

# Um Estudo sobre a Relação Custo-Benefício do Uso de Códigos Corretores de Erro para a Redução do Consumo de Energia em Redes de Sensores Sem Fio

Vanderlei Zarnicinski, Marcelo E. Pellenz, Richard D. Souza e Mauro S. P. Fonseca

**Resumo** — Redes de Sensores Sem Fio (RSSF) são normalmente formadas por pequenos dispositivos alimentados a bateria, sendo um dos principais desafios no desenvolvimento e aperfeiçoamento destas redes a eficiência no uso da energia disponível, buscando um prolongamento da vida útil dos sensores e da rede como um todo. O canal de rádio introduz severas degradações no sinal, aumentando a taxa de erros na recepção e causando diversas retransmissões, que se reflete em maior consumo de energia do nó-sensor. Neste trabalho propomos formas de operação dos sensores que aumentem a eficiência no uso da energia disponível, através de estratégias otimizadas de codificação dos dados transmitidos que os tornem mais robustos aos possíveis erros introduzidos pelo canal e minimizem eventuais retransmissões. Estas estratégias se baseiam no controle de erros híbrido, utilizando a codificação convolucional para correção de erros (FEC) e a retransmissão automática de dados corrompidos (ARQ).

**Palavras-Chave** — ARQ, FEC, Redes de Sensores Sem Fio, Codificação de Canal, Consumo de Energia, Controle de Erros.

**Abstract** — Wireless Sensor Networks (WSN) are usually a set of battery-supplied small devices. One of the main challenges in deployment of WSN is to improve energy-efficiency and lifetime of the nodes while keeping communication reliability. The radio channel introduces impairments in the signal, increasing the error rate at reception and causing several retransmissions, consuming extra energy in the node. In this paper we investigate operation modes for the sensor nodes that improve their energy efficiency, through coding schemes which make the transmitted data less susceptible to errors and reduce occasional retransmissions. These coding schemes are based on hybrid error control, using automatic repeat-request (ARQ) and convolutional coding forward-error-correction (FEC).

**keywords** — ARQ, FEC, Wireless Sensor Networks, Channel Coding, Energy Consumption, Error Control.

## I. INTRODUÇÃO

O tema Rede de Sensores Sem Fio (RSSF) vem recebendo atenção crescente durante os últimos anos, com inúmeros trabalhos publicados que buscam novas soluções para os diversos problemas que se apresentam, como controle de acesso ao meio, formação de topologia, sincronismo, algoritmos de roteamento e controle de erros [1-6].

Uma rede de sensores é formada em geral por pequenos dispositivos eletrônicos, compostos por uma unidade de processamento e por uma unidade de comunicação em rádio frequência, ambas operando de forma colaborativa e alimentadas por bateria.

Vanderlei Zarnicinski, Marcelo E. Pellenz e Mauro S. P. Fonseca, Programa de Pós-Graduação em Informática (PPGIA), Pontifícia Universidade Católica do Paraná, Curitiba, Brasil, e-mails: {marcelo, mauro.fonseca}@ppgia.pucpr.br. Richard Demo Souza, Universidade Tecnológica Federal do Paraná (UTFPR), Paraná, Curitiba, Brasil. e-mail: richard@cpgei.cefetpr.br

Em decorrência disso, uma preocupação comum à maioria dos trabalhos nesta área é a conservação de energia. O objetivo é prolongar a vida útil de cada nó sensor e da rede como um todo, uma vez que a perda de algum nó da rede devido ao esgotamento de capacidade de sua bateria pode deixar a rede inoperante. Geralmente os fabricantes dos dispositivos micro-eletrônicos do sensor propiciam alguma forma de otimização no consumo de energia em seus produtos, sendo a mais comum a possibilidade de deixar o dispositivo (unidade de processamento ou transceptor de rádio) em estado inativo (*idle state*) ou dormente (*sleeping mode*), quando o consumo de energia é bastante reduzido em relação ao modo ativo. Visto que o módulo transceptor de rádio no sensor normalmente é o que demanda maior consumo de energia, muitas das estratégias de conservação de energia buscam minimizar o período de tempo que este módulo permanece ativo, o qual depende da necessidade de envio de dados coletados no sensor ou de ouvir o meio para recepção. Uma destas estratégias é ativar o transmissor somente quando há dados para transmitir, deixando-o dormente no restante do tempo, porém mantendo o receptor ativo para receber qualquer informação endereçada ao nó, exceto durante o tempo de operação do transmissor. Tal solução certamente não é eficaz quanto ao uso de energia, visto que o receptor está ativo independente de haver dados a receber. Algumas soluções procuram contornar este problema mantendo também o receptor inativo, ativando-o em curtos períodos para verificar se há atividade no canal e dados a receber, voltando a seguir ao estado dormente se nenhum sinal for detectado [1]. Outras soluções propostas fixam como objetivo o aperfeiçoamento de estratégias de roteamento na rede [6], distribuindo a carga de encaminhar as informações até o nó de destino num cenário de agregação local de dados (cluster). Neste cenário o encaminhamento dos dados até o destino ocorre em múltiplos saltos através dos nós concentradores (*cluster heads*), os quais sofrem um maior consumo de energia, devido ao maior período ativo e maior potência de transmissão. Para distribuir esta maior carga na rede, em intervalos aleatórios a função de concentrador é assumida por outro nó, buscando um equilíbrio de consumo entre toda a rede [2][3]. Entretanto, estas soluções sofrem os efeitos do canal, uma vez que qualquer comunicação de rádio está sujeita à degradação do sinal decorrente de diversos fatores, principalmente ruído e desvanecimento do sinal [9][14]. Em função da implementação da rede, o nó que recebe dados corrompidos pode simplesmente ignorá-los e aguardar nova transmissão ou solicitar uma retransmissão, em qualquer caso ocorrendo desperdício de energia,

especialmente se o canal se encontrar em condições de degradação severas que possam causar diversas retransmissões. Outros métodos relativos à conservação de energia em nós sensores procuram melhorar a confiabilidade do sinal transmitido, reduzindo a taxa de erros e as decorrentes retransmissões. Uma das formas de melhorar esta confiabilidade é através da codificação dos dados a serem transmitidos com um código corretor de erros [4][5], de modo que possíveis erros introduzidos pelo canal possam ser corrigidos no receptor. As estratégias de correção de erros se dividem basicamente em códigos de bloco e códigos convolucionais. A técnica de codificação convolucional é uma das estratégias mais utilizadas em ambientes de comunicação sem fio como redes de sensores. A codificação convolucional apresenta algumas vantagens, como a facilidade de implementação do codificador e uma estrutura de decodificação bastante simples usando o algoritmo de Viterbi [11]. Como a eficiência do código está relacionada à sua taxa, ou seja, o número de bits gerado para cada bit codificado, outro aspecto importante dos códigos convolucionais é a possibilidade de alteração da taxa do código sem alterar a estrutura do codificador, através da técnica de punção [8][10]. Neste trabalho apresentamos uma forma de controle de erros baseada em um protocolo híbrido, que incorpora tanto a parte de detecção de erros e solicitação de retransmissão (ARQ) como uma parte de correção de erros (FEC) através de codificação convolucional com complexidade variável. A complexidade é variada em função da taxa e da memória do codificador. Para a análise e simulação do cenário proposto, adotamos um código convolucional com taxa 1/2 e diferentes complexidades de acordo com a ordem de memória do codificador. O principal objetivo é demonstrar a melhoria na preservação da energia no sensor, através do reforço da capacidade de correção de erros com codificação de maior complexidade, o que apesar de requerer maior energia de processamento para decodificação reduz o número de retransmissões e conseqüentemente a energia consumida na transmissão e recepção. Ainda que numa primeira análise pareça adequada a utilização de codificação com a maior complexidade possível em todos os nós, visando a máxima correção de erros, isto não se mostra a melhor solução, visto que as taxas de erro entre os nós pode ser distinta, ocorrendo assim um maior consumo de energia de processamento para correção de poucos erros. Como resultado obtemos uma maior preservação da energia disponível na rede, sendo que em ambas as situações a escolha da complexidade do código e o conseqüente consumo de energia de processamento irão pautar a vida útil da rede. Este trabalho está estruturado da seguinte forma: A Seção 2 apresenta o modelo de canal utilizado e discute seus efeitos que afetam a comunicação entre os nós sensores. Na Seção 3 são discutidas as estratégias de controle de erros e os aspectos de complexidade com a utilização de códigos convolucionais. Na Seção 4 é feita a avaliação do consumo de energia para transmissão e recepção, em função do número de retransmissões. Também é analisado o consumo de processamento para decodificação dos dados de acordo com a complexidade do código utilizado. As conclusões são apresentadas na Seção 5.

## II. MODELO DE CANAL

Em qualquer processo de comunicação, o sinal recebido sofre a influência do ruído Gaussiano branco aditivo (AWGN - *Additive White Gaussian Noise*) e também da atenuação do sinal (perda de percurso) ao longo do canal de comunicação. Numa comunicação sem fio através de ondas de rádio ocorre uma degradação adicional causada pela interferência de multipercurso, ou seja, réplicas do sinal transmitido que se propagam por diversos caminhos causando interferências construtivas e destrutivas no receptor, gerando flutuações rápidas na intensidade do sinal recebido. Classicamente este fenômeno é modelado estatisticamente segundo a distribuição de Rayleigh. Estes efeitos do canal de rádio degradam o desempenho da rede de sensores em termos de consumo de energia, pois os pacotes recebidos com erro devem ser retransmitidos. O desempenho do nó sensor receptor em termos da probabilidade de erro de bit ou de quadro, depende portanto de três fatores. O primeiro é a potência média recebida que é determinada pelo modelo de perda de percurso. O segundo fator é o nível de ruído térmico gerado pelo hardware do receptor, que conjuntamente com a potência média recebida do sinal e a taxa de transmissão do sistema, vai definir a relação sinal-ruído média no receptor. O terceiro fator é a condição momentânea do canal de rádio (ganho do canal) durante a transmissão do quadro, que segue a distribuição Rayleigh.

O modelo de perda de percurso [9] utilizado neste trabalho é definido pela equação (1), em que  $PL(d)$  é a perda de percurso calculada para uma distância  $d$  entre transmissor e receptor. O parâmetro  $PL(d_0)$  é a perda de percurso numa distância de referência  $d_0$  e  $n$  é o expoente de perda relativo ao ambiente, que indica a taxa de aumento na perda com a distância, normalmente situado entre 2 e 4 [9]. O parâmetro  $\chi_\sigma$  é uma variável aleatória com distribuição *log-normal*, que representa o efeito de sombreamento no sinal, ou seja, flutuações no valor da perda de percurso para uma mesma distância entre transmissor e receptor.

$$PL(d) \text{ dB} = PL(d_0) + n \cdot 10 \cdot \log_{10} \left( \frac{d}{d_0} \right) + \chi_\sigma \quad (1)$$

A potência do sinal recebido, em função da distância ( $d$ ) entre transmissor e receptor é dada pela equação (2), em que  $P_{tx}$  e  $P_{rx}$  representam a potência transmitida e recebida, respectivamente. A razão entre a potência recebida e a potência do ruído é definida como relação sinal-ruído (*SNR - Signal-to-Noise Ratio*).

$$P_{rx}(d) \text{ dBm} = P_{tx} \text{ dBm} - PL(d) \text{ dB} \quad (2)$$

Para modulações digitais binárias, como por exemplo BPSK e BFSK, a *SNR* pode ser expressa em função da energia de um bit,  $E_b$ , e da densidade espectral de potência unilateral do ruído,  $N_0$  [9]:

$$\frac{E_b}{N_0} = SNR \cdot \frac{B}{R_b} \quad (3)$$

em que  $R_b$  é taxa de transmissão em bits por segundo e  $B$  é a largura de banda de ruído do transceptor, a qual depende da característica construtiva do mesmo.

A probabilidade de erro teórica em um canal com desvanecimento que segue o modelo de Rayleigh depende do tipo de modulação utilizada e da relação sinal-ruído. Neste trabalho adotamos a técnica de modulação binária por comutação de fase (*BPSK – Binary Phase Shift Keying*), que é amplamente utilizada pelos transceptores de rádio usados em nós sensores. A probabilidade de erro de bit (*BER – Bit Error Rate*) da modulação *BPSK* no canal com desvanecimento Rayleigh é dada por [14]:

$$BER = \frac{1}{2} \cdot \left( 1 - \sqrt{\frac{\bar{\gamma}}{1 + \bar{\gamma}}} \right) \quad (4)$$

em que  $\bar{\gamma}$  é o  $E_b/N_0$  médio. Como a ocorrência de pelo menos um bit com erro no quadro transmitido provoca a rejeição de todo o quadro, a taxa de erros dos quadros (*FER – Frame Error Rate*) pode também ser determinada analiticamente a partir da taxa de erro de bit, como

$$FER = 1 - (1 - BER)^f \quad (5)$$

em que  $f$  é o tamanho do quadro codificado em bits. Neste trabalho vamos assumir um tamanho de quadro usado para transmissão dos nós sensores de 288 bits, que é o tamanho típico utilizado no sistema operacional para sensores TinyOS [13].

### III. ESTRATÉGIAS DE CONTROLE DE ERROS

Uma das estratégias mais utilizadas de controle de erros é o protocolo ARQ (*Automatic Repeat Request*), o qual implementa apenas codificação para detecção de dados corrompidos e solicita sua retransmissão. Esta forma de controle de erros numa rede de sensores degrada o consumo de energia, uma vez que pode ser necessário diversas retransmissões para entregar a mesma informação. Um aperfeiçoamento no controle de erros são os protocolos ARQ-Híbridos, que além da detecção de erros incorporam também um código de correção de erros (*FEC – Forward Error Correction*), os quais podem ter sua complexidade variável, tornando assim o controle de erros muito mais robusto [10]. Enquanto códigos com complexidade variável tornam possível o controle de erro adaptativo às condições do canal, por outro lado o esforço computacional da decodificação é proporcional à complexidade, o que implica em maior consumo de energia. Os códigos convolucionais têm a característica de apresentar ganhos de codificação crescentes com a ordem de memória do código. Esta característica pode ser aproveitada para se implementar uma estratégia de codificação adaptativa. Mantendo o mesmo tamanho de quadro é possível alterar a ordem de memória ou complexidade do código, reforçando a capacidade de correção de erros quando a condição do canal se torna adversa. Assim podemos limitar o número de retransmissões e garantir um nível mínimo de consumo para a comunicação entre dois nós sensores da rede.

#### A. Codificação Convolucional

Os códigos convolucionais são representados pelos parâmetros  $(n, k, m, d_{free})$ , em que  $n$  é o número de bits na saída do codificador,  $k$  é o número de bits na entrada,  $m$  é a ordem de memória do codificador e  $d_{free}$  é a distância livre do código, que vai definir a distância de Hamming mínima que pode ser obtida entre duas seqüências de bits codificados. A distância mínima define a capacidade de correção do código. Códigos com a mesma taxa  $r = k/n$  podem apresentar eficiências distintas, de modo que se procura adotar códigos que apresentam máximo  $d_{free}$  [8]. A estrutura do codificador convolucional é determinada pelo arranjo entre os registradores que formam a memória e os somadores binários que formam a saída. As conexões entre os registradores e as saídas são especificadas em termos do polinômio gerador. Para uma seqüência de entrada,  $\bar{u}$ , com tamanho  $k \cdot L$  bits, teremos uma seqüência de saída,  $\bar{v}$ , com tamanho  $n \cdot (L + m)$  bits, uma vez que é necessário zerar a memória do codificador para a entrada de outra seqüência.

#### B. Decodificação

No receptor a seqüência é normalmente decodificada por um decodificador de máxima verossimilhança, sendo geralmente adotado o algoritmo de Viterbi [11], que avalia a transição de estados  $S_i$  representados pelo conteúdo da memória do codificador para cada bit de entrada. Assim o tamanho da memória do código determina o número de estados a ser avaliado, os quais podem ser representados por um diagrama de treliça, na qual constam os possíveis estados e suas transições.

A transição entre dois estados  $(S_i, S_{i+1})$  representa a saída do codificador, sendo avaliado em cada transição o próximo estado mais provável, de acordo com a métrica entre a informação recebida e a esperada transição de estado do codificador. Esta métrica local vai sendo acumulada à medida que se avança na treliça, sendo armazenado o caminho de maior métrica, descartando-se os demais. Desta forma ao final da decodificação o caminho registrado é percorrido em ordem reversa, reproduzindo a informação codificada mais provável [8][10]. Note-se que toda a treliça é uma repetição de módulos idênticos de transição de estados, sendo que em cada módulo o decodificador precisa realizar as operações de comparar a seqüência recebida, selecionar e armazenar a transição mais provável e acumular a métrica calculada, o que demanda esforço computacional.

Uma vez que os estados representam o conteúdo da memória e as transições a saída do codificador de acordo com sua entrada, cada módulo da treliça apresenta  $2^m$  estados iniciais e  $2^m$  estados finais, sendo cada estado inicial conectado a exatamente  $2^k$  estados finais. Cada módulo tem assim  $2^{m+k}$  conexões entre estados, de comprimento  $n \cdot 2^{m+k}$ . A complexidade de decodificação, que determinará o esforço computacional necessário para calcular e escolher o caminho mais provável, é dada pelo número de símbolos gerados por bit codificado [7]:

$$C = \frac{n}{k} \cdot 2^{m+k} \quad \text{símbolos/bit codificado} \quad (6)$$

Este esforço computacional, de acordo com a implementação do algoritmo, determina o número de instruções necessárias para a unidade de processamento decodificar um quadro e conseqüentemente quanta energia de processamento será consumida.

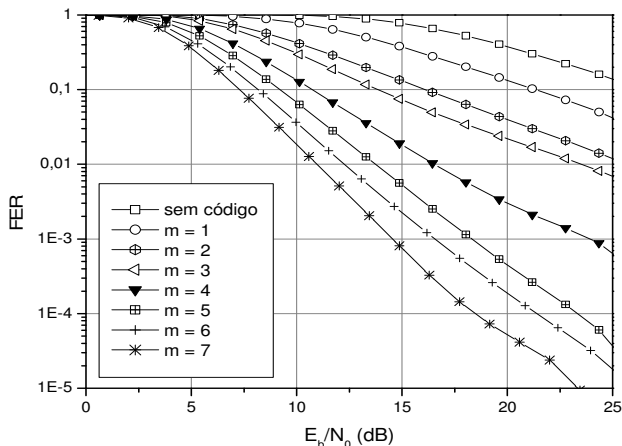


Figura 1. Taxa de erro de quadro para dados codificados no canal Rayleigh

### C. Consumo de Energia de Processamento

O algoritmo de Viterbi, pela sua simplicidade, pode ser implementado em hardware com dispositivos dedicados ou em software, como já desenvolvido em algumas plataformas de microprocessadores. De acordo com as avaliações de desempenho deste decodificador [12], para implementação em microprocessadores adotados em alguns sensores [15][16], podemos estimar o esforço de três ciclos de instrução por símbolo e definir o consumo de energia para decodificar um bit,  $E_{dec}$ , como

$$E_{dec} = \frac{3 \cdot C \cdot P_{proc}}{I} \quad (\text{Joules/bit}) \quad (7)$$

em que  $P_{proc}$  é a potência em watts dissipada pelo processador em estado ativo e  $I$  é a sua capacidade de processamento em instruções/segundo/hertz. Para o processador utilizado na plataforma Mica2 [16] o desempenho de processamento é de 1 MIPS/MHz (Milhões de Instrução por Segundo/Megahertz) [15]. Conforme definido pela equação (6), ocorre um crescimento exponencial da complexidade em função da memória do código, que se reflete no consumo de energia de processamento, conforme a equação (7).

Podemos observar também que a complexidade e o consumo de energia de decodificação crescem em razão inversa da taxa do código. A capacidade de correção também é inversamente proporcional a taxa do código. Outras taxas diferentes de  $r=1/n$ , como  $r=2/3$  ou  $r=3/4$ , podem ser obtidas através da técnica de punção do código, que requer energia adicional de processamento para recomposição do quadro punçado. Esta técnica, porém, não é abordada no escopo deste trabalho, no qual as análises e simulações foram

baseadas num código com taxa  $r=1/2$  e complexidade variável de acordo com o tamanho de memória.

### D. Desempenho da Codificação Convolutional

Com o aumento da complexidade do código ocorre um aumento de consumo de energia de processamento (desvantagem) e uma melhora da capacidade de recuperação de erros (vantagem). Neste sentido o foco principal deste estudo é determinar a melhor escolha das complexidades dos códigos usados na transmissão entre dois nós da rede de sensores, de maneira a se obter a melhor relação entre consumo de processamento e capacidade de correção do código. Desta forma, em determinada transmissão, a taxa de erros será inversamente proporcional à memória do código. A Figura 1 apresenta as curvas de desempenho em termos da taxa de erro de quadro obtidas através de simulação num canal com desvanecimento Rayleigh, usando códigos convolucionais de taxa  $r=1/2$ , tamanho de bloco codificado  $f=288$  bits, com diferentes ordens de memória ( $m=1,2,\dots,7$ ). Estamos considerando o desvanecimento Rayleigh independente de símbolo para símbolo, ou seja, desvanecimento rápido.

## IV. CONSUMO DE ENERGIA DE TRANSMISSÃO

A informação presente no sensor transmissor precisa ser modulada e amplificada pelo módulo transmissor de rádio, antes de ser propagada no meio. Este estágio do sensor apresenta um grande consumo de energia, sendo comum a operação do transmissor somente quando há dados a serem enviados, permanecendo em estado inativo no maior período e preservando energia. De modo similar do lado do nó sensor de destino, o receptor precisa estar ativo para receber e demodular a informação. Assim a energia conjunta consumida na comunicação entre dois nós pode ser definida por:

$$E_{trans} = \frac{P_{tx} + P_{rx}}{R_b} \quad (\text{Joules/bit}) \quad (8)$$

em que  $P_{tx}$  e  $P_{rx}$  são as potências em watts dissipadas pelo transmissor e receptor, respectivamente,  $R_b$  é a taxa de transmissão em bits por segundo. Devemos observar que em determinados transceptores, o valor da potência de transmissão,  $P_{tx}$ , pode ser configurada, tipicamente entre -20 dBm e +10dBm [17].

Quando adotamos a codificação convolutional para correção de erros, além da energia consumida na recepção, temos ainda a energia de processamento necessária para decodificar um bit,  $E_{dec}$ , conforme definido pela equação (7). Portanto, devemos calcular agora qual será o consumo para decodificar um quadro completo recebido. A energia para decodificar o quadro será denotada por  $E_{dq}$ . Podemos calcular a energia necessária para decodificar um quadro,  $E_{dq}$ , como sendo

$$E_{dq} = E_{dec} \cdot f \cdot r \quad (\text{Joules/quadro de informação}), \quad (9)$$

em que  $r$  é a taxa do código e  $f$  é o tamanho do quadro em bits.

Nos protocolos ARQ híbridos, nos quais ocorre a retransmissão automática de quadros com dados corrompidos em que não foi possível a correção pelo FEC, a ocorrência de erros na transmissão pode causar diversas retransmissões e diminuir sensivelmente a vazão da rede (*throughput*), que é definida como o número esperado de transmissões necessárias até que o quadro seja aceito pelo receptor [10],

$$E[Tr] = \frac{1}{1 - FER} \quad (10)$$

em que  $FER$  é a probabilidade de erro do quadro. Assim podemos estimar o número de transmissões do quadro em função da potência do sinal recebido, tanto para dados transmitidos sem codificação como para dados codificados com diversas complexidades, conforme mostrado na Figura 2.

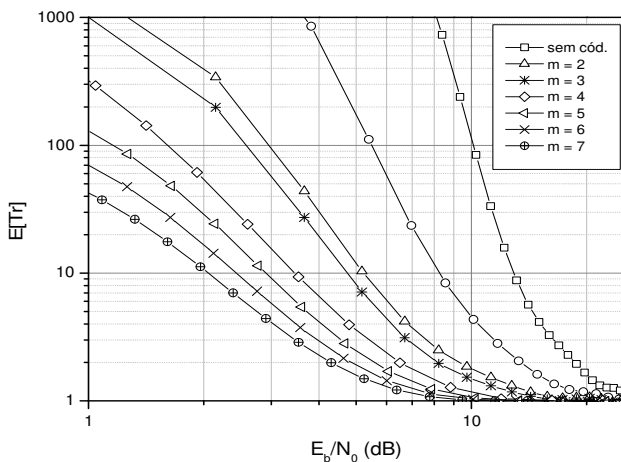


Figura 2. Número esperado de retransmissões do quadro no canal Rayleigh

Cada retransmissão dos dados requer energia adicional do sensor, de forma que a energia total média consumida para transmissão do quadro,  $E_T$ , será determinada pelo número esperado de retransmissões, conforme a equação (10), e pela energia necessária para transmitir e receber um quadro, conforme a equação (9):

$$E_T = E[Tr] \cdot E_{trans} \cdot f \quad (\text{Joules/quadro codificado}) \quad (11)$$

De acordo com as especificações de consumo tanto do rádio [17] como do processador usado na plataforma Mica2 [16], calculamos o consumo de energia na comunicação, ou seja, a energia para transmissão e para decodificação de um quadro, em canal com desvanecimento Rayleigh, para uma potência de transmissão de 0 dBm em 19200 bps. As curvas são apresentadas na Figura 3 para diferentes valores de relação sinal-ruído e os valores de energia estão normalizados para o valor de consumo de comunicação de um quadro. Conforme podemos ver na Figura 2, à medida que aumenta a relação sinal-ruído, o número esperado de transmissões converge para a unidade, para os diversos valores de ordem de memória do codificador. Assim há uma limitação no consumo de energia de transmissão. Entretanto, a energia consumida na decodificação aumenta com a ordem de memória do código. Deste modo, para valores elevados da

relação sinal-ruído, o aumento da complexidade do código aumenta o consumo de energia, conforme a Figura 3, onde podemos perceber que há um *ponto ótimo* de complexidade, onde o consumo de energia é minimizado. Por exemplo, para a curva de  $E_b/N_0 = 10$ , o valor ótimo de ordem de memória ou complexidade do código é aproximadamente  $m=4$ . Para as curvas  $E_b/N_0 = 7$  e  $E_b/N_0 = 5$ , os valores ótimos de memória são  $m=5$  e  $m=6$ , respectivamente. Para condições ruins de relação sinal-ruído, como por exemplo as curva de  $E_b/N_0 = 3$  e  $E_b/N_0 = 1$ , o ponto ótimo de complexidade se encontra num valor de  $m$  bem superior, conforme observado pela tendência das curvas na Figura 3. Isto mostra que a adoção uniforme da maior complexidade possível em todos os nós da rede não é adequada, sendo necessária uma escolha otimizada em cada nó de acordo com a situação do canal. Assim a escolha da melhor complexidade individual e conseqüentemente o consumo de energia de decodificação determinarão a vida útil da rede.

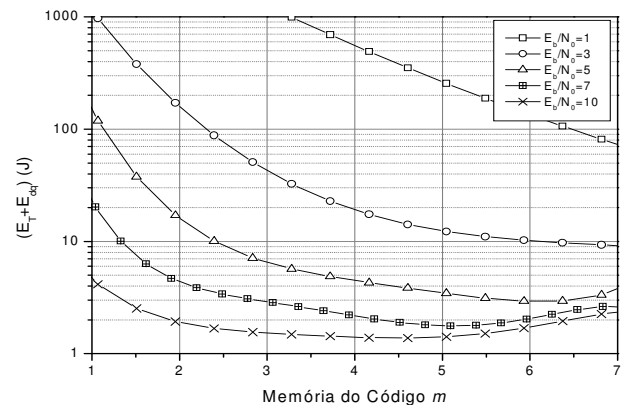


Figura 3. Consumo médio de energia de comunicação para o canal Rayleigh

#### E. Consumo Total de Encaminhamento

O consumo total de energia na comunicação entre dois nós reflete a transmissão de dados num único salto (*single hop*). O encaminhamento de mensagens numa rede, entretanto, normalmente ocorre em múltiplos saltos (*multiple hop*), gerando diversas rotas ao longo das quais os nós podem apresentar diversos perfis de consumo, dependendo da condição do canal entre os nós, conforme mostrado na Figura 4. Assim deve ser avaliada a situação de cada nó na rota para adequar o código que minimize o consumo esperado para encaminhamento da mensagem desde a fonte dos dados até o receptor. Nesta avaliação de consumo eventualmente pode ser vantajosa a eliminação de um salto intermediário. Considere como exemplo o enlace (canal)  $L_{14}$  ilustrado no cenário da Figura 4, com a eliminação do salto intermediário através de  $L_3 - L_4$ . A partir das distâncias entre os nós, aplicando as equações (1), (2) e (3) com  $n=4$  e potência de ruído de  $-115$  dBm, podemos obter os respectivos valores de  $E_b/N_0$  no receptor, conforme mostrado na Tabela 1.

	$L_3$	$L_4$	$L_{14}$
$d(m)$	15	17	18
$E_b/N_0$	$\approx 10$	$\approx 7$	$\approx 5$

Tabela 1. Relação sinal-ruído no nó receptor de cada enlace

Considere as curvas para  $E_b/N_0 = 10$  e  $E_b/N_0 = 7$  da Figura 3. Neste exemplo, supondo que os nós  $C_4$  e  $C_5$  utilizem codificação com memória  $m=1$ , obtemos da Figura 3 um valor total de consumo de aproximadamente  $(E_T + E_{dq})_{m=1}^{C_4} + (E_T + E_{dq})_{m=1}^{C_5} \approx 24,5$ . Se a codificação do nó  $C_5$  utilizar memória  $m=2$ , podemos obter da curva para  $E_b/N_0 = 5$  na Figura 3 o valor de consumo para a transmissão direta entre os nós  $C_5$  e  $C_3$  (enlace  $L_{14}$ ) como sendo  $(E_T + E_{dq})_{m=2}^{C_5} \approx 19$ . Portanto a solução de aumentar a memória do código do nó  $C_5$  para  $m=2$ , produzirá uma redução na energia esperada de comunicação, uma vez que o somatório da energia utilizada nos canais  $L_3$  e  $L_4$ , para  $m=1$ , é maior que a energia utilizada no canal  $L_{14}$ , para  $m=2$ , conforme a Figura 3. Na avaliação desta situação, deve ser satisfeita a seguinte equação:

$$\frac{(E_T + E_{dq})_{m=1}}{(E_T + E_{dq})_{m=2}} \geq 1 \quad (12)$$

De modo similar, deve ser avaliada a rota que apresenta o menor consumo, ou seja, a menor somatória dos consumos dos canais, com eventuais alterações de memória do código. Em determinadas soluções de roteamento, entretanto, que adotam nós concentradores para encaminhamento de informações agrupadas de diversos nós [2], a utilização de codificação mais complexa nos concentradores pode otimizar o consumo de toda a rede.

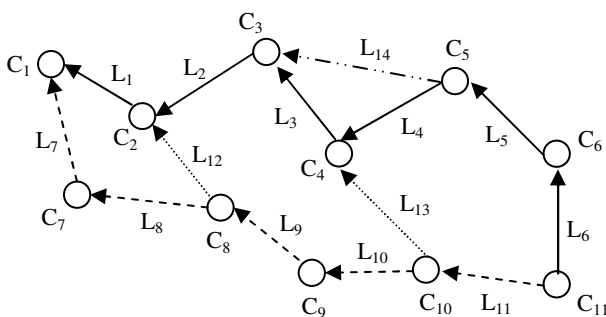


Figura 4. Encaminhamento de mensagens em múltiplos saltos

## V. CONCLUSÃO

Demonstramos neste trabalho que o uso de codificação convolucional para correção de erros em Redes de Sensores Sem Fio, com complexidade variável em cada nó, pode proporcionar ganhos de energia, face às condições do canal que podem provocar retransmissões através do protocolo ARQ-Híbrido. Conforme investigado na Seção 4, a escolha

adequada da ordem de memória do código em cada nó, de acordo com a situação do canal, determinará o consumo de energia e a vida útil do sensor e da rede. Da mesma forma foi demonstrado que através da complexidade variável do código corretor de erro o consumo pode ser otimizado de acordo com a topologia da rede. Outros assuntos relacionados também constituem um amplo campo de investigação, como protocolos de roteamento otimizados em função do consumo. O desenvolvimento de tais protocolos deve ter como objetivo o estabelecimento de rotas adaptativas às condições dos canais, buscando obter a melhor relação entre consumo de codificação e consumo de transmissão de cada canal, conforme a topologia da rede, já que a menor distância pode não ser o melhor parâmetro para o estabelecimento da rota, como exemplificado.

## REFERÊNCIAS

- [1] Polastre, J., Hill, J., Culler, D. (2004), "Versatile Low Power Media Access for Wireless Sensor Networks", Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems (Sensys'04), Baltimore, USA.
- [2] Heinzelman, W. B., Chandrakasan, A. (2000), "Energy-efficient Communication Protocol for Wireless Microsensor Networks", Proceedings of the 33<sup>rd</sup> Hawaii International Conference on System Sciences, Maui, Hawaii.
- [3] Zuniga, M., Krishnamachari, B. (2004) "Analyzing the Transitional Region in Low Power Wireless Links", IEEE Secon04, Santa Clara, California.
- [4] Karvonen, H., Shelby, Z., Pomalaza-Raez, C. (2004) "Coding for Energy Efficient Wireless Embedded Networks". International Workshop on Wireless Ad-Hoc Networks-IWWAN, Oulu, Finland.
- [5] Desset, C., Fort, A. (2003) "Selection of Channel Coding for Low-Power Wireless Systems", The 57th IEEE Semiannual Vehicular Technology Conference (VTC-Spring), Leuven, Belgium.
- [6] Seada, K., Zuniga, M., Helmy, A., Krishnamachari, B. (2004) "Energy-Efficient Forwarding Strategies for Geographic Routing in Lossy Wireless Sensor Networks", Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems (Sensys'04), Baltimore, Maryland, USA.
- [7] McEliece, R. J., Lin, W. (1996) "The Trellis Complexity of Convolutional Codes". IEEE Transactions on Information Theory, vol. 42, no. 6, November, 1996.
- [8] Lin, S., Costello, D. "Error control coding: fundamentals and applications." Englewood Cliffs: Prentice-Hall, 1983.
- [9] Rappaport, T. S. "Wireless communications: principles and practice", Upper Saddle River: Prentice Hall PTR, 1996.
- [10] Wicker, S. B., "Error control systems for digital communication and storage". Upper Saddle River: Prentice Hall, 1995.
- [11] Viterbi, A. J., "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm". IEEE Transactions on Information Theory, vol. IT-13, pp. 260-269, April 1967.
- [12] Hendrix, Henry. Texas Instruments Application Report SPRA071A. January 2002. available <http://focus.ti.com/docs/apps/catalog/resources/appnoteabstract.jhtml?abstractName=spra071a>
- [13] TinyOS 2.0 Documentation. <http://www.tinyos.net/tinyos-2.x/doc/html/tep111.html>
- [14] Proakis, John G., "Digital Communications". Third Edition. McGraw-Hill International Editions, Singapore, 1995.
- [15] Atmel Corporation. [http://www.atmel.com/dyn/products/product\\_card.asp?part\\_id=2018](http://www.atmel.com/dyn/products/product_card.asp?part_id=2018).
- [16] Crossbow Technology, Inc. <http://www.xbow.com/Products/productsdetails.aspx?sid=62>
- [17] Chipcon Products from Texas Instruments [http://www.chipcon.com/index.cfm?kat\\_id=2&subkat\\_id=12&dok\\_id=14](http://www.chipcon.com/index.cfm?kat_id=2&subkat_id=12&dok_id=14)