

Um Modelo Autônomo de Processamento de Consultas Espaciais para Redes de Sensores Urbanas

Marcos A. Carrero¹, Rone I. da Silva², Carmem S. Hara¹, Aldri L. dos Santos¹

¹DINF – Universidade Federal do Paraná – UFPR – Paraná, Brasil

²DTECH – Universidade Federal de São João del-Rei – UFSJ – Minas Gerais, Brasil

{macarrero, carmem, aldri}@inf.ufpr.br, rone@ufs.j.edu.br

Abstract. *Wireless Sensor Networks (WSN) in urban environments manage a large amount of sensing data. The deployment of spatial query processing in a decentralized and autonomous large-scale WSN is a major challenge due to the network resources constraints. This paper proposes ASQPM, a scalable and autonomous model for data storage and spatial query processing. Scalability is provided by grouping sensors into clusters based on the spatial similarity of their readings. The query processing efficiency relies on the concept of repositories, which are regions in the monitored area that concentrate information, storing the readings of a set of clusters. The experimental results show that it is more effective for query processing than classical approaches.*

Resumo. *As Redes de Sensores sem Fio (RSSFs) urbanas lidam com grande quantidade de informações coletadas. O processamento de consultas espaciais em RSSFs de larga escala é um grande desafio devido às limitações de recursos da rede. Este trabalho propõe o ASQPM, um modelo escalável e autônomo para o armazenamento de dados e processamento de consultas espaciais. A escalabilidade resulta da estratégia de formação de agrupamentos de sensores que possuem similaridade espacial de leituras. A eficiência da consulta é determinada pelo conceito de repositório, que são regiões de concentração de dados de um conjunto de agrupamentos. Os resultados experimentais mostram que o seu processamento de consulta é mais eficaz do que as abordagens clássicas.*

1. Introdução

As Redes de Sensores Sem Fio (RSSFs) oferecem uma infra-estrutura para o desenvolvimento de aplicações em várias áreas de interesse. São exemplos de aplicações o monitoramento do ambiente, sistemas militares, de vigilância e de saúde [Yick et al. 2008]. Em geral, os sensores possuem recursos limitados como baixa largura de banda, comunicação de curto alcance, armazenamento limitado e processamento. Em ambientes urbanos, os sensores podem ser densamente distribuídos a fim de coletar diferentes atributos ambientais tais como temperatura, pressão, umidade, luminosidade e poluição. Além disso, os dados coletados podem ser armazenados nos próprios sensores, proporcionando o desenvolvimento de aplicações que executam consultas na rede, sem depender de um servidor central. Em particular, em RSSFs de grande escala, manter todo o processamento centralizado aumenta o custo de comunicação [Can and Demirbas 2013], sendo menos escalável que abordagens descentralizadas [Coman et al. 2007].

As consultas processadas por uma RSSF podem ser classificadas em consultas espaciais e por valor. As consultas espaciais têm como objetivo obter os valores coletados por sensores dentro de uma área geográfica de interesse. As consultas por valor, por outro lado, determinam quais sensores coletaram dados dentro de um intervalo de valores. Neste artigo o foco é sobre consultas espaciais. A disseminação de consultas é uma tarefa difícil devido ao alto custo de comunicação e de restrições dos recursos da rede. No entanto, explorando algumas características encontradas nas RSSFs é possível reduzir a sobrecarga na comunicação [Le et al. 2008, Cheng et al. 2011]. Em algumas aplicações, a correlação espaço-temporal presente nas leituras de sensores próximos possibilita reduzir o número de transmissões na rede. No caso da correlação espacial, leituras de sensores próximos tendem a ser muito similares nos seus valores. Em relação à segunda característica, leituras consecutivas tendem a ser muito próximas no tempo [Vuran et al. 2004]. Explorando-se a correlação espacial, que é o foco deste trabalho, é possível alcançar um requisito muito desejável que é a escalabilidade, organizando em grupos (*clusters*) sensores que apresentarem leituras similares em seus valores [Hung et al. 2012].

Além da característica espacial, a gestão do armazenamento de dados em RSSFs possui impacto significativo no uso de recursos e no desempenho do processamento de consultas [Yu et al. 2010]. Em geral, os dados coletados podem ser armazenados localmente no próprio sensor, em uma base de dados externa, ou em *repositórios* de dados distribuídos na própria rede. Embora a melhor escolha dependa do contexto da aplicação, nota-se que o modelo de armazenamento de dados em *repositórios* oferece uma abordagem interessante entre o custo da consulta e o custo do armazenamento dos dados [Xie et al. 2014]. Logo, a eficiência no processamento de consultas baseia-se no estabelecimento de *repositórios* na rede, reduzindo-se o número de encaminhamento de consultas, isto é, de saltos na rede, para se obter o resultado desejado.

Inspirados por estes desafios, trabalhos recentes que lidam com o agrupamento de sensores com base na similaridade de dados têm sido propostos, como o DCSSC [Le et al. 2008], o SIDS [Furlaneto et al. 2012] e o DCASC [Ma et al. 2011]. Nos trabalhos DCSSC e DCASC, sensores com alta correlação em suas leituras são agrupados e um sensor líder é eleito como representante do grupo. No entanto, nenhum deles especifica um modelo para disseminação de consultas espaciais e nem a criação de *repositórios* na rede para reduzir o número de acessos aos agrupamentos, de modo que o processamento da consulta em uma rede de grande escala pode ser cara. Além disso, outra desvantagem do DCSSC é que as consultas são iniciadas a partir de um único ponto de entrada externo. Por outro lado, o SIDS combina a formação de agrupamentos com base na similaridade espacial dos dados com a criação de *repositórios*, porém não é um modelo autônomo pois depende de uma entidade externa para coordenar o processo de agrupamento.

Este trabalho propõe um modelo autônomo de processamento de consultas espaciais para RSSFs urbanas, chamado ASQPM (*Autonomous Spatial Query Processing Model*). O ASQPM é um modelo hierárquico e distribuído onde cada sensor armazena localmente os dados de monitoramento, e sensores com alta correlação espacial são organizados em grupos. No nível de agrupamento, um sensor é eleito para ser o representante do grupo e sua leitura sensorizada representa a leitura do seu agrupamento. No nível de *repositório*, sensores específicos que servem como centro de dados armazenam informações de um conjunto de agrupamentos distintos, reduzindo o custo

total de comunicação no processamento de consultas. O modelo é autônomo porque a própria rede atua na formação de agrupamentos, na escolha do representante do grupo e na definição de *repositórios*, sem depender de uma entidade externa central. Em nosso entendimento, este é o primeiro modelo autônomo que combina similaridade espacial de leituras com armazenamento distribuído em *repositórios* para reduzir a comunicação na rede no processamento de consultas espaciais no âmbito de RSSFs urbanas. Simulações mostram que o modelo reduz consideravelmente o tempo de processamento das consultas espaciais.

O restante deste artigo está organizado da seguinte forma. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 descreve o funcionamento do modelo ASQPM. A Seção 4 mostra os resultados obtidos na avaliação de desempenho e a conclusão e trabalhos futuros são discutidos na Seção 5.

2. Trabalhos Relacionados

Diferentes abordagens têm sido propostas para tratar o armazenamento de dados, a construção de agrupamentos e o processamento de requisições espaciais para RSSFs. O gerenciamento dos dados na rede possibilita o desenvolvimento de aplicativos eficientes de processamento de consultas espaciais. Além disso, abordagens clássicas frequentemente usadas para tratar a escalabilidade das aplicações são as técnicas de agrupamentos de sensores. Nas redes que apresentam correlação no dado sensoriado, os sensores podem ser agrupados com base na similaridade de suas leituras. Contudo, investigando a revisão literária sobre requisição de dados espaciais apresentada em [Da Silva et al. 2014], verificou-se que nenhum trabalho utiliza agrupamentos por similaridade de dados na rede a fim de melhorar o desempenho do processamento de consultas.

Estratégias usadas por CAG [Yoon and Shahabi 2007], DCSSC [Le et al. 2008] e DCASC [Ma et al. 2011] descrevem mecanismos que lidam com a característica da similaridade de dados sensoriados. Em geral, nestes trabalhos os sensores são agrupados e há seleção de um líder como representante do grupo, de modo que os sistemas sejam escaláveis. Contudo, nenhuma destas abordagens são totalmente autônomas e eficientes no processamento de consultas espaciais. O CAG precisa constantemente reconstruir os agrupamentos da rede para estabelecer novas consultas. O DCSSC, por outro lado, depende da estação base para gerenciar a criação dos agrupamentos. O DCASC apresenta um algoritmo para construção de agrupamentos de dados sensoriados, no qual este trabalho se inspirou, porém o DCASC não oferece suporte para consultas espaciais.

Um modelo de armazenamento de dados que leva em conta a similaridade de leituras é descrita por SIDS [Furlaneto et al. 2012]. Ele estabelece um mecanismo de agrupamentos baseados na similaridade espacial dos dados, um esquema para eleição de líderes e uma estrutura de indexação para evitar inundações. Além disso, locais de concentração de informação, denominados de *repositórios* de dados são formados na rede, consistindo de pelo menos dois sensores líderes em região de borda. No entanto, o SIDS é um modelo centralizado que depende da estação base para criar os agrupamentos.

Entre os algoritmos de consultas espaciais, o IBIS [da Silva et al. 2011] descreve uma proposta para RSSFs eficiente na redução do consumo de energia da rede. O IBIS é um mecanismo para processamento de requisições espaciais irregulares, baseado na criação de itinerários. Durante a requisição de uma consulta espacial, ele cria um itinerário

para encaminhar e agregar os dados sensorizados da região de interesse. No entanto, o IBIS não leva em conta a similaridade de dados e a formação de agrupamentos de sensores. Assim, um modelo de processamento de consultas espaciais que atenda os requisitos encontrados em RSSFs urbanas se faz necessário. Tal modelo deve ser capaz de combinar estratégias de agrupamentos por similaridade de dados e de formação de *repositórios*, adaptando-as para dar suporte a uma rede com alta densidade dos sensores, provendo escalabilidade e mantendo completamente a autonomia da rede.

3. O Modelo ASQPM

Esta seção descreve um modelo hierárquico e distribuído para realizar o processamento de consultas espaciais eficientes em redes que apresentam similaridade espacial nas leituras dos sensores. Este modelo, chamado ASQPM (*Autonomous Spatial Query Processing Model*), organiza em grupos sensores próximos que possuam similaridade em suas leituras. Uma vez definidos os agrupamentos, inicia-se o processo de seleção de sensores denominados de *repositórios*, responsáveis por armazenar informações de um conjunto de agrupamentos distintos. A seguir são detalhados o modelo de rede, os algoritmos de correlação espacial, de formação agrupamentos e de escolha de *repositórios*, bem como o modo de processamento das consultas espaciais.

3.1. Modelo de Rede

Uma RSSF é representada como um grafo $G = (V, L)$, onde $V = \{s_1, \dots, s_n\}$ é um conjunto de sensores dispersos sobre a área monitorada M e L é o conjunto de ligações entre pares de sensores tal que $(s_i, s_j) \in L$ se s_i e s_j estão dentro do raio de comunicação um do outro. Diz-se que a distância entre s_i e s_j é de um *salto* e que s_i e s_j são *vizinhos*. A comunicação entre dois sensores quaisquer requer a existência de um caminho de roteamento $R = \{(s_1, s_2), (s_2, s_3), \dots, (s_{n-1}, s_n)\}$ tal que s_1 é o sensor que origina a mensagem e s_n é o seu destino final. Portanto, a comunicação em RSSFs depende da cooperação do repasse de mensagens na rede, e a escolha da rota R é tarefa do protocolo de roteamento. Neste trabalho, é assumido que os sensores são estáticos e, portanto, possuem um posicionamento geográfico fixo. Assume-se também que cada sensor realiza o sensoriamento de mais de um atributo do ambiente. As leituras de um sensor s são representadas por uma tupla $X = (x_1, x_2, \dots, x_n)$, na qual cada x_i corresponde a um tipo de atributo de dado sensorizado como temperatura, umidade, luminosidade e poluição atmosférica.

Como exemplo, considere uma RSSF urbana monitorando diferentes valores de atributos ambientais, tais como temperatura, umidade, luminosidade e poluição do ar (CO , CO_2). A Figura 1(a) mostra os sensores densamente espalhados sobre algumas regiões da cidade, como parques, jardins, ruas e avenidas. Em cenários complexos, como o cenário urbano, se faz necessário trabalhar com uma granulação fina no monitoramento do ambiente [Thepvilojanapong et al. 2010]. O estudo mostrou que vários fatores ambientais influenciam na variação da temperatura, tais como a presença de árvores, a largura de estradas e diferenças entre as regiões geográficas. Além disso, devido à topografia irregular da cidade, verificam-se diferentes índices de concentração de poluentes mesmo em lados opostos da rua [Resch et al. 2009]. Assim, justifica-se o uso de uma RSSF densa como forma adequada de monitorar tal ambiente complexo [Muller et al. 2013].

Neste trabalho, sensores próximos são agrupados com base na similaridade espacial dos dados, como ilustra a Figura 1(b). Cada agrupamento possui um sensor líder,

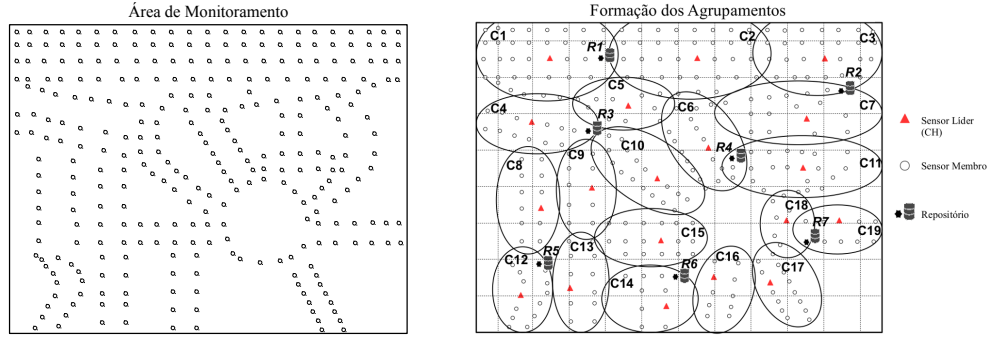


Figure 1. Visão geral do cenário e do modelo ASQPM

que é responsável pelo armazenamento de informações relevantes sobre os demais sensores do grupo. Os agrupamentos são definidos em áreas geográficas contíguas. Os dados amostrados dentro de cada agrupamento apresentam alta correlação entre os seus membros. Portanto, com base no entendimento acima, apenas os dados coletados pelo líder são relevantes durante o processamento de consultas, não necessitando das informações coletadas pelos demais membros. Assim, a sobrecarga no processamento e a comunicação dentro do agrupamento diminuem, resultando em menos uso de recursos.

A seleção do líder foi inspirada pelo método de correlação espacial apresentado em [Ma et al. 2011]. Lembre-se que nós sensores monitoram uma série de informações sobre o ambiente, onde $X = (x_1, x_2, \dots, x_n)$ denota as leituras do sensor s_i , $Y = (y_1, y_2, \dots, y_n)$ denota as leituras do sensor s_j e $N(i)$ denota o conjunto de vizinhos que estão a um *salto* de distância do sensor s_i . Portanto, o líder selecionado será aquele que possuir alta correlação entre suas leituras e as leituras de seus sensores vizinhos. O processo da seleção do líder se faz em várias etapas. Na primeira equação, a distância euclidiana entre as leituras do sensor s_i em relação ao sensor s_j é calculada como

$$d_{ij} = \sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2 + \dots + |x_n - y_n|^2} \quad (1)$$

Então, a média das distâncias \bar{d}_i , entre s_i e seus vizinhos, é dada por

$$\bar{d}_i = \frac{1}{|N(i)|} \sum_{j \in N(i)} d_{ij} \quad (2)$$

Além disso, o desvio médio absoluto de d_i é dado por

$$D(d_i) = \frac{1}{|N(i)|} \sum_{j \in N(i)} (d_{ij} - \bar{d}_i)^2 \quad (3)$$

Portanto, o peso da correlação espacial $w(s_i)$ ($0 \leq w(s_i) \leq 1$) de s_i é dado por

$$w(s_i) = \frac{\left[\sum_{j \in N(i)} |d_{ij} - \bar{d}_i| \right]^2}{|N(i)|^2 D(d_{ij})} = \frac{\left[\sum_{j \in N(i)} |d_{ij} - \bar{d}_i| \right]^2}{|N(i)| \sum_{j \in N(i)} (d_{ij} - \bar{d}_i)^2} \quad (4)$$

De acordo com a equação acima, cada sensor s_i calcula um peso $w(s_i)$ que indica o quão correlacionados estão suas leituras com relação às leituras de seus vizinhos. Grandes valores de $w(s_i)$ indicam que as leituras de s_i e as leituras de seus vizinhos tendem a ser

altamente correlacionadas. De fato, um sensor s_i será selecionado como líder com base em um limite τ definido pelo usuário. Assim, a seleção do líder ocorre se $w(s_i) \geq \tau$. Intuitivamente, em uma vizinhança, o nó com maior peso será eleito como líder de um grupo de sensores que possuem similaridade numérica de leituras.

No modelo ASQPM existem três níveis de componentes: o sensor coletor s , o agrupamento h ao qual s pertence e o *repositório* r no qual os dados do agrupamento h são armazenados. No nível inferior, cada sensor s_i deve armazenar sua leitura atual, o peso da correlação $w(s_i)$, as leituras de seus vizinhos *readingsOfNeighbors*(s_i), uma lista de seus vizinhos $N(i)$ e o líder do agrupamento ao qual ele pertence $CH(s_i)$. Assume-se que o mecanismo de disseminação de consulta depende de um protocolo de roteamento geográfico, de modo que cada sensor tem que armazenar sua própria informação geográfica *position*(s_i). No nível superior, formado pelos agrupamentos e seus líderes, um sensor líder h armazena sua leitura atual, a posição geográfica dos membros do seu agrupamento $GC(h)$ (Geografia do Cluster) e o *repositório* no qual seus dados são armazenados $R(h)$. Observe que a partir do conjunto GC é possível determinar o menor retângulo delimitador que contém todos os sensores de um agrupamento $MBR(h)$. No nível mais alto, definido pelos *repositórios*, um sensor r estabelecido como *repositório* armazena as leituras de um conjunto de líderes, informações sobre as posições geográficas dos seus sensores GR (Geografia do Repositório) e uma lista dos *repositórios* mais próximos *knownRepo*(r). É importante observar que um mesmo sensor pode assumir diversos papéis (*role*) simultaneamente, ou seja, além de um membro de um agrupamento, ele pode ser um líder (CH), um líder isolado (ICH) e/ou um *repositório* (R). Um líder é aquele que possui membros associados a ele. Por outro lado, um líder isolado não possui membros associados. O Algoritmo 1 mostra de forma detalhada o cálculo do peso da correlação espacial e da formação de agrupamentos no ASQPM.

Inicialmente, cada sensor envia a mensagem “SEND_READINGS” por inundação na rede, informando sua leitura atual (l.6). Cada sensor s_i que recebe a mensagem armazena as leituras de seus vizinhos (l.15-17). Após certo período de tempo Γ , cada sensor s_i executa a função “CALCULATE_WEIGHT” (l.7-8) para calcular o peso de sua correlação espacial $w(s_i)$, informando qual o limite (*threshold*) definido pelo usuário. A seleção do líder (CH) ocorre quando o peso $w(s_i)$, calculado pela função *getWeight*(s_i), for maior do que o *threshold* informado. Em seguida, o CH eleito anuncia esta decisão para os demais sensores (l.20-23). Ao receberem mensagens de anúncio, os sensores armazenam os possíveis candidatos a CH (*candidateCHs*) (l.38). Se $w(s_i)$ for menor que o *threshold* informado, o sensor vira membro e espera por anúncios dos líderes (l.24-25). Em seguida, o sensor membro escolhe como CH , dentre os sensores armazenados como possíveis candidatos (*candidateCHs*), aquele que tiver a leitura mais parecida com a sua, e então envia uma mensagem de anúncio para o CH eleito como seu líder (l.26-27). O CH ao receber o ACK de associação, armazena a posição dos sensores membros $GC(s_i)$ (l.32-34), possibilitando ao líder calcular qual a MBR do agrupamento. Por fim, o CH que não recebeu nenhuma mensagem de associação de um membro, tem seu papel alterado para líder isolado (ICH) (l.10-12).

Dado que os sensores formam agrupamentos com base em sua localização espacial e similaridade em suas leituras, resta definir a estrutura do nível de *repositórios*. O *repositório* tem por finalidade minimizar o número de *saltos* durante o processamento de

Algoritmo 1 Peso da Correlação Espacial e Agrupamento

```

1: procedure GROUPING
2:    $CH(s_i) \leftarrow s_i$ 
3:    $role[CH](s_i) \leftarrow FALSE$ 
4:    $role[R](s_i) \leftarrow FALSE$ 
5:    $readingsOfNeighbors(s_i) \leftarrow \{\}$ 
6:    $candidateCHs(s_i) \leftarrow \{\}$ 
7:    $GC(s_i) \leftarrow \{\}$ 
8:
9:   broadcast('SEND.READINGS', currentReadings())
10:  WAIT( $\Gamma$  time units) ▷  $\Gamma < \Delta$ 
11:  CALCULATE.WEIGHT( $Threshold$ )
12:  WAIT( $\Delta$  time units)
13:  if  $(GC(s_i) = \{\})$  then
14:     $role[ICH](s_i) \leftarrow TRUE$  ▷ Cluster-head alone
15:  end if
16: end procedure
17:
18: procedure RECEIVING('SEND.READINGS',  $NeighborReadings$ ) by  $s_i$ 
19:   $readingsOfNeighbors(s_i) \leftarrow readingsOfNeighbors(s_i) \cup \{neighborReadings\}$ 
20: end procedure
21:
22: procedure CALCULATE.WEIGHT( $Threshold$ ) by  $s_i$ 
23:   $w(s_i) \leftarrow getWeight(readingsOfNeighbors(s_i))$ 
24:  if  $(w(s_i) \geq threshold)$  then
25:     $role[CH](s_i) \leftarrow TRUE$ 
26:    broadcast('CH.ANNOUNCEMENT',  $s_i$ )
27:  else
28:    WAIT( $\Gamma$  time units) for CH.ANNOUNCEMENT
29:     $CH(s_i) \leftarrow getCHWithSimilarReadings(candidateCHs(s_i))$ 
30:    send ACK('CH.ANNOUNCEMENT',  $CH(s_i), position(s_i)$ )
31:  end if
32: end procedure
33:
34: procedure RECEIVING('ACK.CH.ANNOUNCEMENT',  $h, pos$ ) by  $s_i$ 
35:  if  $role[CH](s_i) = TRUE$  then
36:     $GC(s_i) \leftarrow GC(s_i) \cup \{pos\}$ 
37:  end if
38: end procedure
39:
40: procedure RECEIVING('CH.ANNOUNCEMENT',  $h$ ) by  $s_i$ 
41:   $candidateCHs(s_i) \leftarrow candidateCHs(s_i) \cup \{h\}$ 
42: end procedure

```

consultas através da agregação de informações de líderes que estão próximos. Portanto, eles são os responsáveis por armazenar as leituras de um grupo de líderes tal que somente os dados dos *repositórios* serão utilizados durante o processamento de consultas espaciais. Intuitivamente, visto que os líderes representam dados de sensoriamento de cada membro do agrupamento, um *repositório* pode atuar como um centro de dados para responder a consultas referentes a quaisquer um dos agrupamentos que o compõem. O Algoritmo 2 apresenta a estratégia de escolha do *repositório*.

Inicialmente, a função *NeighboursInDistinctClusters* (s_i) (l.3) calcula o número de vizinhos de s_i que pertencem a agrupamentos distintos. O objetivo é procurar sensores que se encontram em regiões de borda para escolher como *repositório* aqueles que podem concentrar o maior número de agrupamentos. Assim, a seleção do sensor *repositório* ocorre se o número de agrupamentos em sua vizinhança for maior que um determinado patamar (*threshold*) definido pelo usuário (l.4). Em seguida, o sensor escolhido como *repositório* anuncia esta decisão para os demais sensores (l.5). Os sensores que recebem os anúncios dos *repositórios* armazenam cada *repositório* anunciado em um conjunto *knownRepo* (l.20). Além disso, as mensagens de anúncio somente serão retransmitidas se a distância de s_i para o sensor r que enviou o anúncio for menor que a menor distância de algum *repositório* já conhecido (l.21-23). Após aguardar certo período Δ

Algoritmo 2 Escolha do Repositório

```
1: procedure REPOSITORY_ELECTION( $Threshold$ )
2:    $knownRepo(s_i) \leftarrow \{\}$ 
3:   if ( $NeighboursInDistinctClusters(s_i) > threshold$ ) then
4:      $role[R](s_i) \leftarrow TRUE$ 
5:     broadcast('REPOSITORY_ANNOUNCEMENT',  $s_i$ )
6:     WAIT( $\Delta$  time units)
7:     for all  $r$  in  $knownRepo(s_i)$  do
8:       send('MBR_ANNOUNCEMENT',  $s_i, r$ )
9:     end for
10:  else
11:    WAIT( $\Gamma$  time units)
12:    if  $role[CH](s_i) = TRUE$  then
13:       $R(s_i) \leftarrow getMinDistance(s_i, knownRepo(s_i))$ 
14:      send('CH_JOIN_REPOSITORY',  $R(s_i), GC(s_i)$ )
15:    end if
16:  end if
17: end procedure
18:
19: procedure RECEIVING('REPOSITORY_ANNOUNCEMENT',  $r$ ) by  $s_i$ 
20:    $knownRepo(s_i) \leftarrow knownRepo(s_i) \cup \{r\}$ 
21:   if  $knownRepo(s_i) = \{\}$  OR  $distance(s_i, r) < distance(s_i, getMinDistance(s_i, knownRepo(s_i)))$  then
22:     broadcast('REPOSITORY_ANNOUNCEMENT',  $r$ )
23:   end if
24: end procedure
25:
26: procedure RECEIVING('CH_JOIN_REPOSITORY',  $r, gc$ ) by  $s_i$ 
27:   if  $R(s_i) = r$  then
28:      $GR(s_i) \leftarrow GR(s_i) \cup gc$ 
29:   else
30:     forward('CH_JOIN_REPOSITORY',  $r, gc$ )
31:   end if
32: end procedure
33:
34: procedure RECEIVING('MBR_ANNOUNCEMENT',  $s_j, r$ ) by  $s_i$ 
35:   if  $R(s_i) = r$  then
36:      $knownRepo(s_i) \leftarrow knownRepo(s_i) \cup \{s_j\}$ 
37:   else
38:     forward('MBR_ANNOUNCEMENT',  $s_j, r$ )
39:   end if
40: end procedure
```

de tempo, o *repositório* se anuncia para os seus *repositórios* membros (*knownRepo*) (l.6-9). O *repositório* membro que recebe a mensagem atualiza seu conjunto de *repositórios* conhecidos (l.35-36). Mantendo um conjunto de *repositórios* conhecidos, um *repositório* consegue calcular qual a área de cobertura (MBR) de seus vizinhos. Por outro lado, um *repositório* simplesmente repassa uma mensagem que não foi destinada a ele (l.38). Após certo período Γ de tempo, um *CH* associa como *repositório* aquele que tiver a menor distância dentre os *repositórios* conhecidos (l.11-13). Em seguida, o *CH* envia uma mensagem de associação para o *repositório* informando sua MBR ($GC(s_i)$) (l.14). O *repositório* de destino da mensagem atualiza sua MBR ($GR(s_i)$) com a informação da MBR enviada pelo *CH* (l.27-28). Caso contrário, o sensor retransmite a mensagem para o *repositório* de destino (l.30).

Como exemplo, considere o cenário ilustrado pela Figura 2, no qual o *repositório* *R3* está em uma região de borda compreendida pelos agrupamentos *C4*, *C5*, *C9* e *C10*, enquanto o *repositório* *RA* está em uma vizinhança composta pelos agrupamentos *C6*, *C7* e *C11*. Intuitivamente, uma vez que os *repositórios* armazenam informações sobre mais de um agrupamento, espera-se que o número de *saltos* realizados durante o processamento da consulta será menor.

As técnicas utilizadas pelo modelo de armazenamento distribuído do ASQPM apresentam um bom trade-off entre os custos de consulta e de armazenamento. Sem a

definição de agrupamentos ou *repositórios*, seria preciso inundar a rede com consultas para se obter o resultado desejado, o que é impraticável em redes de grande escala. Embora não detalhado neste artigo, os agrupamentos, CHs e *repositórios* são modificados de forma autônoma e similar aos algoritmos apresentados, sempre que a correlação espacial de um CH deixar de satisfazer o *threshold* pré-estabelecido.

3.2. Processamento de Consultas Espaciais

O processamento da consulta do ASQPM funciona da seguinte forma: qualquer sensor da rede pode iniciar uma consulta espacial, desta forma o modelo não se limita a um único ponto de entrada de consulta. O destino da consulta refere-se à região geográfica de interesse e o resultado refere-se às leituras coletadas por sensores nesta região. Observe que no modelo ASQPM as leituras dos líderes de agrupamentos representam as leituras de sua região geográfica e que *repositórios* armazenam dados sobre um conjunto de líderes. O protocolo de roteamento geográfico adotado é o GPSR [Karp and Kung 2000] para a fase de encaminhamento e resposta da consulta. Na fase de encaminhamento, o sensor que recebe a requisição de consulta encaminha o pedido para um sensor membro cuja região de cobertura se sobrepõe a região delimitada pela consulta. Em seguida, o sensor membro encaminha a mensagem para seu líder, que por sua vez envia a requisição de consulta para o *repositório*.

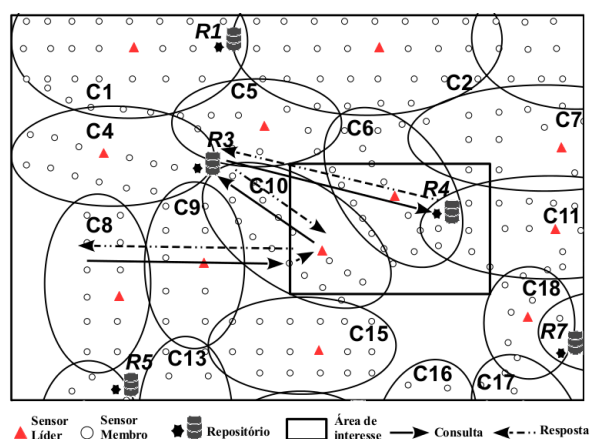


Figure 2. Posicionamento do repositório e processamento de consultas

A Figura 2 ilustra uma consulta espacial, onde neste exemplo, a requisição de consulta quer obter dados de uma região compreendida pelos agrupamentos C_6 , C_7 , C_{10} e C_{11} . Suponha como ponto de entrada da consulta um sensor do agrupamento C_8 . O encaminhamento da consulta se faz da seguinte forma: inicialmente encaminha-se a requisição da consulta em direção à região de interesse até encontrar um sensor membro em C_{10} que está dentro desta região. Ao receber a requisição de consulta, o sensor membro a encaminha para o líder de C_{10} , que por sua vez encaminha a requisição para o *repositório* R_3 . Assim, se a área de R_3 (dada por GR) cobrir inteiramente a área de interesse da consulta, o processamento da consulta termina com o encaminhamento do resultado para o seu ponto de entrada.

Contudo, se a área de R_3 não cobrir inteiramente a área de interesse da consulta, o encaminhamento da consulta direciona a requisição para outros *repositórios* vizinhos que estejam dentro da região de interesse. Dando continuidade ao exemplo acima, claramente

o *repositório R3* pode responder apenas uma parte da consulta, ou seja, apenas os dados referentes ao agrupamento *C10*. Assim, *R3* encaminha a consulta para o *repositório R4*, que pode responder com informações referentes aos agrupamentos *C6*, *C7* e *C11*. Por fim, quando o processamento da consulta coletar todas as informações necessárias, o resultado da consulta retorna até o ponto de origem.

4. Avaliação do ASQPM

As avaliações do ASQPM e do IBIS ocorreram por meio de simulações no simulador de redes NS versão 2.35. No cenário empregado considerou-se uma região retangular de $1400m$ por $1000m$, nós idênticos e estáticos, distância entre os nós em torno de $90m$, com links simétricos, com raio de comunicação de $100m$ e protocolo MAC 802.11. Cada nó possui um GPS que informa sua posição sobre a região monitorada. No entanto podem-se assumir outras formas de localização, como a triangulação. As avaliações consideram também três cenários distintos para analisar a escalabilidade do ASQPM, compostos por 140, 560 e 1260 sensores espalhados sobre a área monitorada. No início das operações da rede, os nós executam o algoritmo para definição dos agrupamentos e dos nós *repositórios* de dados. Em seguida, o nó 28 inicia o processamento de uma consulta, a qual busca as médias das cinco leituras de todos os nós contidos dentro de um retângulo definido pelos vértices opostos $v_1(300, 50)$ e $v_2(950, 500)$, como ilustra a Figura 3. Os resultados mostrados nos gráficos são referentes aos dados coletados a partir do processamento desta consulta. Cada ponto plotado corresponde à média de 35 simulações, com intervalo de confiança de 95%. A Tabela 1 resume os principais parâmetros utilizados na simulação.

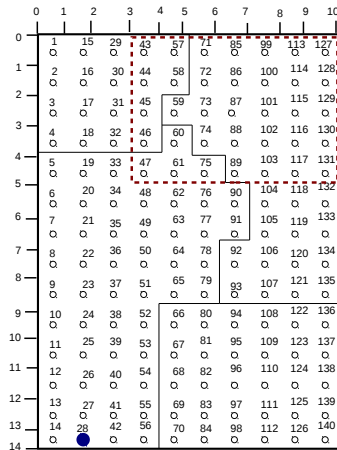


Figure 3. Cenário da simulação

Table 1. Parâmetros da simulação

Parâmetro	Cenário
Quantidade de sensores	140, 560, 1260
Fonte de dados	dados sintéticos
Área do ambiente	$1400m$ por $1000m$
Raio de comunicação do rádio	$100m$
<i>Threshold</i> para agrupamento	$0.8 \leq \tau \leq 0.99$
<i>Threshold</i> para <i>repositório</i>	$\tau \geq 4$
Potência de transmissão	$0.051W$
Potência na recepção	$0.048W$

A avaliação dos experimentos realizou-se em duas etapas. O objetivo da primeira avaliação foi analisar o comportamento do modelo quanto à definição do número de *repositórios*, de líderes e de líderes isolados. Para cada simulação foram geradas novas leituras para os sensores distribuídos na área de monitoramento. Na segunda fase dos experimentos, o objetivo foi comparar o ASQPM com o IBIS [da Silva et al. 2011], analisando-se três métricas: **consumo de energia (Joules)**, **tempo gasto para processamento da requisição (segundos)** e **porcentual de erro nas leituras obtidas**. A energia consumida pela rede para transmitir um pacote (E_{PT}) é igual a energia para transmissão de um pacote (E_{TX}), mais a soma da energia consumida por cada um dos n vizinhos

do emissor para recebê-lo ($\sum_{x=1}^n E_{RE}$). Esse consumo pode ser modelado pela equação: $E_{(i \rightarrow j)PT} = E_{(i)TR} + \sum_{x=1}^n E_{(x)RE}$.

A geração dos dados de simulação levou em consideração a similaridade espacial dos nós na região monitorada. Assim, os nós sensores foram divididos em quatro regiões, como ilustrado pela Figura 3. Cada nó realizou a coleta de cinco leituras distintas, como temperatura e umidade, considerando-se que todas as leituras estão no intervalo entre 0 e 10. Inicialmente, foram criados quatro conjuntos de dados, sendo cada conjunto composto de 5 valores, gerados de forma aleatória. Cada conjunto foi associado a uma região e seus valores correspondem às sementes a partir das quais as leituras de cada nó sensor da região foram geradas. As leituras dos nós de uma mesma região são valores com variação aleatória de no máximo um percentual V a partir do valor semente. Assim, a diferença na leitura dos nós de uma mesma região é no máximo $V\%$.

4.1. Formação de Agrupamentos e Repositórios

Nestes experimentos, o objetivo foi determinar a porcentagem de *repositórios*, líderes e líderes isolados gerados pelo ASQPM, em relação ao número de nós da rede. No gráfico da Figura 4(a), o eixo x representa o patamar de similaridade usado para definir os líderes (*threshold* do Algoritmo 1), o qual variou de 0.8 a 0.99, V foi fixado em 10% e o patamar para escolha de *repositórios* (*threshold* do Algoritmo 2) em todas as simulações foi 4.

Verifica-se que a porcentagem de *repositórios* praticamente não se altera. A definição de um *repositório* leva em consideração o número de líderes que um dado sensor é capaz de escutar. Por isso, a variação da similaridade praticamente não influencia a porcentagem de *repositórios* na rede. Em geral, a porcentagem máxima de nós selecionados como *repositórios* ficou abaixo de 10% em todos os cenários avaliados. Este resultado mostra que a estratégia do modelo de concentrar leituras em *repositórios* ao invés de mantê-los apenas nos líderes dos agrupamentos foi acertada. Como o processamento das consultas requer apenas acessos aos *repositórios*, o número de sensores contactados é pequeno, mesmo que as consultas espaciais cubram regiões geográficas extensas. O gráfico também mostra que, com o aumento do patamar de similaridade para definição de líderes, o número de líderes isolados cresce e de líderes com nós subordinados diminui. Isso ocorre em consequência da diminuição do número de nós com similaridade acima do patamar, o que gera mais líderes isolados. Além disso, o número de *repositórios* necessários para armazenar os dados de agrupamentos vizinhos também tende a ser menor quando há mais agrupamentos com nós subordinados do que agrupamentos com líderes isolados. Isso ocorre porque nesse caso o número total de agrupamentos formados tende a ser menor. Em cenários reais, nos quais há maior similaridade entre sensores vizinhos, acredita-se que a quantidade de líderes isolados seja ainda menor.

No gráfico da Figura 4(b), o eixo X representa a porcentagem de variação máxima V dos dados gerados como entrada para a simulação. Verifica-se que esse parâmetro não altera consideravelmente a formação de agrupamentos e a definição dos *repositórios*. Nestes experimentos, considerou-se um patamar de 0.88 de similaridade para os líderes.

Ainda nesse contexto, é importante analisar a quantidade de memória necessária para o funcionamento do ASQPM. Para isso, analisaremos somente o pior caso, a memória necessária para um *repositório*. Tal nó precisa armazenar sua posição geográfica (2 bytes), sua MBR (4 bytes), suas leituras ($L * TL$ bytes, onde L é o número de leituras e TL o

espaço ocupado por uma leitura), a MBR e as leituras de cada um dos NCH líderes associados a ele e a posição geográfica e a MBR de cada um dos NR repositório conhecidos. Logo, o custo de armazenamento de um repositório pode ser calculado pela seguinte equação: $(6 + (L * TL)) + NCH * (4 + L * TL) + (2 + 4) * NR$. No pior caso em nossos experimentos, ao considerar que um repositório sabe informações de todos os outros repositórios da rede e pela análise do gráfico ilustrado na Figura 4(a), temos $NR = 53$. Em análises dos logs dos experimentos, verificou-se que o maior número de líderes associados a um repositório é $NCH = 54$. Logo, considerando $TL = 1$ byte, o custo máximo de armazenamento de um repositório foi de 834 bytes, menor que os 10 KB de memória RAM da maioria dos nós sensores descritos na literatura.

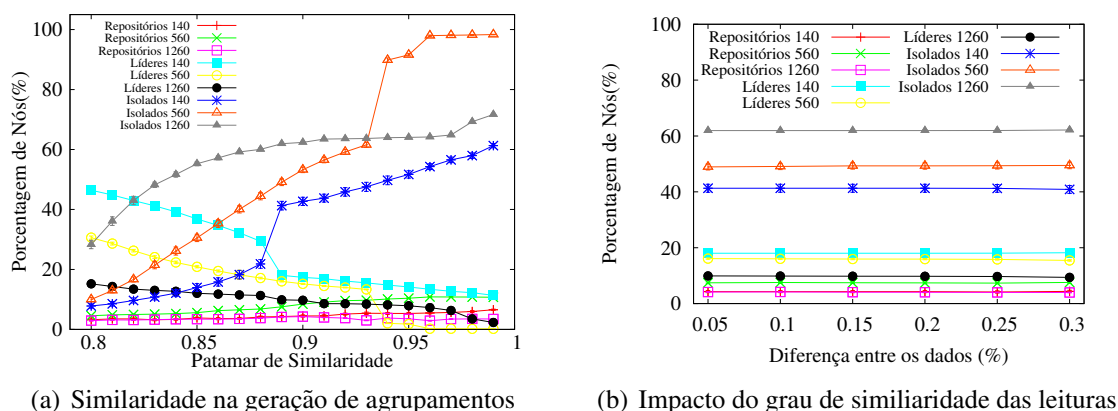


Figure 4. Quantidade de líderes/agrupamentos e repositórios

4.2. Processamento de Consultas ASQPM e IBIS

Nesta seção, o processamento de consultas do ASQPM é comparado com o IBIS, o qual cria um itinerário dentro da região de coleta de forma que todos os sensores contidos nesta região são vizinhos do itinerário ou fazem parte dele. É importante salientar que o processamento de requisições do IBIS se altera somente com alterações na topologia da rede. Como nos experimentos aqui apresentados, a topologia da rede mantém-se inalterada, o comportamento do IBIS foi constante em todos os cenários analisados. Porém, dentre os trabalhos até então encontrados na literatura, tal algoritmo é o que apresenta o melhor desempenho no cenário utilizado na avaliação.

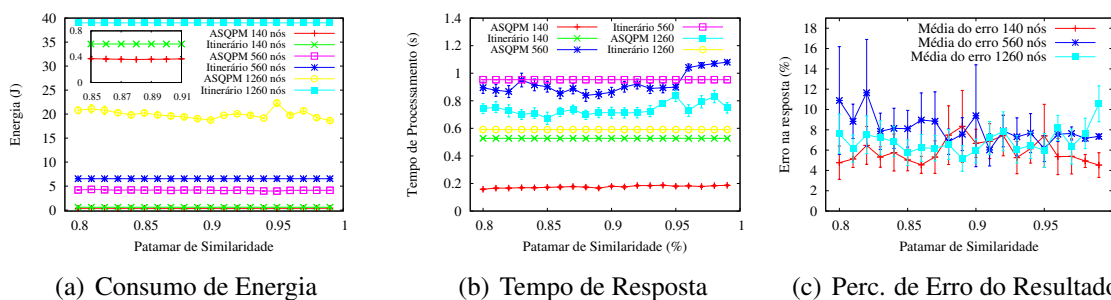


Figure 5. Comparação entre ASQPM e IBIS

O gráfico da Figura 5(a) mostra que o ASQPM consome menos energia que o IBIS, porque as requisições não precisam ser disseminadas para todos os nós na região

de consulta, basta elas alcançarem os *repositórios* que contém dados dessa região para se obter a resposta. Em relação ao tempo de resposta, apresentado pelo gráfico 5(b), para 140 nós o ASQPM é mais eficiente porque o IBIS cria vários *delays* durante a disseminação da consulta pelo itinerário. No cenário de 560 nós, o desempenho do ASQPM mostra-se inferior que o IBIS somente quando a similaridade ultrapassa 0.95, pois a partir deste limiar o ASQPM apresenta o maior número absoluto de *repositórios* de dados dentre os cenários analisados. No entanto, no cenário com 1260 nós, o ASQPM apresenta pior desempenho que o Itinerário. Isto ocorre pois o Itinerário tenta encaminhar a consulta sempre para o vizinho que estiver mais próximo da região de interesse. Dado que no cenário de 1260 cada nó apresenta vizinhos mais próximos dessa região, o encaminhamento e a transmissão da resposta até o nó que iniciou a requisição são realizados de maneira mais rápida.

De acordo com o gráfico 5(c), o ASQPM apresenta uma margem de erro enquanto que o IBIS retorna o resultado exato, uma vez que as leituras são obtidas diretamente dos dispositivos sensores da região de interesse. Os valores apresentados mostram a porcentagem de diferença entre a média das leituras de todos os sensores contidos na região de coleta e a média obtida com o processamento da requisição. Verifica-se que o erro do ASQPM varia em torno de 6% no cenário composto por 140 e 560 sensores, e relativamente um pouco maior para o cenário composto por 1260 sensores. Esse erro vem do fato que as leituras feitas por um líder representam a leitura do seu agrupamento. No entanto, é preciso levar em consideração que a média do erro é calculada a partir das leituras de 5 atributos de sensoriamento, o que geralmente não ocorre nos modelos tradicionais onde a média é calculada a partir de um único atributo do ambiente.

Além das métricas apresentadas, torna-se importante analisar o **custo energético do processo de definição de líderes e repositórios**. O gasto energético do ASQPM ficou em torno de 0.35 Joules. Por outro lado, o consumo do IBIS foi menor, cerca de 0.054 Joules, pois necessita apenas de uma inundação na rede. Entretanto, essa diferença é compensada com o processamento de mais requisições. Como o ASQPM consome menos energia para processar, o consumo da rede durante todo seu período de operação tende a ser menor. Esses dados não são mostrados nos gráficos porque não ocorreram grandes variações dos valores acima mencionados. Estes resultados como um todo mostram que o ASQPM é um modelo que atende os requisitos de **escalabilidade** e **autonomia** para RSSFs urbanas. Mas, é preciso analisar e estender o trabalho com formas alternativas para reduzir o erro relativo das consultas. Observa-se que o ASQPM determina a similaridade entre sensores, considerando um *conjunto* de métricas, e não apenas um tipo de leitura.

5. Conclusão

Este trabalho apresentou um modelo para processamento de requisições espaciais em redes urbanas, chamado de ASQPM. O modelo foi inspirado em características normalmente encontradas em aplicações de sensoriamento urbano, como a alta densidade de sensores na rede para monitorar regiões extensas e a correlação espacial presente no dado sensoriado. Neste modelo, os dados de sensoriamento são distribuídos em estruturas denominadas de *repositórios*. Simulações mostraram que a estratégia adotada pelo ASQPM reduz significativamente o fluxo de mensagens durante o processamento de consultas. No entanto, o ASQPM mostrou uma taxa de erro pequena que deve ser levada em conta de

acordo com os critérios da aplicação executada na rede. Como trabalho futuro, pretende-se analisar o impacto da dinâmica temporal do dado sensoriado.

References

- Can, Z. and Demirbas, M. (2013). A survey on in-network querying and tracking services for wireless sensor networks. *Ad Hoc Networks*, 11(1):596–610.
- Cheng, B., Xu, Z., Chen, C., and Guan, X. (2011). Spatial correlated data collection in wireless sensor networks with multiple sinks. In *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, pages 578–583. IEEE.
- Coman, A., Sander, J., and Nascimento, M. A. (2007). Adaptive processing of historical spatial range queries in peer-to-peer sensor networks. *Distributed and Parallel Databases*, 22(2-3):133–163.
- da Silva, R. I., Macedo, D. F., and Nogueira, J. M. S. (2011). Contornos irregulares no processamento de requisições espaciais para redes de sensores sem fio. In *XXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*.
- Da Silva, R. I., Macedo, D. F., and Nogueira, J. M. S. (2014). Spatial query processing in wireless sensor networks—a survey. *Information Fusion*, 15:32–43.
- Furlaneto, S. S., Dos Santos, A., and Hara, C. S. (2012). An efficient data acquisition model for urban sensor networks. In *Network Operations and Management Symposium (NOMS), 2012 IEEE*, pages 113–120. IEEE.
- Hung, C.-C., Peng, W.-C., and Lee, W.-C. (2012). Energy-aware set-covering approaches for approximate data collection in wireless sensor networks. *Knowledge and Data Engineering, IEEE Transactions on*, 24(11):1993–2007.
- Karp, B. and Kung, H.-T. (2000). Gpsr: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 243–254. ACM.
- Le, T. D., Pham, N. D., and Choo, H. (2008). Towards a distributed clustering scheme based on spatial correlation in wsns. In *Wireless Communications and Mobile Computing Conference, 2008. IWCMC'08. International*, pages 529–534. IEEE.
- Ma, Y., Guo, Y., Tian, X., and Ghanem, M. (2011). Distributed clustering-based aggregation algorithm for spatial correlated sensor networks. *Sensors Journal, IEEE*, 11(3):641–648.
- Muller, C. L., Chapman, L., Grimmond, C., Young, D. T., and Cai, X. (2013). Sensors and the city: a review of urban meteorological networks. *International Journal of Climatology*, 33(7):1585–1600.
- Resch, B., Mittlboeck, M., Girardin, F., Britter, R., and Ratti, C. (2009). Live geography–embedded sensing for standardised urban environmental monitoring.
- Thepvilojanapong, N., Ono, T., and Tobe, Y. (2010). A deployment of fine-grained sensor network and empirical analysis of urban temperature. *Sensors*, 10(3):2217–2241.
- Vuran, M. C., Akan, Ö. B., and Akyildiz, I. F. (2004). Spatio-temporal correlation: theory and applications for wireless sensor networks. *Computer Networks*, 45(3):245–259.
- Xie, L., Lu, S., Cao, Y., and Chen, D. (2014). Towards energy-efficient storage placement in large scale sensor networks. *Frontiers of Computer Science*, pages 1–17.
- Yick, J., Mukherjee, B., and Ghosal, D. (2008). Wireless sensor network survey. *Computer networks*, 52(12):2292–2330.
- Yoon, S. and Shahabi, C. (2007). The clustered aggregation (cag) technique leveraging spatial and temporal correlations in wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 3(1):3.

Yu, Z., Xiao, B., and Zhou, S. (2010). Achieving optimal data storage position in wireless sensor networks. *Computer Communications*, 33(1):92–102.