

Otimização em Redes de Sensores utilizando Evolução Diferencial para determinação de *Clusters-heads*

Patrick Lazzarotto, Myriam Delgado e Anelise Munaretto

Resumo– Este trabalho aborda o problema de formação de *clusters* em redes de sensores sem fio de forma a encontrar, para uma determinada rede, uma configuração que minimize o gasto de energia. Esta configuração, baseada em *clustering*, é determinada de forma remota, utilizando-se um algoritmo de Evolução Diferencial para a obtenção dos *cluster-heads*. Dois algoritmos para associar os nós livres aos respectivos *cluster-heads* são avaliados. Testes são realizados comparando a metodologia proposta com o algoritmo LEACH.

Palavras-chave – *Redes de sensores sem fio, Evolução Diferencial, Clustering.*

Abstract– This paper addresses the problem of clustering in wireless sensor networks in order to obtain, depending on the network topology, a specific configuration that minimizes the total of wasted energy. Such configuration, which is based on clustering, is remotely determined by means of a Differential Evolution algorithm used to provide the set of cluster-heads. Two algorithms to associate the free nodes with the respective cluster-heads are evaluated. Tests are carried out comparing the proposed methods with the LEACH algorithm.

Keywords – *Wireless Sensor Networks, Differential Evolution, Clustering.*

I. INTRODUÇÃO

Redes de Sensores sem Fio (RSSF) são sistemas distribuídos que apresentam, devido às suas peculiaridades, vários desafios em muitas áreas de estudo. Uma RSSF é composta por nós sensores, dispositivos autônomos que cooperam entre si de uma maneira *ad hoc*, com o objetivo de realizar o sensoriamento de uma região, seja esta um ecossistema, uma região de guerra ou mesmo o estoque de uma fábrica. A existência destes pequenos dispositivos se deu graças aos avanços atuais nas áreas de micro-processamento, de materiais de sensoriamento e de sistemas micro-mecânicos. O tamanho reduzido e o baixo custo dos nós sensores em relação aos sensores fixos possibilitam redes densas que por isso podem fornecer resultados com maior grau de precisão [Akyildiz 2002].

Uma RSSF possui características únicas se comparadas com outras redes. A principal dessas características é a restrição de energia, dado que um nó sensor é alimentado apenas pela energia finita de uma bateria [Akyildiz 2002]. Assim, um método de roteamento eficiente é fundamental para o bom funcionamento da rede evitando assim um gasto excessivo de energia.

Na literatura, vários algoritmos foram propostos. Em [Lindsey 2002] e [Huseyin 2003], são apresentados os algoritmos PEGASIS e PEDAP, os quais organizam os nós da topologia em formações de corrente (todos os nós ligados a apenas outros dois, com exceção dos nós das pontas) e árvore, respectivamente. Tais topologias têm o objetivo de diminuir o número de recepções, visto que cada nó estará ligado a poucos vizinhos. Porém, a aplicação de tais algoritmos é limitada a redes em que se considera total redundância dos dados, visto que em cada roteamento os dados são agrupados. Sem esse agrupamento, o desempenho de tais algoritmos é fraco, pois o acúmulo de dados sobrecarrega os nós sensores localizados ao longo da corrente (no caso do algoritmo PEGASIS) ou da árvore (no caso do algoritmo PEDAP).

Outros algoritmos propostos utilizam o método de *clustering* [Lewis 2004]. Esse método consiste em escolher alguns nós da rede (*cluster-heads*) e atribuir aos mesmos a tarefa de transmitir dados para um ponto de coleta. Alguns exemplos são: o algoritmo LEACH [Heinzelman 2000], a sua variante LEACH-C [Heinzelman 2002] e o algoritmo BCDCP [Murugunathan 2005]. O algoritmo LEACH utiliza um conceito no qual os nós se organizam sozinhos, sem auxílio de um controle externo. Tal algoritmo será objeto de comparação com o algoritmo proposto neste artigo. Sua variante, LEACH-C, utiliza um algoritmo de busca local (*Simulated Annealing*) para determinação dos clusters, mas assim como o LEACH, não aborda a questão de equalização da carga entre os *Cluster-head's* (associar nós com um *cluster-head* de acordo com sua energia, questão que é abordada neste artigo). O algoritmo BCDCP (Base-Station Controlled Dynamic Clustering Protocol) [Murugunathan 2005] utiliza um procedimento para balancear o número de nós em cada *cluster* (através da divisão sucessiva de *clusters*), mas também não considera a questão de equalização de acordo com a energia do *cluster-heads*. Além disso, não considera o gasto de energia dos nós comuns como critério para escolha dos *cluster-heads* (o critério adotado é que os recém-formados *clusters* devem ter a máxima separação entre seus *cluster-heads*).

Neste trabalho, a técnica de *clustering* será utilizada. O problema a ser resolvido é a definição dos *cluster-head's*. Para resolvermos tal problema, iremos fazer uso de uma técnica bio-inspirada (*Evolução Diferencial*). Algoritmos bio-inspirados vêm sendo aplicados em vários trabalhos na área de redes de sensores, com destaque para aqueles baseados em Sistemas Imunológicos Artificiais,

Inteligência Coletiva (*Swarm Intelligence*) e Algoritmos Evolucionários [Falko 2006]. Neste trabalho, será utilizado o algoritmo de Evolução Diferencial (ED), o qual é um algoritmo bio-inspirado baseado em evolução de populações, desenvolvido em meados da década de 90 por Price e Storn [Price 1995]. Além de possuir boa propriedade de convergência e possibilidade de paralelização, a ED tem como principais vantagens a simplicidade de conceitos e facilidade de uso [Price 1997]. Pode-se incluir todas as restrições do problema na formulação da função a ser otimizada, de forma que os resultados obtidos já satisfarão as restrições, sem necessidade de processamento posterior.

O restante do artigo está organizado da maneira descrita a seguir. A seção II apresenta os conceitos básicos do algoritmo de Evolução Diferencial utilizado neste trabalho. Na seção III, é apresentado o modelo de redes de sensores adotado, assim como a abordagem proposta para resolver o problema de otimização de energia. A seção IV traz a comparação dos resultados obtidos pela abordagem proposta e o algoritmo LEACH [Heinzelman 2000]. Na seção V, são feitas as considerações finais sobre a metodologia e os resultados alcançados, assim como são apresentadas as perspectivas futuras.

II. EVOLUÇÃO DIFERENCIAL

O algoritmo Evolução Diferencial (ED) foi criado por Storn e Price [Price 1995], e assim como outros algoritmos evolucionários (por exemplo, os Algoritmos Genéticos) trabalha com uma população de soluções, as quais são submetidas aos operadores definidos pelo método. Em cada iteração (ou geração) executa-se a seleção das soluções que irão fazer parte da próxima iteração. O processo é repetido até que um critério de parada seja alcançado. Cada indivíduo da população consiste em um vetor de D dimensões, cada componente contendo um valor no espaço contínuo. O objetivo é minimizar uma dada função objetivo para o problema em questão [Price 1995].

A. Operadores da Evolução Diferencial

Os operadores no algoritmo de Evolução Diferencial (assim como em outros algoritmos evolucionários) têm a função de buscar soluções diferenciadas das soluções da população atual, de forma a explorar todo o espaço de busca da função objetivo. O processo consiste em, para cada indivíduo da população atual (denominado *vetor alvo*), gerar um vetor que será usado para comparação (denominado *vetor experimental*). Os operadores de mutação e cruzamento atuam somente no vetor experimental.

Mutação: Em uma determinada geração q , vetores distintos (X_α , X_β e X_γ) são escolhidos aleatoriamente entre os membros da população atual. O par de vetores (X_β, X_γ) define uma diferença ($X_\beta - X_\gamma$), diferença esta que é multiplicada por um fator $F > 0$ e passa a ser denominada de diferença ponderada. (que pode ser X_α , outro vetor aleatório, ou X_{best} , o melhor vetor da população),

resultando então no vetor experimental $V^{(q+1)}$, como pode ser visto na equação abaixo:

$$V^{(q+1)} = X_\alpha^{(q)} + F(X_\beta^{(q)} - X_\gamma^{(q)})$$

ou

$$V^{(q+1)} = X_{best}^{(q)} + F(X_\beta^{(q)} - X_\gamma^{(q)}) \quad (1)$$

onde α, β, γ devem ser números inteiros, distintos entre si, e pertencer ao intervalo $\{1, \dots, Np\}$. O número de indivíduos da população, Np , deve ser maior ou igual a 4. O fator F deve ser um número real do intervalo $[0, 2]$ [Price 1995]. É possível ainda uma variação do operador de mutação, utilizando-se 2 pares de vetores ao formar a diferença ponderada [Price 1995].

Cruzamento: O operador de cruzamento tem a função de aumentar a diversidade dos indivíduos que sofreram mutação. Os componentes finais do vetor experimental $U^{(q+1)}$ serão formados pela seguinte expressão:

$$u_i^{(q+1)} = v_i^{(q+1)}, \text{ se } r_i \leq CR$$

$$u_i^{(q+1)} = x_i^{(q)}, \text{ se } r_i \geq CR, i=1, \dots, n \quad (2)$$

sendo r_i um número aleatório no intervalo $[0, 1]$. x_i as componentes do vetor alvo X_d e CR a probabilidade do cruzamento ocorrer. O valor de CR é compreendido no intervalo $[0, 1]$, e deve ser fornecido pelo usuário. O método de cruzamento pode ser binomial ou exponencial [Price 1997]. Se após o cruzamento alguma componente do vetor experimental ficar fora da região de busca (limites inferiores e superiores), a mesma deve ser corrigida, truncando-se os valores.

B. Processo de seleção

O processo de seleção consiste em determinar quais indivíduos farão parte da população na próxima geração. Cada indivíduo da população atual (vetor alvo X) tem seu custo comparado com um vetor experimental $U^{(q+1)}$ (fruto dos operadores de mutação e cruzamento). O vetor alvo só avança para a próxima geração se seu custo for menor que o custo do vetor experimental (caso contrário, o vetor experimental avança).

C. Estratégias da Evolução Diferencial

O número de indivíduos da população (NP), a constante que multiplica a diferença de vetores (F) e a probabilidade de cruzamento (CR) são parâmetros que devem ser fornecidos pelo usuário. Outros parâmetros não-numéricos podem ser modificados, e tais configurações constituem as chamadas *estratégias* na Evolução Diferencial. Nas principais estratégias apresentadas na literatura [Price 1995] e [Price 1997], os parâmetros a serem modificados são: vetor a ser perturbado no operador de mutação (X_{rand} ou X_{best}), número de diferenças usadas no operadores de mutação (comumente, 1 ou 2) e operador usado no cruzamento (binominal ou exponencial). A notação utilizada para as estratégias é ED/x/y/z, sendo x, y e z os parâmetros citados. Um resumo das principais estratégias é mostrado na tabela 1:

TABELA 1
Principais estratégias da Evolução Diferencial

Estratégia	Mutação	Estratégia	Mutação
1	best/1/exp	6	best/1/bin
2	rand/1/exp	7	rand/1/bin
3	best/1/exp	8	best/1/bin
4	rand/2/exp	9	rand/2/bin
5	rand-to-best/2/exp	10	rand-to-best/2/bin

A escolha da melhor estratégia a ser utilizada em um problema não é uma tarefa simples e geralmente é feita por tentativa e erro [Price 1996]. Isto porque uma estratégia que funciona bem para um determinado tipo de problema pode não funcionar bem para outro. O mesmo se aplica aos valores dos parâmetros NP , F e CR .

III. METODOLOGIA PROPOSTA

Esta seção apresenta o modelo de redes de sensores sem fio adotado neste trabalho, assim como a abordagem proposta para realizar o processo de otimização de gasto de energia, a qual é baseada em Evolução Diferencial (ED). Embora a ED tenha sido desenvolvida para problemas de otimização contínua, optou-se pelo uso desta técnica devido a sua simplicidade e facilidade de implementação, permitindo-se assim avaliar o desempenho da ED em problemas de otimização inteira.

A. O modelo de Redes de Sensores sem fio

Este trabalho tem como foco a camada de rede de uma rede de sensores sem fio. Para a camada de rede, é necessário usar métodos de roteamento dos dados de forma a otimizar o gasto de energia dos sensores. Neste trabalho será utilizado o método de *clustering* [Lewis 2004], método que consiste em escolher alguns nós da rede (que serão denominados *cluster-head* 's) e atribuir aos mesmos a tarefa de transmitir dados a um ponto de coleta. Os demais nós da rede devem então ser associados a um *cluster-head*, para o qual irão transmitir seus dados.

O problema de otimização tratado neste artigo consiste em encontrar, em um grupo de N sensores, quais são os M sensores que devem ser assinalados como *cluster-head* de forma a atender critérios previamente estabelecidos. Neste trabalho, foi definido que o principal objetivo a ser atingido é minimizar o total de energia gasto na rede. A Figura 1 apresenta um exemplo de uma rede de sensores simples, onde os pontos preenchidos são os *cluster-heads* e a área delimitada por pontilhados representa o *cluster* de cada *cluster-head*.

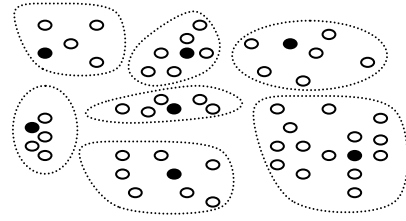


Figura 1. Exemplo de *clustering* em rede de sensores

Como os resultados da metodologia deste artigo serão comparados com o algoritmo de eleição de *cluster-heads* do protocolo LEACH, adota-se o modelo de rádio usado em [Heinzelman 2000], onde os gastos para transmissão e recepção de uma mensagem de k bits é dada por:

$$E_{T_x}(k, d) = E_{elec} * k + \epsilon_{amp} * k * d^2 \quad (3)$$

$$E_{R_x}(k, d) = E_{elec} * k \quad (4)$$

sendo $E_{elec} = 50\text{nJ/bit}$ a energia gasta pelo rádio para transmissão ou recepção, $\epsilon_{amp} = 100\text{ pJ/bit/m}^2$ a energia necessária para o amplificador de transmissão chegar ao nível aceitável E_b/N_0 , e d a distância entre transmissor e receptor.

Assume-se também que a estação base está a uma distância grande dos nós da rede, de forma que a distância de qualquer nó à estação base é praticamente a mesma. Como nos testes realizados o importante é a comparação entre os valores obtidos pelos métodos e não seu valor real, é possível desconsiderar a segunda parcela da equação (3) no cálculo da energia gasta pelos *cluster-head* para transmitir dados até a base. Assim sendo, o gasto total de energia (T_x+R_x) de um nó *cluster-head* associado a n outros nós (sem considerar redundância dos dados) é dado por:

$$E_{CH} = \sum_{i=1}^n 2 * E_{elec} * k \quad (5)$$

B. Abordagem baseada em Evolução Diferencial

Na notação do algoritmo de Evolução Diferencial (ED), o número de *cluster-head*'s na solução é definido como a dimensão D do vetor, conforme explicado na seção III. Neste trabalho adotou-se a codificação inteira.

Portanto, para uma rede com $N = 100$ nós e $M = 5$, um vetor com as componentes $V = [21 \ 34 \ 45 \ 78 \ 96]$ indica que os nós 21, 34, 45, 78 e 96 serão *cluster-heads*.

Definida a codificação, cada vetor sofrerá a ação dos operadores mutação e cruzamento, conforme descrito na seção II.A. No momento da seleção (seção II.B.), deve-se calcular então os custos dos vetores alvo e experimental, de forma a determinar qual irá permanecer na população na próxima geração. A avaliação do custo dos vetores é feita da seguinte forma: primeiro, associa-se cada nó que não é *cluster-head* a um nó que seja *cluster-head*, de acordo com um algoritmo estabelecido. Dois métodos serão testados na abordagem que está sendo proposta: o

primeiro considera o critério de distância para associar nó ao *cluster-head* e o segundo considera o critério de equalização de energia. Estes dois métodos serão detalhados a seguir. Após a etapa de associação dos nós calcula-se, para este arranjo, o total de energia gasto para transmissão de dados. Este valor será então usado para obter o custo do vetor.

C. Métodos para associação dos nós livres

A associação de cada nó comum com um nó *cluster-head* deve ser feita de forma a minimizar o gasto total de energia da rede. Como o gasto de energia é diretamente proporcional à distância entre o nó e o *cluster-head* (equações 3 e 4), uma primeira metodologia a ser seguida consistirá em associar cada nó ao *cluster-head* que estiver mais perto (tal processo é o mesmo adotado pelo protocolo LEACH, descrito em [Heinzelman 2000]).

Neste trabalho, uma metodologia adicional será utilizada, no sentido de tentar equalizar o número de nós associados a cada *cluster-head*. Para alcançar tal objetivo, introduz-se a restrição de que um *cluster-head* poderá ter um número máximo de nós associados a ele, e tal número deve ser **diretamente** proporcional à energia disponível neste nó.

Definidos então estes critérios inicia-se o cálculo do número máximo de nós (n_i) em cada *cluster-head*, i e da energia de cada nó da rede. Por exemplo, para uma rede simples, com $N=10$ nós (sendo $M=2$ *cluster-head* e $N-M=8$ nós que não são *cluster-head*), calcula-se o valor de n_1 e n_2 através das seguintes equações:

$$\frac{n_1}{E_1} = \frac{n_2}{E_2} \quad (6) \quad n_1 + n_2 = 8 \quad (7)$$

onde E_1 , E_2 são as energias disponíveis. A eq (8) identifica a condição de equalização energia/número de associações. Já a equação 7 determina que o total de nós livres é igual à soma dos nós de cada *cluster-head*. Os valores de n_1 e n_2 devem ser inteiros, portanto resultados fracionários devem sofrer arredondamento (sempre verificando, evidentemente, se respeita a segunda condição). É possível estender estes conceitos para uma rede com N nós:

$$\frac{n_1}{E_1} = \frac{n_2}{E_2} = \frac{n_3}{E_3} = \dots = \frac{n_n}{E_n} \quad (8)$$

$$n_1 + n_2 + n_3 + \dots + n_n = N - M \quad (9)$$

Sabendo o máximo de nós que cada *cluster-head* pode ter associado, iniciam-se as associações: primeiramente, tenta-se associar cada nó da rede com o *cluster head* que está à menor distância. Se tal *cluster head* não pode mais receber associações, tenta-se com o 2º *cluster-head* mais próximo em distância, e assim por diante, até conseguir associar o nó a algum *cluster-head*. A ordem dos nós no momento da associação é aleatória, de forma a garantir justiça nas associações (caso contrário, sempre os primeiros nós conseguiriam associar-se com os *cluster-*

head mais próximos de si, enquanto que os demais teriam que buscar outras associações).

D. Cálculo efetivo do custo

Para obter os valores reais de gastos de energia, seria necessário calcular os gastos de cada nó da rede. Se o nó é um *cluster-head*, calcular os gastos de recepção (equação 4), devido aos dados recebidos dos nós associados ao mesmo, e os gastos de transmissão (equação 3), devido à transmissão de dados para a estação base. Se o nó é um nó comum, deve ser calculado seu gasto de transmissão (equação 3), devido à transmissão de dados ao seu *cluster-head*. Neste trabalho, optou-se por fazer uma simplificação neste cálculo, considerando somente os gastos de energia dos nós comuns, desconsiderando os gastos de transmissão e recepção dos *cluster-head*'s, devido às seguintes razões:

- Os custos de recepção são proporcionais ao número de nós associados a cada *cluster-head*, conforme equação (4). Como o número de nós da rede é constante, a soma dos gastos de recepção de todos os *cluster-head* também será constante.

- Como a distância de todos os nós à estação base foi considerada a mesma, e não se considera a redundância dos dados a serem transmitidos (seção III.A), a soma dos custos de transmissão dos *cluster-head* também é constante, sendo possível desconsiderar os mesmos na avaliação do custo.

Desta forma, o cálculo do custo é feito através da equação (3), para cada nó comum da rede. Neste caso assume-se $n=1$ e desconsideraram-se as constantes E_{elec} e \mathcal{E}_{amp} , sem perda de informações.

$$E_T = \sum_{j=1}^q \sum_{i=1}^{n_j} (d_i^j)^2 \quad (10)$$

onde d_i^j é a distância do nó i ao *cluster-head* j (cada *cluster* possui n_j sensores, e existem q *clusters*). O valor numérico obtido da equação (10) é, portanto, o custo final do vetor em avaliação.

IV. SIMULAÇÕES E RESULTADOS

Com o intuito de verificar a eficácia da metodologia proposta, foram executadas simulações de uma rede de sensores na qual são aplicadas três possíveis técnicas para a eleição dos *cluster-heads* que otimizam o roteamento dos dados: o algoritmo utilizado pelo LEACH, proposto em [Heinzelman 2000], e as duas técnicas baseadas no algoritmo ED apresentadas neste artigo (ED com associação simples dos nós, na qual cada nó se associa ao *cluster-head* mais próximo, referenciada adiante como “ED simples” e ED com associação proporcional, na qual os *cluster-head* têm uma limitação no número de nós associados, de acordo com sua energia, referenciada adiante como “ED proporcional”). Para as simulações das técnicas baseadas em ED, definiram-se, em testes

preliminares, quais os melhores valores a serem usados nos parâmetros do algoritmo, cujos resultados foram: Estratégia ED-1 (best/1/exp), constantes $F=0.9$, $CR=0.8$ e $NP=10*D$ (tamanho da população 10 vezes o número de elementos do vetor elementar).

Definiram-se três cenários para simulações, cada um contendo, respectivamente, 100, 200 e 400 nós dispostos de forma aleatória em uma região retangular de 1x1 unidades de comprimento (u.c.). Para cada nó, foi atribuída uma energia inicial, também de forma aleatória. Desta forma, simula-se o ambiente de uma rede de sensores real, na qual os nós estão dispostos de forma aleatória e a energia de cada nó é diferente para cada instante de tempo considerado.

Conforme descrito em [Heinzelman 2000], o algoritmo LEACH é executado em ciclos, cada ciclo composto por um número específico de rodadas. O número de rodadas é definido como $(1/p)$, sendo p a porcentagem de *cluster-heads* que compõem a rede. Nos testes deste artigo seguiu-se o mesmo critério adotado em [Heinzelman 2000], definindo $p=5\%$. Desta forma, cada ciclo será composto por $(1/0.05 = 20)$ rodadas.

Para poder comparar os gastos de energia nos modelos testados, simulou-se 1 ciclo completo do algoritmo LEACH (20 rodadas) em cada topologia. Nas metodologias propostas, o processo é semelhante ao algoritmo LEACH: em cada rodada o algoritmo da Evolução Diferencial é executado, de forma a determinar quais nós serão *cluster-head's* naquela rodada. Definiu-se o critério de parada para o algoritmo Evolução Diferencial como sendo o número máximo de 50 gerações.

Definida a topologia (quais nós são *cluster-head*, e para qual *cluster-head* cada nó comum deve se reportar), computa-se o “gasto” de energia na rede da seguinte forma: considera-se que cada nó comum irá reportar uma quantidade fixa de bits para o seu *cluster-head*. Os gastos são computados de acordo com as equações (3) (para nós comuns) e (5) (para *cluster-head's*), atualizando-se o valor da energia de cada nó. Inicia-se então uma outra rodada, utilizando os valores de energia dos nós atualizados na rodada anterior, e uma nova configuração na topologia é calculada pelo algoritmo de ED. O processo se repete até a última rodada.

Para cada topologia, o algoritmo da ED foi rodado 10 vezes com sementes diferentes (gerando diferentes populações iniciais), a fim de diminuir os efeitos da aleatoriedade. Os resultados obtidos são resumidos na Tabela 2, que apresenta a soma da energia gasta ao fim das 20 rodadas (os valores representam a média e desvio padrão considerando-se as 10 execuções com sementes diferentes). A coluna *EDsimp* (ED simples) apresenta os resultados da metodologia na qual cada nó se associa ao *cluster-head* mais próximo, enquanto que a coluna *EDprop* (ED proporcional) apresenta os resultados no caso em que o número de nós associados é proporcional à energia do *cluster-head*.

TABELA 2
Resultados das Simulações

Topologia	EDsimp	EDprop	LEACH
100 nós	$\mu=86,756$ $\sigma=0,154$	$107,776$ $\sigma=0,174$	169,903
200 nós	$\mu=98,354$ $\sigma=0,175$	$\mu=137,335$ $\sigma=0,165$	175,718
400 nós	$\mu=109,457$ $\sigma=0,162$	$\mu=178,300$ $\sigma=0,186$	176,679

A Tabela 3 mostra a representação dos ganhos médios obtidos em cada topologia pelas metodologias propostas, em comparação com o algoritmo LEACH:

TABELA 3
Ganhos relativos ao algoritmo LEACH

Topologia	EDsimp	EDprop
100 nós	50,90%	39,00%
200 nós	44,03%	21,84%
400 nós	35,58%	-4,94%

Pelos resultados, pode-se notar que, quanto menor a densidade de nós da rede, maior a economia de energia obtida pelas modelagens propostas. Evidentemente, a metodologia ED simples sempre obtém menores valores absolutos de gastos de energia, visto que na mesma os nós sempre se associam ao *cluster-head* que estiver mais próximo. Já a metodologia ED proporcional obtém resultados inferiores, principalmente na topologia com maior densidade (400 nós), onde o resultado do algoritmo LEACH foi melhor (a “economia negativa” apresentada na tabela 3 deve-se a isso). Este resultado é compreensível uma vez que, para o algoritmo usado por LEACH, quanto maior a densidade, mais fácil de agrupar os nós em *clusters* “evidentes” (agrupamentos de nós claramente identificáveis), pois cada nó se associa ao *cluster-head* mais próximo. Já a metodologia ED proporcional sempre se preocupa em distribuir a carga, mesmo que isso signifique abrir mão de configurar *clusters* teoricamente evidentes.

Porém, deve-se analisar outro aspecto dos resultados obtidos pela metodologia ED proporcional: tais resultados foram obtidos em configurações de topologia na qual a carga foi uniformemente distribuída entre os *cluster-head*. As Tabelas 4, 5 e 6, evidenciam esta característica mostrando os resultados para o caso da topologia com 100 nós, após a execução do algoritmo LEACH, ED simples e ED proporcional, respectivamente:

TABELA 4
Exemplo de configuração - LEACH

<i>Cluster-head</i>	22	55	61	1	80
Nº nós associados	21	17	22	28	7
Energia disponível	8.38	7.6	6.52	3.46	7.77

TABELA 5

Exemplo de configuração - ED simples

<i>Cluster-head</i>	76	22	85	27	89
Nº nós associados	28	17	20	11	19
Energia disponível	3.92	4.70	6.27	8.19	6.62

TABELA 6

Exemplo de configuração - ED proporcional

<i>Cluster-head</i>	65	10	24	73	18
Nº nós associados	12	17	18	21	27
Energia disponível	3.63	5.07	5.24	6.06	8.02

Verifica-se nos resultados do LEACH e ED simples, que nós com a menor energia entre todos os nós escolhidos para serem *cluster-head's* (como o nó 76 no ED simples e nó 1 no LEACH) tiveram o maior número de nós associados. Ou seja, serão os nós que mais gastarão energia nesta rodada. Isso certamente terá impactos em uma execução longa (como é o caso de uma rede de sensores real), pois nós com menor energia irão se esgotar antes. Já na metodologia ED proporcional (Tabela 6), verifica-se a distribuição de nós entre os *cluster-head's* é feita de forma proporcional à energia de cada *cluster-head*, proporcionando então um balanceamento da carga.

V. CONCLUSÕES E TRABALHOS FUTUROS

Este artigo utilizou a técnica de formação de *clusters* em redes de sensores sem fio. Foi apresentada uma proposta na qual a configuração da rede é feita remotamente, utilizando-se o algoritmo de Evolução Diferencial como ferramenta de otimização, de forma a obter uma configuração de baixo custo para uma determinada topologia.

Duas formas de configuração da topologia foram propostas: uma forma simples, na qual cada nó se associa ao *cluster-head* mais próximo, e uma forma proporcional, na qual a atribuição dos nós aos *cluster-head's* é proporcional a sua energia restante. Foram executados testes para comparar as duas metodologias com o algoritmo LEACH. Os resultados mostraram que as metodologias propostas obtiveram melhor desempenho referente ao gasto de energia local (transmissão de dados dos sensores até os *cluster-head's*) em redes com baixa densidade de nós, sendo que tal diferença diminuiu conforme aumentou a densidade da rede. Notou-se também que para a metodologia de atribuição proporcional, obteve-se uma equalização de gastos de energia entre os nós escolhidos como *cluster-head's*, pois se tomou como princípio que a distribuição de carga entre os mesmos deve ser proporcional à sua energia. Em redes onde a estação base receptora dos dados está longe da rede de sensores, tal equalização tem fundamental importância no prolongamento da vida útil da rede, pois se evita a morte prematura de nós.

Como trabalhos futuros, algoritmos específicos para otimização combinatória poderão substituir a Evolução Diferencial, a fim de verificar os seus desempenhos e os custos computacionais. Também pretende-se modificar o

procedimento para equalização da carga entre os *cluster-head's*, considerando como critério não somente o número de nós associados, mas também o grau de redundância dos dados dos nós comuns. Novas simulações de comparação com outros protocolos (além do LEACH) deverão ser realizadas de forma a verificar o real ganho da vida útil da rede incluindo a fase de roteamento dos dados.

REFERÊNCIAS

- Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E. (2002). "A survey on Sensor Networks", IEEE Communications Magazine, August 2002.
- Ali, M. Montaz, Khompatraporn, Charoenchai, Zabinsky, Zelda B. (2005). "A numerical Evaluation of Several Stochastic Algorithms on Selected Continuous Global Optimization Test Problems", In Journal of Global Optimization, 2005.
- Falko Dressler, "Benefits of Bio-inspired Technologies for Networked Embedded Systems: An Overview", In Proceedings of Dagstuhl Seminar, May 2006.
- Heinzelman, W., Chandrakasan, A., Balakrishnan, H. (2000). "Energy-Efficient Communication Protocol for Wireless Microsensor Networks". Maui, Hawaii, Jan 2000. In Proceedings of the 33rd International Conference on System Sciences (HICSS '00).
- Heinzelman, W., Chandrakasan, A., Balakrishnan, H. (2002). "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," IEEE Trans. Wireless Commun., vol. 1, no. 4, Oct. 2002, pp. 660-70.
- Huseyin Ozgur Tan et. al., "Power Efficient Data Gathering and Aggregation in Wireless Sensor Networks", SIGMOD Record, Vol. 32, No. 4, December 2003.
- Lewis, F. L. (2004). "Wireless Sensor Networks". In Smart Environments: Technologies, Protocols and Applications, ed. D.J. Cook, New York, 2004.
- Lindsey, S., C. S. Raghavendra, C. S. "Pegasis: Power-efficient gathering in sensor information systems," in IEEE Aerospace Conference, March 2002.
- Murugunathan, Siva D., Daniel C. F. Ma, Rolly I. Bhasin and Abraham O. Fapajuwo, "A Centralized Energy-Efficient Routing Protocol for Wireless Sensor Networks" IEEE Radio Communications, March 2005 pp. S8-S13.
- Storn, R., Price, K (1995). "Differential Evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces", Technical Report TR-95-012, International Computer Science Institute, Berkeley.
- Storn, R., Price, K (1997). "Differential Evolution: a simple and efficient heuristic for global optimization over continuous spaces", Journal of Global Optimization vol. 11, pp. 341-359.