

Evaluation of an RSSI-based Location Algorithm for Wireless Sensor Networks

G. Gracioli, A. A. Fröhlich, *Member IEEE*, R. P. Pires and L. Wanner

Abstract— Position awareness is a desirable feature for many applications of Wireless Sensor Networks. The Received Signal Strength Indication (RSSI) of a radio channel provides a feasible way of estimating distance between nodes because its use does not require any additional hardware but only a radio transceiver. The main drawback of using RSSI is its instability and interference susceptibility noticed in real environments. This work presents an evaluation of the HECOPS implementation, a location algorithm for wireless sensor networks that uses RSSI to determine the distance among nodes in an efficient way. Based on the analysis of the HECOPS behavior in practice, this paper proposes to change the way that the distance among nodes are estimated. The proposal consists in using the mean of the last n estimatives, instead of using only the RSSI value in the moment that the information are received, as in the original algorithm. The results show that, with this change, HECOPS has obtained an improvement of up to 100% in terms of reliability in the nodes position estimation.

Keywords— Wireless Sensor Networks, Location Algorithms, Embedded Systems, Ad Hoc Networks

I. INTRODUÇÃO

UMA Rede de Sensores Sem Fio (RSSF) consiste em uma grande quantidade de sensores autônomos espalhados em um determinado ambiente. Com os recentes avanços tecnológicos nas áreas de comunicação sem fio e eletrônica, tornou-se possível o desenvolvimento de pequenos sensores de baixo custo e de baixa potência, dotados de comunicação sem fio com pequeno alcance e alimentados por bateria [13]. Devido a essas características, RSSFs são utilizadas em uma ampla variedade de aplicações, desde monitoramento de ambientes até controle de casas inteligentes.

A localização dos nodos em uma RSSF é uma funcionalidade extremamente importante. Em aplicações como rastreamento de objetos, monitoramento do meio-ambiente, rastreamento de inimigos em campo de batalha e sensoriamento para previsão de tempo, as informações coletadas pelos sensores apenas possuem sentido se atreladas a uma posição geográfica.

Neste contexto, os algoritmos de localização para RSSFs devem possuir determinadas características para atender aos requisitos específicos dessas redes (e.g. baixo custo, pequeno alcance e recursos limitados). Entre as principais características estão:

- **Independência de infra-estrutura:** os sistemas de localização devem ser auto contidos e de baixo custo, ou seja, não depender de nenhuma infra-estrutura fixa, como por exemplo, o GPS (*Global Positioning System*);
- **Robustez:** devem tolerar falhas nos nodos sensores e erros nas estimações de distância;
- **Economia de recursos:** devem economizar recursos, como processamento e comunicação, para aumentar o tempo de vida dos sensores.

Um algoritmo de localização especificamente projetado para atender os requisitos de uma RSSF é o HECOPS (*Heuristic Environmental Consideration Over Positioning System*) [1]. O conceito básico do HECOPS é utilizar a leitura da intensidade do sinal do rádio (RSSI - *Received Signal Strength Indication*) dos sensores de maneira eficiente e assim estimar a distância entre os nodos. Através de trilateração, o HECOPS utiliza nodos mais confiáveis como base para os cálculos das posições dos nodos da rede e também para corrigir as estimativas de distância entre eles.

O HECOPS foi validado através de extensivos testes e avaliações em simulação [1]. Porém, sua implementação em uma rede de sensores real ainda não havia sido realizada. Este trabalho apresenta uma avaliação da implementação do HECOPS para o sistema operacional EPOS [2] utilizando nodos sensores reais. Com base na análise do comportamento dos nodos nesta implementação, notou-se que a acurácia das estimativas das distâncias era afetada pela instabilidade nas leituras do RSSI, devido à reflexão, difração e degradação do sinal de rádio [3].

Neste sentido, este artigo propõe a alteração na forma de como estimar as distâncias entre nodos no HECOPS. Notamos que uma alteração na forma com que as correções das distâncias são realizadas melhora consideravelmente a acurácia das estimativas de posição e viabiliza sua utilização em situações reais. A proposta consiste em usar a média das últimas n estimativas, ao contrário de utilizar apenas a leitura do RSSI no momento da recepção das informações como no algoritmo original. Esta mudança melhora em cerca de 100% a confiabilidade na estimativa das posições dos nodos na rede.

O restante deste artigo está organizado da seguinte maneira. A próxima seção apresenta uma descrição do HECOPS. Na seção III, a infra-estrutura utilizada é descrita juntamente com as considerações e adaptações do algoritmo. A seção IV apresenta uma avaliação dos resultados obtidos. Finalmente, a seção V conclui o artigo.

R. P. Pires, Universidade Federal de Santa Catarina (UFSC), Santa Catarina, Brasil, rafaelp@lisha.ufsc.br

G. Gracioli, Universidade Federal de Santa Catarina (UFSC), Santa Catarina, Brasil, giovani@lisha.ufsc.br

L. Wanner, Universidade da Califórnia (UCLA), Los Angeles, Estados Unidos da América, wanner@ucla.edu

A. A. Fröhlich, Universidade Federal de Santa Catarina (UFSC), Florianópolis, Santa Catarina, Brasil, gutto@lisha.ufsc.br

II. O ALGORITMO DE LOCALIZAÇÃO HECOPS

O objetivo do HECOPS é determinar as coordenadas de todos os nodos sensores da rede via comunicação nodo a nodo, considerando uma RSSF formada por um conjunto de nodos com coordenadas desconhecidas e equipados com um mecanismo que possa estimar suas distâncias (i.e. rádio) a alguns nodos vizinhos e também por um conjunto de poucos nodos que possuem o conhecimento *a priori* das suas posições (chamados de nodos âncoras) [1]. Um nodo âncora tem ciência da sua localização baseando-se em algum sistema de coordenadas global. Este sistema pode ser um GPS, pré-estabelecido no código fonte, ou ainda através de processos automáticos de estabelecimento de sistemas de coordenadas executados antes do serviço de localização [4]. Com base nas coordenadas dos nodos âncoras, todos os outros nodos da rede estimam suas posições através de comunicação nodo a nodo.

Um fator importante em um algoritmo distribuído de localização é sua capacidade de medir distâncias entre pares de nodos diretamente conectados. Existem técnicas baseadas no tempo de propagação de mensagens como ToA (*Time of Arrival*) e TDoA (*Time Difference of Arrival*), mas necessitam de dispositivos com temporizadores de alta resolução. Outras técnicas, como AoA (*Angle of Arrival*), dispositivos ópticos e ultrassom também são dependentes de hardware adicional. Pela independência de quaisquer outros dispositivos além de um transmissor/receptor de rádio, já necessário à comunicação sem fio, o HECOPS utiliza o RSSI para estimar as distâncias. A distância estimada é inversamente proporcional à intensidade do sinal recebido.

O principal problema da utilização do RSSI é que sua intensidade não varia da mesma maneira em todas as direções, sendo facilmente afetada por interferências eletromagnéticas e barreiras físicas. Essas características introduzem bastante imprecisão na forma de estimar a distância, sendo que o algoritmo de localização deve ser capaz de tratá-la de maneira eficiente. A Fig. 1 mostra o contorno de probabilidade na recepção de um pacote em função da distância a um nodo emissor x .

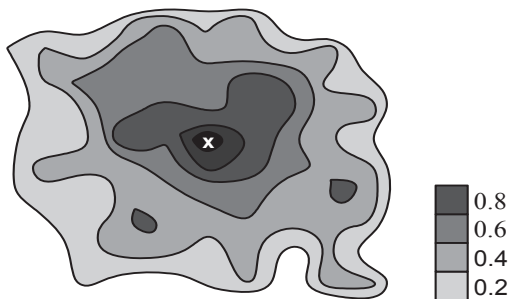


Figura 1. Contorno de probabilidade na recepção de um pacote por um nodo central [1].

Apesar da instabilidade do RSSI, nota-se que sua variação está diretamente relacionada à direção do sinal, ou seja, nodos que estejam na mesma direção farão leituras similares do RSSI, com uma variação quase linear em relação

à distância. A Fig. 2 exemplifica o conceito de direção. Os nodos X e B estão na mesma direção do nodo A, cercados pelo triângulo pontilhado. Por outro lado, o nodo D não está na mesma direção que X e B.

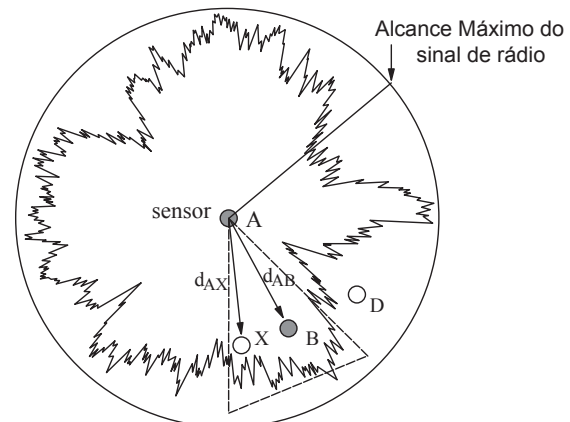


Figura 2. Padrão irregular do sinal de rádio e sistema de calibragem baseado em direção [1].

Com base na variação quase linear do RSSI em relação à direção, o HECOPS estima as distâncias de um nodo que ainda desconhece sua posição através da comunicação com nodos que estejam na mesma direção que a sua. É usado um fator de correção, denominado desvio, que é definido pelo produto entre a distância real, obtida entre nodos que conhecem suas posições, e a leitura do RSSI desses mesmos nodos. Na Figura 2 por exemplo, os nodos B e X, por estarem na mesma direção em relação ao nodo A, sofrem o mesmo desvio. Considerando que os nodos A e B sejam âncoras, o desvio seria dado por:

$$dev_{AB} = d_{AB} \times RSSI_{AB} \quad (1)$$

onde:

- dev_{AB} é o desvio;
- d_{AB} a distância euclidiana entre A e B; e
- $RSSI_{AB}$ a leitura do RSSI feita pelo nodo B de uma mensagem enviada por A.

Sendo assim, a distância entre A e X seria dada por $d_{AX} = dev_{AB} / RSSI_{AX}$. O nodo D não seria afetado por dev_{AB} por não estar na mesma direção.

Espera-se que a quantidade de nodos âncoras necessários, sem que se comprometa a precisão dos resultados, seja a mínima possível. Para tanto, o HECOPS não exige que as 3 distâncias necessárias para o cálculo da posição sejam em relação a âncoras, afinal, tal medida demandaria um grande número deles, espalhados uniformemente pela rede. Desta forma, nodos que estimaram suas posições também podem ser escolhidos como *pontos de referência*.

Para que isso não prejudique de forma considerável o desempenho do algoritmo, um método baseado em heurísticas que quantizam a confiabilidade das informações de localização providas pelos nodos foi desenvolvido para

escolher os melhores pontos de referência. Cada nodo, ao calcular sua posição, define um valor de *confiança* no resultado obtido. Esse valor define a ordem de escolha dos pontos de referência por um nodo que deseja estimar sua posição.

O cálculo da confiança é baseado nos valores de confiança dos nodos escolhidos como pontos de referência e nos valores de confiança dos nodos utilizados para calibrar a distância em relação àqueles pontos. Numa escala de 0 a 1,0, nodos âncoras têm confiança máxima na sua posição, igual a 1,0. Os demais têm confiança, de no máximo 0,8, dada por:

$$c_x = \frac{0,8 \times \sum_{i=1}^n (C_i \times 0,75 + C_{ix} \times 0,25)}{n} \quad (2)$$

onde:

- C_x é a confiança na posição que está sendo calculada por um nodo X;
- C_i a confiança de cada ponto de referência escolhido por X, sendo um total de n nodos;
- C_{ix} a confiança do nodo que, juntamente com o nodo i , definiu o *desvio* aplicado na distância entre o nodo i e X, se houver. Por exemplo, na Fig. 2, C_{ix} seria a confiança do nodo B, considerando como i o nodo A.

Um sensor não âncora nunca atinge confiança máxima de 1, não importa quão confiável sejam as suas informações disponíveis. Isso é devido ao fato de somente os sensores âncoras conhecerem exatamente suas posições, sendo que os outros nodos sempre terão valores estimados. A medição de alcance até um nodo âncora é uma informação confiável. Porém, a confiabilidade pode ser aumentada se esta medição for calibrada pelo desvio (*dev*), como explicado na Equação (1) [1]. As constantes 0,75 e 0,25 asseguram que o valor calculado para a confiança dependa mais da confiança dos sensores vizinhos, como esperado. A primeira constante se refere à confiança do ponto de referência, e a segunda ao nodo com o qual foi feita a calibragem da distância à esse ponto de referência [1].

Assim que um nodo que deseja estimar sua posição escolher seus pontos de referência e possuir distâncias estimadas a todos eles, basta aplicar algum método de cálculo da posição, como lateração [5] ou min-max [6].

III. LOCALIZAÇÃO NO EPOS

A fim de avaliar a viabilidade e os resultados da implantação do HECOPS em sensores reais, implementamos o algoritmo a nível de aplicação no sistema operacional EPOS (*Embedded Parallel Operating System*) [2], por ter uma infra-estrutura de comunicação consolidada [7], [8] e uma camada MAC (*Medium Access Control*) que fornece a leitura do RSSI para as mensagens recebidas. Os protótipos alvo da avaliação foram os sensores Mica2 [9], que possuem um microcontrolador Atmel ATmega128 com palavra de 8 bits (arquitetura AVR8) e frequência de 7.37 Mhz.

Por ser uma arquitetura de 8 bits, o AVR8 não possui uma unidade de ponto flutuante. Portanto, todos os cálculos que

utilizam ponto flutuante devem ser realizados via *software*. Existem bibliotecas implementadas para esse fim, porém elas são grandes e têm custo computacional considerável. Para contornar o problema, decidi-se utilizar aritmética inteira. No entanto, para que não se perdesse muita precisão, os valores utilizados nos cálculos foram aumentados em algumas ordens de magnitude, de forma que, depois de processados, ainda tivessem precisão suficiente para não comprometer os resultados.

Outro ponto importante no HECOPS é o cálculo das distâncias entre nodos que também necessitaria de bibliotecas matemáticas para a sua resolução, devido à raiz quadrada (Equação (3)). Entretanto, para evitar o uso dessas bibliotecas, utilizou-se o método numérico iterativo de Newton-Raphson para encontrar raízes de funções baseadas nas suas derivadas. Pelo comportamento da função quadrática $f(x) = x^2 - n$, $n \in \mathfrak{R}$ e $n > 0$, a aplicação desse método tem convergência rápida (até 20 iterações para obter-se uma precisão de, no mínimo, uma unidade). A raiz positiva da função é o valor da raiz quadrada de n .

Inicialmente no HECOPS, somente os nodos âncoras conhecem suas posições. Eles começam o processo de localização enviando mensagens a todos os nodos ao seu alcance (*broadcast*). Uma mensagem contém seu identificador (ID), posição (x,y) e confiança (conforme a Figura 3(a)). Os nodos que receberem essa mensagem guardam as informações juntamente com a medida do RSSI lida no seu recebimento. Se o nodo que recebe a mensagem já possui ciência da sua posição, ele calcula a sua distância e o seu desvio em relação ao nodo emissor e após isso, envia essas informações a todos os nodos dentro do seu alcance (conforme a Figura 3(b)), que também são recebidas e armazenadas pelos nodos que desejam estimar suas posições.

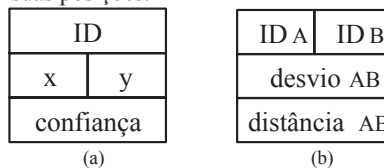


Figura 3. Conteúdo das mensagens trocadas: (a) Mensagem com informação de posição. (b) Mensagem com informação de desvio.

Quando um nodo que deseja estimar sua posição recebe uma mensagem com informação de desvio, ele verifica se está na mesma direção que o emissor em relação ao terceiro nodo descrito no conteúdo da mensagem. Caso estiver, ele calibra a leitura do RSSI de uma mensagem enviada por esse terceiro nodo com o desvio.

A verificação de um nodo para descobrir se ele se encontra na mesma direção que outro em relação a um nodo emissor é feita com base na proximidade entre esses dois nodos. Na Figura 4, o nodo C recebe uma mensagem de B informando o desvio entre A e B. Assim, o nodo C verifica se a sua distância ao nodo B é menor do que a metade da distância até A. Em caso positivo, o nodo C calibra a medição RSSI da última mensagem recebida de A com o desvio entre A e B.

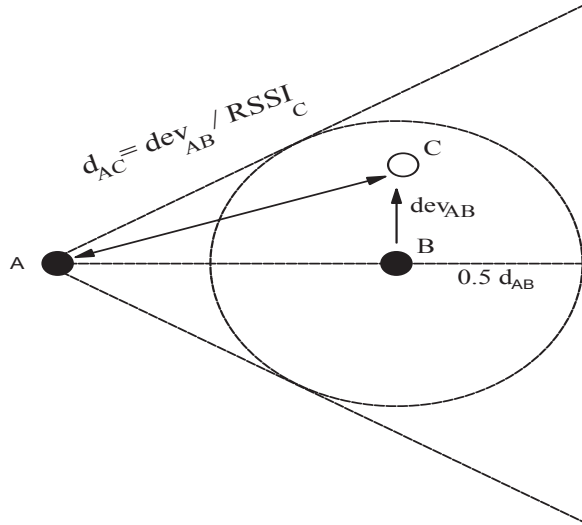


Figura 4. Definição dos nós que estão na mesma direção, a fim de utilizar a calibragem.

As mensagens com informação de posição são mantidas em uma lista ordenada pelo valor de confiança nos nós que devem estimar suas coordenadas. Assim que a lista alcançar tamanho 3, é possível fazer o cálculo da posição. Os três nós da lista que tiverem maior confiança em sua posição são escolhidos como pontos de referência.

O método utilizado para o cálculo das coordenadas é o *min-max*, por ter menor complexidade computacional que a lateração (que exigiria a resolução de sistemas de equações lineares) [10]. Nele, é definida uma área onde provavelmente está localizado o nó que estima sua posição. Assim que a área é definida, ela é dividida em um certo número de pontos e, então, calcula-se o resíduo para cada um desses pontos, com base na Fórmula (3):

$$Resíduo = \sum_{j=1}^n \left(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} - d_j \right) \quad (3)$$

onde:

- (x_i, y_i) corresponde às coordenadas de cada ponto dentro da área definida;
- (x_j, y_j) são as coordenadas de cada âncora (n no total); e
- d_j a distância estimada do nó que está calculando sua posição e o âncora j .

A posição do nó é então definida como sendo o ponto que apresentar o menor valor de resíduo.

A Fig. 5 mostra como é definida a área onde estarão os pontos que serão usados no cálculo do resíduo. A área é retangular e definida pela intersecção dos intervalos $\cap_{j=1}^n [x_j - d_j, x_j + d_j]$ e $\cap_{j=1}^n [y_j - d_j, y_j + d_j]$ nos eixos x e y , respectivamente. A área é então dividida em um número parametrizável de pontos, para que se possa adequar à carga computacional suportada pelo dispositivo. Finalmente, quando é definida a posição do nó, ele envia uma mensagem como a da Fig. 3(a) a todos ao seu alcance, de forma que ele possa servir como ponto de referência para outros nós.

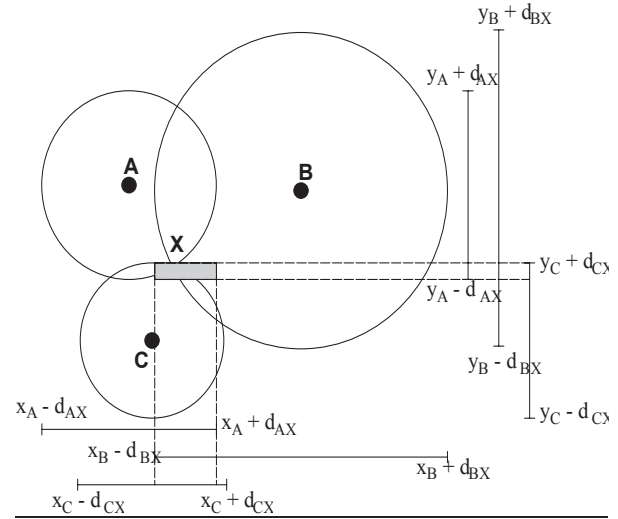


Figura 5. Método de cálculo da posição Min-Max.

A Fig. 6 ilustra o particionamento da área e os pontos dos quais seriam calculados os resíduos. A Equação (4) mostra como são calculados os parâmetros que definem o particionamento. O laço que calcula os resíduos tem N iterações, nx colunas e ny linhas ($nx \times ny = N$).

$$nx = \sqrt{\frac{(N \times largura)}{altura}} \quad ny = \sqrt{\frac{(N \times altura)}{largura}} \quad (4)$$

$$dx = \frac{largura}{nx} \quad dy = \frac{altura}{ny}$$

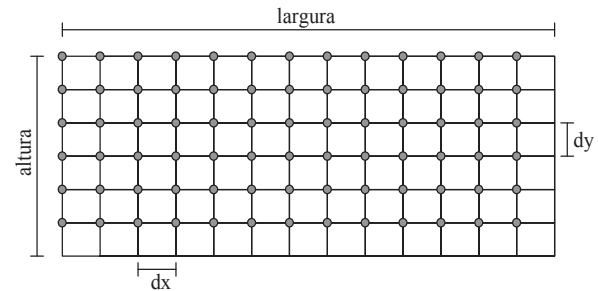


Figura 6. Particionamento da área baseado no número de pontos a serem testados.

IV. AVALIAÇÃO

Já que a instabilidade do RSSI é bastante alta, decidiu-se coletar dados em campo e executar o algoritmo em estações de trabalho, para que se pudesse avaliar, sobre os mesmos dados, seu comportamento sob diferentes aspectos variando alguns detalhes de implementação. Para tanto, foi necessário desenvolver um *wrapper*, onde o código desenvolvido para o sistema operacional EPOS pudesse ser executado da mesma forma que ocorreria nos dispositivos móveis. Através dele, o código que executaria em cada sensor é atrelado a uma *thread*, e o envio e recebimento de mensagens que seriam trocadas pelo rádio são feitas na memória. Além disso, uma aplicação gráfica que mostra a disposição geográfica dos nós também foi desenvolvida, para que se pudesse observar o comportamento do algoritmo visualmente.

Para a execução do algoritmo no cenário descrito acima, foram coletadas as medidas RSSI de todas as combinações entre pares de nodos dispostos numa grade 3×3 , com um espaço de $5m$ entre eles. Essas medições serviram como dados de entrada para a implementação nas estações de trabalho. Foram escolhidos como âncoras 3 dos 9 nodos possíveis e, como nodos que estimam suas posições, outros 3. A Fig. 7 ilustra a disposição geográfica dos nodos após a estabilização das localizações. As setas indicam a distância entre a posição estimada dos nodos A, B e C e a posição real em que eles se encontram. Os círculos hachurados mostram a posição dos âncoras. Nota-se, pela disposição dos nodos, que a posição relativa entre eles foi mantida parecida com a original, o que encoraja sua utilização em aplicações que possuem como requisito a localização dos sensores, como roteamento geográfico [11] ou consultas de aquisição de dados [12].

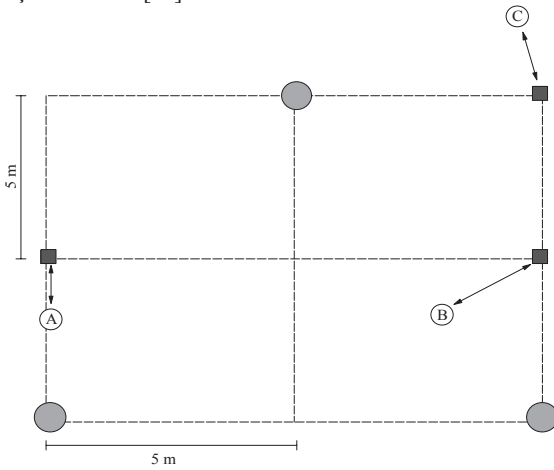


Figura 7. Disposição geográfica dos nodos após um tempo de execução do algoritmo.

A Fig. 8 mostra o gráfico com a média entre as distâncias de cada nodo que estima sua posição e o ponto real em que se encontram, em função do número de iterações, ou seja, o número de vezes que cada nodo fez o cálculo da sua posição. A curva da variação do RSSI mostra o comportamento do algoritmo em face às variações altamente instáveis do RSSI, fazendo com que essas flutuações impossibilitem a estabilização das posições estimadas com o tempo. Entretanto, na curva da média das estimativas, uma média histórica do fator de correção das medidas do RSSI foi mantida pelos nodos, fazendo com que instabilidades momentâneas da leitura do RSSI não afetassem a estabilidade do resultado. As outras duas curvas mostram a convergência do erro quadrático médio de estimação, medido pela seguinte equação:

$$\sqrt{\frac{\sum_{i=1}^n (X_e - X_r)^2}{n}} \quad (5)$$

onde:

- X_e é a posição estimada do ponto i ;
- X_r é a posição real do ponto i ; e
- n é o número de pontos i usados no somatório.

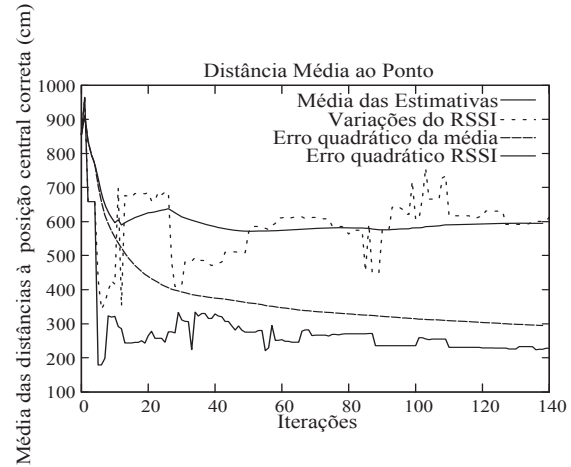


Figura 8. Gráfico da distância média ao ponto correto de acordo com o número de iterações.

Percebe-se que existe uma convergência gradual da média das estimativas, enquanto que o erro quadrático das variações do RSSI não é linear. Esta análise comprova que a estimativa usando as médias é mais eficiente que apenas o cálculo do RSSI utilizando uma medida. Por exemplo, o erro quadrático médio da médias das estimativas com 20 iterações é cerca de 50% do valor do erro quadrático das variações do RSSI. Considerando o valor absoluto, com 20 iterações, houve uma melhoria de 100% no erro de estimação.

Considerando uma RSSF composta por nodos móveis, onde as variações do RSSI têm maior influência na estimativa das distâncias dos nodos, a introdução da média das estimativas pode ter como efeito colateral a invariabilidade dos resultados com o passar do tempo. No entanto, uma alternativa seria manter também uma média das últimas n iterações e, quando essa média se distanciasse consideravelmente da que estivesse sendo utilizada, substituiria-se a primeira por esta. Pelo gráfico, nota-se que logo nas primeiras iterações as posições se estabilizam e, portanto, caso os nodos fossem móveis, a convergência para posições com menores erros seria rápida.

O número de pontos de referência utilizados na avaliação foi fixado em 3, sem variação, devido a escassez de recursos da plataforma utilizada. A variação no número de pontos de referência não apresenta grande influência no erro de estimação, mantendo-se quase que constante, conforme mensurado e avaliado no HECOPS original [1].

V. CONSIDERAÇÕES FINAIS

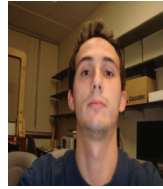
Este trabalho descreveu a implementação e avaliou o algoritmo de localização HECOPS para redes de sensores sem fio sobre a infra-estrutura oferecida pelo sistema operacional EPOS. Observou-se que, mesmo com as restrições impostas pelos sensores de baixo custo e baixo poder computacional, é possível usar o HECOPS em dispositivos reais desde que a forma de estimar as distâncias entre os nodos seja modificada para utilizar a média das últimas n estimativas. Ao usar essa média, o algoritmo obteve uma melhora de 100% na

confiabilidade e precisão da estimativa das posições dos nodos sensores da rede.

Embora a utilização do RSSI como medida de estimar a distância entre nodos em uma rede sem fio tem sido desencorajada pela sua alta instabilidade e suscetibilidade a interferências, pôde-se observar que sua utilização pode apresentar resultados aceitáveis em algoritmos de localização. Estes resultados têm como consequência a possibilidade da implantação de algoritmos de localização em redes sem fio ad hoc mesmo que os dispositivos não possuam hardware adicional dedicado para esse fim. Como trabalho futuro, pretende-se analisar o custo de comunicação do algoritmo em termos de consumo de energia em nodos sensores reais.

REFERÊNCIAS

- [1] R. Reghelin and A. A. Fröhlich, "A decentralized location system for sensor networks using cooperative calibration and heuristics," in MSWiM '06: Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems. Terromolinos, Spain. ACM Press, 2006, pp. 139-146.
- [2] A. A. Fröhlich, "Application-Oriented Operating Systems", ser. GMD Research Series. Sankt Augustin, Germany: GMD - Forschungszentrum Informationstechnik, Aug. 2001, no. 17.
- [3] J. Ma, Q. Chen, D. Zang, and L. M. Ni, "An Empirical Study of Radio Signal Strength in Sensor Networks Using MICA2 Nodes", Department of Computer Science and Engineering - Hong Kong University of Science and Technology. Technical Report, Mar. 2006.
- [4] H. Wu, C. Wang, and N.-F. Tzeng, "Novel self-configurable positioning technique for multihop wireless networks," IEEE/ACM Transactions on Networking, vol. 13, no. 3, pp. 609-621, 2005.
- [5] K. Langendoen and N. Reijers, "Distributed localization in wireless sensor networks: a quantitative comparison," Comput. Networks, vol. 43, no. 4, pp. 499-518, 2003.
- [6] A. Savvides, H. Park, and M. B. Srivastava, "The bits and flops of the n-hop multilateration primitive for node localization problems," in Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA-02). New York: ACM Press, Sep. 28 2002, pp. 112-121.
- [7] L. Wanner, A. Junior, F. Polpeta, and A. Fröhlich, "Operating system support for handling heterogeneity in wireless sensor networks," in Proceedings of 10th IEEE Conference on Emerging Technologies and Factory Automation, vol. 2, Sep. 2005.
- [8] L. Wanner, A. Junior, A. B. de Oliveira, and A. Fröhlich, "Operating system support for data acquisition in wireless sensor networks," in Proceedings of 11th IEEE Conference on Emerging Technologies and Factory Automation, Sep. 2006, pp. 582-585.
- [9] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, "System architecture directions for networked sensors," in Proceedings of the ninth international conference on Architectural support for programming languages and operating systems, Cambridge, Massachusetts, United States, 2000, pp. 93-104.
- [10] A. Savvides, C.-C. Han, and M. B. Srivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," in MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking. New York, NY, USA: ACM, 2001, pp. 166-179.
- [11] I. Stojmenovic, "Position-based routing in ad hoc networks," IEEE Communications Magazine, vol. 40, pp. 128-134, Jul. 2002.
- [12] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "The design of an acquisitional query processor for sensor networks," in SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data. New York, NY, USA: ACM Press, 2003, pp. 491-502.
- [13] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques", Computer Networks: The International Journal of Computer and Telecommunications Networking, v.51 n.10, Jul. 2007, pp. 2529-2553.



Giovani Gracioli graduou-se Bacharel em Ciência da Computação pela Universidade Federal de Santa Maria em 2007 e Mestre em Ciência da Computação pela Universidade Federal de Santa Catarina em 2009. Atualmente é doutorando em Engenharia de Automação e Sistemas na Universidade Federal de Santa Catarina e pesquisador do Laboratório de Integração em software e hardware.



Antônio Augusto Fröhlich graduou-se Bacharel em Ciência da Computação pela Universidade Federal do Rio Grande do Sul em 1989, Mestre pela Universidade Federal de Santa Catarina em 1994, e Doutor pela Technische Universität Berlin, Alemanha, em 2001. Desde 1995 é Professor do Departamento de Informática e Estatística da Universidade Federal de Santa Catarina, e desde 2001 é coordenador do Laboratório de Integração em Software e Hardware. É autor de um livro e de uma série de artigos na área de sistemas operacionais e sistemas embarcados, áreas nas quais coordena e executa uma série de projetos de pesquisa e desenvolvimento.



Rafael Pereira Pires graduou-se Bacharel em Ciência da Computação pela Universidade Federal de Santa Maria em 2006 e Mestre em Ciência da Computação pela Universidade Federal de Santa Catarina em 2009.



Lucas Francisco Wanner graduou-se Bacharel e Mestre em Ciência da Computação pela Universidade Federal de Santa Catarina em 2004 e 2007. Atualmente é pesquisador do Laboratório de Sistemas Embarcados e Redes da Universidade da Califórnia em Los Angeles.