UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

Carlos Giovanni Nunes de Carvalho

Correlação Espaço-Temporal Multivariada na Melhoria da Precisão de Predição para Redução de Dados em Redes de Sensores sem Fio

FORTALEZA
Março 2012

CARLOS GIOVANNI NUNES DE CARVALHO

CORRELAÇÃO ESPAÇO-TEMPORAL MULTIVARIADA NA MELHORIA DA PRECISÃO DE PREDIÇÃO PARA REDUÇÃO DE DADOS EM REDES DE SENSORES SEM FIO

Tese apresentada ao Curso de Doutorado em Engenharia de Teleinformática do Centro de Tecnologia da Universidade Federal do Ceará, como parte dos requisitos para obtenção do título de Doutor em Engenharia de Teleinformática. Área de concentração: Sinas e Sistemas.

Orientador: Prof. Dr. José Neuman de Souza

Coorientador: Prof. Dr. Danielo Gonçalves Gomes

FORTALEZA

Março 2012

Dados Internacionais de Catalogação na Publicação

Universidade Federal do Ceará

Biblioteca do Centro de Tecnologia

Tese (doutorado) – Universidade Federal do Ceará, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia de Teleinformática, Fortaleza, 2012.

Área de Concentração: Sinais e Sistemas.

Orientação: Prof. Dr. José Neuman de Souza.

Coorientação: Prof. Dr. Danielo Gonçalves Gomes

CARLOS GIOVANNI NUNES DE CARVALHO

CORRELAÇÃO ESPAÇO-TEMPORAL MULTIVARIADA NA MELHORIA DA PRECISÃO DE PREDIÇÃO PARA REDUÇÃO DE DADOS EM REDES DE SENSORES SEM FIO

Tese apresentada ao Curso de Doutorado em Engenharia de Teleinformática do Centro de Tecnologia da Universidade Federal do Ceará, como parte dos requisitos para obtenção do título de Doutor em Engenharia de Teleinformática. Área de concentração: Sinas e Sistemas.

Orientador: Prof. Dr. José Neuman de Souza
Coorientador: Prof. Dr. Danielo Gonçalves Gomes

Aprovada em: ___/___/_____.

BANCA EXAMINADORA

_____

Prof. Dr. José Neuman de Souza (Orientador)

Universidade Federal do Ceará (UFC)

_____

Prof. Dr. Danielo Gonçalves Gomes (Coorientador)

Universidade Federal do Ceará (UFC)

_____

Prof. Dr. Augusto José Venâncio Neto

Universidade Federal do Ceará (UFC)

_____

Prof. Dr. Angelo Roncalli Alencar Brayner

Universidade de Fortaleza (UNIFOR)

_____

Prof. Dr. Stênio Flávio de Lacerda Fernandes

Universidade Federal de Pernambuco (UFPE)

A Deus.

À minha esposa Fabíola e ao nosso filho David.

AGRADECIMENTOS

À Fundação de Amparo à Pesquisa do Estado do Piauí – FAPEPI, pelo apoio financeiro com concessão da bolsa de doutorado.

À Universidade Estadual do Piauí – UESPI, pela autorização de meu afastamento de forma remunerada.

À Professora Rossana Maria de Castro Andrade, por ter aberto as portas do GREat/UFC para que eu pudesse trabalhar no doutorado, desde a seleção até a defesa.

Aos Professores José Neuman de Souza e Danielo Gonçalves Gomes, pela oportunidade de me aceitar como orientando e co-orientando, respectivamente, e ter acreditado em meu trabalho.

Aos Professores Ahmed .Karmouch e Stênio Flávio de Lacerda Fernandes, por terem me recebido em Ottawa-ON (Canadá) e me orientado por um curto, mas valoroso período de 6 meses.

Aos Professores Augusto José Venâncio Neto e Angelo Roncalli Alencar Brayner, por ter participado e colaborado com a minha defesa.

Aos Amigos do GREat, especialmente a Atslands do Rego, os quais trocamos experiências durante os quatro anos de curso.

Aos meus familiares, os quais deram muito apoio, possibilitando que nós (eu, Fabíola e David) conseguíssemos morar em Fortaleza-CE e depois em Ottawa-ON.

RESUMO

A predição de dados não enviados ao sorvedouro é uma técnica usada para economizar energia em RSSF através da redução da quantidade de dados trafegados. Porém, os dispositivos devem rodar mecanismos simples devido as suas limitações de recursos, os quais podem gerar erros indesejáveis e isto pode não ser muito preciso. Este trabalho propõe um método baseado na correlação espacial e temporal multivariada para melhorar a precisão da predição na redução de dados de Redes de Sensores Sem Fio (RSSF). Simulações foram feitas envolvendo funções de regressão linear simples e regressão linear múltipla para verificar o desempenho do método proposto. Os resultados mostram um maior grau de correlação entre as variáveis coletadas em campo, quando comparadas com a variável tempo, a qual é uma variável independente usada para predição. A precisão da predição é menor quando a regressão linear simples é usada, enquanto a regressão linear múltipla é mais precisa. Além disto, a solução proposta supera algumas soluções atuais em cerca de 50% na predição da variável umidade e em cerca de 21% na predição da variável luminosidade.

Palavras-chave: Redução de dados. Precisão da predição. Rede de Sensores Sem Fio.

ABSTRACT

Prediction of data not sent to the sink node is a technique used to save energy in WSNs by reducing the amount of data traffic. However, sensor devices must run simple mechanisms due to its constrained resources, which may cause unwanted errors and this may not be very accurate. This work proposes a method based on multivariate spatial and temporal correlation to improve prediction accuracy in data reduction for Wireless Sensor Networks (WSN). Simulations were made involving simple linear regression and multiple linear regression functions to assess the performance of the proposed method. The results show a higher correlation between gathered inputs when compared to variable time, which is an independent variable widely used for prediction and forecasting. Prediction accuracy is lower when simple linear regression is used, whereas multiple linear regression is the most accurate one. In addition to that, the proposed solution outperforms some current solutions by about 50% in humidity prediction and 21% in light prediction.

Keywords: Data reduction. Prediction accuracy. Wireless Sensor Network.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

1.1 Contextualização da problemática em foco

As Redes de Sensores Sem Fio (RSSF) são exemplos de redes limitadas de recursos, as quais os recursos de processamento, armazenamento e energia são escassos. As informações são coletadas por nós sensores e enviadas ao nó sorvedouro da rede salto a salto (*hop-by-hop*). Porém, o tráfego de dados gerado por estes encaminhamentos exige um consumo de energia incompatível com as limitações das RSSF (GAMA e GABER, 2007; VURAN *et al*, 2004; WANG, 2010).

Entre as informações que um sensor pode coletar em campo, destacamos os mais comuns, que são luminosidade, temperatura, umidade, pressão barométrica, velocidade, aceleração, acústica e campo magnético (AKYILDIZ *et al*, 2002). Dependendo da aplicação, as implantações podem ser terrestres, subterrâneas, subaquáticas, multimídia ou móvel, onde um sistema de monitoramento pode utilizar uma ou a combinação delas.

Os sensores podem ser usados em muitas aplicações tais como detecção de evento, localização e em particular, monitoramento e controle de ambientes (AKYILDIZ *et al*, 2002), em que coletas de dados periódicas geram um grande volume de dados na rede. Nestes cenários, os nós sensores frequentemente enviam os mesmos dados coletados de uma área específica. A sobreposição de informação enviada ao sorvedouro implica em gasto de energia desnecessário e consequentemente, diminuição do tempo de vida da rede. O problema agrava-se conforme o número de nós aumenta (escalabilidade), porque a comunicação de dados é responsável pela maior parte do consumo de energia em RSSF (AKYILDIZ *et al*, 2002; KOSHY *et al*, 2008; TAHIR e FARRELL, 2009).

Uma forma de diminuir o problema do consumo de energia é reduzir o tráfego de dados entre nós sensores. A adoção de alguns mecanismos de redução de dados pode ser inadequada para determinadas aplicações. Portanto, apresentamos alguns exemplos de aplicações (AKYILDIZ *et al*, 2002) e seus requisitos:

a) na detecção e exploração de ataque nuclear, biológico e químico existe a necessidade de detecção precisa, uma vez que a propagação deste tipo de ataque é rápida e envolve vidas;

b) a agricultura de precisão pode monitorar níveis de pesticidas na água, o nível de erosão do solo e o nível de poluição do ar em tempo real, com alto requisito de precisão dos dados. Assim, quanto menor o erro provocado pelo mecanismo de redução de dados, menor a chance de sistema de agricultura provocar desastres naturais no meio ambiente;

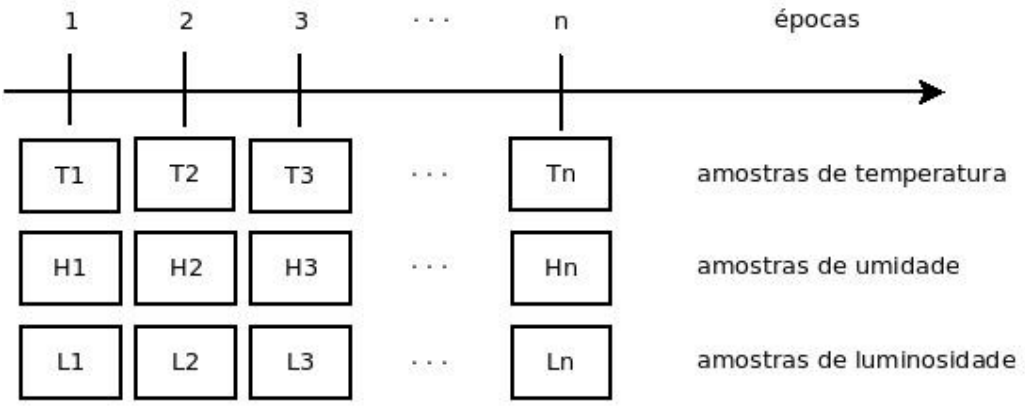
c) a área da saúde demanda alta precisão em operações como a de monitoramento de dados fisiológicos de humanos. O monitoramento e a detecção do comportamento antecipadamente (por exemplo: queda, detectar sintomas pré-definidos) é essencial ao funcionamento deste tipo de aplicação. O rastreamento e monitoramento de pacientes e médicos dentro de um hospital (por exemplo: localização e detecção de taxas de batimentos cardíacos e pressão do sangue) também envolvem dados com informações vitais ao paciente, demandando confiabilidade do sistema. A administração de remédios no hospital (por exemplo: evitar a administração errada de medicamentos e o consumo de medicamentos que produzam reações alérgicas no paciente) é outro exemplo de aplicação que exige cuidados na adoção do mecanismo de redução de dados;

d) os sistemas de controle do ambiente em prédios requerem precisão para controlar o fluxo de ar e a temperatura em diferentes partes. O nível de precisão não pode ser negligenciado a ponto de causar desconforto às pessoas e prejudicar o ambiente;

e) aplicações de localização como de museus interativos, alarmes contra roubo de carros, rastreamento e monitoramento de veículos, e controle de inventário requerem baixo nível de precisão, pois são sistemas que não envolvem vidas, demandando menos precisão quando comparados a aplicações como as de saúde e militares.

O sistema de monitoramento de ambiente convencional (Figura 1) funciona da seguinte forma. Cada nó sensor coleta amostras de uma variável (tal como temperatura) e envia-as ao sorvedouro em cada ciclo (época). A frequência de envio das informações afetará diretamente o tráfego de dados, gerando sobreposição de informações à medida que a rede fica mais densa.

Figura 1. Operação do sistema de monitoramento.



Cada nó sensor coleta amostras de uma variável (tal como temperatura) e envia-as ao sorvedouro em cada ciclo

Para os cenários de monitoramento de ambientes, tais como monitoramento de vegetação e clima, um protocolo de comunicação eficiente precisa ser implantado, para diminuir o consumo de energia e aumentar o tempo de vida da rede. Isto é possível explorando a correlação que existe entre os nós sensores envolvidos no sistema de monitoramento.

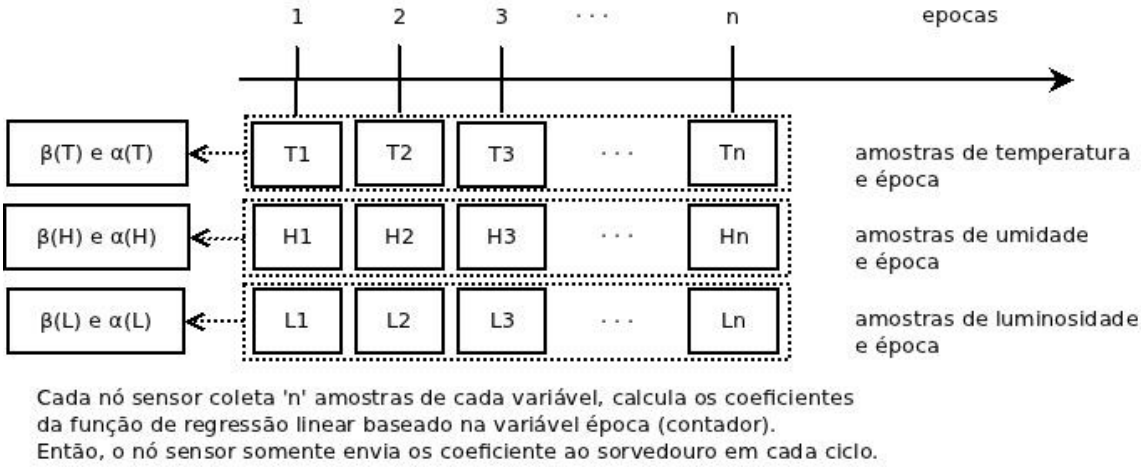
A correlação entre os dados coletados por um nó sensor e seus vizinhos, bem como a correlação entre os dados coletados pelo nó sensor sobre um dado tempo (VURAN *et al*, 2004) devem ser exploradas por protocolos eficientes para melhorar o consumo de energia. Normalmente, técnicas de redução de dados contemplam mecanismos que suportam a exploração das correlações. Estas correlações são conhecidas como correlação espacial e temporal, respectivamente. Quando mais de uma variável na correlação é levada em conta, a abordagem é chamada de correlação multivariada.

Para explorar a correlação temporal, ou seja, um conjunto de amostras de um mesmo nó sensor relacionadas no tempo, pode ser usada à predição de dados, a qual reduz o tráfego de dados ao sorvedouro. Isto tem sido adotado em vários trabalhos na literatura (GOEL e IMIELINSKI (2001); XU e LEE (2003); MATOS *et al*, 2010; JIANG *et al.* 2011). Ela ajuda a reduzir o consumo de energia geral da rede, gerando modelos de dados os quais são enviados ao invés das próprias amostras.

Em alguns casos, um algoritmo é embutido dentro do nó sensor para calcular os coeficientes de uma função de regressão linear. Estes coeficientes são chamados de β e α e

representam o modelo de dados da variável coletada pelo nó sensor, tal como a temperatura, através de uma sequência de amostras. Quando β e α chegam ao sorvedouro, eles são usados pela função de regressão linear embutida no sorvedouro para predição. Então, a sequência de amostras é gerada através da predição conforme o modelo de dados obtido pelo sorvedouro (Figura 2).

Figura 2. Operação do sistema de monitoramento baseado em predição através de regressão linear simples.



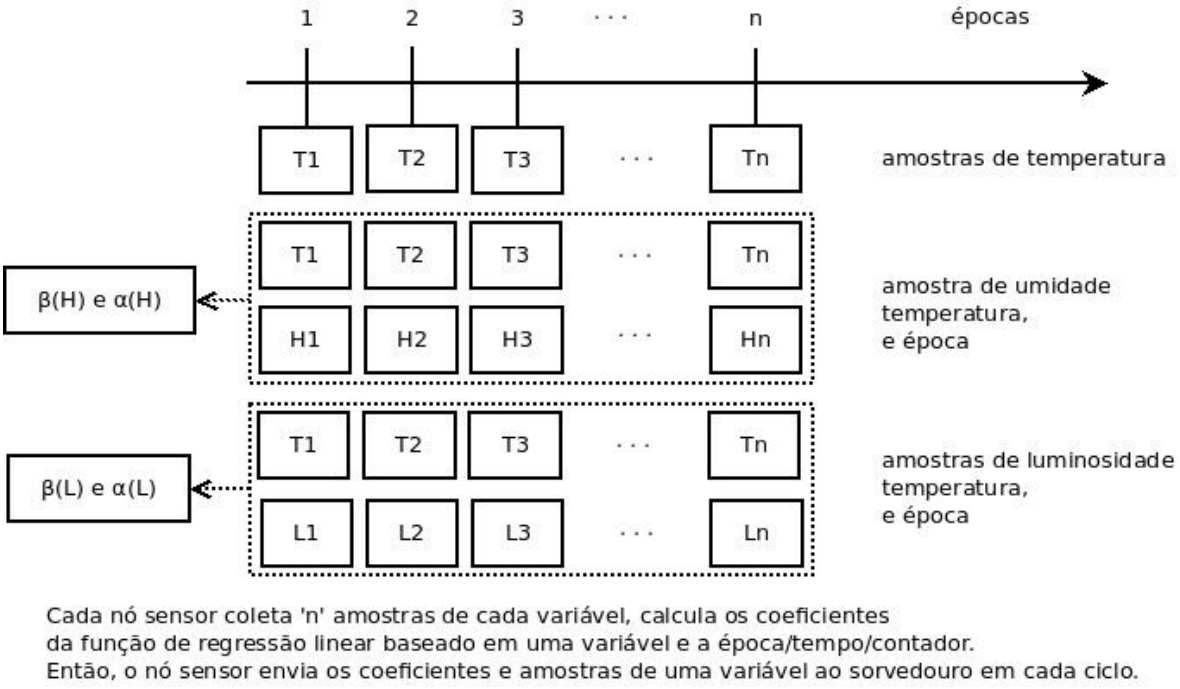
A abordagem usando regressão linear simples leva em conta a correlação de somente uma variável a ser predita (chamada variável dependente, por exemplo: a temperatura) e somente uma variável para predizer a variável dependente (chamada variável independente, por exemplo: época). Porém, a variável época não é a variável mais correlacionada que outras, tais como temperatura, umidade e luminosidade. Assim, a predição adotada por soluções baseada na regressão linear simples, nestas situações, não é precisa.

Consequentemente, as questões que nós abordamos aqui são: “nós podemos usar a correlação entre as variáveis coletadas pelo mesmo nó sensor, para melhorar a precisão da predição?” e “a predição multivariada consome mais energia que a predição univariada para alcançar níveis melhores de precisão?”. O uso de análise de correlação para identificar quais variáveis são mais adequadas como variável independente em uma função de regressão linear e o melhoramento de precisão de predição são discutidos por HAIR *et al* (1998).

Nossa hipótese é que a regressão linear seja usada para executar predição de dados, baseado na correlação multivariada. Em nosso método, nós levamos em conta a correlação entre duas leituras de dados coletados pelo nó sensor e também a variável época

(Figura 3). Nosso método é diferente da abordagem usando regressão linear simples, a qual usa a correlação entre uma variável coletada e a variável época (contador).

Figura 3. Operação do sistema de monitoramento baseado em predição, proposto neste trabalho (regressão linear múltipla).



1.2 Motivação e justificativa

Aumentar a precisão da predição na redução de dados é importante para que tenhamos soluções com menor consumo de energia, mas que sejam capazes de recriar no sorvedouro, as amostras coletadas em campo, com o menor erro possível. Como toda redução de dados gera perda de informação, quanto mais preciso for o mecanismo de predição, mais próximo os dados estarão dos valores coletados em campo. Portanto, diminui o erro para a aplicação do usuário, a qual pode exigir mais dos requisitos de precisão das informações.

O principal desafio é usar um mecanismo de redução de dados que seja preciso, sem exigir complexidade de implementação e implantação em RSSF. Existem vários mecanismos de redução de dados, mas os mais simples, normalmente são mais imprecisos. A complexidade pode inviabilizar a adoção destes mecanismos em dispositivos que já são limitados de recurso.

Os mecanismos utilizados para redução de dados em RSSF normalmente são complexos ou negligenciam a precisão dos dados reconstruídos no sorvedouro. Portanto, nós estudamos o funcionamento de um mecanismo simples, usando regressão linear simples para predizer dados não enviados ao sorvedouro, a qual reduz a quantidade de transmissão e recepção dos dados coletados em campo. Este mecanismo serve de *benchmark* para que possamos comparar a nossa proposta.

A opção de melhorar um mecanismo simples é importante para que nossa solução seja viável em RSSF. Além de melhorar o mecanismo simples, adotando uma técnica estatística mais eficiente na precisão, nós também exploramos a sobreposição espacial dos dados coletados pelos nós sensores vizinhos, aplicando um mecanismo de detecção da correlação destes dados baseado na distância Euclidiana. A distância Euclidiana permite verificar o quanto um vetor está próximo a outro, ou seja, em nosso caso, o quanto correlacionado está os coeficientes multivariados de um nó sensor com seu vizinho.

1.3 Objetivos

O objetivo principal desta tese é propor uma solução de redução de dados que não seja complexa de implementar e implantar, e que seja mais precisa que uma solução usando regressão linear simples.

Diante disto, os objetivos específicos para alcançarmos o objetivo principal são:

(i) investigar o mecanismo de redução de dados baseado em regressão linear simples;

(ii) realizar experimentos com este mecanismo para identificarmos o nível de precisão;

(iii) explorar as correlações espaciais e temporais para auxiliar na diminuição do consumo de energia nas RSSF;

(iv) propor uma solução que seja simples e ao mesmo tempo, que seja capaz de melhorar a precisão do mecanismo identificado anteriormente.

1.4 Publicações relacionadas à tese

Nós publicamos recentemente dois artigos que descrevem nossos avanços com a solução proposta, um deles foi aprovado e apresentado no LANOMS2011 em Quito

(CARVALHO *et al*, 2011) e o outro no periódico Sensors (CARVALHO *et al*, 2008b). O primeiro artigo, intitulado “Multiple linear regression to improve prediction accuracy in WSN data reduction”, mostra a fase inicial de nossos experimentos, a qual nossa solução, embora gaste mais energia que a solução baseada em regressão linear simples, melhora a precisão de predição. No segundo artigo, intitulado “Improving Prediction Accuracy for WSN Data Reduction by Applying Multivariate Spatio-Temporal Correlation”, nós detalhamos os experimentos e assim, provamos que é possível implementar a nossa solução em RSSF.

1.5 Estrutura da tese

Esta tese é organizada em seis capítulos, conforme descrito a seguir. No Capítulo 2, são descritos os conceitos utilizados neste trabalho, através de uma revisão bibliográfica e posteriormente são discutidos os trabalhos correlatos, os quais auxiliaram na pesquisa desenvolvida. A solução proposta para o melhoramento da precisão da predição em RSSF é mostrada no Capítulo 3, abordando o mecanismo usado, como são exploradas as correlações espaciais e temporais, e como acontece a recuperação dos dados. Os materiais e métodos são abordados no Capítulo 4, detalhando os experimentos realizados. O Capítulo 5 descreve os resultados obtidos nos experimentos e suas análises. Encerramos no Capítulo 6 com as conclusões e os direcionamentos para os trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA E TRABALHOS CORRELATOS

Neste capítulo, inicialmente é destacada a necessidade de usar mecanismos de redução de dados em RSSF através de um exemplo. Também são abordadas as algumas técnicas e trabalhos relacionados à redução de dados. Em seguida, discutimos como o coeficiente de Pearson (r) é usado para identificar o grau de correlação entre duas variáveis e como é explorada a correlação espacial e temporal de dados em RSSF.

2.1 Redução de dados em RSSF

O problema crucial em RSSF é o consumo de energia. Na maioria dos dispositivos, o tempo de vida do nó sensor depende do tempo de vida da bateria, o qual é fundamental neste tipo de rede. Portanto, vários mecanismos de redução de dados (tais como compressão e agregação) são propostos para tentar diminuir os problemas de consumo de energia. Mas, outro problema surge ao realizarmos redução de dados, que é o erro causado por tais mecanismos.

Alguns dispositivos possuem fontes de energia constante ou alternativa, como sensores fixos em paredes alimentados por tomadas ou sensores alimentados por energia solar, mas nas soluções de RSSF é comum o uso de pilhas alcalinas do tipo AA. Portanto, os dispositivos são equipados com fonte de energia limitada (<0,5 Ah; 1,2 V) (AKYILDIZ *et al*, 2002, 2004). Dessa forma, o uso de mecanismos de redução de dados é imprescindível. O consumo de energia pode ser dividido em três domínios: sensoriamento, comunicação e processamento de dados.

A energia gasta com sensoriamento varia de acordo com a natureza da aplicação, pois o sensoriamento esporádico (por exemplo: coleta de amostras de temperatura do ambiente) consome menos energia que o monitoramento de evento constante (por exemplo: rastreamento de objetos). Outro fator que influencia o gasto de energia é a complexidade da detecção do evento, pois um ambiente com maior nível de ruído pode gerar muitos dados corrompidos, demandando um algoritmo de detecção mais complexo.

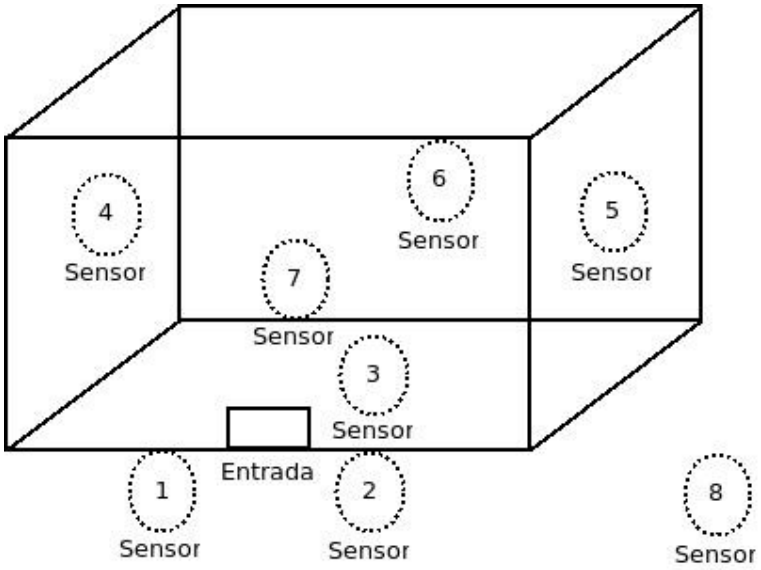
O maior gasto de energia é com a comunicação de dados, ou seja, na transmissão e recepção. Em situações de comunicação com curtas distâncias e baixo poder de radiação (aproximadamente 0 dbm), os custos de transmissão e recepção são próximos (AKYILDIZ *et*

al, 2002, 2004). Por isso, as soluções de redução de dados têm focado em reduzir o tráfego entre dispositivos.

O custo de processamento dos dados é bem menor comparado à comunicação. O custo de transmitir 1 KB a uma distância de 100 m é aproximadamente igual à execução de três milhões de instruções por um processador de 100 milhões de instruções por segundo (MIPS)/W (AKYILDIZ *et al*, 2002, 2004).

Nós concentramos nosso trabalho na redução de dados trafegados na rede, pois assim economizamos energia na comunicação, onde o problema das RSSF é mais acentuado. A redução de dados é necessária para aplicações de RSSF, as quais têm grande quantidade de dados para enviar sobre a rede. Principalmente as aplicações de dados de fluxo contínuo, as quais geram um grande volume de dados para serem encaminhados do nó sensor até chegar ao sorvedouro, passando por nós intermediários (*relays*).

Figura 4. Exemplo de sistema de monitoramento: colmeia de abelhas.



As aplicações de monitoramento de ambientes produzem uma ou mais leituras de variáveis como temperatura, umidade e luminosidade a cada tempo determinado pelo usuário. Supondo que estamos monitorando o comportamento de colmeias de abelhas para saber o motivo do abandono de enxames, o qual é comum em regiões que ocorrem alterações drásticas no clima, fizemos uma comparação para termos a ideia do impacto do tráfego de dados na rede.

De acordo com o trabalho de MEITALOVA *et al* (2009), para medir o microclima dentro da colmeia são necessários oito nós sensores por colmeia, coletando dados de temperatura e umidade relativa do ar em tempo real (Figura 4). Os nós sensores 1 e 2 são implantados na entrada da colmeia, os nós sensores de 3 a 7 são implantados nas paredes internas da colmeia e o nó sensor 8 é implantado fora da colmeia.

As medições são feitas dentro e fora da colmeia para ser possível perceber a atividade das abelhas de controlar o clima dentro, ajustando às condições climáticas de fora. Se a cada segundo as leituras forem enviadas ao sorvedouro, teremos duas variáveis por segundo para serem gravadas em um banco de dados no computador do usuário.

Para garantir melhor precisão das leituras de temperatura e umidade relativa do ar, é comum o uso de variáveis de 16 bits cada uma. Então, se usarmos o TinyOS 2.x[1] (sistema operacional de sensores que permite até 20 bytes para informações do usuário) neste exemplo, o pacote de dados terá 32 bits (4 bytes) de dados das variáveis para enviar a cada segundo por nó sensor, ou seja, 32 bits/s são trafegados por cada nó sensor. Se temos 8 nós sensores para realizar a tarefa, consequentemente 256 bits/s (32 bytes/s) de dados do usuário serão enviados da origem para a rede.

Desconsiderando a necessidade de *relays* para encaminhar os dados, devido à proximidade das origens com o sorvedouro, o consumo de energia da rede para enviar as amostras das duas variáveis do nosso exemplo será de 53,44 μJ a cada segundo, conforme o modelo de energia proposto por Jurdak *et al.* (1,67 μJ/Byte enviando) (JURDAK *et al*, 2008). Como algumas técnicas de redução de dados normalmente atuam somente nos dados da aplicação, fizemos as contas somente com base no tamanho dos dados das variáveis.

Agora, consideramos que um nó sensor use duas pilhas AA, as quais têm a capacidade de 0,6 J (volt = energia/corrente) por uma hora cada uma, o tempo de vida útil dessa rede, levando em conta somente os dados da aplicação (na verdade existem mais informações a serem trafegadas pelas outras camadas da tecnologia), será algo em torno de 50 horas (2% da energia da pilha gasta a cada 1 hora de coleta da rede).

Como vimos no exemplo acima, a aplicação gastará toda a energia das duas pilhas a cada 50 horas. Se levarmos em consideração um ambiente de experimento real, as pilhas gastarão em menos tempo, pois não contabilizamos os outros bytes provenientes das

[1] Site Web do TinyOS 2.x em http://www.tinyos.net/tinyos-2.x/doc/

informações de roteamento e embora a rede seja pequena, os nós sensores vizinhos recebem por difusão, os dados enviados por outros nós sensores, consumindo mais energia (recepção de dados). Portanto, teremos que trocar todas as pilhas da rede em menos de 50 horas. No nosso exemplo já são 16 pilhas para serem trocadas, mas quando aumentamos à quantidade de nós sensores, a manutenção da rede se tornará inviável.

2.1.1 Técnicas mais comuns de redução de dados

A redução de dados, segundo ANASTASI *et al.* (2009), é uma das abordagens de conservação de energia, a qual é baseada nos dados do usuário. Em sua taxonomia, as técnicas de redução de dados baseadas nos dados do usuário são divididas em processamento intra-rede, compressão de dados e predição de dados. Todas estas técnicas têm o objetivo de reduzir a quantidade de dados enviados ao sorvedouro, embora os princípios de cada uma sejam diferentes.

2.1.1.1 Processamento intra-rede

O processamento intra-rede consiste em executar agregação de dados em nós intermediários, diminuindo a quantidade de dados enquanto os mesmos trafegam na rede. Esta técnica depende da aplicação a ser implantada. Na agregação, o dado é coletado de múltiplos sensores e combinados juntos para transmitir ao sorvedouro. O dado agregado é mais importante que as amostras de leituras individuais, pois ele é encaminhado após a junção e representa um conjunto de leituras.

Normalmente esta técnica é usada em abordagens baseadas em nó central (*cluster*). A agregação tem a desvantagem de diminuir o tráfego somente no encaminhamento dos dados após a junção em um *cluster*, ou seja, no momento do envio dos dados individualmente (de cada nó sensor) ao nó central, o consumo de energia na transmissão equivale ao de uma solução sem redução de dados neste mesmo momento. Outro problema está relacionado à perda da informação após a junção, a qual não pode ser desprezada pelo sorvedouro.

2.1.1.2 Compressão de dados

A compressão é uma técnica de redução de dados que acontece normalmente antes da transmissão da informação do nó sensor ao próximo nó da rede, para reduzir o tamanho das amostras coletadas. Neste caso, o nó sensor é capaz de codificar as amostras e a

decodificação ocorre no sorvedouro para a descompressão. Esta técnica requer alto poder de processamento e maior complexidade de implantação, sendo necessário adapta-las para suportar as limitações dos dispositivos, inviabilizando a adoção em alguns cenários de RSSF.

2.1.1.3 Predição de dados

A predição de dados gera uma abstração das amostras coletadas através de um modelo, o qual descreve a evolução dos dados. O modelo é capaz de predizer as amostras com certo grau de precisão. Neste caso, o nó sensor normalmente gera coeficientes para descrever o modelo e envia-os ao sorvedouro, ao invés de enviar as amostras. Um problema crítico para este tipo de técnica é que os coeficientes não podem ser perdidos no encaminhamento ao sorvedouro, pois eles representam um conjunto de amostras. Além disto, o nó sensor tem que ter a capacidade de armazenar temporariamente as leituras que serão processadas, antes de realizar a compressão.

Seguindo a taxonomia proposta por ANASTASI *et al.* (2009), a predição de dados pode ser dividida em: abordagens estocásticas, previsão de séries temporais e abordagens algorítmicas.

a) abordagens estocásticas – são baseadas nas propriedades probabilísticas e/ou estatísticas do fenômeno envolvido, ou seja, é possível mapear os dados em um processo randômico descrito através de uma função de densidade da probabilidade.

b) previsão de séries temporais – um histórico das amostras coletadas por um nó sensor por um determinado período de tempo pode ser usada para prever as amostras futuras. Após o modelo dos dados ser obtido, as previsões podem ser feitas com uma margem predefinida de erro.

c) abordagens algorítmicas – uma heurística ou um modelo de transição de estados descreve o fenômeno observado. Estas abordagens são dependentes da aplicação, sedo necessário estudas cada caso para implantá-las.

2.2 Correlação espacial e temporal de dados em RSSF

Várias técnicas têm sido definidas para otimizar o consumo de energia em aplicações para reduzir dados enviados ao sorvedouro. As mais comuns são compressão e agregação (YICK *et al,* 2008; GAMA e GABER, 2007; AKYILDIZ *et al,* 2002). Tais

técnicas são normalmente usadas sem levar em conta a correlação espacial e temporal multivariada das leituras coletadas pelos nós sensores em campo. Porém, muitos nós sensores implantados em campo são normalmente capazes de monitorar mais que uma variável, e são assim chamados multisensores.

No trabalho de VURAN *et al*. (2004), os autores definiram a teoria da correlação de dados voltada para RSSF e realizaram testes para demonstrar que devido a alta densidade na topologia da rede, os dados coletados pelos nós sensores são altamente correlacionadas no espaço de domínio. Além disto, eles observaram que a natureza do fenômeno físico constitui a correlação temporal entre cada observação consecutiva de um nó sensor, pois dependendo da aplicação, teremos ou não correlação entre as variáveis coletadas em campo.

Para VURAN *et al.* (2004), as correlações espaciais e temporais, e a natureza colaborativa das RSSF, devem ser aproveitadas para a implantação de protocolos de comunicação mais eficientes. A correlação espacial normalmente ocorre quando as aplicações de RSSF requerem implantação de nós sensores em um cenário denso espacialmente, para alcançar uma cobertura satisfatória do evento a ser monitorado.

A implantação de RSSF usando topologia randômica é forte candidata à sobreposição espacial de dados, pois os nós sensores são normalmente jogados em áreas próximas ou dentro do evento. Com isso, múltiplos nós sensores capturam e enviam ao sorvedouro a mesma informação sobre um único evento. Devido a esta alta densidade de nós sensores, os dados coletados espacialmente próximos, são altamente correlacionadas, com o grau de correlação aumentando à medida que se diminui a separação entre os nós sensores.

Quanto à correlação temporal, ela ocorre em várias aplicações de RSSF, tais como rastreamento de evento e monitoramento de ambientes, podendo requerer nós sensores para periodicamente executar coletas de dados e em seguida, transmitir as características do evento monitorado. Além disso, as variáveis coletadas podem ter um alto grau de correlação entre si. No exemplo do monitoramento de ambientes, as variáveis temperatura e umidade normalmente são inversamente correlacionadas, onde quando uma aumenta seu valor, ao outra diminui.

Claro que em algumas situações há uma alta discrepância, como por exemplo, no monitoramento da luminosidade em ambientes fechados, como prédios. A luminosidade é alterada com a simples presença de uma pessoa próxima ao dispositivo sensor ou até mesmo o

procedimento de acender ou apagar as luzes da sala. Este comportamento das leituras das variáveis leva a resultados de correlação baixos.

Nós encontramos trabalhos tais como o de SKORDYLIS *et al.* (2006) o qual usa a técnica adotada para redução de dados correlacionado espacialmente através do coeficiente de Pearson (r). Também, nós encontramos trabalhos tais como o de MATOS *et al.* (2010) o qual usa uma técnica adotada para redução de dados correlacionados temporalmente por regressão linear simples.

Contudo, pelo melhor de nosso conhecimento, não existe outro trabalho de redução de dados distribuído (localmente em cada nó sensor) que usa regressão linear múltipla para executar predição e distância Euclidiana para verificar a correlação entre leituras de nós sensores vizinhos. A solução proposta se beneficia da correlação multivariada para aumentar a precisão da predição e consequentemente diminuir o erro no processo de redução de dados.

2.2.1 Coeficiente de Pearson

O coeficiente de Pearson [Equação (1)] é usado para identificar a correlação espacial da mesma variável entre dois nós sensores (SKORDYLIS *et al*, 2006), por exemplo, a correlação entre as amostras de temperatura de dois nós sensor vizinhos. Mas, ele também pode ser usado para identificar a correlação temporal entre duas variáveis do mesmo nó sensor e assim descobrirmos qual o nível de correlação entre as variáveis coletadas em campo pelo dispositivo.

$$r_{X_1,X_2} = \frac{\sum\left(x_{1_i} - \bar{X}_1\right) * \left(x_{2_i} - \bar{X}_2\right)}{\sqrt{\sum(x_{1_i} - \bar{X}_1)^2 * \sum(x_{2_i} - \bar{X}_2)^2}} \quad (1)$$

onde r_{X_1,X_2} representa o relacionamento entre dois vetores unidimensionais X_1 e X_2, para ser comparado em termos de suas correlações. Eles contêm janelas de amostras de duas variáveis, $X_1 = x_{1_1}, \dots, x_{1_i}$ e $X_2 = x_{2_1}, \dots, x_{2_i}$, onde $i = 1, \dots, n$ é o número de amostras. $\overline{X_1}$ e $\overline{X_2}$ representam a média de amostras de cada vetor de variável.

O coeficiente de Pearson (r) mede o grau de relação entre dois vetores unidimensionais e seus resultados podem ser entre -1 e 1 (números reais, exemplo: $0{,}9$ é altamente correlacionado e $-0{,}9$ também é altamente correlacionado e 0 é pouco

correlacionado). Existe relacionamento linear perfeito (dois vetores aumentam ou diminuem seus valores) quando o valor de correlação é 1. Por outro lado, existe um relacionamento linear perfeito inverso (um vetor aumenta seus valores quando o outro diminui seus valores) quando o valor de correlação é -1. Não existe relacionamento linear entre dois vetores se o valor de correlação é 0 (zero).

Portanto, quando o coeficiente de Pearson (r) está próximo ao maior ou menor valor (1 or-1), então a correlação entre os vetores é alta. Assim, nós podemos calcular a correlação espacial e temporal das leituras de apenas uma variável entre dois nós sensores (SKORDYLIS *et al*, 2006). O problema é que nós não podemos calcular a correlação espacial multivariada usando este método [Equação (1)], a qual é necessária para nossa proposta. Porém, na próxima seção mostramos como a distância Euclidiana é usada para identificar a correlação espacial multivariada em nossa proposta.

Além disto, nós podemos gerar uma tabela a qual determina quanto uma variável está relacionada à outra. A tabela de correlação para variáveis de um *trace* de dados reais é mostrada na próxima seção. O coeficiente de Pearson (r) é usado para identificar qual variável é mais correlaciona em relação às outras. Esta variável altamente correlacionada é usada para calcular os coeficientes β e α da regressão linear múltipla e também para recuperar dados no sorvedouro aos quais não são enviados.

2.3 Trabalhos correlatos

A redução de dados em RSSF foi pioneiramente discutida por GOEL e IMIELINSKI (2001). Eles aplicaram os conceitos de compressão MPEG para reduzir dados e consequentemente diminuir o consumo de energia em RSSF. O trabalho deles propôs um mecanismo de monitoramento baseado em predição, chamado PREMON, o qual abstrai o fluxo de dados enviado pelos nós sensores ao sorvedouro, como se fosse um fluxo de vídeo codificado pelo padrão MPEG.

A abordagem da solução adotada no PREMON é centralizada, onde o modelo de predição é gerado pelo sorvedouro, o qual tem que ser enviado para todos os nós sensores na rede. Assim, o consumo de energia gerado pela difusão destes modelos de predição tende a aumentar de acordo com o comportamento dos dados coletados e com a quantidade de nós sensores na rede. A cada alteração na correlação das variáveis, é necessário um novo cálculo do modelo de predição e novamente eles serão enviados a cada nó sensor.

Diferente da abordagem centralizada, a abordagem distribuída elimina a necessidade de enviar modelos de predição para os nós sensores. Ao contrário disto, os próprios nós sensores calculam seus modelos de predição. Os algoritmos são embutidos dentro dos nós sensores para reduzir a transmissão de dados ao sorvedouro. Este tipo de abordagem vem sendo adotada em vários trabalhos (LI *et al*, 2010; JIANG *et al*, 2011; LIU *et al*, 2007; SANTINI e ROMER, 2006; SKORDYLIS *et al*, 2006). O problema agora é que mecanismos robustos vêm sendo utilizados para serem executados em dispositivos que são limitados em recurso.

O mecanismo proposto por JIANG *et al*. (2011) para realizar predição obedecendo a um limite predefinido usa abordagem adaptativa. O mecanismo habilita e desabilita a predição de acordo com o seu desempenho. Este tipo de mecanismo é muito útil em cenários onde a correlação constantemente muda entre as variáveis em um determinado tempo. Mas, devido a sua dificuldade de implementação, ainda permanece como uma solução desvantajosa para dispositivos limitados de recursos.

LIU *et al* (2007) propõem um método de coleta de dados onde o mecanismo dinamicamente particiona os nós sensores em clusters, tal que os nós pertencentes ao mesmo cluster contém dados de coleta similares. Eles aplicaram um mecanismo adaptativo, o qual dinamicamente ajusta a taxa de amostragem temporal e espacial de acordo com as mudanças do fenômeno físico monitorado.

Da mesma forma, um mecanismo de predição localizado foi proposto por XU e LEE (2003). O mecanismo é dividido em duas partes: uma arquitetura de rede de sensores localizada, a qual permite os nós sensores entrarem em modo *sleep* e reduz a quantidade de transmissão de longo alcance; e a predição dupla, a qual é um mecanismo que calcula predições de movimentos dos objetos no nó sensor e no seu líder de *cluster*.

Estas duas abordagens acima são um misto de centralizada com a distribuída, caracterizando uma abordagem híbrida. O problema reside no procedimento da escolha dos nós sensores para serem os líderes de clusters, pois tem que ser levado em conta, a correlação entre eles. SKORDYLIS *et al.* (2006) sugerem que antes do desenvolvimento da solução de redução de dados, seja feita a implantação de vários nós sensores no ambiente de monitoramento para que se produzam dados suficientes para realizar um planejamento adequado e eficiente. Com isto, é possível identificar o mecanismo mais adequado para cada aplicação.

Muitos cenários envolvem dados coletados que são altamente correlacionados e até mesmo os mecanismos mais simples, conseguem atingir uma precisão de acordo com os requisitos da aplicação. Por exemplo, o coeficiente de Pearson (*r*) é usado no trabalho de SKORDYLIS *et al.* (2006) para observar as correlações entre duas séries temporais do mesmo nó sensor e também entre duas séries temporais de diferentes nós sensores.

O mecanismo adotado por SANTINI e ROMER (2006) é simples e adaptativo, baseado nos mínimos quadrados (*Least-Mean-Square – LMS*). A desvantagem desta abordagem é que os mecanismos simples normalmente produzem muitos erros de predição, comprometendo a precisão para algumas aplicações. Um mecanismo usando abordagem distribuída baseada em regressão linear simples, porém eficaz nos experimentos envolvendo consultas de banco de dados em RSSF, pode ser observado no trabalho de MATOS *et al.* (2010). Os autores propõem uma solução para reduzir dados nas consultas ao banco de dados usando os dados coletados pelos nós sensores.

O problema da abordagem simples é que a precisão da predição baseada na regressão linear simples depende de somente uma variável, a qual em muitas situações não está correlacionada com qualquer outra variável. A variável época (contador) é normalmente menos correlacionada que as outras variáveis coletadas em campo pelo nó sensor, tais como temperatura, umidade e luminosidade. Portanto, os erros de predição tendem a serem altos, ou seja, menos preciso. As abordagens multivariadas permitem aumentar a precisão da predição baseada em um conjunto de variáveis coletadas, diminuindo o erro gerado pela redução de dados.

Várias técnicas de redução de dados multivariados foram avaliadas por SEO *et al.* (2005). Entre estas técnicas estão métodos baseado em *Wavelet*, amostragem, clusterização hierárquica e *Singular Value Decomposition (SVD)*. As características mais importantes nestas técnicas avaliadas em seu trabalho são:

(i) a técnica baseada em *Wavelet* possui rapidez computacional e pequeno espaço de complexidade, mas normalmente é ineficiente com atributos multivariados;

(ii) a técnica de clusterização hierárquica com árvore de índice multidimensional pode ser usada para redução de dados hierárquicos, como também para rápidas respostas de aproximação para consultas, mas a complexidade da solução pode comprometer a precisão;

(iii) a técnica de amostragem permite um número menor de amostras representarem o conjunto de dados completo. A vantagem é que o custo de obter a amostra é proporcional ao tamanho das amostras. A complexidade da amostragem é potencialmente linear, mas é ineficiente para junção relacional *ad hoc* sobre esquema arbitrário e a efetividade para consultas de aproximação não é clara.

Assim como qualquer abordagem usada para redução de dados em RSSF, SEO *et al* (2005) sugerem que sejam feitas avaliações de desempenho na abordagem multivariada, de acordo com o contexto da aplicação. As aplicações em RSSF produzem grande impacto no desempenho das técnicas de redução de dados, tornando-se fundamental realizar análises antes da implementação. Como as aplicações contêm diferentes requisitos e características de geração de dados, a mesma técnica pode acarretar ganho de desempenho em um tipo de aplicação e perda de desempenho em outro tipo de aplicação diferente.

A Análise de Componentes Principais (PCA) foi aplicada por SILVA *et al.* (2009) para reduzir a dimensionalidade multivariada de dados coletados por nós sensores. Como a abordagem multivariada tende a ser complexa devido a quantidade de variáveis envolvidas, neste caso, o algoritmo identifica as amostras mais significativas e então as envia ao sorvedouro. Para aplicações onde ocorre correlação entre as variáveis por um longo período de tempo, o PCA pode reduzir consideravelmente as amostras reais, caso contrário, o erro da predição tende a aumentar. Dependendo dos requisitos de precisão da aplicação, o erro pode gerar imprecisão nos resultados da aplicação.

Os três métodos básicos para realizar fusão de dados são inferência, agregação e estimação (NAKAMURA *et al.*, 2007b; NAKAMURA e LOUREIRO, 2008). Nos métodos de inferência, tais como a inferência Bayesiana e Dempster-Shafer, o objetivo é processar os dados e abstrair conclusões acerca deles. Os métodos de agregação são mais simples e produzem um dado de menor representatividade para reduzir o conjunto de amostras, normalmente baseados na média, máximo e mínimo. Por outro lado, os métodos de estimação (e.g. quadrados mínimos, filtros de média móvel, filtros de Kalman e filtros de partículas) têm o objetivo de estimar o vetor de estado de um processo a partir das amostras.

A estimação através do filtro de média móvel e a inferência de Dempster-Shafer foram usados por NAKAMURA *et al.* (2007a, 2007b) e NAKAMURA *et al.* (2008) em uma solução de roteamento para RSSF tolerante a falhas chamada Diffuse, a qual consegue

detectar de maneira automática a necessidade de reconstrução da topologia de disseminação baseada em predições. O problema destacado pelos autores é a desvantagem da solução deles apresentar um custo computacional não desprezível, devido às operações de ponto flutuante e às características exponenciais das técnicas usadas.

Nós usamos a ideia da predição baseada na regressão linear simples adotada no trabalho de MATOS *et al.* (2010) para desenvolvermos uma solução simples e portanto, servir como nosso *benchmark* na avaliação de desempenho de nossa solução. Como descrito por HAIR *et al* (1998), a regressão linear múltipla utiliza mais de uma variável para predizer outra variável, com um nível maior de precisão, quando comparado à regressão linear simples.

2.4 Características de alguns trabalhos

A correlação espacial e temporal multivariada é assunto chave para resolver problemas de precisão da predição, mas deve ser observado o aumento do consumo de energia através de técnicas de redução de dados mais complexas que a abordagem usando regressão linear simples. Diante disso, nosso trabalho tem a vantagem de executar uma análise de correlação (Tabela 1) de variáveis coletadas pelos nós sensores, antes da predição ser implementada. Também, nós verificamos os efeitos de usar predição baseada em correlação espacial e temporal multivariada em RSSF.

Tabela 1. Comparação das principais características das soluções.

Trabalho	**Principais Características**					
	Topologia	**Correl. Espacial.**	**Correl. Temporal**	**Mecanismo**	**Multivariada**	**Análise Correlação**
Goel e Imielinski (2001)	Centralizada	Sim	Não	MPEG Standard—like	Não	Não
Xu e Lee (2003)	Localizada	Sim	Sim	Predição dupla	Não	Não
Matos *et al.* (2010)	Distribuída	Não	Sim	Regressão Linear Simples	Não	Não
Silva *et al.* (2009)	Distribuída	Não	Sim	Análise dos Principais Componentes	Não	Não
Nossa proposta	Distribuída	Sim	Sim	Regressão Linear Múltipla	Sim	Sim

Detalhes da implementação de nossa solução são destacados, revelando os desafios de embutir regressão linear simples e multivariada em RSSF. Além disto, nós mostramos quando o uso de predição baseada no método de correlação multivariada é mais apropriado, de acordo com os resultados.

2.5 Considerações finais

As abordagens observadas na literatura tendem a usar algoritmos que requerem poder computacional que exigem muito dos recursos limitados das RSSF para melhorar a precisão da predição e com isso, exigir mais da implementação e implantação dos mecanismos de redução de dados. Por outro lado, as abordagens mais simples diminuem os requisitos computacionais, mas normalmente comprometem a precisão da predição, inviabilizando a sua adoção em aplicações sensíveis ao erro.

Aumentar a precisão da predição é um desafio que resolvemos atacar através da regressão linear múltipla. Este mecanismo produz um melhoramento na precisão da predição baseado na correlação entre as variáveis coletadas em campo. Nós resolvemos aproveitar a vantagem da simplicidade da predição baseada na regressão linear simples e aplicarmos um mecanismo estatístico multivariado para melhorar a precisão.

Além de explorar a correlação temporal multivariada através da regressão linear múltipla, nós também aplicamos um mecanismo baseado na distância Euclidiana. Dessa forma, nós exploramos também a correlação espacial multivariada, visto que o coeficiente de Pearson (r) tem limitações para trabalhar com dados multivariados.

3 MELHORAMENTO DA PRECISÃO DO MECANISMO DE REDUÇÃO DE DADOS ATRAVÉS DA PREDIÇÃO MULTIVARIADA

O Capítulo 3 descreve a forma como exploramos o mecanismo de regressão linear simples adotado na literatura e como adicionamos o mecanismo de regressão linear múltipla para aumentarmos a precisão da predição em nossa proposta. Detalhes da solução proposta são apresentados passo a passo, mostrando como nosso mecanismo funciona.

Além disto, apresentamos a aplicação da distância Euclidiana para detectar a presença da correlação espacial multivariada e a forma que usamos a função de regressão linear múltipla para calcular os seus coeficientes. O mecanismo de recuperação dos dados omitidos pelos nós sensores e executado no lado do sorvedouro, é mostrado neste capítulo, onde nós usamos os coeficientes gerados pela função de regressão linear múltipla.

3.1 Visão geral

A predição de dados em RSSF é normalmente usada para recuperar dados perdidos por falhas de transmissão ou de dispositivos nas RSSF. Em nossa proposta, usamos a predição para recuperar os dados omitidos pelo nó sensor, os quais dão origem a coeficientes de uma função de regressão linear múltipla, calculados a partir de um conjunto de amostras. A nossa ideia é usar a predição para reduzir dados a serem transmitidos pelos nós sensores em direção ao sorvedouro. Com isso, diminuímos o consumo de energia na transmissão ou recepção de dados.

As abordagens de predição podem ser classificadas em três categorias, de acordo com a topologia dos mecanismos. Estas abordagens podem ser distribuídas, centralizadas ou híbridas. Na distribuída, o algoritmo do mecanismo de redução de dados é embutido nos nós sensores para diminuir o tráfego de dados entre os nós sensores vizinhos e os *relays*. Na centralizada, também existe uma diminuição no tráfego, principalmente de envio das informações, mas acontece com o algoritmo sendo executado em um determinado nó fora da RSSF e os parâmetros sendo enviados a cada nó sensor para que ele decida transmitir os dados ou não. A híbrida tenta se beneficiar das vantagens das duas abordagens anteriores, com a decisão de transmitir ou não as amostras, sendo feita pelo nó sensor, mas com intervenções baseadas nas informações provenientes dos dados que chegam e são analisados no sorvedouro.

Os algoritmos da abordagem distribuída, normalmente, exigem processamento em dispositivos limitados de recursos, como nos casos dos nós sensores. Outro problema desta abordagem ocorre nos momentos onde a correlação é baixa ou não existe, pois o custo de processamento para redução de dados dentro dos dispositivos pode ser prejudicial (JIANG *et al.* 2011), devido ao compromisso (*trade-off*) entre o custo da técnica adotada e o custo do recurso disponível.

Por outro lado, os algoritmos da abordagem centralizada necessitam informar a todos os nós sensores da rede o que eles devem fazer, normalmente com base em parâmetros do estimador. Assim, no caso da abordagem centralizada, ocorre um aumento na recepção de dados pelos nós sensores, embora consuma menos energia que enviar as amostras.

A abordagem híbrida herda as vantagens e também algumas desvantagens das duas outras, tornando-se inclusive, mais complexa para desenvolver e implementar. Os mecanismos híbridos exigem mais conhecimento do projetista, devido à escolha correta dos nós sensores que irão enviar as informações por um período de tempo ser uma tarefa árdua.

Nossa proposta utiliza a abordagem distribuída, mas minimizando as desvantagens através de técnicas existentes na literatura, como o uso da distância Euclidiana para detectar e evitar a transmissão de dados sobrepostos devido à correlação espacial entre nós sensores e o uso da regressão linear multivariada para evitar a transmissão de dados sobrepostos devido à correlação espacial. O mecanismo de regressão linear múltipla é menos complexo que vários mecanismos adotados na literatura, que requerem uma grande quantidade de dados para serem armazenados e posteriormente processados.

3.2 Mecanismo com abordagem simples (regressão linear simples)

As soluções atuais de redução de dados por meio de regressão linear são executadas usando regressão linear simples baseada nos mínimos quadrados, conforme foi aplicado por MATOS *et al.* (2010). Neste caso, cada nó sensor calcula os coeficientes β e α da função de regressão linear usando uma variável, normalmente época (contador). Então, o nó sensor envia seus coeficientes β e α ao sorvedouro, ao invés de enviar as amostras das leituras. A vantagem desta solução é que o consumo de energia é reduzido, uma vez que somente o modelo das amostras é enviado ao sorvedouro, mas por outro lado, a predição nem sempre é precisa.

Duas versões de aplicações baseada em regressão linear simples foram desenvolvidas por nós para compararmos a avaliação de desempenho de nossa proposta, as quais usam precisão baseada na correlação univariada (regressão linear simples baseada nos mínimos quadrados). Uma destas versões da aplicação é baseada no trabalho de MATOS *et al.* (2010), a qual usa o tempo como variável independente e é baseada na regressão linear simples. Outra versão da aplicação usa a temperatura como variável independente e é também baseada na regressão linear simples. A ideia é experimentarmos o mecanismo simples trocando somente a entrada da função de regressão de uma variável menos correlacionada (por exemplo: tempo) por uma mais correlacionada (por exemplo: temperatura).

Os coeficientes β e α da função de regressão linear simples são calculados de acordo com as Equações (2) e (3), como segue:

$$\beta = \frac{\sum_{i=1}^{n}(x_i - \bar{X}) * (y_i - \bar{Y})}{\sum_{i=1}^{n}(x_i - \bar{X})^2} \quad (2)$$

$$\alpha = \bar{Y} - \beta * \bar{X} \quad (3)$$

onde β representa uma constante que é multiplicada pelo valor de cada variável independente. α é uma constante adicionada à multiplicação anterior, resultando no valor predito. X e Y são dois vetores unidimensionais, os quais respectivamente representam a janela de amostras das variáveis dependentes e independentes, com $X = x_1, \dots, x_i$ e $Y = y, \dots, y_i$, onde $i = 1, \dots, n$ e n é o numero de amostras. $\bar{X}$ e $\bar{Y}$ representam a media de amostras de cada vetor.

Os coeficientes β e α são calculados por cada nó sensor e quando chegam ao sorvedouro, eles são usados para recuperar dados omitidos para reduzir a quantidades de dados, de acordo com a Equação (4):

$$Y_{q_i} = \alpha + \beta * X_{p_i} \quad (4)$$

onde Y_{q_i} e X_{p_i} representam vetores unidimensionais, os quais respectivamente contêm os valores das predições feitas por uma variável dependente q e a janela de amostras de uma variável independente p, respectivamente. $Y_{q_i} = y_{q_1}, \dots, y_{q_i}$ e $X_{p_i} = x_{p_1}, \dots, x_{p_i}$, onde $i = 1, \dots, n$ e n é o numero de amostras. β e α respectivamente representam os coeficientes calculados pelas Equações (2) e (3).

Esta abordagem é simples, mas nós propomos o uso da regressão linear múltipla ao invés da regressão linear simples, devido ao fato que a precisão da predição na correlação multivariada é melhor. Na próxima seção, nos descrevemos como calcular os coeficientes β e α para executar nosso método.

A predição baseada na regressão linear simples é fácil de implementar, mas produz erros de predição os quais, para aplicações sensíveis à precisão, tornam-se inviáveis. A solução que nós encontramos foi melhorar a precisão da predição usando uma técnica estatística multivariada que é mais precisa que a regressão linear simples. A regressão linear múltipla, embora seja mais complexa que a simples, permite aumentarmos a precisão da predição e requer operações com vetores (*arrays*), as quais são também de fácil implementação em nós sensores. Mesmo em dispositivos com limitação de recursos como os das RSSFs, a regressão linear múltipla requer o armazenamento de uma pequena quantidade de amostras e consequentemente, o processamento também é pequeno.

Um total de dez amostras foram armazenadas e posteriormente processadas, em nossos experimentos, usando o Tossim para *motes* do tipo *micaz*, que são bastante limitados de recursos (Processador ATMega128, 7.3828 MHz;. 4KB de memória RAM e 128KB de memória ROM).

3.3 Solução proposta

O propósito de nossa proposta é melhorar a precisão da predição na redução de dados em RSSF. Então, nós usamos correlação multivariada para diminuir os erros de predição por meio da regressão linear múltipla, como segue:

a) a princípio todos os nós sensores calculam e enviam seus coeficientes através de difusão;

b) correlação temporal multivariada é aplicada para executar a predição de leituras consecutivas por meio da regressão linear múltipla em cada nó sensor;

c) cada nó sensor calcula seus coeficientes β e α e envia-os ao sorvedouro, ao invés de enviar todas as leituras de campo;

d) correlação espacial multivariada é usada para detectar sobreposição de dados por meio da distância Euclidiana. Assim, nós evitamos que a mesma informação seja enviada por vários nós sensores vizinhos; e

e) os dados omitidos (não enviados) podem ser gerados pelo sorvedouro através da função de regressão linear.

3.3.1. Mecanismo da solução

Nossa solução proposta é composta de oito etapas, mas algumas premissas são assumidas para que possamos implementá-la em nós sensores, tais como:

a) uma tabela de coeficientes de vizinhos é criada em cada nó sensor quando ele inicializa;

b) uma tabela de coeficientes é criada no sorvedouro quando ele inicializa;

c) as interfaces de rede de todos os nós sensores permanecem em modo promíscuo, armazenando os coeficientes dos seus nós sensores vizinhos para serem usados pela distância Euclidiana para descobrir se um nó vizinho já enviou a mesma informação calculada atualmente;

d) a janela de amostras deve ser adequada ao tamanho máximo do pacote suportado pelo TinyOS 2.x e definido anteriormente pelo desenvolvedor.

Os passos os quais nossa solução proposta é implementada estão resumidos no diagrama da Figura 5 e detalhados a seguir:

(i) Passo 1: o nó sensor armazena um número fixo de amostras das leituras, coletadas de todas as variáveis em cada ciclo;

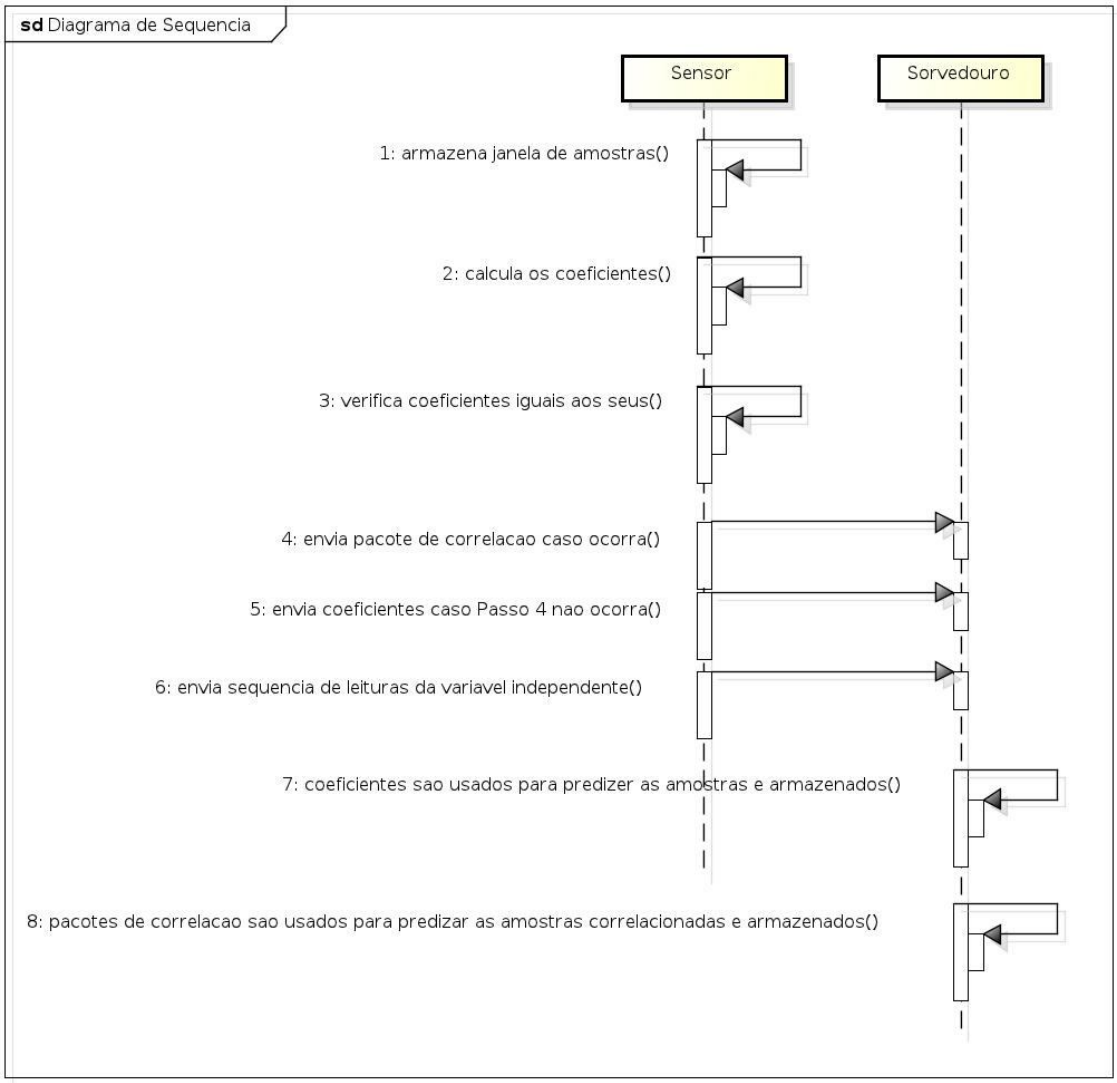
(ii) Passo 2: cada nó sensor calcula os coeficientes β e α da função de regressão linear múltipla, quando a janela de amostras alcança o limite máximo de armazenamento previamente definido;

(iii) Passo 3: antes de enviar seus coeficientes β e α ao sorvedouro, o nó sensor calcula a distância Euclidiana e verifica se estes coeficientes estão na sua tabela de coeficientes de vizinhos. Estes coeficientes são recebidos de seus nós sensores vizinhos por difusão;

(iv) Passo 4: se os valores gerados pelo nó sensor já têm sido enviados por um nó sensor vizinho, o nó sensor descarta seus coeficientes β e α. Então, ele envia

um pacote especial de tamanho reduzido, chamado pacote de correlação. Esse pacote anuncia que o nó sensor está correlacionado ao outro nó sensor vizinho;

Figura 5. Diagrama do mecanismo proposto.



(v) Passo 5: se os coeficientes β e α não tenham sido enviados ainda por outro nó sensor vizinho, o nó sensor envia-os a seu nó sensor pai, até o sorvedouro ser alcançado;

(vi) Passo 6: o nó sensor também envia a sequência de leituras da variável a qual é usada como variável independente. Conforme mencionado anteriormente, esta variável é calculada usando o coeficiente de Pearson [Equação (2)]. Em nossos

experimentos, a variável independente é a temperatura, devido ela ser a com maior grau de correlação com as demais;

(vii) Passo 7: quando os coeficientes β e α alcançam o sorvedouro, eles são usados na função de regressão linear múltipla para predizer as leituras as quais não têm sido enviadas. Além disto, estes coeficientes são armazenados para uso pelos pacotes de correlação ao chegarem ao sorvedouro;

(viii) Passo 8: se um pacote de correlação alcança o sorvedouro, ao invés dos coeficientes, o sorvedouro procura por uma entrada do nó sensor correlacionado na sua tabela de coeficientes (Passo 7). Então, os coeficientes β e α previamente armazenados, são usados para predizer as leituras.

3.3.2 Correlação espacial multivariada

A RSSF consiste de múltiplos nós espalhados de forma redundante. Assim, é possível termos um sistema tolerante a falhas através de redes densas. Por outro lado, estas redes são normalmente compostas de dispositivos com limitações de recursos. A energia é alimentada por baterias e o consumo de energia pode ser mais bem gerenciado, quando as correlações provenientes de aplicações, como à de monitoramento, são levadas em conta.

Como mencionado anteriormente, o coeficiente de Pearson (*r*) apresentado [Equação (1)], não calcula a correlação espacial multivariada. Então, nós propomos o uso da distância Euclidiana para determinar a correlação espacial multivariada entre dois vetores multidimensionais, ao invés de usar o coeficiente de Pearson (*r*), devido à necessidade de comparação entre dois vetores contendo os coeficientes β e α da regressão linear múltipla. A distância Euclidiana mostra quanto próximo um vetor multidimensional é de outro. A distância Euclidiana é definida como segue:

$$d_{X_N,X_V} = \sqrt{\sum_{j=1}^{k} (x_{N_j} - x_{V_j})^2} \tag{5}$$

onde $X_N = x_{N_1}, \dots, x_{N_j}$ e $X_V = x_{V_1}, \dots, x_{V_j}$. Em nosso caso, d_{X_N,X_V} representa a correlação entre dois vetores multidimensionais, de dimensões k com $j = 1, \dots, k$ para ser comparado em termos de suas correlações. Cada vetor contém os valores dos coeficientes β e α de cada

variável coletada pelo nó sensor N e seu nó sensor vizinho V, com exceção da variável independente.

Quanto menor a distância Euclidiana é, maior é a correlação entre os dois vetores. Assim, nós podemos comparar os coeficientes β e α da regressão linear múltipla, gerada a partir de consecutivas leituras coletadas pelo nó sensor e os coeficientes β e α dos seus nós sensores vizinhos, em um dado momento.

O nó sensor verifica se existe correlação entre ele e seus nós sensores vizinhos (Passo 3), antes de enviar um pacote contendo os coeficientes β e α da função de regressão linear múltipla. Se a distância Euclidiana é próxima a 0 (zero), então significa que um pacote com o mesmo conteúdo foi previamente enviado por qualquer outro nó sensor vizinho(Passo 4).

Em nossa solução proposta, o nó sensor detecta se existe correlação espacial multivariada entre ele e seu nó sensor vizinho por roteamento baseado em árvore. O mecanismo de roteamento é similar ao adotado por LI *et al.* (2010) para compressão de dados. Onde cada nó sensor envia seus coeficientes ao seu nó pai na árvore de roteamento até o pacote chegar ao sorvedouro por difusão.

O nó sensor verifica o grau de relacionamento dos coeficientes β e α calculando os valores de d_{X_N,X_V} [Equação (5)]. Ele não envia os coeficientes β e α das amostras de leituras atuais ao sorvedouro, se a distância Euclidiana for 0 (zero), ou seja, caso os coeficientes sejam iguais entre ele e seu vizinho. Assim, elimina a sobreposição de informação entre nós sensores vizinhos com base nos pacotes de coeficientes enviados por difusão e armazenados na tabela de vizinhos de cada. Portanto, isto reduz a difusão entre nós sensores vizinhos e também o encaminhamento de dados por *relays*.

3.3.3 Correlação temporal multivariada

A correlação temporal multivariada acontece devido ao fato que o nó sensor coleta dados correlacionados, de um ou mais variáveis em dado momento. Este tipo de correlação é observado devido à natureza do fenômeno físico (VURAN *et al*, 2004) (por exemplo: a temperatura do ambiente muda lentamente de acordo com o tempo).

A função de regressão linear simples é capaz de trabalhar sobre a correlação temporal, mas ela não é capaz de trabalhar sobre a correlação temporal multivariada (mais que

uma variável ao mesmo tempo). Nós propomos usar a função de regressão linear múltipla para trabalhar sobre a correlação multivariada. Nossa solução de redução de dados ocorre de forma distribuída, onde cada nó sensor calcula os coeficientes β e α da função de regressão linear múltipla (Passo 2). Então, ele somente envia β e α se não existir correlação espacial com outro nó sensor vizinho.

Os coeficientes β e α não são calculados pela regressão linear simples, quando a quantidade de variáveis independentes for superior a um. A regressão linear múltipla é descrita abaixo:

$$\beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_j \end{pmatrix}, X' = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_{11} & x_{12} & \cdots & x_{1i} \\ \vdots & \vdots & \vdots & \vdots \\ x_{j1} & x_{j2} & \cdots & x_{ji} \end{pmatrix}, X = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1j} \\ 1 & x_{21} & \cdots & x_{2j} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{i1} & \cdots & x_{ij} \end{pmatrix} \text{ e } Y = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_i \end{pmatrix}$$

com:

$$\beta = (X'X)^{-1}X'Y \tag{6}$$

onde β representa o vetor de coeficientes da função de regressão linear múltipla. Nós usamos $\beta_0 = \alpha$ por simplicidade e compatibilidade com os coeficientes β e α da regressão linear simples. X é um vetor multidimensional, o qual representa a janela de amostras das variáveis independentes, junto com seu vetor transposto X'. Y é o vetor de uma dimensão, o qual representa a janela de amostras da variável dependente. $i = 1, \ldots, n$ e n é o número de amostras, e $j = 1, \ldots, k$ onde k é a dimensão do vetor X.

3.3.4 Recuperação de dados

O sorvedouro recebe os coeficientes β e α, ou o pacote de correlação, para recuperar os dados por meio da predição. Ele distingue isto baseado no tamanho do pacote. Portanto, o estimador calcula os valores das amostras não enviadas, baseado nos coeficientes β e α da função de regressão linear múltipla [Equação (6)].

Porém, se o pacote de correlação chega ao sorvedouro, ao invés dos coeficientes, o sorvedouro usa os coeficientes β e α do nó sensor correlacionado, armazenado na tabela de coeficientes do sorvedouro. A predição das variáveis usando regressão linear múltipla é calculada de acordo com a Equação (7):

$$Y_{q_{ij}} = \beta_0 + \beta_1 * X_{p_{i1}} +, \dots, + \beta_j * X_{p_{ij}} \quad (7)$$

onde $Y_{q_{ij}}$ representa um vetor de uma dimensão, o qual contém os valores da predição feita por uma variável dependente q e $X_{p_{ij}}$ representa o vetor multidimensional, o qual contém o histórico de valores das amostras de mais de uma variável independente p. $Y_{q_{ij}} = y_{q_{i1}}, \dots, y_{q_{ij}}$ e $X_{p_{ij}} = x_{p_{i1}}, \dots, x_{p_{ij}}$, com $i = 1, \dots, n$, onde n é o número de amostras, e $j = 1, \dots, k$, onde k é a dimensão do vetor $X_{p_{ij}}$. β e α, respectivamente, representam os coeficientes calculados usando a Equação (6). Relembrando que, $\beta_0 = \alpha$ devido à compatibilidade com a notação dos coeficientes β e α usada neste trabalho.

A predição por regressão linear simples é calculada pela Equação (4), mas nossa solução proposta usa uma correlação multivariada, ao invés de uma univariada. Então, nossa proposta usa a Equação (7) para executar as predições dos valores das variáveis no sorvedouro.

3.4 Considerações finais

O melhoramento da precisão da predição de dados em RSSF proposto em nossa solução é um método estatístico multivariado que tem a vantagem de aumentar a precisão diminuindo o erro. Mas para isto, nós tivemos que aumentar o consumo de energia em relação à solução de regressão linear simples.

Isto acontece porque as entradas da função de regressão linear múltipla para diminuir o erro têm que utilizar as variáveis mais correlacionadas. Como discutido anteriormente, a variável época (contador) é menos correlacionada, quando comparamos com as variáveis coletadas em campo. Assim, para calcular os coeficientes da função de regressão linear múltipla, nós precisamos usar uma ou mais variáveis coletadas em campo. A desvantagem é que as variáveis usadas como entradas da função têm que ser enviadas ao sorvedouro para recuperar os dados omitidos.

Embora o consumo de energia seja maior na regressão linear múltipla em relação à regressão linear simples, nós propomos explorar a correlação espacial através da distância Euclidiana para evitar que os coeficientes da equação e as amostras usadas nos seus cálculos, sejam enviados quando um nó sensor vizinho já tenha enviado ao sorvedouro.

4 MATERIAIS E MÉTODOS

Neste capítulo abordamos a forma como os nossos experimentos são realizados para que possamos avaliar o desempenho de nossa proposta. Inicialmente iremos descrever alguns princípios básicos para o entendimento da nossa proposta. Em seguida mostramos a metodologia utilizada para alcançar nossos objetivos através de simulações de experimentos envolvendo as versões das aplicações de monitoramento com e sem redução de dados, as quais foram desenvolvidas para comparar seus resultados. Logo depois, detalhamos cada aplicação que foi criada para realizar as simulações e quais as métricas que avaliamos. Além disso, detalhamos os cenários que foram utilizados nas simulações.

4.1 Princípios

Em nossa abordagem, nós usamos um protocolo de roteamento baseado em árvore para encaminhar o tráfego de dados dos nós sensores ao sorvedouro, uma abordagem similar a adotada por Li *et al.* (2010). A abordagem de roteamento em árvore é importante para que seja possível detectar a correlação espacial e posteriormente descartar os pacotes com informações sobrepostas.

Para evitar a sobreposição espacial, cada nó sensor verifica se existe um grau de correlação multivariada entre os pacotes previamente enviados por seus vizinhos. Isto é feito antes de cada nó sensor enviar os coeficientes de regressão linear. Além disto, nós também usamos o método de correlação multivariada para evitar a sobreposição temporal no mesmo nó sensor.

Neste trabalho, simulações com funções de regressão linear simples e múltipla são executadas para avaliar a solução de predição. A regressão linear simples serve de *benchmark* para compararmos o desempenho de nossa proposta. Inicialmente o grau de correlação das variáveis coletadas pelos nós sensores é medido para decidir qual variável será a variável independente.

Aqui neste trabalho, o coeficiente de Pearson (*r*) (HAIR *et al*, 1998) em um *trace* de dados reais indica a força do relacionamento linear entre duas variáveis, isto é, se as variáveis são independentes, o coeficiente de Pearson (*r*) é zero. Nós avaliamos o consumo de energia e a precisão da predição em todas as soluções, nas quais os nós sensores carregam funções de regressão linear simples (soluções atuais) ou regressão linear múltipla (nossa proposta).

Uma aplicação original para coleta de dados sem qualquer mecanismo de predição foi desenvolvida para que possamos observar a economia de energia alcançada pelas versões da aplicação com redução de dados. Esta aplicação emula uma coleta real de dados de temperatura, umidade e luminosidade. A Tabela 2 resume as características de cada versão da aplicação.

Então, a versão original desta aplicação é comparada a três versões melhoradas, onde duas usam regressão linear simples e uma usa regressão linear múltipla. O desempenho da precisão de predição é avaliado por meio do Quadrado da Soma dos Erros (SSerr) e o coeficiente de determinação (R^2).

Tabela 2. Versões da aplicação desenvolvidas para o experimento

Aplicação	Redução de dados	Descrição
Versão 1 (Original)	Sem redução de dados	Aplicação de monitoramento de temperatura, umidade e luminosidade
Versão 2 (SimpleCount)	Regressão linear simples	Versão melhorada da versão original usando contador como variável independente
Versão 3 (SimpleTemperature)	Regressão linear simples	Versão melhorada da versão 2 usando temperatura como variável independente
Versão 4 (Multiple)	Regressão linear múltipla	Versão melhorada da versão 3 usando temperatura e contador como variáveis independes

4.2 Metodologia

Nós usamos simulação para provar o desempenho de nossa proposta. A ferramenta de simulação adotada foi o Tossim[2], porque nós temos em laboratório os kits de desenvolvimento da Crossbow[3] para posteriormente executar *testbeds* em campo e melhorar

[2] Site Web do Tossim em http://docs.tinyos.net/tinywiki/index.php/TOSSIM
[3] Site Web da Crossbow em http://www.xbow.com

nossa proposta. Este tipo de dispositivo suporta TinyOS 2.x e o Tossim é a ferramenta padrão para fazer as simulações.

Todo o código foi desenvolvido para simulação em nesC para TinyOS 2.x. Eles podem ser embutidos dentro dos nós sensores do simulador do Tossim e também dentro de nós sensores reais. Isto garante que o mesmo código usado para similar os experimentos é capaz de executar testes em cenários reais no futuro.

Os cenários de simulação envolvem diferentes situações de densidade da rede, valores de aplicações de dados (variáveis coletadas, correlacionadas ou não correlacionadas) e forma de implantação do nó. Assim, nós verificamos possíveis cenários do mundo real por simulações.

As versões das aplicações foram criadas para verificar a eficácia e eficiência de nossa proposta em face da abordagem simples, como descrito a seguir:

(i) a primeira versão serve de base (guia) para comparar o consumo de energia. O objetivo desta versão é medir o consumo de energia sem predição e verificar quanto cada proposta de predição gastará de energia, quando a redução de dados é usada por regressão linear simples e múltipla;

(ii) a segunda versão é uma versão baseada na técnica de redução de dados adotada por MATOS *et al.* (2010) usando regressão linear simples. Ela é uma versão de predição básica na qual nós verificamos os erros de predição e o consumo de energia. Esta versão é baseada na variável tempo, a qual não é altamente correlacionada com as variáveis coletadas. Portanto, nós acreditamos que o erro de predição tende a aumentar;

(ii) a terceira versão é uma forma de verificar se é possível melhorar a precisão de predição mudando somente a variável independente. Nós usamos a variável temperatura, ao invés da variável tempo, porque ela é mais correlacionada com as outras variáveis. A melhor maneira de melhorar a precisão de predição é diminuindo os erros de predição, usando a mesma quantidade de energia que a segunda versão, mas existe um compromisso entre precisão de predição e consumo de energia;

(iv) a última versão é nossa proposta, a qual usa as variáveis época (contador) e temperatura juntas na predição das variáveis umidade e luminosidade. A correlação entre as variáveis coletadas é mais alta que a variável tempo, e então nós acreditamos que o erro de predição diminuirá, mesmo que ele gaste mais energia.

Cada versão da aplicação tem diferentes comprimentos de pacotes, os quais determinam quanto de energia será gasto na comunicação de dados, ou seja, quanto maior o pacote, maior será o consumo de energia.

4.3 Avaliação de desempenho das versões da aplicação

A avaliação de desempenho foi feita através das quatro versões da aplicação, as quais nós usamos para simular e comparar a regressão linear múltipla à regressão linear simples e à versão original de uma aplicação de monitoramento. Esta aplicação de monitoramento simula a coleta de três variáveis de ambiente: temperatura, umidade e luminosidade.

As características das versões da aplicação para as simulações com os objetivos de cada uma são apresentadas a seguir. Como pretendemos mostrar a evolução dos estudos desde a abordagem simples até chegarmos à nossa proposta, nós apenas realizamos modificações na versão original da aplicação.

A primeira versão, que chamamos de versão original da aplicação, envia leituras de temperatura, umidade e luminosidade periodicamente a cada 1024 disparos de relógio do nó sensor, sem executar predição. Esta versão foi criada para servir como uma aplicação de referência para nós compararmos o consumo de energia das versões seguintes, as quais usam predição para redução de dados.

A segunda versão é uma versão melhorada da aplicação original através de um modelo de regressão linear simples. Ela envia os coeficientes β e α para cada variável dependente. Ela usa um contador (variável tempo) como variável independente para predizer a temperatura, a umidade e a luminosidade. Esta versão foi desenvolvida para verificar o consumo de energia quando a regressão linear simples é usada para reduzir dados enviados ao sorvedouro. Ela foi também implementada para calcular a soma do erro (SSerr) e o coeficiente de determinação/melhoramento (R^2) para comparar à próximas versões. O

contador é usado como variável tempo, então o nó sensor não envia qualquer amostra de variáveis ao sorvedouro.

A terceira versão é uma versão melhorada da segunda versão através de uma função de regressão linear simples, mas usando a temperatura como variável independente, ao invés da variável tempo. O nó sensor envia amostras de leituras da variável temperatura e os coeficientes β e α para cada variável dependente (exceto a temperatura) para predizer as variáveis dependentes umidade e luminosidade. Esta versão foi desenvolvida para verificar o impacto deste modelo no consumo de energia quando a regressão linear simples, usando uma variável dependente coletada em campo, para reduzir a comunicação de dados. Ela também foi criada para verificar o SSerr e R^2 comparado às segunda e terceira versões. A temperatura foi escolhida como variável independente, devido aos resultados obtidos do coeficiente de Pearson (r), os quais podem ser vistos posteriormente na próxima seção.

A quarta versão é uma versão melhorada da terceira versão através de uma função de regressão linear múltipla, usando o contador e a temperatura como variáveis independentes. O nó sensor envia amostras de leituras de temperatura e os coeficientes β e α para cada variável dependente (exceto a temperatura) com $\beta = (\beta_0, \beta_1, \beta_2)$ onde $\alpha = \beta_0$. O sorvedouro estima as variáveis dependentes umidade e luminosidade. Esta versão foi desenvolvida para verificar o SSerr e R^2 comparado às segunda e terceira versões. Nosso método proposto é baseado nesta versão.

4.4 Implementação

Para cada versão da aplicação, nós usamos diferentes tipos de pacotes de acordo com cada situação. O TinyOS 2.x provê, por padrão, pacotes de até 28 bytes para serem enviados por aplicações de RSSF, onde somente 20 bytes podem ser usados para dados dos usuários e informações extras, com as de roteamento. Portanto, nós desenvolvemos mensagens da aplicação com tamanhos ajustados ao tamanho máximo aceitável pelo TinyOS 2.x. As características de cada versão da aplicação são discutidas a seguir:

4.4.1 Primeira versão da aplicação

Para esta versão existe somente um tipo de pacotes da aplicação de 14 bytes (Figura 6) contendo leituras das variáveis temperatura (Temp), umidade (Humid) e luminosidade (Light). O tamanho do campo das variáveis é 16 bits devido ao fato que o

pacote de dados no TinyOS não suporta valores de ponto flutuante. Então, para configurar algumas variáveis, tais como temperatura, o valor é convertido em inteiro.

Além disto, este pacote contém informações para serem manipuladas pela camada de rede, tais como nó origem (Origin), métrica de estimação de rota (Etx), valor da rota (Lr_value) e próximo salto (Lr_addr). Em cada rodada (ciclo) de coleta, um pacote com dez leituras é enviado pelos nós sensores ao sorvedouro, ou seja, um total de 140 bytes/rodada/nó.

Figura 6. Comprimento do pacote de leituras (versão 1).

Origin	Temp	Humid	Light	Etx	Lr_value	Lr_addr	Não usado
16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	(48 bits)

4.4.2 Segunda versão da aplicação

Nós criamos dois tipos de pacotes da aplicação: um pacote de 20 bytes (Figura 7) contendo os coeficiente β (bT – temperatura, bH – umidade e bL – luminosidade) e α (aT – temperatura, aH – umidade e aL – luminosidade) calculados para cada variável dependente; e um pacote de tamanho reduzido de 10 bytes (Figura 8) para enviar a mensagem que o nó sensor é espacialmente correlacionado a um nó sensor vizinho (Correlated).

Além disto, os dois pacotes acima contêm informações para serem manipuladas pela camada de rede, tais como nó origem (Origin), métrica de estimação de rota (Etx), valor da rota (Lr_value) e próximo salto (Lr_addr). Em cada rodada (ciclo) de coleta, um pacote de coeficientes ou um pacote de correlação é enviado pelos nós sensores ao sorvedouro, ou seja, 20 bytes/rodada/nó ou 10 bytes/rodada/nó.

Figura 7. Comprimento do pacote de coeficientes (versão 2).

Origin	aT	bT	aH	bH	aL	bL	Etx	Lr_value	Lr_addr
16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits

Figura 8. Comprimento do pacote de correlação (versão 2).

Origin	Correlated	Etx	Lr_value	Lr_addr	Unused
16 bits	16 bits	16 bits	16 bits	16 bits	80 bits

4.4.3 Terceira versão da aplicação

Três tipos de pacotes da aplicação foram criados nesta versão: um pacote de 16 bytes (Figura 9) contendo os coeficientes β (bH – umidade e bL – luminosidade) e α (aH – umidade e aL – luminosidade) calculados para cada variável dependente (exceto a variável temperatura); um pacote de tamanho reduzido de 10 bytes (Figura 10) para enviar mensagem que o nó sensor é espacialmente correlacionado a um nó sensor vizinho (Correlated); e um pacote de 18 bytes (Figura 11) contendo dez leituras de temperatura (T1 a T10) em sequência, para serem usados na predição das variáveis umidade e luminosidade.

Além disto, os três pacotes acima contêm informações para ser manipuladas pela camada de rede, tais como nó origem (Origin), métrica de estimação da rota (Etx), valor da rota (Lr_value) e próximo salto (Lr_addr). A variável temperatura é enviada em sequência em um único pacote, porque ela não é predita pelo sorvedouro e é também usada para predizer as outras duas variáveis. O número de leituras enviadas depende do tamanho máximo do pacote do TinyOS 2.x. Em cada rodada (ciclo) de coleta, um pacote de coeficientes e um pacote de leituras, ou somente um pacote de correlação é enviado pelos nós sensores ao sorvedouro, ou seja, um total de 34 bytes/rodada/nó ou 10 bytes/rodada/nó.

Figura 9. Comprimento do pacote de coeficientes (versão 3).

Origin	aH	bH	aL	bL	Etx	Lr_value	Lr_addr	Não usado
16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	(32 bits)

Figura 10. Comprimento do pacote de correlação (versão 3).

Origin	Correlated	Etx	Lr_value	Lr_addr	Não usado
16 bits	16 bits	16 bits	16 bits	16 bits	80 bits

Figura 11. Comprimento do pacote de leituras (versão 3).

Origin	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	Etx	Lr_value	Lr_addr	Não usado
16 bits	10 × 8 bits = (80 bits)										16 bits	16 bits	16 bits	(16 bits)

4.4.4 Quarta versão da aplicação

Três tipos de pacotes da aplicação foram criados nesta versão: um pacote de 20 bytes (Figura 12) contendo os coeficientes β (b1H – umidade e b1L – luminosidade, e b2H – umidade e b2L – luminosidade) e α (aH – umidade e aL – luminosidade) calculados para cada variável dependente (exceto a variável temperatura), com $\beta = (\beta_0, \beta_1, \beta_2)$ onde $\alpha = \beta_0$; um pacote de tamanho reduzido de 10 bytes (Figura 13) para enviar a mensagem que o nó sensor é espacialmente correlacionado a um nó sensor vizinho; e um pacote de 18 bytes (Figura 14) contendo dez leituras de temperatura (T1 a T10) em sequência, para serem usados na predição das variáveis umidade e luminosidade.

Além disto, os três pacotes acima contêm informações para serem manipuladas pela camada de rede, tais como nó origem (Origin), métrica de estimação da rota (Etx), valor da rota (Lr_value) e próximo salto (Lr_addr). A variável temperatura é enviada em sequência em um mesmo pacote, como na terceira versão, porque ela não é predita pelo sorvedouro e é também usada para predizer as outras duas variáveis. O número de leituras depende do tamanho máximo do pacote do TinyOS 2.x. Em cada rodada (ciclo) de coleta, um pacote de coeficientes e um pacote de leituras, ou somente um pacote de correlação é enviado pelos nós sensores ao sorvedouro, ou seja, um total de 38 bytes/rodada/nó ou 10 bytes/rodada/nó.

Figura 12. Comprimento do pacote de coeficientes (versão 4).

Origin	Ah	b1H	b2H	aL	b1L	b2L	Etx	Lr_value	Lr_addr
16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits	16 bits

Figura 13. Comprimento do pacote de correlação (versão 4).

Origin	Correlated	Etx	Lr_value	Lr_addr	Não usado
16 bits	16 bits	16 bits	16 bits	16 bits	80 bits

Figura 14. Comprimento do pacote de leituras (versão 4).

Origin	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	Etx	Lr_value	Lr_addr	Não usado
16 bits	10 × 8 bits = (80 bits)										16 bits	16 bits	16 bits	(16bits)

4.5 Configuração das simulações

As implementações das aplicações foram carregadas no Tossim. Nós usamos um arquivo de *trace* do Intel Berkeley Research Lab[4] contendo leituras de temperatura, umidade e luminosidade coletadas por multisensores em um prédio. Assim, os dados coletados para nossa simulação vêm de um cenário próximo à realidade. Ele contém leituras de 54 nós sensores implantados em laboratórios com intervalos de 31 segundos. As leituras usadas na simulação correspondem a coletas de um dia deste arquivo.

Nós embutimos as quatro versões da aplicação dentro dos nós sensores no Tossim. Então, o desempenho da precisão da predição das diferentes aplicações foi medido. Também, o consumo de energia da comunicação de dados na versão da aplicação original foi verificado. O consumo de energia da versão original com as três versões melhoradas foi comparado, onde em duas versões usamos regressão linear simples e na outra, usamos regressão linear múltipla (nossa solução proposta). O modelo de energia adotado nos experimentos foi gerado pela ferramenta PowerTossim-Z[5] em todos os cenários.

Os dois parâmetros usados para revelar o melhor ou pior desempenho da precisão da predição de nossa proposta comparado aos trabalhos atuais são o Quadrado da Soma dos Erros (SSerr) e o coeficiente de determinação (R^2). O SSerr [Equação (8)] é a soma da potência dos erros de predição para cada variável dependente usando regressão linear simples ou múltipla. O R^2 [Equação (9)] representa o melhoramento da soma da potência dos erros de predição.

Mais detalhes sobre estes parâmetros podem ser encontrados em Hair *et al* (1998) e suas equações podem ser vistas a seguir:

$$SSerr = \sum_{i=1}^{n} (Y_i - Y_{q_i})^2 \tag{8}$$

com:

$$SSreg = \sum_{i=1}^{n} (Y_{q_i} - \bar{Y})^2 \; ; \; SStot = \sum_{i=1}^{n} (Y_i - \bar{Y})^2$$

[4] Site Web do arquivo de trace em http://db.csail.mit.edu/labdata/labdata.html
[5] Site Web do PowerTossim-Z em https://www.scss.tcd.ie/~carbajor/powertossimz/index.html

$$R^2 = \frac{SSreg}{SStot} \quad (9)$$

onde Y_{q_i} representa um vetor unidimensional, o qual contém os valores das predições feitas por uma variável dependente q. $Y_{q_i} = y_{q_1}, \dots, y_{q_i}$, onde $i = 1, \dots, n$ e n é o número de amostras. Y é um vetor unidimensional, o qual representa a janela de amostras das variáveis independentes, com $Y = y, \dots, y_i$, onde $i = 1, \dots, n$ e n é o número de amostras. $\bar{Y}$ representa a media das amostras do vetor. SSreg é a soma da regressão dos quadrados e SStot é a soma total dos quadrados.

A avaliação de desempenho de nossa proposta foi também medida variando a quantidade de amostras. Isto mostra quanto nossa proposta é afetada pelo compromisso entre precisão da predição e consumo de energia. Nós repetimos os cenários que têm os melhores resultados entre os cenários simulados, para verificar o comportamento de nossa proposta.

4.6 Métricas de avaliação

As métricas de avaliação adotadas por este trabalho são: (1) métricas da eficiência do consumo de energia; (2) e métricas da eficiência do preditor.

4.6.1 Métricas da eficiência do consumo de energia

As métricas da eficiência do consumo de energia são definidas como:

(i) a média total do consumo de energia na rede em Joule, da transmissão dos pacotes da aplicação (Etrans);

(ii) a média total do consumo de energia da rede in Joule, da recepção dos pacotes da aplicação por difusão dos nós sensores vizinhos – *gossiped* (Erecp);

(iii) o número de vezes que a correlação espacial multivariada foi detectada pelos nós sensores (Cspatial);

(iv) e a porcentagem de energia economizada nas versões com regressão linear (versões de 2 a 4) em face da versão original (Esaved).

A energia gasta na comunicação de dados é analisada pela métrica do consumo de energia. De acordo com cada versão da aplicação, o comprimento de pacote é menor na

versão inicial e é maior na versão final. Assim, o consumo de energia tende a ser maior na última versão da aplicação.

A correlação espacial é medida pela quantidade de vezes que ela é detectada, mostrando quanto uma versão da aplicação economiza de energia por não enviar um pacote de dados mais largo. Talvez não exista diferença significativa entre as versões da aplicação, visto que este mecanismo não foi modificado, mas somente adaptado para cada versão da aplicação.

Nosso trabalho objetiva melhorar a precisão da predição e não está focado na economia de energia em relação aos trabalhos atuais, mas nós temos verificado que o impacto de nossa proposta em face de soluções atuais para medir qual a viabilidade nas RSSF.

4.6.2 Métricas da eficiência do preditor

As métricas da eficiência do preditor são definidas como:

(i) o erro da predição (SSerr);

(ii) e o melhoramento do preditor baseado no coeficiente de determinação (R^2).

O SSerr mostra quanto erro cada versão da aplicação supera outra. Provavelmente, as versões iniciais têm um maior erro de predição que as últimas versões, porque o uso de variáveis correlacionadas na predição garante poucos erros. O coeficiente de determinação (R^2) mede o melhoramento do preditor em relação ao seu erro. Diferente do SSerr, o melhoramento tende a ser melhor nas versão finais.

4.7 Cenários de simulação

Três características são importantes para configurar os cenários em nossas simulações. Após alguns testes iniciais, detectamos a necessidade de considerar estas características como necessárias nos cenários de nossos experimentos. Vejamos cada uma a seguir:

4.7.1 Comportamento da variável descorrelacionada

A primeira é o comportamento da variável luminosidade, que em alguns momentos nos experimentos permanece descorrelacionada das demais variáveis. Prejudicando o desempenho da precisão da predição. Às vezes, a variável luminosidade muda facilmente e

leva a diferentes resultados na predição, devido à variação da correlação entre variáveis coletadas. Ela pode ser apresentada de duas formas, constante e não constante.

As variáveis temperatura e umidade são normalmente correlacionadas, ou seja, quando uma aumenta a outra diminui de valor e vice-versa. Portanto, seus comportamentos são constantes, com seus valores mudando simultaneamente e lentamente. As versões da aplicação de 2 a 4 usam predição e podem aumentar o erro de predição quando uma ou mais variáveis mudam seus valores rapidamente, gerando falta de correlação entre elas.

4.7.2 Topologia da rede

A segunda característica é a topologia, a qual pode aumentar o consumo de energia nas implantações randômicas. Normalmente, todas as versões da aplicação sofrem os mesmos efeitos no consumo de energia, visto que a topologia não afetará a predição (correlação temporal). Mas temos que levar em consideração o tipo de topologia para observar a influência da distância Euclidiana na correlação espacial. Na topologia em grade é possível planejarmos e influenciarmos o desempenho da rede, mas na topologia randômica isto não acontece.

4.7.3 Densidade da rede

A última é a densidade da rede, a qual também influencia o consumo de energia, mas não afeta a predição. Quando a densidade da rede é alta, ou seja, muitos nós sensores próximos dos outros, o consumo de energia aumenta, devido à recepção de pacotes por difusão. As versões da aplicação de 2 a 4 devem sofrer o mesmo efeito da densidade da rede, mas tem que ser verificado se a comunicação entre nós sensores com o mais baixo erro da predição pode otimizar o consumo de energia.

Então, para explorar os cenários das simulações, nós resumimos as características na Tabela 3. Estas características tentam emular as circunstâncias do mundo real, tal que podemos simular cenários próximos a uma implantação de nós sensores para aplicações de monitoramento de ambientes.

Nós definimos seis diferentes cenários que têm sido carregados 30 vezes cada um. Todos os cenários usam as quatro versões da aplicação e o número de nós é de 4 a 100 (para medir a escalabilidade). A escalabilidade é importante para verificar o consumo de energia em todas as versões da aplicação.

Tabela 3. Características dos cenários de simulação.

Cenários	Características						
	Variável luminosidade		Topologia		Densidade da rede		
	Constante	Não constante	Grade	Randômica	1 nó/5 m	Variando	Fixo
1	X		X		X		
2		X	X		X		
3		X		X		X	
4	X			X		X	
5		X		X			X
6	X			X			X

Todos os resultados dos experimentos têm intervalo de confiança de 95%. A ferramenta Link Layer Model do TinyOS 2.x foi usada para criar as topologias em grade e randômicas. Em cada cenário, várias densidades de nós sensores são usadas e resumidas na Tabela 4. O modelo do consumo de energia adotado aqui é o mesmo de Jurdak *et al.* (2008), onde o rádio gasta 1,67 μJ/Byte enviando e 1,89 μJ/Byte recebendo dados usando mote micaz da Crossbow.

Tabela 4. Densidade da rede nos cenários de simulação.

Nós	Densidade (nó/m^2) por cenários					
	#1	#2	#3	#4	#5	#6
4	0,1600	0,1600	0,2500	0,2500	0,2500	0,2500
9	0,0900	0,0900	0,1111	0,1111	0,2500	0,2500
16	0,0711	0,0711	0,0625	0,0625	0,2500	0,2500
25	0,0625	0,0625	0,0400	0,0400	0,2500	0,2500
36	0,0576	0,0576	0,0278	0,0278	0,2500	0,2500
49	0,0544	0,0544	0,0204	0,0204	0,2500	0,2500
64	0,0522	0,0522	0,0156	0,0156	0,2500	0,2500
81	0,0506	0,0506	0,0123	0,0123	0,2500	0,2500
100	0,0494	0,0494	0,0100	0,0100	0,2500	0,2500

4.8 Considerações finais

Para avaliarmos o desempenho de nossa proposta, inicialmente desenvolvemos e realizamos experimentos com a regressão linear simples para detectar os erros provocados pela redução de dados. Esta abordagem simples é muito útil em cenários onde não requerem altos níveis de precisão da predição. Mas por tratar-se de uma abordagem simples, para alguns tipos de aplicações ela não é confiável, gerando problemas de consistência dos dados na aplicação. Então, resolvemos analisar o nível de correlação entre as variáveis coletadas em campo para detectar qual a mais correlacionada com as outras.

O objetivo é usar esta variável como entrada na função de regressão linear simples e acompanha seu desempenho. Se for favorável, ou seja, o erro diminuir em relação a regressão linear simples usando a variável tempo como entrada da função, então teremos a solução de nosso problema de forma rápida. Mas isto não aconteceu, como veremos no capítulo seguinte, onde detalhamos os resultados dos experimentos. Portanto, resolvemos aplicar a regressão multivariada para melhorarmos a precisão da predição.

A partir disto, os resultados dos experimentos foram favoráveis para que nós apontássemos esta solução como a nossa proposta. A desvantagem desta abordagem é que aumentamos o consumo de energia em detrimento disto.

5 RESULTADOS

Conforme vimos no capítulo anterior, para avaliarmos a nossa proposta, nós realizamos experimentos para identificar nos resultados o desempenho do consumo de energia e da precisão da predição. Antes de qualquer simulação das aplicações criadas para avaliar estes desempenhos, nós fizemos a análise de correlação a qual serve para apontarmos qual variável será usada como variável independente. Após isto, realizamos os experimentos e acompanhamos os resultados, que nos levaram a um comprometimento tradicional neste tipo de abordagem, que existe entre a precisão e o consumo de energia.

5.1 Avaliação da análise de correlação

Os resultados do coeficiente de Pearson (*r*) (Tabela 5) mostram que existe uma maior correlação entre a variável temperatura e as outras variáveis coletadas pelos nós sensores (tais como umidade e luminosidade) que com a variável época (contador).

Tabela 5. Resultados da análise de correlação.

	Temperatura	**Umidade**	**Luminosidade**	**Época**
Temperatura	1,0000	−0,7987	0,4550	−0,2681
Umidade	−0,7987	1,0000	−0,2489	0,1987
Luminosidade	0,4550	−0,2489	1,0000	−0,1807
Época	−0,2681	0,1987	−0,1807	1,0000

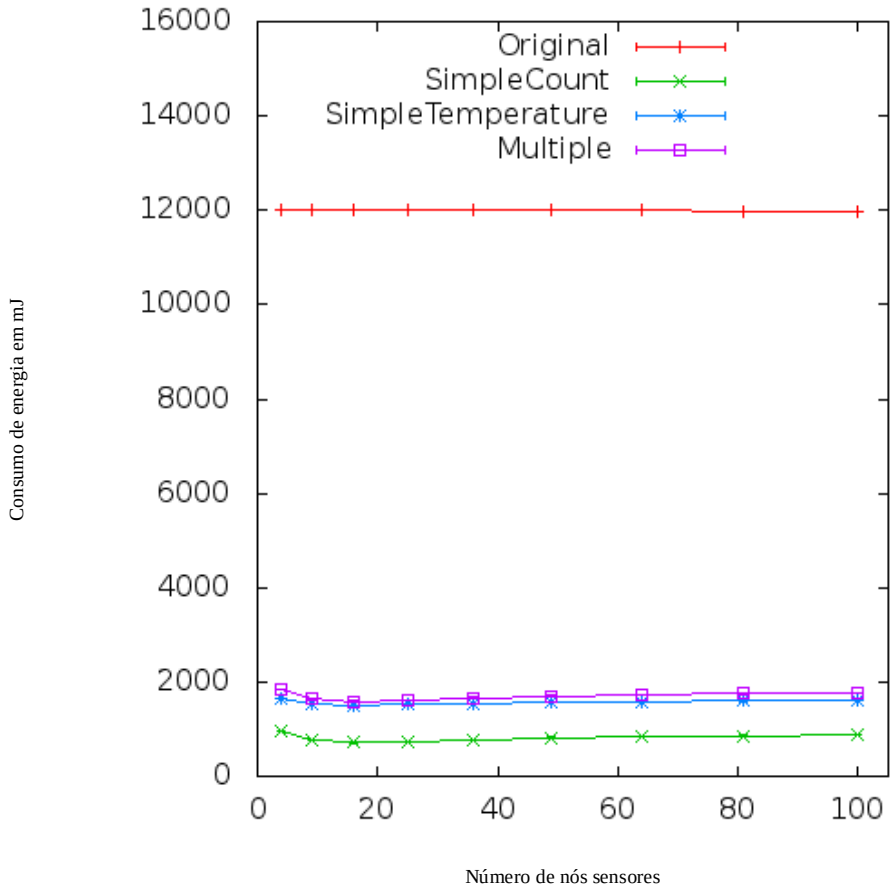
Dado os resultados desta análise de correlação, a variável temperatura foi usada como a variável dependente para as versões da aplicação 3 e 4. A versão da aplicação 2 usa somente a variável época como variável independente e a versão da aplicação 3 usa somente a variável temperatura como variável independente, ao invés de variável época (contador). Por outro lado, a versão da aplicação 4 usa as variáveis época e temperatura como variáveis independentes.

5.2 Consumo de energia

O principal objetivo de nossa solução proposta não é reduzir o consumo de energia comparado a abordagens baseadas na regressão linear simples, mas encontrar o melhor compromisso entre consumo de energia e precisão da predição.

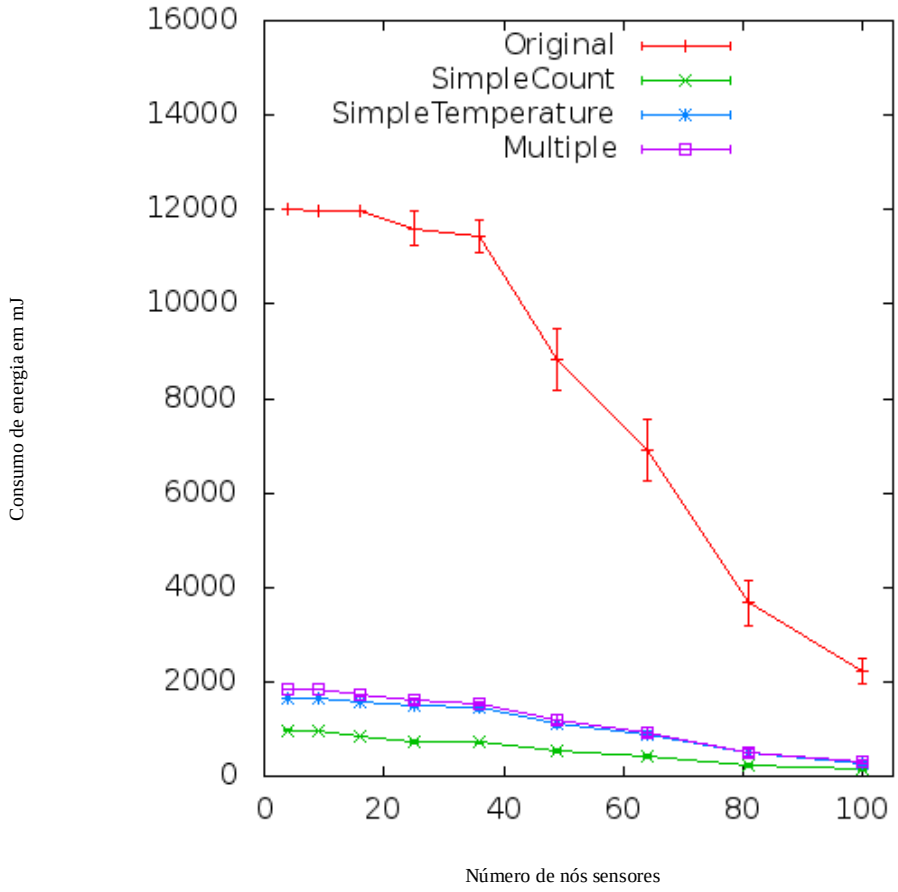
Em nosso método, nós usamos amostras da variável temperatura para predizer as variáveis umidade e luminosidade. Enquanto nós, aumentamos o consumo de energia comparado à regressão linear simples, nós aumentamos a precisão da predição causada pela regressão linear múltipla. Como o maior consumo de energia nas RSSF é alcançado pela comunicação dos pacotes na rede, acompanhamos o desempenho do consumo de energia no rádio quando ele realiza transmissão e recepção. Quando nós planejamos nossos experimentos, previámos que os resultados de economia de energia não seriam favoráveis a nossa proposta quanto aos resultados obtidos pela regressão linear simples. Mas de qualquer forma, teríamos que obter os valores para comparar o quanto uma solução gasta mais que a outra.

Figura 15. Média da energia do rádio em mJ, consumida pelas mensagens enviadas ao sorvedouro variando o número de nós sensores: (a) cenários #1, #2, #5 e #6; (b) cenários #3 e #4.



(a)

Figura 15. Continuação.

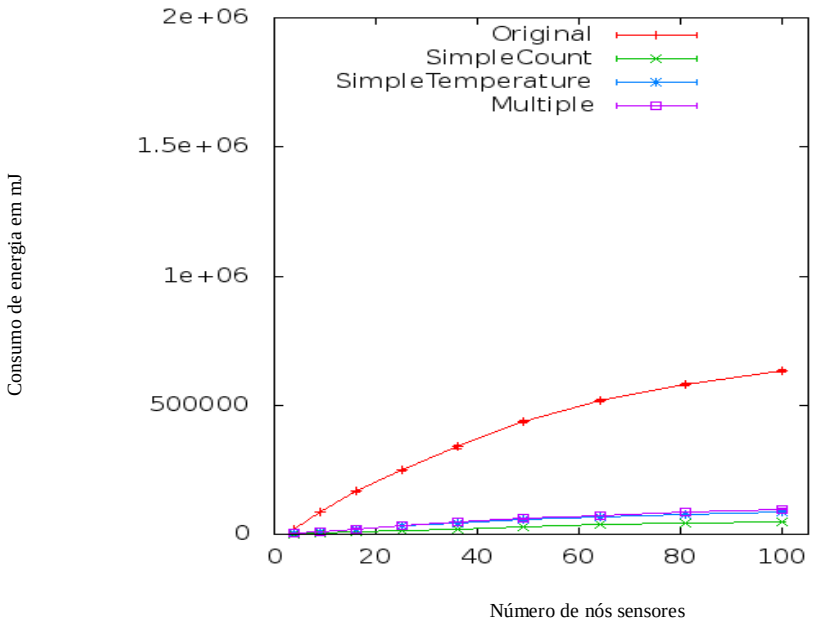


(b)

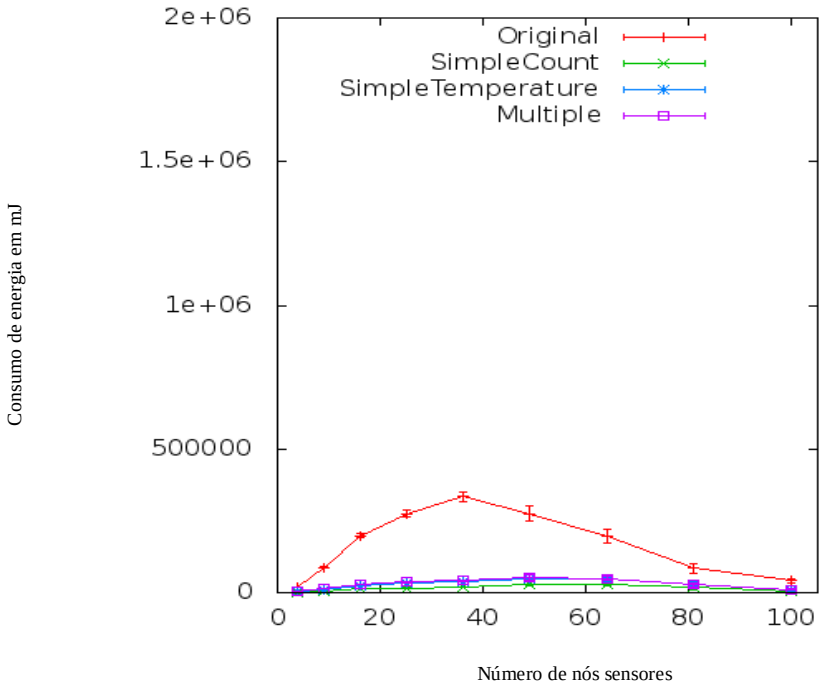
As Figuras 15 e 16 mostram os resultados do consumo de energia obtidos das simulações das quatro versões da aplicação. Elas descrevem o desempenho do consumo de energia para transmissão (Etrans) e recepção (Erecp) dos dados pelos nós sensores. Nós observamos o impacto do nosso método comparando o consumo de energia da regressão linear múltipla (nossa proposta) à regressão linear simples (trabalhos atuais).

Sob todas as condições, o consumo de energia é maior nas versões da aplicação que usam regressão linear simples ou múltipla baseadas na variável temperatura ao invés da variável tempo. Isto acontece porque quando usamos a variável independente coletada pelos nós sensores, suas amostras de leituras têm que ser enviadas ao sorvedouro. Visto isto, elas consomem mais energia que a versão que a variável tempo (contador) como variável independente.

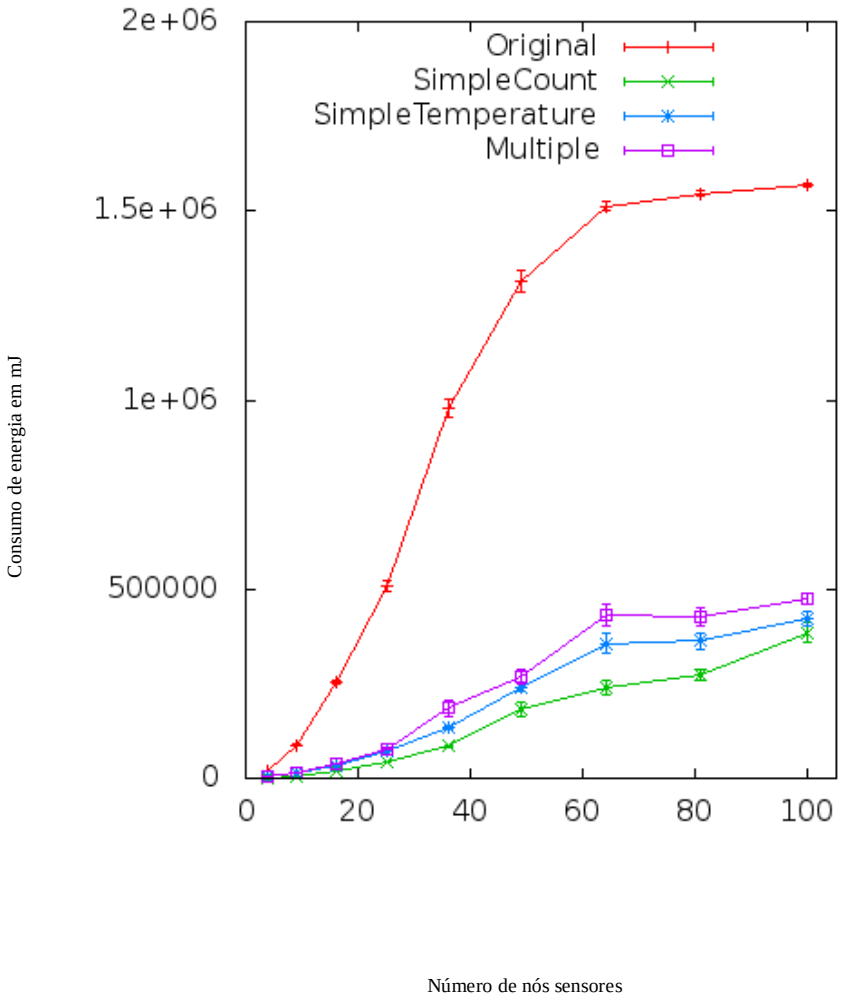
Figura 16. Média da energia do rádio em mJ, consumida pelas mensagens recebidas por roteamento gossip variando o número de nós sensores: (a) cenários #1 e #2; (b) cenários #3 e #4 e (c) cenários #5 e #6.



(a)



(b)



(c)

O consumo de energia devido à troca de mensagens entre nós sensores nos cenários #1, #2, #5 e #6 é representado na Figura 15 (a). A diferença da E_{trans} permanece constante em todas as versões da aplicação, mesmo quando a escalabilidade muda e as versões da aplicação que usam variáveis coletadas consomem duas vezes a E_{trans} da versão que não usa.

A relação entre a E_{trans} da aplicação original e as versões da aplicação com variáveis coletadas é cerca de 0,17 e com a versão da aplicação atual é cerca de 0,08. Nos cenários #3 e #4, a falha de comunicação afeta o consumo de energia [Figura 15(b)] de todas as versões da aplicação quando a densidade cai abaixo de 0,0278 (de 36 a 100 nós sensores). Nota-se também que o consumo de energia quando os cenários contendo mais nó sensores, ou

seja, quando se aproximam de 100, tende a ser menor devido à densidade mais alta resultar em um tempo de vida da rede menor.

Nós verificamos que o consumo de energia dos dados enviados pelos nós sensores na segunda versão da aplicação (SimpleCount) é a menor [Figura 15(a,b)], devido ao fato que esta aplicação não envia amostras de leituras ao sorvedouro. Esta aplicação é uma adotada por trabalhos atuais. Contudo, nós podemos também ver que o consumo de energia das terceira e quarta versões (SimpleTemperatura e Multiple, respectivamente) são os mais próximos do SimpleCount, em face da primeira versão da aplicação (Original). Nossa proposta usa o dobro da energia da solução baseada em regressão linear, mas seu consumo de energia é ainda baixo quando comparado à versão da aplicação sem predição (versão Original).

A quantidade de energia gasta para receber mensagens (E_{recp}) da difusão da aplicação na transmissão dos nós sensores vizinhos (roteamento *gossip*) é observada na Figura 16. Em alguns cenários [Figura 16(c)], a E_{recp} de nossa abordagem é cerca de três vezes menor que a versão da aplicação original, mas ainda consumindo mais energia que a abordagem usando regressão linear simples.

Nós podemos ver mais detalhes da porcentagem da economia de energia das três versões da aplicação que usam regressão linear simples ou múltipla em face da versão da aplicação original na Tabela 6.

Tabela 6. Porcentagem da economia de energia para enviar e receber dados em face da versão da aplicação original.

Versão	**Cenário #1**		**Cenário #2**		**Cenário #3**		**Cenário #4**		**Cenário #5**		**Cenário #6**	
da App.	**Env.**	**Gossiped**	**Env.**	**Gossiped**	**Env.**	**Gossiped**	**Env.**	**Gossiped**	**Env.**	**Gossiped**	**Env.**	**Gossiped**
2	0.93	0.93	0.93	0.93	0.93	0.89	0.93	0.92	0.92	0.87	0.92	0.87
3	0.87	0.87	0.87	0.87	0.87	0.82	0.87	0.85	0.86	0.82	0.86	0.82
4	0.86	0.86	0.86	0.86	0.86	0.81	0.86	0.84	0.85	0.79	0.85	0.80

Obs: Env. significa enviados e Gossiped significa recebidos.

Os resultados da correlação espacial ($C_{spacial}$) não mostraram diferença entre nossa abordagem e abordagem atuais, mas eles apontam para o fato que é essencial para a economia de energia. A quantidade de vezes que a correlação foi detectada é maior nos cenários onde existe densidade fixa de 0,25 nós sensores por m^2, ou seja, nos cenários #5 e #6.

Isto mostra que em situações de alta densidade, os pacotes não serão enviados duas vezes ao sorvedouro. Assim, nós evitamos a sobreposição e economizamos mais energia.

5.3 Avaliação de desempenho da precisão da predição

A Figura 17 mostra o desempenho da predição das três versões da aplicação às quais usam regressão linear sobre um dia de coleta de dados do *trace* do Intel Research Lab. O desempenho do erro e do melhoramento para as variáveis umidade e luminosidade garante que nossa proposta é melhor que atuais soluções.

Os resultados de SSerr e R^2 da predição da umidade [Figura 17(a,c)] mostra, para todos os cenários, que a menor precisão da predição foi obtida quando nós comparamos a regressão linear simples baseada nas variáveis tempo e temperatura como variáveis independentes.

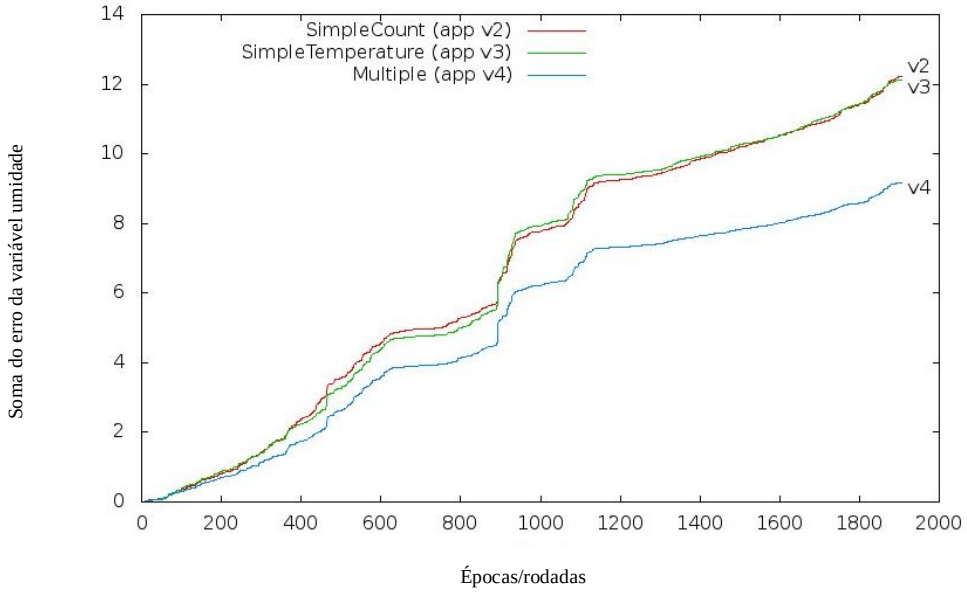
A melhor precisão de predição foi obtida quando a regressão linear múltipla foi usada. Porém, o consumo de energia é mais alto nas versões que usam regressão linear simples ou múltipla, baseada na variável temperatura como variável independente, ao invés da variável tempo, embora elas ainda obtenham melhores valores que a versão original.

Os resultados de SSerr e R^2 da predição da luminosidade [Figura 17(b,d)] mostram, para todos os cenários, que o maior erro de predição foi obtido quando nós comparamos a regressão linear simples baseada nas variáveis tempo e temperatura como variáveis independentes. O menor erro de predição foi obtido quando a regressão linear múltipla foi usada.

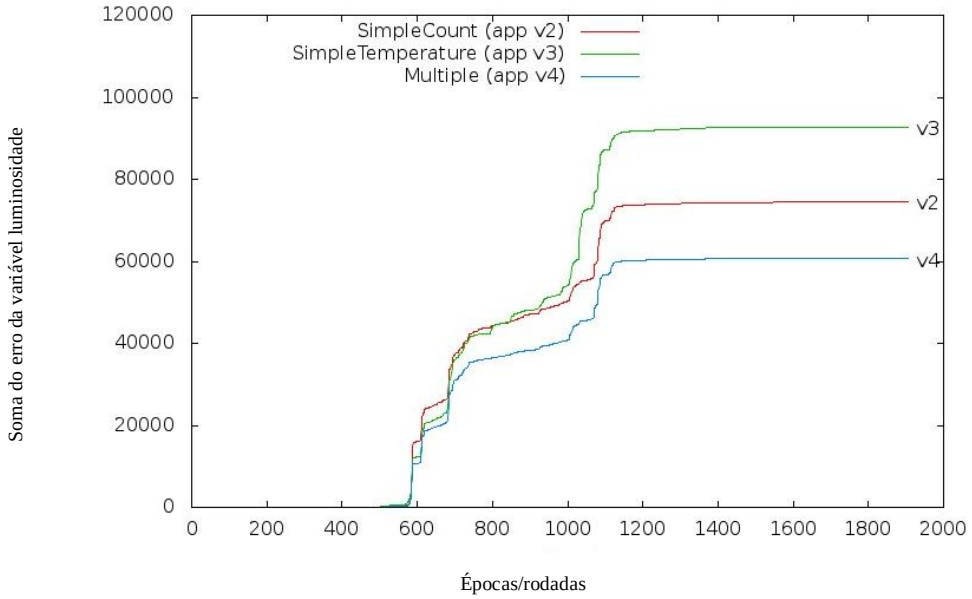
Nós também observamos que existem diferentes comportamentos nos resultados [Figura 17(b,d)] onde a variável luminosidade é irregular. Como visto anteriormente, as leituras coletadas da variável luminosidade no *trace* são irregulares, ou seja, os valores no *trace* não seguem uma sequência (aumenta ou diminui).

Isto provavelmente demonstra ruídos ou procedimentos de ligar e desligar luzes, e a alta sensibilidade do sensor de luminosidade.

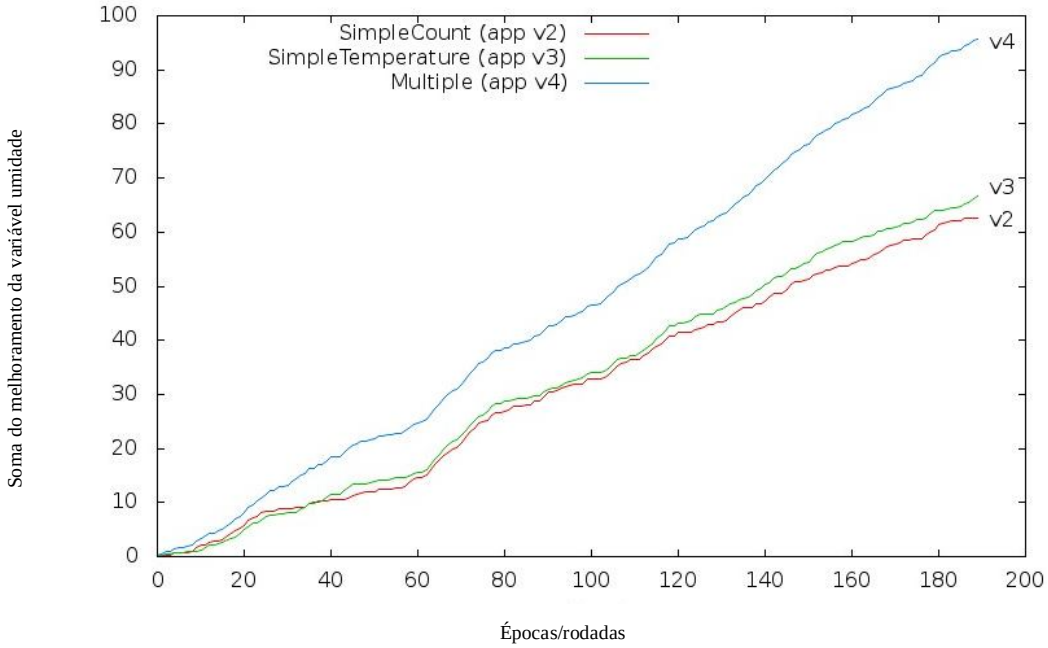
Figura 17. Soma do erro pelas rodadas em um dia do *trace* para as versões da aplicação às quais usam regressão linear (app v2 a app v4): (a) erro da umidade; (b) erro da luminosidade; (c) melhoramento da umidade; e (d) melhoramento da luminosidade.



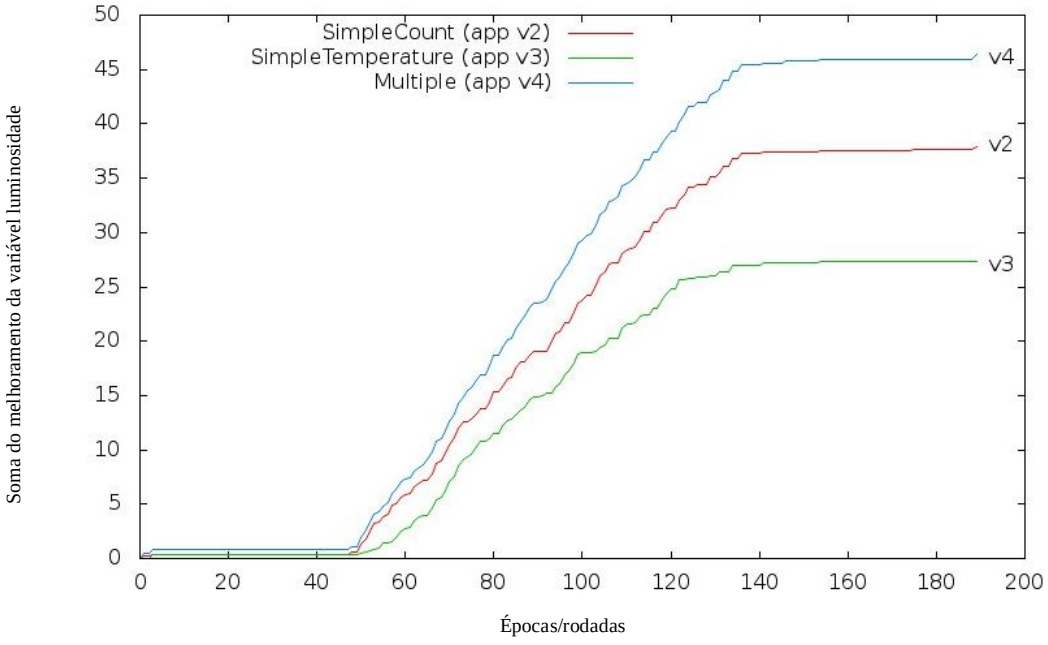
(a)



(b)


SimpleCount (app v2)
SimpleTemperature (app v3)
Multiple (app v4)
v4
v3
v2
Soma do melhoramento da variável umidade
Épocas/rodadas

(c)


SimpleCount (app v2)
SimpleTemperature (app v3)
Multiple (app v4)
v4
v2
v3
Soma do melhoramento da variável luminosidade
Épocas/rodadas

(d)

Assim, os resultados da predição da variável luminosidade mostram a desvantagem da regressão linear múltipla neste caso, embora ela ainda obtenha melhores resultados que a abordagem atual. Quando não existe correlação entre as variáveis, a precisão da predição diminui ou não trabalha corretamente. Portanto, nós sugerimos que ao usar predição baseada na regressão linear múltipla, o nó sensor verifique os melhoramentos e tome decisões de forma adaptativa, semelhante ao trabalho de Jiang *et al.* (2011) que determina o momento de usar a predição caso um determinado limiar seja atingido.

As Tabelas 7 e 8 mostram mais detalhes dos resultados de SSerr e R^2. Note que na maioria dos casos, o SSerr é maior na regressão linear simples e menor na regressão linear múltipla e o R^2 é inversamente proporcional ao SSerr. Isto mostra que é mais vantajoso usarmos regressão linear múltipla para aumentar a precisão da predição, do que continuar com a abordagem simples.

Tabela 7. Resultados de desempenho do SSerr e R^2 de todas as versões nos cenários #1, #4 e #6.

	Variável independente					
	Contador (Tempo)		**Temperatura**		**Contador e Temperatura**	
	Versão 2		**Versão 3**		**Versão 4**	
	SSerr	**R^2**	**SSerr**	**R^2**	**SSerr**	**R^2**
Temperatura	0,210300	0,296891	–	–	–	–
Umidade	9,355700	0,025813	2,033940	0,788210	0,203488	0,978811
Luminosidade	2,121380	0,000000	0,073135	0,965525	0,054342	0,974384

Tabela 8. Resultados de desempenho do SSerr e R^2 de todas as versões nos cenários #2, #3 e #5.

	Variável independente					
	Contador (Tempo)		**Temperatura**		**Contador e Temperatura**	
	Versão 2		**Versão 3**		**Versão 4**	
	SSerr	**R^2**	**Sserr**	**R^2**	**SSerr**	**R^2**
Temperatura	10.321800	0.290535	–	–	–	–
Umidade	4.964100	0.476813	8.583820	0.095316	0.185308	0.980470
Luminosidade	140.150060	0.869629	794.135000	0.261311	1075.060000	0.000000

5.4 *Trade-off* de nossa proposta

Depois dos resultados acima, nós decidimos repetir as simulações para avaliar o desempenho do consumo de energia e da precisão da predição e analisar o comportamento de nossa proposta. O compromisso entre estes dois desempenhos é natural, porque para aumentar a precisão da predição, nossa proposta envia mais amostras coletadas de uma variável. Portanto, nossa proposta consome mais energia que atuais soluções.

O relacionamento entre o consumo de energia e a precisão da predição não depende da quantidade de nós sensores na rede, porque a predição é feita de forma distribuída e localizada. Nós aprendemos que ela depende da quantidade de amostras. Portanto, quando nós aumentamos a quantidade de amostras, o consumo de energia diminui, o SSerr aumenta e o R^2 diminui, mas a RSSF não pode gastar muita energia.

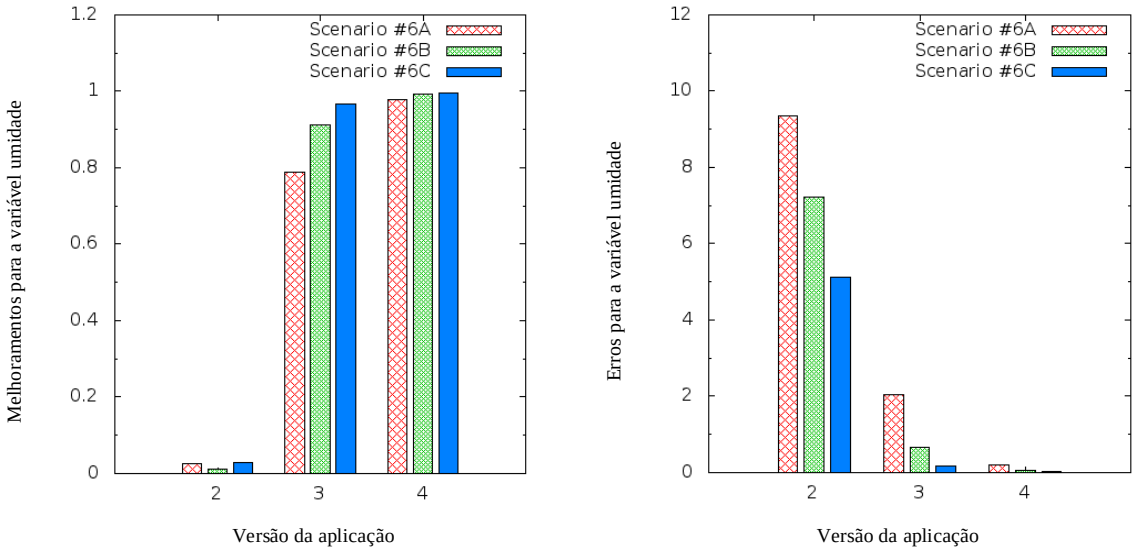
Assim, o cenário #6 foi simulado novamente, devido ao fato que ele obteve melhores resultados de desempenho que os outros cenários. Nesta nova fase de simulação, a quantidade de amostra variam de 6 (seis), 8 (oito) e 10 (dez), as quais nós respectivamente chamamos Cenário #6C, Cenário #6B e Cenário #6A.

Os resultados do consumo de energia nestes cenários de mensagens enviadas pelos nós sensores mostram que, para diminuir a quantidade de amostras de 10 (cenário #6A com 100 nós sensores) para 6 (cenário #6C com 100 nós sensores), a E_{trans} da rede aumenta de 1.834,32 μJ para 2.465,70 μJ. Isto acontece porque, reduzindo a quantidade de amostras, mais pacotes serão enviados. Os resultados de E_{recp} mostram que o consumo de energia aumenta de 489.567,40 μJ (cenário #6A com 100 nós sensores) para 578.866,80 μJ (cenário #6C com 100 nós sensores).

O melhoramento da predição da umidade para a versão da aplicação 4 (regressão linear múltipla) diminui de 0,995868 para 0,978811 [Figura 18(a)] e o SSerr da umidade aumenta de 0,021840 para 0,203488 [Figura 18(b)]. Deve-se também ser notado que a versão da aplicação 4 sempre tem melhores resultados que as outras versões.

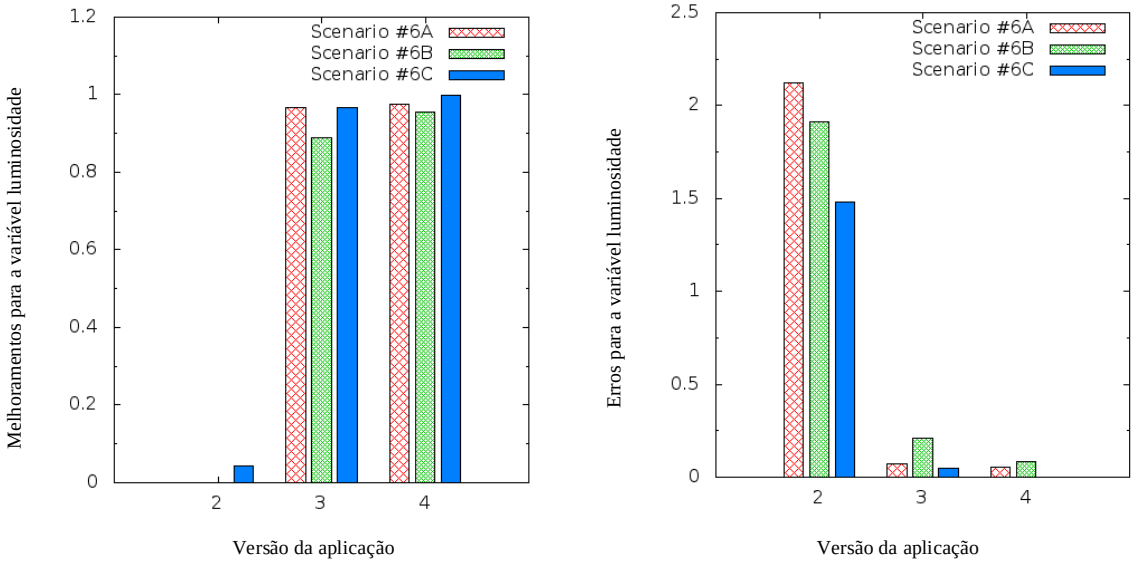
Os resultados para a predição da luminosidade são um pouco diferentes dos resultados para umidade, mas eles mostram o mesmo comportamento. O melhoramento da predição da luminosidade para a versão da aplicação 4 (regressão linear múltipla) diminui de 0,999752 para 0,974384 [Figura 19(a)] e o SSerr da luminosidade aumenta de 0,000384 para 0,054342 [Figura 19(b)].

Figura 18. Melhoramento e SSerr da predição executada por versões da aplicação para a variável umidade, alterando a quantidade de amostras (Cenário #6A – dez amostras, Cenário #6B – oito amostras e Cenário #6C – seis amostras): (a) Melhoramento para umidade; e (b) SSerr para umidade.



(a) (b)

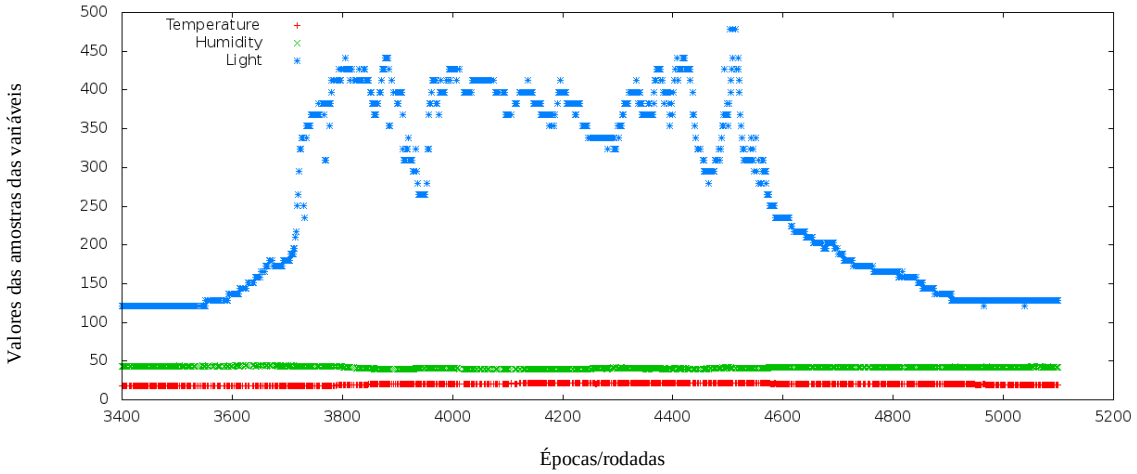
Figura 19. Melhoramento e SSerr da predição executada por versões da aplicação para a variável luminosidade, alterando a quantidade de amostras (Cenário #6A – dez amostras, Cenário #6B – oito amostras e Cenário #6C – seis amostras): (a) Melhoramento para luminosidade; e (b) SSerr para luminosidade.



(a) (b)

Os resultados obtidos da predição da variável luminosidade foram diferentes dos resultados obtidos da predição da variável umidade. Então, nós verificamos o comportamento das três variáveis coletadas e usamo-las na nossa avaliação de desempenho. A Figura 20 mostra épocas de um dia de coletas de dados, onde a correlação entre as variáveis é baixa.

Figura 20. Valores da variável luminosidade por épocas de um dia de coleta, onde a variável luminosidade é menos correlacionada com as variáveis temperatura e umidade.



Note que nas épocas que variam de 3.550 a 4.900, a variável luminosidade aumenta muito. Consequentemente, as regressões lineares simples e múltipla tendem a piorar a precisão da predição. Isto explica alguns resultados anormais quando nós usamos a variável luminosidade como variável independente. Devido à sensibilidade desta variável e os momentos de falta de correlação, a variável luminosidade gera erros de predição que são difíceis de recuperar.

Ainda estamos realizando experimentos usando outros mecanismos de predição para melhorar a precisão das variáveis descorrelacionadas e alguns testes apontam para abordagens usando inteligência computacional para resolver tal problema em aberto.

5.5 Considerações finais

O melhoramento da precisão da predição ainda é um desafio para a comunidade científica, uma vez que questões como complexidade da solução e redução no consumo de energia ainda permanecem abertas.

Quando melhoramos a precisão da predição, automaticamente aumentamos o consumo de energia e podemos comprometer os recursos limitados das RSSF. Mas sob esta ótica, nós estamos realizando testes no Scilab usando o algoritmo *K-means* para separar as amostras descorrelacionadas em *clusters* e somente depois calcularmos os coeficientes da função de regressão linear simples.

Os resultados têm sido melhor que a regressão linear simples, mas ainda necessitam de ajustes para implantarmos em nós sensores, como a adequação da quantidade de amostras ideais para compensar os gastos com energia devido ao envio dos centroides pelos nós sensores ao sorvedouro.

Embora os resultados iniciais apontem para uma melhora na abordagem simples, a necessidade de acrescentarmos um mecanismo para auxiliar a abordagem simples gera maior consumo de energia.

6 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho propõe uma solução de redução de dados não complexa e mais precisa que as soluções mais simples encontradas na literatura, baseada na predição de dados usando regressão linear múltipla.

Inicialmente nós identificamos uma abordagem simples de predição baseada na regressão linear simples e estudamos as suas desvantagens em relação à precisão da predição. O estudo serviu de base para apontarmos uma solução multivariada de dados explorando as correlações espaciais e temporais, com o objetivo de diminuir o erro provocado no processo de redução de dados.

Nossa proposta supera a abordagem simples em cerca de 50% na precisão da predição da variável umidade e cerca de 21% na precisão da predição da variável luminosidade. Mas como resultado desse melhoramento da precisão da predição, a nossa proposta gasta cerca do dobro da energia consumida pela a abordagem simples. Embora esta seja uma desvantagem, as aplicações que requerem alto grau de precisão podem não suportar os erros provocados pela redução de dados da abordagem simples.

Para diminuir os efeitos do aumento de consumo de energia, nós acrescentamos um mecanismo de identificação da correlação espacial para evitar que dados sobrepostos sejam enviados ao sorvedouro. Portanto, ao calcular os coeficientes da função de regressão linear múltipla, cada nó sensor verifica se os mesmos já foram enviados por algum nó sensor vizinho. Caso afirmativo, o nó sensor descarta o pacote contendo os coeficientes da função e as amostras necessárias como entrada (variável independente) da equação.

O problema de precisão da predição descrito na seção 1.1 (Capítulo 1) foi resolvido conforme a nossa proposição descrita na seção 3.3 e comprovado através dos resultados dos experimentos na seção 5.3, nas quais aplicamos de forma inédita a correlação espaço-temporal multivariada para redução de dados em RSSF.

6.1 Contribuições da proposta

As principais contribuições deste trabalho são:

(i) proposição de um método de redução de dados baseado em regressão linear múltipla para explorar a correlação temporal multivariada melhorando a precisão da predição, o qual ainda não foi aplicado a RSSF;

(ii) proposição de um mecanismo de redução de dados que auxilia o método citado acima, o qual é baseado na distância Euclidiana para explorar a correlação espacial multivariada e ameniza o gasto de energia provocado por este método;

(iii) discussão da necessidade de planejarmos a implementação da solução de redução de dados baseada em predição usando análise de correlação nas variáveis coletadas em campo, para definir a variável independente mais correlacionada e com isso aumentar a precisão.

6.2 Trabalhos futuros

Nós estamos na fase inicial de aplicar a nossa solução de redução de dados para auxiliar o monitoramento de colméias de abelhas em parceria com a Embrapa Meio Norte em Teresina. Este monitoramento atualmente é feito através de equipamentos que coletam dados de uma estação meteorológica e nós iremos modificar para implantar a tecnologia de RSSF.

A substituição de tecnologia irá demandar de nossa proposta um melhor nível de precisão para coletar variáveis internas e externas às colméias, tais como: temperatura, umidade e rastreamento do comportamento das abelhas.

Portanto, iremos descobrir qual o comportamento de nossa proposta na prática em cenários real, ao invés de apenas simulações.

Quanto ao problema de aumento do consumo de energia de nossa proposta em relação à abordagem simples, nós estamos estudando uma forma de incrementar a precisão da predição na própria abordagem simples através do uso do algoritmo *K-means*.

Como as amostras de luminosidade tendem a serem descorrelacionadas às outras variáveis e até mesmo com as suas próprias amostras anteriores, o *K-means* pode separar as amostras mais correlacionadas em *clusters* e assim diminuir o erro da predição.

REFERÊNCIAS

AKYILDIZ, I.F.; SU, W.; SANKARASUBRAMANIAM, Y.; CAYIRCI, E. **Wireless sensor networks: A survey**. Computer Network. 2002, 38, 393-422.

AKYILDIZ, I.F.; VURAN, M.C.; AKAN, O.; SU, W. **Wireless Sensor Networks: A Survey Revisited**. In: Computer Networks Journal (Elsevier Science), Vol. 45, no. 3, 2004.

ANASTASI, G.; CONTI, M.; DI FRANCESCO, M.; PASSARELLA, A. **Energy conservation in wireless sensor networks: A survey**, Ad Hoc Network, Vol. 7, Issue 3, May, 2009, pp. 537-568, Elsevier Science Publishers B. V., Amsterdam, The Netherlands.

CARVALHO, C.G.N.; GOMES, D.G.; AGOULMINE N.; SOUZA, J.N. **Multiple linear regression to improve prediction accuracy in WSN data reduction**. In Proceedings of 7th Latin American Network Operations and Management Symposium, Quito, Ecuador, 10–11 October 2011.

CARVALHO, C.G.N.; GOMES, D.G.; AGOULMINE, N.; SOUZA, J.N. **Improving Prediction Accuracy for WSN Data Reduction by Applying Multivariate Spatio-Temporal Correlation**. Sensors (Basel), v. 11, p. 10010-10037, 2011.

GAMA, J.; GABER, M.M. **Learning from Data Streams: Processing Techniques in Sensor Networks**, Springer: Berlin/Heidelberg, Germany, 2007.

GOEL, S.; IMIELINSKI, T. **Prediction-based monitoring in sensor networks: Taking lessons from MPEG**. SIGCOMM Computer Communication. 2001, 31, 82-98.

HAIR, J.; BLACK, W.; BABIN, B.; ANDERSON, R. **Multivariate Data Analysis**, Prentice Hall: Englewood Cliffs, NJ, USA, 1998.

JIANG, H.; JIN, S.; WANG, C. **Prediction or not? An energy-efficient framework for clustering-based data collection in wireless sensor networks**. IEEE Trans. Parall. Distrib. Syst. 2011, 22, 1064-1071.

JURDAK, R.; RUZZELLI, A.; O'HARE, G. **Adaptive radio modes in sensor networks: How deep to sleep?** In Proceedings of the 5th Annual IEEE Communications Society

Conference on Sensor, Mesh and Ad Hoc Communications and Networks, San Francisco, CA, USA, 16–20 June 2008.

KOSHY, J.; WIRJAWAN, I.; PANDEY, R.; RAMIN, Y. **Balancing computation and communication costs: The case for hybrid execution in sensor networks**. Ad Hoc Netw. 2008, 6, 1185-1200.

KRIEF, F.; BENNANI, Y.; GOMES, D.G.; SOUZA, J.N. **LECSOM: A low-energy routing algorithm based on SOM clustering for static and mobile wireless sensor Networks**. Int. J. Commun. Antenna Propag. 2011, 1, 55-63.

KULKARNI, R.; FANDRSTER, A.; VENAYAGAMOORTHY, G. **Computational intelligence in wireless sensor networks: A survey**. IEEE Communication Survey. Tutor. 2011, 13, 68-96.

LI, J.; DESHPANDE, A.; KHULLER, S. **On computing compression trees for data collection in wireless sensor networks**. In Proceedings of INFOCOM'10: the 29th Conference on Information Communications, San Diego, CA, USA, 15–19 March 2010; pp. 2115-2123,

LIU, C.; WU, K.; PEI, J. **An energy-efficient data collection framework for wireless sensor networks by exploiting spatiotemporal correlation**. IEEE Trans. Parall. Distrib. Syst. 2007, 18, 1010-1023.

MATOS, T.B.; BRAYNER, A.; MAIA, J.E.B. **Toward in-network data prediction in wireless sensor networks**. In Proceedings of the ACM Symposium on Applied Computing, Sierre, Switzerland, 22–26 March 2010; pp. 592-596.

MEITALOVS, J.; HISTJAJEVS, A; STALIDZANS, E., **Automatic Microclimate Controlled Beehive Observation System**, 8th International Scientific Conference 'Engineering for Rural Development, Latvia University of Agriculture, pp. 265-271, 2009.

NAKAMURA, E.F., FIGUEIREDO, C.M.S., NAKAMURA, F.G., LOUREIRO, A.A.F., **Diffuse: A Topology Building Engine for Wireless Sensor Networks**. Signal Processing, v. 87, p. 2991-3009, 2007.

NAKAMURA, E.F., LOUREIRO, A.A.F., FRERY, A.C., **Information fusion for wireless sensor networks: Methods, models, and classifications**, ACM Comput. Surv., 2007, vol. 39, number 3, Sep, ACM, New York, NY, USA.

NAKAMURA, E.F., LOUREIRO, A.A.F., **Information fusion in wireless sensor networks**. In Proceedings of the 2008 ACM SIGMOD, International Conference on Management of Data (SIGMOD '08). ACM, New York, NY, USA, 1365-1372.

PERLA, E., CATHÁIN, A.Ó., CARBAJO, R.S., HUGGARD, M., Mc GOLDRICK, C., **PowerTOSSIM z: realistic energy modeling for wireless sensor network environments**, Proceedings of the 3nd ACM Workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks, PM2HW2N '08, Vancouver, British Columbia, Canada, p.p 35-42, ACM, New York, NY, USA.

ROCHA, A.; SANTOS, I.; PIRMEZ, L.; DELICATO, F.; GOMES, D.; SOUZA, J. **Semantic clustering in wireless sensor networks**. IFIP Adv. Inform. Commun. Tech. 2010, 327, 3-14.

ROCHA, A.R.; DELICATO, F.C.; SOUZA, J.N.; GOMES, D.G.; PIRMEZ, L. **A semantic middleware for autonomic wireless sensor networks**. In Proceedings of the Workshop on Middleware for Ubiquitous and Pervasive Systems, Dublin, Ireland, 16–19 June 2009; pp. 19-25.

SANTINI, S.; ROMER, K. **An adaptive strategy for quality-based data reduction in wireless sensor networks**. In Proceedings of INSS 2006: 3rd International Conference on Networked Sensing Systems, Chicago, IL, USA, 31 May–2 June 2006.

SEO, S.; KANG, J.; RYU, K.H. **Multivariate stream data reduction in sensor network applications**. In Proceedings of EUC Workshops, Nagasaki, Japan, 6–9 December 2005; pp. 198-207.

SILVA, O.; AQUINO, A.; MINI, R.; FIGUEIREDO, C. **Multivariate reduction in wireless sensor networks.** In Proceedings of IEEE Symposium on Computers and Communications, Sousse, Tunisia, 5–8 July 2009; pp. 726-729.

SKORDYLIS, A.; GUITTON, A.; TRIGONI, N. **Correlation-based data dissemination in traffic monitoring sensor networks**. In Proceedings of CoNEXT ’06, Lisbon, Portugal, 4–7 December 2006.

TAHIR, M.; FARRELL, R. **Optimal communication-computation tradeoff for wireless multimedia sensor network lifetime maximization**. In Proceedings of WCNC'09: the IEEE Conference on Wireless Communications & Networking Conference, Budapest, Hungary, 5–8 April 2009.

VURAN, M.C.; AKAN, O.B.; AKYILDIZ, I.F. **Spatio-temporal correlation: Theory and applications for wireless sensor networks**. Computer Network. 2004, 45, 245-259.

WANG, H.; AGOULMINE, N.; MA, M.; JIN, Y. **Network lifetime optimization in wireless sensor networks**. IEEE J. Sel. Areas Communication. 2010, 28, 1127-1137.

XU, Y.; LEE, W.-C. **On localized prediction for power efficient object tracking in sensor networks**. In Proceedings of 23rd International Conference on Distributed Computing Systems Workshops, Providence, RI, USA, 19–22 May 2003; pp. 434-439.

YICK, J.; MUKHERJEE, B; GHOSAL, D., **Wireless sensor network survey**, Computer Networks, Volume 52, Issue 12, 22 August 2008, pp. 2292-2330, 2008.