo intervalo, passarão a compor um único *cluster*, sendo o *Cluster Head* escolhido atráves da voltagem e/ou distância do *sink* e os novos coeficientes atualizados serão disparados para todos os sensores na rede.

A rede então se manterá automaticamente até que o sink seja incapaz de se comunicar com pelo menos um sensor da rede. Em

um sistema de transmissão de mensagens via *multi-hop*, é possível que sensores ainda ativos sejam incapazes de se comunicar-se com o *sink*, se seus nós intermediários não forem capazes de fazê-lo.

Para tornar possível este trabalho, os testes se baseiam sobre dados de sensores reais, avaliados em ambiente controlado e a linguagem de programação Java foi escolhida por explorar a estrutura da plataforma SinalGo para algoritmos em RSSF. As leituras utilizados foram retiradas a partir de sensores reais de uma base de dados disponibilizado pela *Intel Berkeley Research Laboratory*, disponibilizando parâmetros de leituras sobre temperatura, umidade, luminosidade e voltagem de 54 sensores fisicamente distribuídos em laboratório, transmitindo mensagens ao sink a cada 31 segundos entre 28/02/2004 e 04/05/2004. O algoritmo se encontra disponível no GitHub, em [8].

III. Resultados e Discussão

Para realizar os testes foram executadas uma rodada de 70 leituras iniciais, seguidas de 1.000 ciclos para comparar o tamanho das mensagens enviadas na rede e a quantidade de mensagens na rede para comparar a eficácia das abordagens da correlação de Pearson e medida de similaridade, respectivamente. Inicialmente, os testes são executados com o modo de controle, denominado *naive*, onde as mensagens serão todas enviadas ao *sink*, como mostram as Tabelas I, II e III:

Tabela I: EXECUÇÃO DA SIMULAÇÃO EM MODO NAIVE

Resultados			
Rodadas	RMSE	NumMsg	SRead
70	0	13697	3522
250	0	54017	13062
500	0	110017	26312
750	0	166017	39562
1000	0	222017	52812

Tabela II: EXECUÇÃO DA SIMULAÇÃO UTILIZANDO MEDIDA DE SIMILARIDADE

Resultados			
Rodadas	RMSE	NumMsg	SRead
70	0	234	3780
250	39,006	4562	13460
500	35,497	10841	26932
750	32,801	17142	40388
1000	30,711	23462	53872

Tabela III: TAMANHO DO PACOTE UTILIZANDO A CORRELAÇÃO DE PEARSON, EM BYTES

Resultados		
Rodadas	Naive	Pearson
70	719040	441280

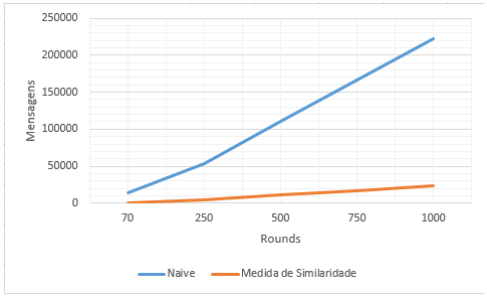

Figura 4: Quantidade de mensagens na rede entre o modo naive e a medida de similaridade

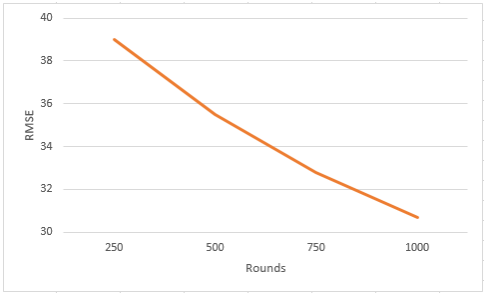

Figura 5: RMSE utilizando a medida de similaridade

A Tabela I e a Tabela II demonstram que o paramêtro *NumMsg*, que corresponde ao número de mensagens, atingiu uma redução de 89,4%, comparando o modo *naive* com a medida de similaridade após o milésimo round, demonstrado através do gráfico da Figura 4. A raiz do erro quadrático médio (do inglês, Root Mean Square Error, ou RMSE) apresentado é uma medida que procura observar as diferenças entre estimadores, ou mesmo comparar as diferenças entre o valor real e o valor observado. De acordo com o teorema do limite central [9], dado um número de observações suficientemente grande, a amostra passa a se comportar como uma distribuição uniforme, o que explica a queda do RMSE mostrado na Figura 5, ao passo que as observações crescem, o valor do RMSE diminui. O Parâmetro *SRead* corresponde ao número total de leituras dos sensores, utilizado para controle da simulação. A Tabela III e a Figura 6 apresentam, por sua vez, o tamanho do pacote enviado durante as 70 leituras iniciais, onde há todos os dados sensoriados

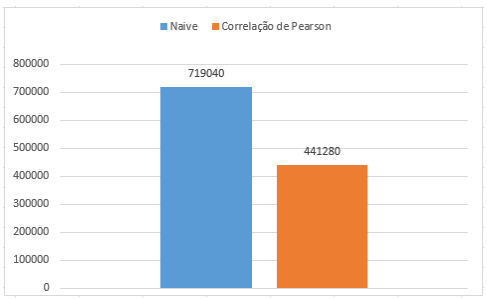

Figura 6: Comparação do tamanho das mensagens em bytes após a fase de *flooding*

no modo *naive* e apenas as mensagens estatisticamente distintas através da correlação de Pearson. Para esta abordagem, não foi necessário utilizar a correlação de Pearson após a fase inicial dado que novas mensagens seriam enviadas apenas na ocorrência de novidades.

IV. Conclusão

Este trabalho apresentou duas abordagens para um problema amplamente discutido no âmbito das RSSF, o consumo energético dos dispositivos. em busca deste objetivo foram apresentadas as aplicações de métodos estatísticos como parte de um algoritmo que busca otimizar o envio de mensagens em uma RSSF de um sensor para o *sink*. Houve uma redução significativa tanto na quantidade de mensagens, através do agrupamento por medida de similaridade, quanto no tamanho dos pacotes de leituras de dados, através da correlação de Pearson. E por consequência, um aumento na vida útil do sensor. Este algoritmo pode ser aplicado à sensores que realizam a leitura de um ou muitos parâmetros simultâneamente, sendo possível variar a sensibilidade da correlação e do agrupamento para aproveitamento em diferentes ambientes que necessitam de monitoramento e redução no consumo de energia.

Referências

[1] C. Carvalho, D. G. Gomes, N. Agoulmine, and J. N. de Souza, "Improving prediction accuracy for WSN data reduction by applying multivariate spatio-temporal correlation," *Sensors*, vol. 11, no. 11, pp. 10010–10037, 2011.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393 – 422, 2002.

[3] A. a.F. Loureiro, J. M. S. Nogueira, L. B. Ruiz, R. A. D. F. Mini, E. F. Nakamura, and C. M. S. Figueiredo, "Redes de Sensores Sem Fio," *XXI Simpósio Brasileiro de Redes de Computadores*, pp. 179–226, 2003.

[4] J. Cohen, "{CHAPTER} 3 - the significance of a product moment rs," in *Statistical Power Analysis for the Behavioral Sciences (Revised Edition)* (J. Cohen, ed.), pp. 75 – 107, Academic Press, revised edition ed., 1977.

[5] F. R. Almeida, A. Brayner, J. J. P. C. Rodrigues, and J. E. B. Maia, "Fractal Clustering and Similarity Measure : Two New Approaches for Reducing Energy Consumption in Wireless Sensor Networks," *Icufn2016*, pp. 288–293, 2016.

[6] T. Appelhans, E. Mwangomo, I. Otte, F. Detsch, T. Nauss, and A. Hemp, "Eco-meteorological characteristics of the southern slopes of Kilimanjaro, Tanzania," *International Journal of Climatology*, vol. 36, no. 9, pp. 3245–3258, 2016.

[7] F. R. Almeida, A. Brayner, J. Rodrigues, and J. E. U. Maia, "Improving Multidimensional Wireless Sensor Network Lifetime Using Pearson Correlation and Fractal Clustering," *Sensors (Basel, Switzerland)*, vol. 17, no. June, p. 1317, 2017.

[8] F. R. Almeida, "Cluster wsn." https://github.com/fernandoraj/Cluster_WSN, 2016.

[9] H. Fischer, *A History of the Central Limit Theorem: From Classical to Modern Probability Theory*. Sources and Studies in the History of Mathematics and Physical Sciences, Springer New York, 2010.