

Capítulo

5

Fusão de dados para Ambientes Inteligentes

Claudio M. de Farias, Gabriel Caldas, Gabriel Costa, Luis Filipe Kopp, Beatriz A. Campos

Abstract

This chapter proposal is to present the main multisensor data fusion concepts and its application on smart environments. It will be shown the main concepts about data multisensor fusion techniques, models, classification and main definitions about smart environments, its relation to multisensor data fusion, multisensor data fusion applications to smart environments, as well as open research opportunities for future research.

Resumo

A proposta do minicurso é apresentar os principais conceitos sobre fusão de dados e a aplicação desses conceitos em Ambientes Inteligentes. Serão abordados os conceitos fundamentais sobre técnicas de fusão de dados, modelos de fusão de dados, suas classificações, e as principais definições de Ambientes Inteligentes, sua relação com fusão de dados, aplicações para fusão de dados em Ambientes Inteligentes, bem como questões em aberto que apontem possibilidades de pesquisas futuras.

5.1. Introdução

Os recentes avanços nas tecnologias de comunicação e computação fomentaram um enorme crescimento no número de dispositivos inteligentes disponíveis para uso [Farias et al.2016]. A integração desses objetos inteligentes na Internet originou o conceito de Internet das Coisas (IoT - *Internet of Things*) [Whitmore et al.2015]. A IoT pode ser compreendida como um mundo de objetos interligados, capazes de serem identificados, endereçados, controlados e acessados via Internet. Esses objetos podem se comunicar uns com os outros, com outros recursos disponíveis na web, com sistemas de informação e usuários humanos. As aplicações de IoT envolvem interações entre vários dispositivos heterogêneos, a maioria deles interagindo diretamente com seu ambiente físico, seja coletando variáveis ou atuando sobre o meio. Entre essas aplicações pode-se citar o monitoramento

e sistemas de decisão presentes ambientes que vão desde: (i) pessoas e partes do corpo, (ii) peças de equipamentos, (iii) eletrodomésticos e carros, (iv) estruturas civis de grande porte como prédios, pontes e plataformas petrolíferas. Essas conjuntos de aplicações e sensores criam o que se conhece como Ambiente Inteligentes [Liu et al.2019].

Novos desafios emergem neste cenário, bem como várias oportunidades a serem exploradas. Uma dessas oportunidades refere-se extração de informação útil para os usuários finais a partir dos grandes volumes de dados produzidos por esses ambientes inteligentes. Nesse contexto, são necessárias técnicas para promover a descoberta do conhecimento a fim de explorar plenamente o uso dos dispositivos da IoT. As técnicas de fusão de dados [Nakamura et al.2007] tratam da associação, correlação e combinação de dados e informações de fontes únicas e múltiplas para obter uma representação consistente e precisa de um objeto ou ambiente do mundo real. Uma vez que os dados produzidos pelos ambientes inteligentes são geralmente dinâmicos e heterogêneos, torna-se importante investigar técnicas de fusão de dados nesse contexto. O emprego dessas técnicas de fusão de dados é útil para revelar tendências nos dados amostrados, diminuir o volume de dados trafegados (e assim minimizar o consumo dos recursos), descobrir novos padrões de variáveis monitoradas, realizar previsões, e com isso aumentar a eficácia dos processos de tomada de decisão, reduzindo os tempos de resposta de decisões e permitindo uma percepção mais inteligente e rápida do ambiente monitorado.

As Tecnologias da Informação e da Comunicação (TIC) desempenham um papel vital na solução dos problemas ambientais causados pela degradação da natureza. Apesar de serem partes do problema por também consumirem energia e serem fontes de poluição, as TICs possuem potencial de contribuir para a redução do consumo de energia através da otimização das operações em diversas áreas (como a geração e distribuição de energia elétrica, controle de tráfego, construção, controle industrial) e, conseqüentemente, diminuir o desperdício [Huang et al.2018]. Dessa forma, as TICs assumem um importante papel na busca por soluções para o crescimento sustentável e verde das nações.

Um dos campos de pesquisa relacionado ao uso das Tecnologias da Informação e Comunicação (TICs) como provedoras de soluções para os desafios ambientais são os espaços inteligentes (*smart spaces*) [Marchenkov2018]. Um espaço inteligente (ou ambiente de computação pervasiva) pode ser caracterizado como um ambiente com diversos dispositivos, conectados em rede, os quais possuem capacidade de processamento e sensoriamento e que auxiliam os usuários finais na execução de suas tarefas de forma mais eficiente.

As redes elétricas inteligentes (*smart grid*) [Zame et al.2018] podem ser citadas como um dos exemplos de ambientes inteligentes. A rede elétrica inteligente é um tipo de rede de fornecimento e distribuição de energia elétrica (composta por subsistemas para geração, distribuição, transmissão e consumo de energia elétrica) que utiliza as TICs para prever o comportamento do sistema elétrico e, em caso de problemas (como quedas de energia), atuar (como, por exemplo, ligar sistemas de refrigeração de baterias) [Zame et al.2018].

Há diversos trabalhos na literatura discutindo a questão dos ambientes inteligentes que lidam com o aparecimento de objetos com capacidade de monitoramento, processamento e comunicação nos últimos anos [Culman et al.2019], [Zame et al.2018],

[Santos et al.2019], [de Farias et al.2019], [dos Santos et al.2018], [de Farias et al.2018], [Rogova and Snidaro2019] e [Jorge et al.2018]. Diante disto, aparece o cenário da Internet das Coisas (*Internet of Things* – IoT) [dos Santos et al.2018] onde objetos podem se conectar à Internet e prover comunicação entre usuários, dispositivos (D2D), máquinas (M2M) e formarem novas aplicações. Recentemente, a computação de borda (*Edge Computing*) é vista como o próximo passo dos sistemas de Internet das Coisas [Li et al.2018] [Liu et al.2019]. A computação na borda é uma arquitetura de TIC aberta e distribuída que apresenta poder de processamento descentralizado, capacitando tecnologias de computação móvel e Internet das Coisas (IoT). Na computação na borda, os dados são processados pelo próprio dispositivo, computador ou servidor local, em vez de serem transmitidos para um *data center*. Essa arquitetura têm se mostrado promissoras para o processo de tomada de decisão em ambientes inteligentes [Huang et al.2018].

Nos últimos anos o campo de IoT tem observado diversas mudanças que impactaram o projeto e operação dessas redes. Dentre as diversas mudanças apresentadas há o surgimento das redes de sensores compartilhadas, que ao invés de assumir um projeto de rede tradicional específico para uma única aplicação alvo, a infraestrutura de sensoriamento e comunicação das redes IoT é compartilhada por múltiplas aplicações que podem pertencer a usuários diferentes, otimizando assim a utilização de recursos. Pelo fato de compartilhar a mesma infraestrutura de sensoriamento e comunicação por diversas aplicações, esse tipo de rede passa a ser uma das mais promissoras soluções para as aplicações voltadas para ambientes inteligentes. Há uma série de potenciais vantagens de um projeto de rede de múltiplas aplicações, como a redução significativa nos custos de implantação da rede por permitir que múltiplas aplicações dividam os mesmos nós e infraestrutura de comunicação e sensoriamento, melhorando a utilização global de recursos. Porém, apesar desse potencial, a adoção das redes compartilhadas apresenta novos desafios, os quais devem ser suplantados para se usufruir plenamente de suas vantagens. Tais desafios estão relacionados a funções básicas necessárias para a operação e a gerência das redes, como fusão de dados, escalonamento de tarefas, integração de tarefas, segurança de rede dentre outras, que devem ser adaptadas para esse novo ambiente compartilhado.

Com o aumento no número de aplicações dividindo uma mesma infraestrutura de IoT, há conseqüentemente um aumento na quantidade de dados coletados pelos sensores (e por conseqüência um aumento no número de transmissões). A partir dos dados obtidos pelos sensores informações úteis podem ser extraídas. Nós sensores possuem limitações críticas de recursos (baterias removíveis, pouco poder de processamento e memória), apesar disso eles podem realizar ações tais como o processamento de dados como uma forma de solucionar o problema do aumento do número de transmissões em redes de sensores.

Uma forma promissora de reduzir o número de transmissões na rede e, conseqüentemente, de reduzir o consumo de energia dos nós sensores consiste na utilização de técnicas de fusão de dados [Chakraborty et al.2019]. Fusão de dados pode ser definida como o processamento de múltiplas fontes de dados a fim de obter um único dado de saída considerado melhor em termos de precisão ou custo [Chakraborty et al.2019]. Tradicionalmente, as técnicas de fusão de dados processam múltiplas fontes de dados de uma única aplicação para gerar um único dado de saída. Entretanto, podem existir situações nas quais algumas aplicações, que compartilham uma mesma rede IoT, demandam da-

dos que podem ser adquiridos por uma mesma unidade de sensoriamento. Por exemplo, considere duas aplicações dentro do cenário de redes elétricas inteligentes: uma aplicação de monitoramento de linhas de transmissão suspensas (MLTS) e uma aplicação de monitoramento de baterias (MB). A aplicação de MLTS deve suspender a transmissão de energia por uma dada linha de transmissão caso o valor de temperatura ultrapasse um determinado limite (65°C). A aplicação de monitoramento de bateria deve suspender o armazenamento de energia de uma bateria caso o valor da temperatura esteja acima de 144°C . Nota-se que as duas aplicações compartilhariam o sensor de temperatura (fazem uso dos mesmos dados gerados por ele). Nessas situações, não é desejável que os dados relativos às mesmas fontes de dados (unidades de sensoriamento) sejam coletados repetidamente para ambas as aplicações. Bastaria que, para uma mesma fonte de dados e para um determinado instante de tempo, o dado fosse coletado uma única vez e armazenado para posterior consulta e uso pelas respectivas aplicações. Entretanto, as técnicas de fusão de dados tradicionais, quando aplicadas aos dados de uma mesma fonte destinados a múltiplas aplicações, podem resultar na perda da semântica dos dados para algumas destas aplicações. Tal perda se dá porque as aplicações possuem diferentes intervalos de dados. Ao realizar a fusão de dados de aplicações com intervalos tão distintos para o mesmo tipo de dados acabam-se gerando resultados em intervalos que não pertencem a nenhuma das aplicações. Em outras palavras, a semântica dos dados é diferente em aplicações distintas e, por isso, quando a fusão de dados é realizada com técnicas tradicionais há perda da semântica das aplicações (já que os dados resultantes não estão nos intervalos de nenhuma aplicação).

A fim de superar este desafio, a fusão de conhecimento [Dong et al.2014a] surge como uma importante ferramenta para descobrir e limpar erros presentes nas fontes de dados. Os erros podem ser cometidos no processo de extração de conhecimento das fontes. Comparando com a fusão de dados, que visa resolver conflitos de fontes, a fusão de conhecimento considera uma dimensão adicional de erros - os erros cometidos pelos extratores de conhecimento. Entretanto, a fusão do conhecimento foi proposta para um cenário rico em memória e processamento: a web [Preece et al.2000]. Geralmente, os dispositivos na área de ambientes inteligentes têm restrições severas de recursos [Nakamura et al.2007], como energia e processamento. Ainda, de acordo com o modelo de dados de Dasarathy [Nakamura et al.2007], a fusão do conhecimento foi aplicada principalmente em cenários com maiores níveis semânticos de dados (como decisões) [Nakamura et al.2007], enquanto em ambientes inteligentes os dispositivos frequentemente produzem dados em baixos níveis semânticos.

Portanto, o desafio consiste em criar/adaptar técnicas de fusão de dados e ou conhecimento que considerem os dados coletados de uma mesma fonte e destinados a diferentes aplicações realizando tantas reduções quanto possível sem que haja perda da semântica dos dados.

Adicionalmente, a existência de um maior número de aplicações em uma rede IoT pode possibilitar que o processo de tomada de decisão de uma aplicação influencie no processo de tomada de decisão de outra aplicação provocando alterações no comportamento das aplicações. Por exemplo, em geral uma aplicação de MLTS deve deixar que a transmissão de energia elétrica aconteça enquanto o valor informado da temperatura da linha de transmissão indicar que a temperatura está em um valor considerado ideal

(entre 40°C e 65 °C). No entanto, caso o valor da temperatura esteja acima de 65 °C, a aplicação de MLTS indicará que a transmissão de energia deve ser temporariamente interrompida. A aplicação de MLTS decide que deverá avisar a aplicação MB sobre a interrupção da transmissão de forma que essa possa comandar o desligamento da bateria, evitando assim que a bateria fique desnecessariamente ligada, o que pode diminuir a vida útil da mesma. Portanto, a fim de gerir as aplicações em uma rede IoT de forma eficiente surge a noção de integração. A integração é definida como sendo a capacidade de realizar a troca de informação e a colaboração entre aplicações para atingir objetivos comuns [Martins et al.2018]. Nesse contexto, mecanismos responsáveis pela integração de aplicação devem ser concebidos de forma a tornar possível a realização da comunicação entre aplicações distintas através da troca de informação (como a decisão da aplicação de MLTS enviar alerta de interrupção de transmissão para a aplicação MB). Além disso, esse mecanismo de integração de aplicações deve levar em consideração também que decisões de aplicações consideradas mais prioritárias devem ser tratadas em primeiro lugar. Portanto outro desafio consiste em desenvolver metodologias/mecanismos descentralizados voltados para aplicações de ambientes inteligentes que fazem uso de RSAC os quais possibilitem integrar diferentes aplicações dentro da própria rede.

Na busca de investigar soluções para os desafios expostos, este minicurso envisa apresentar os conceitos fundamentais atrás da fusão de dados para ambientes inteligentes bem como expôr seus principais desafios. Também buscar-se-á apresentar novas técnicas de fusão de dados que busquem resolver os problemas apresentados acima.

O minicurso está organizado da seguinte forma: (i) na seção 2 serão apresentados os conceitos fundamentais de fusão de dados; (ii) na seção 3 serão apresentados os conceitos básicos e exemplos de ambientes inteligentes; (iii) na seção 4 serão apresentadas técnicas de fusão de dados que atendam às demandas dos ambientes inteligentes; (iv) na seção 5 será apresentado o *framework* micropython bem como o desenvolvimento de um caso de exemplo usando o referido *framework*. Por fim na seção 6 serão apresentadas breves conclusões.

5.2. Fusão de Dados

Nesta seção serão apresentados os principais conceitos relacionados à fusão de dados. Também serão apresentados os desafios existentes quando se utilizam técnicas de fusão de dados. Serão apresentadas as classificações das técnicas, as técnicas mais tradicionais e os modelos de fusão de dados.

De uma forma mais geral, a fusão de dados pode ser vista como “um processo de múltiplos níveis que lida com a detecção, associação, correlação e estimação de dados provenientes de múltiplos sensores” (Departamento de Defesa dos EUA 1991). No domínio das IoT, técnicas de agregação de dados simples (médias aritméticas, a busca por máximos e mínimos, dentre outras) têm sido usadas para a redução do tráfego de dados com o intuito de reduzir o consumo de energia dos nós sensores. A agregação de dados pode ser definida como a combinação de dados de diferentes nós fontes usando funções triviais (*i.e.*, máximo, mínimo, média) que realizam a supressão de mensagens redundantes, e consequentemente, reduzem a quantidade de dados. A eficiência dos algoritmos de agregação de dados depende da correlação entre os dados gerados pelas diferentes fontes

de informação [de Farias et al.2019]. A correlação pode ser espacial, quando os valores gerados por sensores próximos são relacionados; temporal, quando as leituras de sensores mudam lentamente ao longo do tempo; ou semântica, quando as informações de diferentes pacotes de dados podem ser classificadas sob o mesmo grupo semântico, como por exemplo os dados que são gerados por sensores colocados em uma mesma sala. Esse aspecto favorece a eliminação de redundância (uma das metas das técnicas de agregação de dados), mas garante também a acurácia dos dados. Isso é importante, pois a sumarização dos dados pode representar uma perda na acurácia [de Aquino et al.2018], que é um requisito típico para muitas aplicações RSSFs. A acurácia pode ser definida como o grau de proximidade entre a medição observada e o seu real valor esperado. Com uma correlação eficiente dos dados originais é possível alcançar uma maior redução da quantidade de dados para uma mesma acurácia dos dados agregados.

Outros dois conceitos importantes para a eficiência do mecanismo de agregação de dados são: grau e latência [Nakamura et al.2007]. O grau de agregação é definido como o número de pacotes agregados em um único pacote de transmissão; enquanto a latência pode ser medida como o tempo entre os pacotes recebidos no nó sorvedouro e os dados gerados nos nós fontes [de Farias et al.2016]. É importante que a relação entre esses dois conceitos seja equilibrada para que haja eficiência na redução da quantidade de dados por um lado, mas que também não haja atrasos exagerados na entrega final dos dados, por outro lado. A fusão de dados pode ser categorizada em diversos aspectos, a saber: relacionamento entre fontes de dados, nível de abstração e o propósito da fusão de dados. De acordo com o relacionamento entre fontes de dados, a fusão de dados pode ser classificada como complementar, redundante e cooperativa [Farias et al.2016]:

- **Complementar:** Quando a informação provida pelas fontes representa pedaços de um cenário maior, a fusão pode ser aplicada para obter informações mais completas a cerca do cenário. A fusão complementar busca a completude, formando uma nova informação através da composição de diversas outras (como sensores que verificam a presença de pessoas em quatro cantos diferentes de um aposento e ao fundir essa informação temos a visão completa do aposento).
- **Redundante:** Se duas ou mais fontes independentes proveem o mesmo pedaço de informação, estes pedaços podem ser fundidos para aumentar a confiabilidade da informação. A fusão de redundância pode ser usada para aumentar a confiabilidade, precisão e credibilidade da informação. Em RSSF, a fusão de redundâncias pode prover informação de alta qualidade e prevenir nós sensores de transmitirem dados iguais (vários sensores de temperatura avaliando a temperatura de uma caldeira industrial).
- **Cooperativo:** Duas fontes são cooperativas quando a informação provida por elas é fundida em uma nova informação (normalmente mais complexa do que a original), que do ponto de vista da aplicação representa melhor a realidade (um sensor de temperatura e um sensor de fumaça combinando informações para detectar um incêndio).

Quanto ao nível de abstração, [Nakamura et al.2007] a fusão de dados pode ser classificada em quatro níveis:

- Sinal (*signal*): lida com sinais uni ou multi-dimensionais vindos dos sensores (em geral dados brutos vindos dos sensores). Pode ser usado em aplicações de tempo real ou como um passo intermediário entre fusões.
- Pixel: usado em imagens que podem ser utilizadas em tarefas de processamento de multimídia.
- Característica (*feature*): lida com características (ou atributos) extraídas de sinais (como, por exemplo, a temperatura de uma sala) e imagens, como forma e velocidade.
- Símbolo (*symbol*): neste tipo de fusão, a informação é um símbolo que representa uma decisão (como por exemplo, há um símbolo indicando a ação acionar alarme em caso de incêndio), e portanto, esse tipo de fusão também é conhecida como fusão de decisões.

De acordo com o nível de abstração dos dados manipulados, a fusão da informação também pode ser classificada em 4 categorias:

- Fusão de baixo nível: Também conhecida como fusão em nível de sinal. Dados sem processamento são usados como entrada da fusão, combinados em novos dados mais precisos que os originais (aqui se enquadram os dados extraídos das unidades de sensoriamento, como voltagem, amperagem ou campo eletromagnético).
- Fusão de nível médio: atributos ou características de uma entidade (como a forma, textura e posição) são fundidos para a obtenção de um mapa de características que pode ser útil em outras tarefas. Este tipo de fusão também é conhecida como fusão de atributos (dados de temperatura e campo eletromagnético para encontrar danos em linhas de transmissão).
- Fusão de alto nível: decisões ou representações simbólicas são usadas como entrada e são combinadas para obter uma decisão mais confiável ou com uma visão mais ampla do cenário (por exemplo, a decisão de que uma linha de transmissão está danificada e a decisão de que a bateria está danificada gerando a decisão de usar outra linha de transmissão de energia elétrica).
- Fusão de múltiplos níveis: acontece quando a fusão de dados utiliza dados de diferentes níveis de abstração (por exemplo, quando dados de decisão são fundidos com dados do tipo característica, como um dado vindo de um sensor de presença, indicando que não há pessoas na sala, combinado com uma decisão de aquecimento ao realizar a fusão pode concluir que nenhuma ação deve ser dada).

Dasarathy [Martins et al.2018] apresenta outra bem-conhecida classificação para fusão de dados que leva em consideração a abstração dos dados de entrada e saída.O modelo Dasarathy identifica 5 categorias:

- *Data In–Data Out* (DAI-DAO): Nessa classe, a fusão de dados lida com dados em nível de sinal e o resultado também em nível de sinal, possivelmente mais preciso ou confiável.
- *Data In–Feature Out* (DAI-FEO): Nessa classe, a fusão de dados usa dados brutos como entrada para extrair atributos ou características que descrevem uma atividade como saída.
- *Feature In–Feature Out* (FEI-FEO): nessa classe, a fusão de dados trabalha sobre um conjunto de características para melhorar ou refinar uma característica ou atributo, ou para extrair novos.
- *Feature In–Decision Out* (FEI-DEO): nessa classe, a fusão de dados usa uma série de características de uma entidade gerando uma representação simbólica ou uma decisão.
- *Decision In–Decision Out* (DEI-DEO): nessa classe, decisões podem ser fundidas de forma a obter novas decisões ou dar ênfase a decisões anteriores.

Ainda outra forma de classificação é baseada no propósito dos métodos de fusão, ou seja, que tipo de informação busca-se extrair dos dados coletados [Nakamura et al.2007]. De acordo com esse critério a fusão de dados pode ser realizada com diferentes objetivos como inferência, estimação, classificação, agregação e compressão.

Métodos de inferência, estimação e classificação são muitas vezes aplicados em fusões de decisão. Nesse caso, uma decisão é tomada baseada no conhecimento da situação percebida. A inferência se refere a transição de uma proposição provavelmente verdadeira, a qual a veracidade é creditada como resultado de uma inferência anterior. Métodos clássicos de inferência são a Inferência Bayesiana [de Farias et al.2019] e a teoria da acumulação de crenças de Dempster-Shafer [de Farias et al.2017].

Métodos de compressão e agregação são usados apenas para redução do volume de dados. A agregação é usada para resolver os problemas de implosão e *Overlapping*. No primeiro, os dados sensorizados são duplicados na rede devido a alguma estratégia de roteamento. O *overlapping* acontece quando dois nós diferentes disseminam os mesmos dados. Os métodos de compressão não são métodos de fusão de dados propriamente ditos, uma vez que eles apenas consideram as estratégias de codificação dos dados. O código de Huffman se enquadra nos métodos de compressão de dados.

5.3. Ambientes Inteligentes e a Internet das Coisas

Conforme apresentado na seção 5.1, um espaço inteligente (ou ambiente de computação pervasiva - *smart space*) pode ser caracterizado como um ambiente com diversos dispositivos, conectados em rede, os quais possuem capacidade de processamento e sensoriamento e que auxiliam os usuários finais na execução de suas tarefas de forma mais eficiente.

Diversos ambientes inteligente surgiram recentemente de forma a gerar mais conforto e/ou segurança para os usuários. A questão ao se trabalhar com ambientes inteligentes é que ambientes distintos possuem requisitos distintos em termos de suas aplicações.

Claramente há uma disparidade na importância entre uma aplicação de controle de temperatura e um sistema de alarme de incêndio. Ou ainda entre um sistema de controle cardíaco de pacientes em um hospital e o controle de iluminação. Ainda uma mesma aplicação pode possuir requisitos diferentes em ambientes distintos. O controle de refrigeração é menos importante em uma casa do que é um hospital.

O termo edifício inteligente (*Smart Building*) segundo [Liu et al.2019] é definido como edifícios equipados com dispositivos inteligentes instalados de forma a minimizar o consumo de energia e sem comprometer o conforto e a segurança do usuário. Portanto, um passo importante rumo a um estilo de vida mais sustentável é melhorar a eficiência energética dos edifícios. Embora os avanços recentes no campo da ciência dos materiais tenham conseguido reduzir o consumo diretamente na estrutura dos edifícios, ainda existem problemas a serem superados [Marchenkov2018]. Um deles relaciona-se a grande quantidade de energia que continua sendo consumida por diversos equipamentos, tais como os aparelhos de controle de temperatura e de iluminação, tanto em edifícios residenciais como comerciais.

Aplicações para edifícios inteligentes mais comumente encontradas na literatura [de Farias et al.2019] [Whitmore et al.2015] incluem: aplicações de controle de aquecimento, ventilação e sistemas de ar condicionado (HVAC); aplicação de iluminação; aplicação de sombreamento; aplicação de qualidade do ar e controle de janelas; aplicações de desligamento de dispositivos; aplicações domésticas (por exemplo, televisores, máquinas de lavar); aplicações de segurança (controle de acesso) e segurança de dispositivos. Tais aplicações muitas vezes fazem uso do mesmo tipo de dado ambiental, como luz, vibração, temperatura, presença, químicos (como fumaça ou gases) e voltagem.

No contexto de *smart building*, recentemente, foram apresentadas na literatura novas propostas, [de Farias et al.2017] [de Farias et al.2019] [Caldas et al.2015] propondo soluções de controle e monitoramento que fazem uso de RSSF. No contexto de *smart grid* não existem soluções de monitoramento devido ao seu alto custo [Gungor et al. 2009]. Nesses ambientes *smart grid* a instalação e manutenção de cabos de comunicação são procedimentos custosos e por isso que os sistemas de monitoramento cabeados não são amplamente utilizados atualmente. Portanto, há uma necessidade urgente por sistemas de monitoramento e diagnóstico sem fio de baixo custo que melhorem a confiabilidade e a eficiência do sistema através da melhora do gerenciamento dos sistemas de geração e transmissão de energia.

O sistema de monitoramento da integridade da estrutura (SHM, do inglês *Structural Health Monitoring*) permite prever danos (fraturas) e, conseqüentemente, antecipar consertos evitando acidentes. Em aplicações construídas para esse fim, os dispositivos de sensoriamento são usados para capturar medidas relacionadas à própria estrutura bem como os eventos externos que afetam a estrutura monitorada e enviar tais medidas para uma entidade centralizada de coleta de dados. A tomada de decisão do SHM quanto a existência de dano é feito de forma centralizada e é efetuada nessa entidade. Normalmente, o monitoramento de estruturas (SHM) é efetuada utilizando-se sensores analógicos cabeados. Entretanto, ao transferir parte do processo de decisão quanto a existência de um dano ou quanto ao mau funcionamento de um dispositivo para dentro dos sensores se obtêm um SHM inteligente. Esse SHM inteligente faz uso de mecanismos descentraliza-

dos capazes de detectar e prever danos (fraturas) ou mal funcionamento de dispositivos de forma precisa e confiável dentro da própria rede. Ao empregar SHM inteligente por exemplo em estruturas ou dispositivos ligados a uma usina eólica inserida no contexto de *smart grid* permite que tais sistemas gerem energia de forma mais confiável e economicamente eficaz.

O termo Rede elétrica inteligente (*Smart Grid*) segundo [de Farias et al.2018] é definida como sendo uma rede transmissão bidirecional (do gerador para o consumidor e vice-versa) de eletricidade que faz uso das TICs nos sistemas de geração, transmissão, distribuição e uso final de energia com o intuito de melhorar a eficiência, confiabilidade e segurança da geração e fornecimento de energia. Uma vantagem adicional do *smart grid* é que a geração de energia pode ser feita através do uso de energias renováveis e não-poluíntes como a energia solar e a energia eólica. A transmissão ser bidirecional implica que será possível conhecer em tempo real a condição de fios, cabos, transformadores e o consumo até de dispositivos específicos instalados em fábricas, escritórios ou domicílios. Com isso será possível controlar esses dispositivos e usar tarifas que variem de acordo com a carga do sistema, hora do dia ou estação para incentivar a conservação de energia, reduzindo seu uso não econômico em horas de demanda máxima.

De acordo com os estudos realizados pela EPRI (*Electric Power Research Institute*) [Whitmore et al.2015], a rede elétrica inteligente irá gerar uma redução de emissão de gases que varia de 60 a 211 milhões de toneladas de CO₂ nos Estados Unidos até 2030. A comunicação dentro da rede elétrica pode ser feita através fibras ópticas e via tecnologias que usam fios elétricos de média e baixa voltagem como canais para comunicação IP.

Outra aplicação que vêm ganhando destaque é a aplicação de Fazendas Inteligentes [Culman et al.2017]. Estas são fazendas controladas por sensores e ajudam os agricultores em melhorar o desempenho das colheitas e/ou rebanho. Tipicamente, aplicações para fazendas Inteligentes incluem sensores para a medição de umidade do solo, controle de temperatura, irrigação, avaliação da evapotranspiração dos ambientes.

Nas fazendas Inteligentes [Culman et al.2019] [de Farias et al.2017] atuais diversos algoritmos de predição tentam prever como os fatores edafoclimáticos (do clima e do solo) podem interferir na produção e alterar decisões de irrigação e insumos com base nelas. Alguns algoritmos fazem uso de lógica *fuzzy* como no trabalho de [Culman et al.2019] que modela as variáveis ambientais como variáveis *fuzzy* e tenta alterar em tempo real o comportamento dos sistemas de irrigação.

Em [de Farias et al.2017] os autores usam uma técnica de fusão de dados entre os dados de sistemas meteorológicos e dos sensores na fazenda de forma a controlar a irrigação. Neste trabalho usou-se um modelo de fusão de dados baseado em teoria de crença no qual conforme os dados meteorológicos acertavam ou erravam, ajustavam os pesos de suas informações na decisão de irrigação.

Há ainda o conceito de estufas inteligentes (*smart greenhouses*) que tentam retirar as complexidades impostas pelas intempéries do ambiente através do total isolamento das culturas plantadas em estufas controladas através de sensores [de Farias et al.2017]. Nesse tempo de ambiente o controle de temperatura bem como o controle de iluminação são extremamente importantes.

Hospitais Inteligentes [Hassan et al.2019] são hospitais monitorados por sensores e câmeras que tentam monitorar as condições dos pacientes e das instalações dos hospitais. Várias aplicações são similares à das casas inteligentes. Entretanto há diversas aplicações distintas como a verificação de infecções e contaminação de ambientes. Sensores químicos são muito importantes. Ainda dentro deste contexto o uso de sensores multimídia como câmeras é deveras importante para a predição de quedas de pacientes e monitoramento de movimentação.

Uma segunda categoria de aplicações de hospitais inteligentes diz respeito ao monitoramento dos sinais vitais dos pacientes através das redes de sensores corporais (do inglês *body sensor networks*). O monitoramento de sinais vitais é extremamente complexo e muito suscetível a falsos alarmes. Os falsos alarmes são alarmes para a equipe do hospital sem que haja uma emergência ocorrendo. Esses falsos alarmes levam ao problema da fadiga de alerta no qual as equipes deixam de responder aos alarmes de uma rede corporal pensando ser um falso alarme. Sendo assim técnicas de fusão de dados para esse ambiente específico devem buscar reduzir os falsos alarmes tanto quanto possível.

Há ainda aplicações militares fazendo uso de fusão de dados para a verificação de saúde de soldados em tempo real. Além disso o uso de fusão de dados para a navegação em veículos autônomos [Blasch et al.2018].

Como pode ser observado cada ambiente possui particularidades que devem e desafios que devem ser tratados com extremo cuidado. Na próxima seção serão apresentadas diversas técnicas que buscam resolver esses desafios ainda que parcialmente.

5.4. Domínios de Aplicação de Fusão de Dados em Ambientes Inteligentes

Nesta seção serão apresentados diversas técnicas de fusão de dados usadas em cenários apresentados na seção 5.3.

Conforme dito na seção 5.2 as técnicas de fusão de dados podem ser organizadas de acordo com as saídas dos métodos. Uma técnica a nível de característica famosa é o filtro de média móvel. O Filtro de média móvel é um método amplamente usado no processamento de sinais digitais por ser simples e capaz de reduzir o ruído do sinal. O filtro computa a média aritmética de um número de sinais de entrada para produzir cada ponto do sinal de saída. Dado um sinal $z = (z(1), z(2), \dots)$, o verdadeiro sinal $x = (x(1), x(2), \dots)$ é estimado por:

$$x(k) = \frac{1}{N} \sum_{i=0}^{M-1} z(k-i), \forall k \geq M, \quad (1)$$

Onde M é o tamanho da janela de fusão, $z = z(1), z(2), z(3) \dots$ são os dados de entrada e $x = x(1), x(2), x(3) \dots$ são os dados estimados pelo método.

A janela de fusão é o parâmetro mais importante para essa equação uma vez que M é usado para a detecção de qualquer mudança no nível do sinal; quanto maior o valor de M , mais claro será o sinal.

Apesar de fácil compreensão e baixa complexidade, o filtro de média móvel lida apenas com medições em nível de sinal e característica, mas não consegue lidar com

níveis semânticos mais altos, como decisões.

No filtro de média móvel, todos os dados possuem o mesmo peso. Em um ambiente com múltiplas aplicações, algumas medidas podem ser mais importantes para uma determinada aplicação do que para outra aplicação. Por exemplo, uma aplicação de detecção de incêndio é mais importante do que uma aplicação de AVAC e valores de temperatura mais altos são mais importantes para a detecção de incêndio do que para a aplicação de AVAC. Um valor de temperatura de 50 C possui pouco valor para a aplicação de AVAC, uma vez que o intervalo de operação desta aplicação não considera valores de temperatura tão altos, entretanto é um ótimo indicativo de incêndio. Se todas as medidas forem consideradas da mesma forma, pode-se encontrar um resultado que não reflete o ambiente sensoriado corretamente, um resultado tendencioso.

No Filtro de média móvel aperfeiçoado, há a necessidade de avaliar determinada medida considerando uma aplicação alvo. De forma a tornar essa abordagem possível, pode-se modificar a abordagem tradicional para a apresentada na equação 10:

$$x(k) = \frac{1}{N} \sum_{i=0}^{M-1} \mu z(k-i), \forall k \geq M, \mu > 0, N = \sum_{i=0}^{M-1} \mu \quad (4)$$

Onde M é o tamanho da janela de fusão, $z = z(1), z(2), z(3) \dots$ são os dados de entrada, $x = x(1), x(2), x(3) \dots$ são os dados estimados pelo método, μ é o peso dado baseado na importância da aplicação e N é a soma de todos os pesos. Um especialista de domínio deve escolher os pesos apropriadamente.

Um outro trabalho já usa o conceito de distribuir pesos diferentes no filtro de média móvel. O EWMA (Exponentially weighted moving average), uma variante do filtro de média móvel apresentada em [Farias et al.2016], distribui pesos para os dados de cada sensor como forma de melhor avaliar o ambiente. Nossa proposta é diferente do EWMA uma vez que ao invés de distribuir pesos para os dados de diferentes sensores, assume-se que todos os sensores são capazes de monitorar o mesmo tipo de informação (temperatura por exemplo), mas a relevância dessa informação variará de acordo com os requisitos de cada aplicação que for utilizá-la.

Entretanto dentro dos ambientes inteligentes as técnicas de fusão de dados a nível de decisões possuem ainda mais destaque uma vez que evitam que os dados coletados precisem sair da rede. Entretanto essas técnicas devem ser capazes de ser implementadas em sensores.

Uma das técnicas de fusão de dados mais utilizadas a nível de decisão é a Inferência Bayesianas (muitas vezes chamada de *Naive Bayes*). A inferência bayesiana utiliza uma combinação de evidências de acordo com certas regras de probabilidades. O grau de incerteza é representado por probabilidades condicionais como mostrado na equação 1:

$$P(Y | X) = \frac{P(X | Y)P(Y)}{P(X)} \quad (1)$$

Onde $Pr(Y | X)$ representa a probabilidade de que a hipótese Y seja verdadeira dada a informação X. Essa probabilidade é obtida através da multiplicação de $Pr(Y)$, a

probabilidade anterior da hipótese Y , por $\Pr(X|Y)$, a probabilidade de que X seja verdadeiro dado que Y seja verdadeiro; $\Pr(X)$ pode ser considerada uma constante normalizadora. O maior problema da Inferência Bayesiana é que as probabilidades $\Pr(X)$ e $\Pr(X|Y)$ devem ser conhecidas a priori sem nenhuma garantia dos valores reais. Não há testes com dados reais para garantir que as probabilidades representem cenários reais. Apesar disso, a Inferência Bayesiana é usada no cenário de IoT devido à sua simplicidade e baixo consumo de recursos.

Para estender os métodos probabilísticos, tais como a Inferência Bayesiana, uma abordagem recursiva é proposta neste trabalho. Na Inferência Bayesiana aprimorada (equação 8), a expressão $BI(Y|X)$ representa a crença de que a aplicação Y terá um determinado comportamento dado o resultado da aplicação X . Por exemplo, sempre que uma aplicação de detecção de incêndio detecta um incêndio em potencial, a aplicação de AVAC será desligada, uma vez que quando um determinado limiar de temperatura é encontrado, e esse limiar representa a ocorrência de um incêndio, não há necessidade de manter a aplicação de AVAC (que tem como objetivo prover conforto aos ocupantes do edifício) em operação. Nesse momento, todos os dados de temperatura providos pelos sensores serão relevantes somente para a aplicação de detecção de incêndio. [Farias et al.2014]:

$$BI(A|B) = \frac{BI(B|A)BI(A)}{BI(B)} \quad (2)$$

os termos $BI(A)$ e $BI(B)$ são inferências bayesianas que consideram o conjunto de estados de cada aplicação separadamente.

Entretanto essa extensão ainda sofre com a limitação da Inferência Bayesiana tradicional: as probabilidades devem ser conhecidas a priori, ou sejam antes do início da operação do sistema de decisão [Farias et al.2014].

Esse método é baseado na teoria de acumulação de crença de Dempster-Shafer, que é uma teoria introduzida por Dempster-Shafer (DEMPSTER E SHAFER 1974), e generaliza a teoria bayesiana. Dempster-Shafer lida com crenças ou funções de massa da mesma forma que a teoria Bayesiana lida com probabilidades. A teoria de probabilidades provê um formalismo que pode ser usado para a representação de conhecimentos incompletos, combinação de evidências e atualização de crenças.

Um conceito fundamental no sistema de inferência de Dempster-Shafer é o quadro de discernimento, que é definido como segue. Seja $\Theta = \{\theta_1, \theta_2, \dots, \theta_N\}$ o conjunto de todos os estados possíveis que descrevem o sistema, dado que θ é exaustivo e mutuamente exclusivo, no sentido em que o sistema está em apenas um dado estado θ_i , onde $1 \leq i \leq N$. O intervalo 1 até N é chamado de quadro de discernimento porque seus elementos são usados para discernir os estados do sistema.

Os elementos do conjunto 2^Θ são chamados de hipóteses. Na teoria de Dempster-Shafer, baseado na evidência E , uma probabilidade é relacionada para cada hipótese $H \in 2^\Theta$, de acordo com a função de massa $m : 2^\Theta \rightarrow [0, 1]$ que satisfaz as condições abaixo:

$$m(\emptyset) = 0 \quad (3)$$

$$m(H) \geq 0, \forall H \in 2^{\Theta} \quad (4)$$

$$\sum_{H \in 2^{\Theta}} m(H) = 1 \quad (5)$$

O grau de crença de uma hipótese é definido pela função $bel(H)$ como visto em seguida $bel : 2^{\Theta} \rightarrow [0, 1]$ sobre Θ como:

$$bel(H) = \sum_{A \subseteq H} m(A), \quad (6)$$

onde $bel(\emptyset) = 0$, e $bel(\Theta) = 1$.

A partir da crença podemos calcular dois outros graus de pertencimento: dúvida (dou) e plausibilidade (pl) como mostrado nas equações 4 e 5:

O grau de dúvida em H pode ser expressado de forma intuitiva em termos da função de crença $bel : 2^{\Theta} \rightarrow [0, 1]$ como:

$$dou(H) = bel(\neg H) = \sum_{A \subseteq \neg H} m(A). \quad (7)$$

Para expressar a plausibilidade de cada hipótese, a função $pl : 2^{\Theta} \rightarrow [0, 1]$ sobre Θ é definido como:

$$pl(H) = 1 - dou(H) = \sum_{A \cap H = \emptyset} m(A) \quad (8)$$

A plausibilidade intuitivamente diz que quanto menor a dúvida na hipótese H , mais plausível é.

Dessa forma, para combinar o efeito de dois conjuntos de probabilidades (m_1 e m_2) sobre uma mesma hipótese, a teoria de Dempster-Shafer define a seguinte regra de combinação satisfazendo $m_1 \oplus m_2(\emptyset) = 0$:

$$m_1 \oplus m_2(H) = \frac{\sum_{X \cap Y = H} m_1(X)m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X)m_2(Y)} \quad (9)$$

Na inferência de Dempster-Shafer, probabilidades não são associadas as hipóteses a priori. Ao invés disso, probabilidades são associadas somente quando a informação de suporte está disponível, ou seja, quando há dados coletados para serem avaliados. Por outro lado, a inferência de Dempster-Shafer é totalmente dependente dos dados coletados, o que significa que se houver poucos dados coletados o resultado pode ser impreciso ou tendencioso. Para superar esse problema deve-se analisar quantos dados de entrada são necessários para cada aplicação.

Escolher entre a inferência Bayesiana e Dempster-Shafer não é uma decisão trivial, uma vez que há uma troca entre a acurácia da inferência Bayesiana e a flexibilidade trazida por Dempster-Shafer [Nakamura et al.2007]. Em um cenário não estático (como o das RSAC), a acurácia da Inferência Bayesiana é reduzida devido as mudanças no ambiente. Nesse caso, a flexibilidade trazida pela inferência de Dempster-Shafer passa a ser interessante, uma vez que a função de crença se adaptará aos novos dados enquanto as probabilidades da inferência Bayesiana se mantêm estáticas [Suganuma et al.2018].

O método Dempster-Shafer foi aperfeiçoado, pois é uma variante dos métodos probabilísticos que não necessita conhecer as probabilidades dos eventos a priori, conforme apresentado no capítulo 5.2. No Dempster-Shafer aperfeiçoado, cada aplicação terá seu próprio conjunto de hipóteses. Nessa abordagem, a função belief representará a crença que uma aplicação Y apresentará um determinado comportamento dado o resultado de outra aplicação X. Nesse contexto, H é o conjunto de estados que representam o comportamento de uma aplicação. Assim deriva-se a equação 9:

$$m_1 \oplus m_2(H) = \frac{\sum_{X \cap Y = H} DS_1(X)DS_2(Y)}{1 - \sum_{X \cap Y = \emptyset} DS_1(X)DS_2(Y)} \quad (10)$$

Onde DS1 e DS2 são inferências Dempster-Shafer sobre as condições das aplicações isoladas e o numerador da equação representa a crença de que as duas aplicações estarão em um dado estado simultaneamente.

Considerando o exemplo da ocorrência de um incêndio, X é uma aplicação de detecção de incêndio e Y é uma aplicação de AVAC. O numerador da equação 9 representa o grau de dúvida que aparece quando DS1 infere sobre um estado da aplicação, por exemplo que em uma dada condição de temperatura deve haver uma alerta de incêndio, DS2 indicará que na mesma condição de temperatura, a aplicação Y irá desligar o ar-condicionado. Não há necessidade da aplicação de AVAC permanecer ativa durante um incêndio, uma vez que o uso de um sistema de ventilação aumentaria os riscos de problemas na fiação elétrica que poderiam aumentar o incêndio. Isto é importante pois será possível evitar o desperdício de recursos através da eliminação de estados repetidos de duas ou mais aplicações. O denominador significa a plausibilidade dessa hipótese.

Uma outra família de técnicas que surgiu com o intuito de lidar com as incertezas sobre os dados coletados é a lógica subjetiva. Lógica Subjetiva (LS) é um tipo de lógica que utiliza probabilidade para calcular o grau de certeza ou crença em determinado indivíduo ou na ocorrência de um evento. A ideia é que para a ocorrência de um dado evento, em que não é possível estimar ao certo sua probabilidade de ocorrência, seja adicionado um valor de incerteza [Jøsang2001].

A LS trabalha utilizando quatro parâmetros, e são eles [Jøsang et al.2006]: *belief* (*b*, crença) e *disbelief* (*d*, descrença), que são influenciados por eventos passados provenientes do indivíduo, sendo estes eventos positivos e negativos, onde eventos positivos aumentam a crença e vice-versa; *uncertainty* (*u*) (incerteza), grau de incerteza (ou ignorância) que se possui acerca de um evento, este parâmetro diminui conforme o número de observações do indivíduo aumenta; α ou *a*, pode ser considerado como a confiança inicial

que se pode ter sobre um novo nó da rede ou como a probabilidade real de acontecimento de um evento, caso este valor seja conhecido.

$$\omega_x^A = (b, d, u, a) \quad (11)$$

Juntos, os quatro parâmetros formam a *opinião* de um indivíduo em outro. A expressão presente em 11 denota a opinião de um indivíduo A acerca de x [Jøsang et al.2006]. As propriedades dos parâmetros estão presentes nas equações 12 e 13. Note que, para o maior grau possível de incerteza, $u = 1$, os valores de b e d são zero e pode-se dizer que é o estado de completa ignorância acerca da ocorrência de um determinado evento. Em contra partida, para o menor grau possível de incerteza sobre um evento ($u = 0$), diz-se que a opinião sobre este é dogmática e pode obter este valor após infinitas observações [Jøsang et al.2006].

$$b, d, u, a \in [0, 1] \quad (12)$$

$$b + d + u = 1 \quad (13)$$

Para calcular os parâmetros b, d e u são utilizadas as fórmulas presentes em 14. Onde p corresponde ao número de eventos positivos gerados por um indivíduo e n o número de eventos negativos gerados pelo mesmo indivíduo. Em 14, k é uma constante, com valor geralmente 1 ou 2 que determina o quão rápido a crença no elemento avaliado é construída [?].

$$b = \frac{p}{p+n+k}, d = \frac{n}{p+n+k}, u = \frac{k}{p+n+k}. \quad (14)$$

É possível ilustrar a utilização da LS através de dois nós A e B , que se relacionem de alguma forma. No exemplo, A possui uma opinião acerca de B . Com valores fictícios, esta opinião pode ser expressa pela equação $w_B^A = (0.24, 0.43, 0.34, 0.5)$. Neste caso, $b = 0.24$, $d = 0.43$, $u = 0.34$ e $a = 0.5$. O espaço de opinião de A com relação a B pode ser mapeado no interior de um triângulo equilátero, onde b, d e u identificam a posição da opinião no espaço [Jøsang et al.2006], como mostra a Figura 5.1.

Caso existam, para um mesmo evento, duas opiniões diferentes, o operador de ‘consenso’ tem por objetivo calcular um acordo entre as duas opiniões [Jøsang et al.2006]. Para tal, supõe-se que A e B possuem cada um, uma opinião acerca de x , onde estas opiniões são: $\omega_x^A = (b_x^A, d_x^A, u_x^A, a_x^A)$ e $\omega_x^B = (b_x^B, d_x^B, u_x^B, a_x^B)$. Segundo [Jøsang2016, Jøsang et al.2006] o cálculo do consenso entre duas opiniões se divide em dois casos, e são descritos como as equações 15 e 5.4. A equação 5.4 também pode ser utilizada nos casos em que uma das opiniões é dogmática [Martinsson2005]. Uma demonstração gráfica do consenso aplicado é demonstrada na Figura 5.2, onde ele é aplicado em para a junção de duas opiniões distintas.

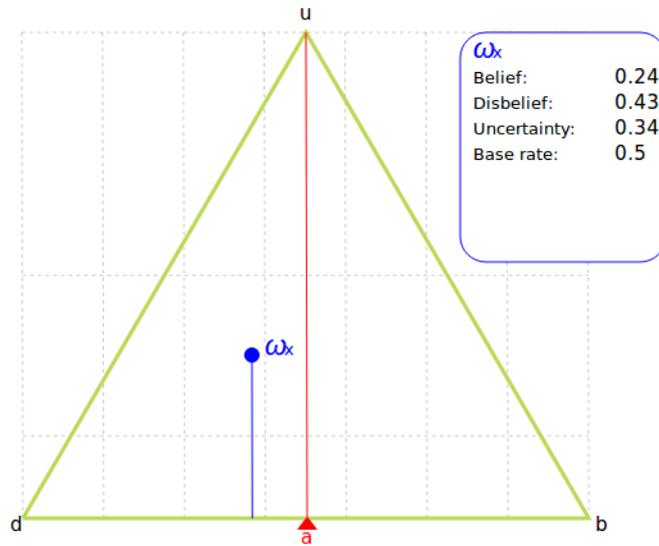


Figura 5.1. Espaço de opinião de w_B^A [Jøsang et al.2006].

Caso I: $u_x^A + u_x^B - u_x^A u_x^B \neq 0$:

$$b^{AB} = \frac{b_x^A u_x^B + b_x^B u_x^A}{u_x^A + u_x^B - u_x^A u_x^B}, d^{AB} = \frac{d_x^A u_x^B + d_x^B u_x^A}{u_x^A + u_x^B - u_x^A u_x^B}, u^{AB} = \frac{u_x^B u_x^A}{u_x^A + u_x^B - u_x^A u_x^B}, a^{AB} = \frac{a_x^A + a_x^B}{2}. \quad (15)$$

Caso II: $u_x^A + u_x^B - u_x^A u_x^B = 0$:

$$b^{AB} = \gamma_x^A b_x^A + \gamma_x^B b_x^B, d^{AB} = \gamma_x^A d_x^A + \gamma_x^B d_x^B, u^{AB} = 0, a^{AB} = \gamma_x^A a_x^A + \gamma_x^B a_x^B.$$

Onde:

$$\gamma_x^A = \lim_{u_x^B \rightarrow 0, u_x^A \rightarrow 0} \frac{u_x^B}{u_x^A + u_x^B},$$

$$\gamma_x^B = \lim_{u_x^B \rightarrow 0, u_x^A \rightarrow 0} \frac{u_x^A}{u_x^A + u_x^B}. \quad (16)$$

Os trabalhos recentes de [de Andrade Campos et al.2019] e [de Andrade Campos et al.] utilizaram a lógica subjetiva para detecção de anomalias. Nesse trabalhos os autores colocaram pesos nos canais de comunicação que eram modificados conforme as decisões se provavam corretas ou não. Dessa forma canais de comunicação pouco confiáveis podiam ser ignorados.

Finalmente, a fusão de conhecimento [Dong et al.2014a] surge como uma importante ferramenta para descobrir e limpar erros presentes nas fontes de dados. Os erros podem ser cometidos no processo de extração de conhecimento das fontes. Comparando com a fusão de dados, que visa resolver conflitos de fontes, a fusão de conhecimento considera uma dimensão adicional de erros - os erros cometidos pelos extratores de conhecimento. Entretanto, a fusão do conhecimento foi proposta para um cenário rico em recursos (memória e processamento): a web [Dong et al.2014a] [Dong et al.2014b] [Oliveira et al.2019]. Geralmente, os dispositivos na área de ambientes inteligentes têm

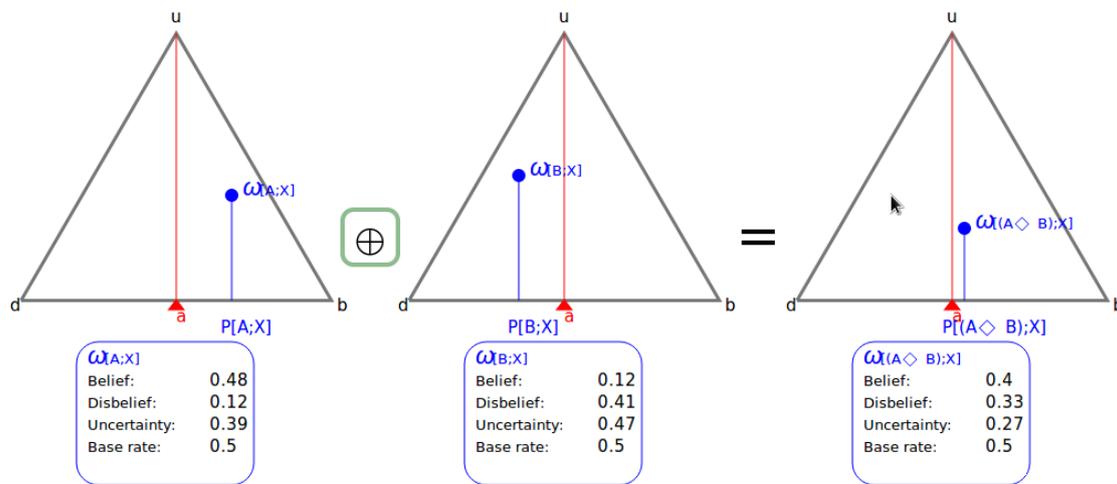


Figura 5.2. Representação gráfica do consenso entre duas opiniões.

restrições severas de recursos [Nakamura et al.2007], como energia e processamento. Ainda, de acordo com o modelo de dados de Dasarathy [Nakamura et al.2007], a fusão do conhecimento foi aplicada principalmente em cenários com maiores níveis semânticos de dados (como decisões) [Nakamura et al.2007], enquanto em ambientes inteligentes os dispositivos frequentemente produzem dados em baixos níveis semânticos (como dados brutos).

Recentemente, o termo *Ai on Edge (Artificial Intelligence)* passou a ser um dos mais requisitados em ambientes inteligentes e significa o uso de técnicas de aprendizado de máquina e/ou de Inteligência Artificial nas bordas das redes IoT. Vale destacar que apesar das redes neurais serem opções naturais neste tipo de aplicação há outras opções viáveis como redes neurais sem peso [Cardoso et al.2018] que são mais rápidas e consomem menos recursos do que as redes neurais tradicionais.

Outro ponto relevante é que as técnicas apresentadas podem fazer parte de técnicas mais complexas. Por exemplo nos trabalhos de [de Farias et al.2016] [de Farias et al.2019] [de Farias et al.2018] [de Farias and Pirmez2017] os autores buscam diferenciar eventos dentro de massas de dados para só então aplicar técnicas de fusão de dados. Ou seja pode haver momentos em que as técnicas apresentadas sejam somente mais uma etapa do sistema de decisão.

5.5. Construindo um sistema de decisão simples usando Fusão de Dados e Micropython

Nesta seção será apresentado como construir uma aplicação que faça uso de fusão de dados. Usar-se-á como domínio de aplicação um edifício inteligente. Como ambiente de desenvolvimento será usado o *framework* Micropython [Norris2016].

A ideia é implementar um filtro de média móvel bem simples que possa ser utilizado para controlar uma aplicação de controle de ventilação. A implementação do algoritmo para esta solução foi dada em Micropython – uma implementação do software da linguagem de programação baseada em Python, escrita em C, otimizada para microcon-

troladores.

O MicroPython é uma implementação enxuta e eficiente da linguagem de programação Python 3 que inclui um pequeno subconjunto da biblioteca padrão do Python e é otimizada para ser executada em microcontroladores e em ambientes restritos.

```
import pyb

# turn on an LED
pyb.LED(1).on()

# print some text to the serial console
print('Hello MicroPython!')
```

Figura 5.3. Hello world!

O MicroPython é um compilador e um tempo de execução completos do Python que são executados no *bare-metal*. Você recebe um *prompt* interativo (o REPL) para executar comandos imediatamente, com a capacidade de executar e importar scripts do sistema de arquivos interno. O REPL possui histórico, preenchimento de guias, modo de recuo automático e colar para uma ótima experiência do usuário.

O MicroPython se esforça para ser o mais compatível possível com o Python normal (conhecido como CPython), de modo que, se você conhece Python, já conhece o MicroPython. Por outro lado, quanto mais você aprende sobre o MicroPython, melhor você se torna no Python. Bibliotecas tradicionais do Python como a biblioteca *Math* já possuem versões equivalentes para Micropython. Recentemente a biblioteca de cálculos matriciais Numpy teve uma implementação feita para Micropython.

```
1 # a subset of the Python Math library
2
3 import math
4 import cmath
5
6 print(math.sqrt(5))
7 print(math.log10(100))
8 print(math.sin(12345) ** 2 + math.cos(12345) ** 2)
9 print(math.cosh(1) ** 2 - math.sinh(1) ** 2)
10 print(cmath.polar(1 + 1j))
11
```

Figura 5.4. Operações Matemáticas

Além de implementar uma seleção das principais bibliotecas Python, o MicroPython inclui módulos como "máquina" para acessar hardware de baixo nível. Dessa forma com o Micropython é possível acessar dados de sensores não inclusos com a placa bem como extrair informações do hardware das placas.

Pode-se simular o funcionamento de uma placa usando um simulador online através da página <https://micropython.org/unicorn/>. Nela pode-se ver um emulador da placa pyboard (placa desenvolvida pelo projeto micropython) em pleno funcionamento. Diversos algoritmos e exemplos estão presentes para estudo.

Há uma IDE (*Integrated Development Environment*) chamada upcraft que permite o desenvolvimento em micropython bem como automatiza o processo de instalação de drivers e afins. Tendo disponibilidade para Windows, Linux e MacOSX a IDE é uma opção que facilita o desenvolvimento de aplicações em micropython.

```

from machine import Pin, I2C

# creat an I2C bus
i2c = I2C(scl=Pin('X1'), sda=Pin('X2'))

# scan for list of attached devices
dev_list = i2c.scan()

# write to and read from a device
i2c.writeto(0x42, b'\04')
data = i2c.readfrom(0x42, 4)

# memory transactions
i2c.writeto_mem(0x42, 0x12, b'\0')
data = i2c.readfrom_mem(0x42, 0x12, 2)

```

Figura 5.5. Trabalhando com a máquina

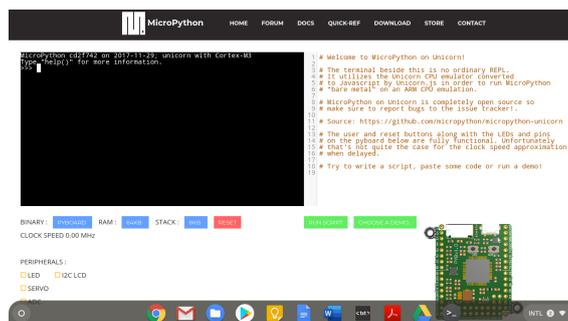


Figura 5.6. Emulador Pyboard

Com a finalidade de conseguir executar os casos de teste foi preciso um microcontrolador para compilar e executar o código construído a partir do algoritmo. O microcontrolador utilizado foi o ESP8266 (da plataforma NodeMCU), muito utilizado para desenvolvimentos IoT e difundido pela facilidade na sua utilização e instalação. A ESP8266 (Figura 5.9) possui um processador ESP8266-12E, que pode operar em 80MHz/160MHz, 4Mb de memória flash, 64Kb para instruções e 96Kb para dados.

Para o desenvolvimento do algoritmo do filtro de média móvel escolheu-se o uso de sensores DHT que medem a temperatura do ambiente. A ideia seria replicar o comportamento de um sistema de controle de ventilação. O ideal seria permanecer entre 18 e 23 graus. Portanto o sistema deveria ler algumas medidas de temperatura para encher a janela de fusão, realizar o filtro de média móvel, e se a temperatura estiver fora da faixa de conforto acionar o sistema de ventilação.

O algoritmo pode ser visto na Figura 5.8. Note que não necessariamente este é o modo mais eficiente de implementação mas para critérios didáticos o algoritmo é suficiente para demonstrar o quão simples é fazer um sistema com MicroPython.



Figura 5.7. Interface do upycraft

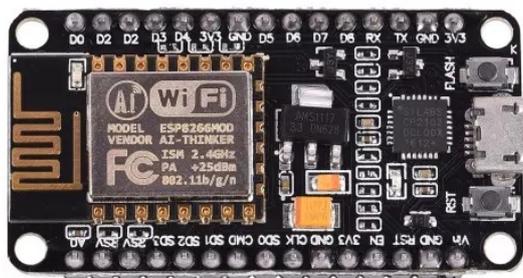


Figura 5.8. NodeMCU: ESP8266 microcontrolador

5.6. Conclusões

Nesta seção serão apresentadas as considerações finais referentes aos assuntos abordados, realizando uma retrospectiva das principais questões discutidas e ressaltando os benefícios trazidos pela fusão de dados para Ambientes Inteligentes.

Conforme pode ser observado diversas aplicações de ambientes inteligentes possuem requisitos em relação ao tempo de resposta das aplicações. Nesses casos o tempo necessário para coleta e processamento fora da rede pode ser maior do que o requerido pela aplicação. Não somente esse fato mas conforme as redes IoT se popularizam aumenta o número de aplicações que compartilham a mesma infraestrutura. Com isso além do aumento do número de mensagens na rede e do maior consumo de energia há a possibilidade de conflito entre diferentes aplicações em um mesmo ambiente.

Apesar da fusão de dados aparecer como uma das possíveis soluções para os desafios apresentados, as técnicas tradicionais não são adequadas para lidar com os múltiplos requisitos dessas múltiplas aplicações presentes nos ambientes inteligentes. Sur-

```

1 from machine import Pin
2 from time import sleep
3 import dht
4
5 sensor = dht.DHT22(Pin(14))
6 fus_wnw = []
7 maf = 0
8
9 while True:
10     try:
11         sleep(2)
12         sensor.measure()
13         temp = sensor.temperature()
14         temp_f = temp * (9/5) + 32.0
15         fus_wnw.add(temp_f)
16         # escolheu-se uma janela de 5 elementos
17         if (fus_wnw.lenght() == 5):
18             # calculando o filtro
19             for i in fus_wnw:
20                 temp +=i/5
21         if (maf >= 23 and maf <= 21):
22             activate_HVAC()
23     except OSError as e:
24         print('Failed to read sensor.')
```

Figura 5.9. Algoritmo do Filtro em Micropython

giram diversas novas técnicas capazes de lidar com esse desafio [de Farias et al.2019] [Caldas et al.2015] [de Farias et al.2018] [de Andrade Campos et al.] [de Farias et al.2016] [Farias et al.2016] [Farias et al.2014].

Embora tenham havido esforços ainda existem muitos desafios em aberto na área como: (i) fusão em *Streams* de dados, (ii) técnicas de fusão que lidam com os dados de múltiplas aplicações simultaneamente, (iii) mineração de dados, (iv) Sensoriamento Participativo, (v) predição de dados, (vi) *Ai on Edge* e (vii) fusão de dados com fontes heterogêneas, tanto no nível de dispositivo quanto no nível de aplicações.

Referências

- [Blasch et al.2018] Blasch, E., Boril, J., Smrz, V., and Leuchter, J. (2018). Pilot interface considerations using high level information fusion. In *2018 IEEE Aerospace Conference*, pages 1–8. IEEE.
- [Caldas et al.2015] Caldas, G., de Farias, C. M., Pirmez, L., and Delicato, F. C. (2015). S-leach: A leach extension for shared sensor networks. In *Proceedings of the International Conference on Wireless Networks (ICWN)*, page 121. The Steering Committee of The World Congress in Computer Science, Computer
- [Cardoso et al.2018] Cardoso, D. O., Gama, J., and França, F. (2018). Weightless neural modeling for mining data streams. *SERIES IN MACHINE PERCEPTION AND ARTIFICIAL INTELLIGENCE*, 83(1):26–43.
- [Chakraborty et al.2019] Chakraborty, I., Chakraborty, A., and Das, P. (2019). Sensor selection and data fusion approach for iot applications. In *Recent Developments in Machine Learning and Data Analytics*, pages 17–33. Springer.
- [Culman et al.2019] Culman, M., de Farias, C. M., Bayona, C., and Cruz, J. D. C. (2019). Using agrometeorological data to assist irrigation management in oil palm crops: A

decision support method and results from crop model simulation. *Agricultural water management*, 213:1047–1062.

- [Culman et al.2017] Culman, M., Portocarrero, J. M., Guerrero, C. D., Bayona, C., Torres, J. L., and de Farias, C. M. (2017). Palmnet: An open-source wireless sensor network for oil palm plantations. In *2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC)*, pages 783–788. IEEE.
- [de Andrade Campos et al.] de Andrade Campos, B., de Farias, C. M., and da Costa Carmo, L. F. R. Using trusted networks to detect anomaly nodes in internet of things. In *Proceedings of the 22nd International Conference on Information Fusion (FUSION 2019)*, volume 1.
- [de Andrade Campos et al.2019] de Andrade Campos, B., de Farias, C. M., and da Costa Carmo, L. F. R. (2019). Utilização de redes de confiança para detecção de anomalias em rssf. In *IV Workshop sobre Regulação, Avaliação da Conformidade, Testes e Padrões de Segurança 2018*, volume 1. Galoá.
- [de Aquino et al.2018] de Aquino, G. R. C., de Farias, C. M., and Pirmez, L. (2018). Data quality assessment and enhancement on social and sensor data. In *BiDu-Posters@ VLDB*.
- [de Farias et al.2016] de Farias, C. M., Li, W., Delicato, F. C., Pirmez, L., Pires, P. F., and Zomaya, A. Y. (2016). Seraph: Service allocation algorithm for the execution of multiple applications in heterogeneous shared sensor and actuator networks. In *Management of Cyber Physical Objects in the Future Internet of Things*, pages 93–113. Springer.
- [de Farias and Pirmez2017] de Farias, C. M. and Pirmez, L. (2017). A multisensor data fusion technique for multiapplication wireless sensor networks based on overlapping intervals. In *2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, pages 804–811. IEEE.
- [de Farias et al.2018] de Farias, C. M., Pirmez, L., and Delicato, F. C. (2018). Density based multisensor data fusion for multiapplication wireless sensor networks. In *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, pages 814–821. IEEE.
- [de Farias et al.2017] de Farias, C. M., Pirmez, L., Delicato, F. C., Pires, P. F., Li, W., Zomaya, A. Y., de LF Jorge, E. N., and Juarez-Ramirez, R. (2017). Grown: A control and decision system for smart greenhouses using wireless sensor networks. In *Proceedings of the Australasian Computer Science Week Multiconference*, page 48. ACM.
- [de Farias et al.2019] de Farias, C. M., Pirmez, L., Fortino, G., and Guerrieri, A. (2019). A multi-sensor data fusion technique using data correlations among multiple applications. *Future Generation Computer Systems*, 92:109–118.

- [Dong et al.2014a] Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., and Zhang, W. (2014a). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610. ACM.
- [Dong et al.2014b] Dong, X. L., Gabrilovich, E., Heitz, G., Horn, W., Murphy, K., Sun, S., and Zhang, W. (2014b). From data fusion to knowledge fusion. *Proceedings of the VLDB Endowment*, 7(10):881–892.
- [dos Santos et al.2018] dos Santos, I. L., Costa, G. M. d. O., de Farias, C. M., Pirmez, L., Delicato, F. C., and Diego, M. d. A. (2018). A holistic approach to challenges in industry 4.0. In *Anais da V Escola Regional de Sistemas de Informação do Rio de Janeiro*, pages 24–29. SBC.
- [Farias et al.2014] Farias, C., Pirmez, L., Delicato, F., Carmo, L., Li, W., Zomaya, A. Y., and de Souza, J. N. (2014). Multisensor data fusion in shared sensor and actuator networks. In *17th International Conference on Information Fusion (FUSION)*, pages 1–8. IEEE.
- [Farias et al.2016] Farias, C. M. D., Li, W., Delicato, F. C., Pirmez, L., Zomaya, A. Y., Pires, P. F., and Souza, J. N. D. (2016). A systematic review of shared sensor networks. *ACM Computing Surveys (CSUR)*, 48(4):51.
- [Hassan et al.2019] Hassan, M. K., El Desouky, A. I., Elghamrawy, S. M., and Sarhan, A. M. (2019). Big data challenges and opportunities in healthcare informatics and smart hospitals. In *Security in Smart Cities: Models, Applications, and Challenges*, pages 3–26. Springer.
- [Huang et al.2018] Huang, J., Liu, Z., Duan, Q., Atiquzzaman, M., Jo, M., and Haas, Z. J. (2018). Green computing and communications for smart portable devices. *Wireless Communications and Mobile Computing*, 2018.
- [Jorge et al.2018] Jorge, E. N. d. L. F., de Farias, C. M., dos Santos, I. L., and de Souza Pereira, M. S. (2018). An analysis of the network of rural producers in the state of rio de janeiro. In *BiDu-Posters@ VLDB*.
- [Jøsang2001] Jøsang, A. (2001). A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(03):279–311.
- [Jøsang2016] Jøsang, A. (2016). *Subjective Logic: A formalism for reasoning under uncertainty*. Springer Publishing Company, Incorporated.
- [Jøsang et al.2006] Jøsang, A., Hayward, R., and Pope, S. (2006). Trust network analysis with subjective logic. In *Proceedings of the 29th Australasian Computer Science Conference-Volume 48*, pages 85–94. Australian Computer Society, Inc.
- [Li et al.2018] Li, H., Ota, K., and Dong, M. (2018). Learning iot in edge: Deep learning for the internet of things with edge computing. *IEEE Network*, 32(1):96–101.

- [Liu et al.2019] Liu, Y., Yang, C., Jiang, L., Xie, S., and Zhang, Y. (2019). Intelligent edge computing for iot-based energy management in smart cities. *IEEE Network*, 33(2):111–117.
- [Marchenkov2018] Marchenkov, S. (2018). Infrastructure multi-layer model for smart spaces middleware development. In *Proceedings of the 22st Conference of Open Innovations Association FRUCT*, page 51. FRUCT Oy.
- [Martins et al.2018] Martins, G., de Farias, C. M., and Pirmez, L. (2018). Athena: A knowledge fusion algorithm for the internet of things. In *Proceedings of the 14th ACM International Symposium on QoS and Security for Wireless and Mobile Networks*, pages 92–99. ACM.
- [Martinsson2005] Martinsson, H. (2005). An evaluation of subjective logic for trust modelling in information fusion.
- [Nakamura et al.2007] Nakamura, E. F., Loureiro, A. A., and Frery, A. C. (2007). Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Computing Surveys (CSUR)*, 39(3):9.
- [Norris2016] Norris, D. (2016). *Python for Microcontrollers: Getting Started with MicroPython*. Mcgraw-hill Education-Europe.
- [Oliveira et al.2019] Oliveira, J., Farias, C. M., Pacitti, E., and Fortino, G. (2019). *Big Social Data and Urban Computing*. Springer.
- [Preece et al.2000] Preece, A., Hui, K., Gray, A., Marti, P., Bench-Capon, T., Jones, D., and Cui, Z. (2000). The kraft architecture for knowledge fusion and transformation. In *Research and Development in Intelligent Systems XVI*, pages 23–38. Springer.
- [Rogova and Snidaro2019] Rogova, G. L. and Snidaro, L. (2019). Quality, context, and information fusion. In *Information Quality in Information Fusion and Decision Making*, pages 219–242. Springer.
- [Santos et al.2019] Santos, I. L., Pirmez, L., Delicato, F. C., Oliveira, G. M., Farias, C. M., Khan, S. U., and Zomaya, A. Y. (2019). Zeus: A resource allocation algorithm for the cloud of sensors. *Future Generation Computer Systems*, 92:564–581.
- [Suganuma et al.2018] Suganuma, T., Oide, T., Kitagami, S., Sugawara, K., and Shiratori, N. (2018). Multiagent-based flexible edge computing architecture for iot. *IEEE Network*, 32(1):16–23.
- [Whitmore et al.2015] Whitmore, A., Agarwal, A., and Da Xu, L. (2015). The internet of things—a survey of topics and trends. *Information Systems Frontiers*, 17(2):261–274.
- [Zame et al.2018] Zame, K. K., Brehm, C. A., Nitica, A. T., Richard, C. L., and Schweitzer III, G. D. (2018). Smart grid and energy storage: Policy recommendations. *Renewable and Sustainable Energy Reviews*, 82:1646–1654.