

Using fuzzy logic in smart homes lighting controllers

Erika Hagenbeck
PROCC, Universidade Federal
de Sergipe
São Cristóvão, Sergipe, Brasil
erikahagenbeck@hotmail.com

Nilson Barros Santos
PROCC, Universidade Federal
de Sergipe
São Cristóvão, Sergipe, Brasil
nilsonbs75@gmail.com

Rodrigo Cavalcante
PROCC, Universidade Federal
de Sergipe
São Cristóvão, Sergipe, Brasil
rodrigo@isx.com.br

Hendrik Macedo
DCOMP, Universidade Federal
de Sergipe
São Cristóvão, Sergipe, Brasil
hendrik@ufs.br

ABSTRACT

Intelligent control systems are dramatically changing domestic technology. The so-called smart homes are environments where technological tools provide convenience and comfort to residents and contribute to the reduction of energy consumption. The term smart is used due to high communication capability between devices and the use of algorithms that analyze the profile of each resident in order to optimize the use of resources and provide custom routines. The aim of this study is to verify the effectiveness of a control system that makes use of fuzzy logic to control the brightness level to be applied in accordance with the amount of light captured in the environment. An application which enables the automation of the lighting system has been created. The results of the experiments show that the system is efficient and the fuzzy logic allows the brightness control properly, reducing or increasing the incidence of illumination without abrupt changes.

Keywords

Smart Home, intelligent controls, fuzzy Logic, luminosity

1. INTRODUÇÃO

Smart city geralmente remete ao uso de dispositivos inteligentes amplamente distribuídos para monitorar o ambiente urbano em tempo real, de reagir a tempo para estabelecer o controle automatizado, para coletar informações para a tomada de decisão, para facilitar variados serviços e melhorar a qualidade de vida [11]. A ideia de ambiente inteligente implica numa intrínseca ligação do indivíduo e seu ambiente de maneira que este interage com dispositivos computacionais de maneira intuitiva sem que haja interferência em suas atividades diárias [12].

Neste contexto, podemos ainda definir a chamada Internet das Coisas (IoT) – sigla proveniente do inglês *Internet of Things* –, que é um paradigma recente cujo conceito se baseia na presença ubíqua de objetos – sensores, atuadores, etiquetas RFID (Radio-Frequency Identification), dispositivos móveis etc – que interagem entre si, através de endereçamento único, para atingir objetivos comuns [5].

As chamadas *Smart Homes* são residências que fazem uso dessa tecnologia, quer seja, através de sensores de presença para fins de segurança, automação de processos como con-

trole de temperatura ou luminosidade, etc., trazendo assim mais conforto e economia para vida das pessoas.

1.1 Revisão de Literatura

A literatura traz alguns trabalhos relacionados a esta proposta. Fazendo uso de um ambiente de simulação denominado Astecca, [7] avaliou o projeto de um controlador PI (proporcional e integral) nebuloso aplicado a um sistema HVAC (*Heating, Ventilation and Air Conditioning*) de uma edificação. Este sistema foi desenvolvido com o software Matlab/Simulink, e seus resultados mostraram que a estratégia de controle nebuloso proposta gerou economia de energia do sistema HVAC.

[13] utilizou um foto-resistor *LDR* (*Light Dependent Resistor*, um dispositivo semiconductor de dois terminais, cuja resistência varia de forma linear à medida que a intensidade de luz incidente sobre sua superfície se altera) e um *Arduino* para desenvolver um sistema para controle de iluminação através de um *software* embarcado escrito a partir de conceitos da lógica *fuzzy*.

1.2 Objetivo

O objetivo desse trabalho é analisar a utilização da *Lógica Fuzzy* em sistemas de sensores e atuadores instalados em *smart homes* com o intuito de tornar mais eficiente a automação de processos como o controle de iluminação, por exemplo. Para tanto, implementamos uma aplicação que controla um sistema de iluminação, podendo facilmente ser adaptado para o controle de outros sistemas como, por exemplo refrigeração, som e tv.

Para tanto, foi implementado um sistema fazendo uso da lógica *fuzzy* que é executado em um computador pessoal. As regras da lógica nebulosa foram esquematizadas de forma que não ocorresse mudanças bruscas na iluminação. O aumento ou a diminuição da intensidade de luz aplicada ao ambiente ocorre de maneira gradual, sendo que o sistema realiza pequenas mudanças na potência do *dimmer*, quase imperceptíveis aos usuários.

1.3 Lógica Fuzzy

Segundo [6] os princípios de lógica *fuzzy* foram desenvolvidos primeiramente por Jan Lukasiewicz (1878-1956), que em 1920 desenvolveu e introduziu conjuntos com grau de pertinência que combinados aos conceitos da lógica clássica, desenvolvida por Aristóteles.

Esses princípios serviram de base para professor Lofti A. Zadeh no início de 1960 publicar o artigo *Fuzzy Sets* criando assim a lógica *fuzzy* como conhecemos atualmente.

As primeiras aplicações da lógica *fuzzy* datam de 1974, sendo hoje aplicada nas ciências ambientais, medicina, engenharia e em outras ciências [9].

A lógica nebulosa é largamente aplicada a sistemas complexos que não podem ser tratados eficazmente pela utilização de abordagens baseadas em matemática clássica, tendo em vista que estas não são suficientemente expressivas para servir como um meio de caracterização de *input-output*, dada as relações estabelecidas em um ambiente de imprecisão, incerteza e incompletude de informação [3].

Para [8] o que diferencia a lógica *fuzzy* da lógica booleana é a capacidade desta de se aproximar do mundo real onde não existe somente respostas extremas. A lógica *fuzzy* da espaço ao meio termo apresentando ainda a possibilidade de mensurar o grau de aproximação da solução exata e assim inferir algo que seja necessário.

A *Lógica Fuzzy* pode ser esquematizada através da figura 1.

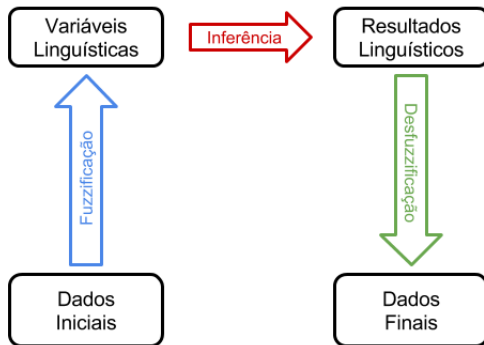


Figura 1: Sistema lógico fuzzy. Fonte: [8]

Para [6], o conceito fuzzy pode ser entendido como uma situação em que não é possível responder simplesmente “sim” ou “não”. Mesmo conhecendo as informações necessárias sobre a situação, dizer algo entre “sim” e “não”, como “talvez” ou “quase”, torna-se mais apropriado, sendo, portanto, uma técnica adequada para o processamento de dados reais, capturados por sensores instalados em *Smart Homes*.

2. MATERIAIS E MÉTODO

Nesse experimento foi criada uma aplicação que automatiza o sistema de iluminação, com o intuito de verificar a eficiência da lógica *fuzzy* quando aplicada à sistemas de controles de *smart homes*. A aplicação em questão requer interação mínima do ser humano, – precisando apenas que execute o aplicativo a partir de um computador pessoal – o que, consequentemente, torna mais eficiente o controle do consumo de energia, haja vista que diminui a possibilidade de falhas causadas por erros de operação.

O sistema proposto é composto de um sensor de presença e um de luminosidade, além de um atuador que regula a potência do *dimmer* conectados a um *gateway open source* que atua a partir de um conjunto de regras *fuzzy* previamente definidas.

A aplicação capta informações do ambiente, lidas através da conexão com o *gateway* via *REST* e realiza inferência

fuzzy sobre os dados captados para definir como o sistema irá atuar, ou seja, ligar, não ligar ou desligar o sistema de iluminação e a iluminância a ser aplicada.

REST é um estilo arquitetônico proposto por [10]. Representational State Transfer (*REST*) é definido como um conjunto coordenado das seguintes restrições arquiteturais: *Client-Server, Stateless, Cache, Uniform Interface, Layered System, and the optional Code-on-Demand* [4]. O *REST* foca nos papéis dos componentes, nas restrições sobre sua interação com outros componentes e na sua interpretação de elementos de dados significantes.

Para o desenvolvimento do sistema foi utilizado como *gateway* o Software Livre *openHAB 1.7.1*, que viabilizou a interligação entre os dispositivos que se comunicam através de protocolos distintos entregando uma *API RESTful*.

OpenHAB é um software para a integração de diferentes sistemas e tecnologias de automação doméstica em uma única solução que permite mais abrangente regras de automação e que oferece interfaces de usuário uniformes [2].

O *openHAB* foi instalado em um *Raspberry PI* que por sua vez estava conectado à rede *Wifi* através de um *access point* e conectado à rede *Z-wave* através de um *Z-wave USB Stick* da Aeon.

Os sensores de luminosidade e presença foram conectados a um arduino com *Wifi* integrado, que se comunica com o *gateway* através do protocolo *MQTT*. O atuador *dimmer* utilizado foi o *Micro Smart Energy Illuminator* da Aeon Labs que se comunica com o *gateway* através do protocolo *Z-wave*.

O protocolo *IBM MQ Telemetry Transport (MQTT)* foi projetado para trocar mensagens entre pequenos dispositivos em baixa largura de banda, ou conexões caras, e para enviar mensagens no modo confiável fazendo uso de *TCP/IP* [1].

A arquitetura do sistema de teste pode ser melhor compreendida observando a figura 2.

Por fim usamos um notebook conectado a esta rede *Wifi* para executar o sistema de controle do experimento que foi implementado com a utilização da Linguagem *Python 2.7*, utilizando a *API PyFuzzy* para a geração do conjunto de regras, *fuzzificação* e *desfuzzificação* do sistema.

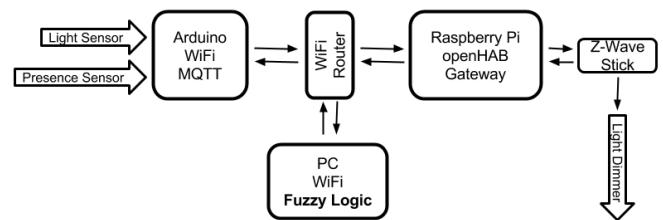


Figura 2: Arquitetura do Sistema de Controle

Os componentes do sistema descrito acima pode ser visto na figura 3.

A utilização da *Lógica Difusa* foi motivada pela sua adequação a processamento de dados reais captados de sensores cuja leitura modifica-se de forma gradual, como por exemplo a quantidade de luz de um ambiente, que gradativamente altera-se durante o dia.

O sistema projetado para esse experimento, capta o nível de luminosidade do ambiente através de um sensor que mede a luminosidade, retornando valores que variam numa escala de 0 a 100, processa através da máquina de inferência *fuzzy*



Figura 3: Sistema de controle fuzzy

retornando um valor na mesma escala que então é aplicado ao *dimmer* regulando sua potência.

No entanto, o sistema somente permanecerá ligado se for captada presença no ambiente, pois acoplado ao sistema existe ainda um outro sensor (de presença) que retorna o valor 0 ou 1. O valor 1 indica que existe alguém presente e, portanto, a iluminação será mantida ligada e a potência do *dimmer* será regulada de acordo com o grau de pertinência estabelecido pela máquina de inferência. O valor 0 indica que não há ninguém presente no ambiente e, então, a iluminação será desligada.

O sistema acima descrito pode ser facilmente adaptado para um sistema de segurança, fazendo com que sejam ligadas as luzes do ambiente – servindo assim de alerta – caso haja presença de intruso(s).

O processo de *fuzzificação* foi definido com as seguintes variáveis nominais:

Variável de entrada (input) *Sensorstate* : Leitura do sensor de luminosidade enviada através do *REST* via Web – valores variando de 0 a 100; *Sensorpres* : Leitura do sensor de presença enviada através do *REST* via Web – valores 0 ou 1.

Variável de saída (output): *Atua dim* : Variável que recebe o valor *desfuzzificado* – valores variando de 0 a 100.

A função de pertinência foi definida numa escala (*range*) a partir da leitura do sensor de luminosidade que varia de 0 a 100 e do sensor de presença que assume os valores 0 ou 1. As relações de pertinências das variáveis linguísticas foram definidas, conforme segue:

```
\small{
Sensor_state (Luz ambiente)
Low := (-1, 0) (0, 1) (10, 0);
Med := (5, 0) (25, 1) (45.5, 0);
Hig := (30.9, 0) (60, 1) (70.5, 0);
Hhig := (55.5, 0) (85, 1) (86, 0);

Sensor_pres (Presença)
Des := 0;
Lig := 1;

Atua_dim (Potência do Dimmer)
Desliga := (0, 0) (0, 1) (0, 0);
Baixa := (0, 0) (1.5, 1) (10.5, 0);
Media := (9.9, 0) (25, 1) (50.5, 0);
Alta := (40.9, 0) (60, 1) (80.5, 0);
H_alta := (70.9, 0) (85, 1) (100, 0);}
```

Foram testadas as diversas formas de *desfuzzificação* disponibilizada pela API *PyFuzzy* – *COG* (centro de massa), *LM* (valores máximos a esquerda), *RM* (valores máximos a direita) e etc. – contudo o método que apresentou melhor resultado, para a aplicação implementada foi o centróide (*COG*).

O conjunto de regras *difusas* foi definido conforme tabela 1.

Tabela 1: Regras

Condição 1	Condição 2	Consequência
Luz Amb. Baixa	E Presença Sim	Ilum. Sup Alta
Luz Amb. Baixa	E Presença Não	Desliga
Luz Amb. Média	E Presença Sim	Ilum. Alta
Luz Amb. Média	E Presença Não	Desliga
Luz Amb. Alta	E Presença Sim	Ilum. Média
Luz Amb. Alta	E Presença Não	Desliga
Luz Amb. Sup Alta	E Presença Sim	Ilum. Baixa
Luz Amb. Sup Alta	E Presença Não	Desliga

3. RESULTADOS

Conforme podemos observar na figura 3, fazendo o sistema inferir sobre todos os valores possíveis do domínio da função de pertinência – de 0 a 100 –, o sistema comportou-se de forma adequada, diminuindo de gradualmente a luminosidade aplicada ao ambiente através de pequenas alterações na potência do *dimmer* modificando assim, o nível de luminosidade aplicada ao ambiente, de acordo com a quantidade de luz captada pelo sensor de luminosidade. Ou seja, se o ambiente estiver bem iluminado (com luz natural) o *dimmer* será regulado com menor potência e vice versa.

A figura 3 abaixo demonstra o decaimento gradual da potência repassada para o *dimmer* à medida que o nível de iluminação natural aumenta gradativamente.

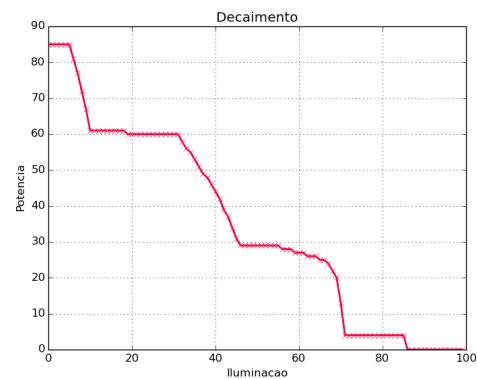


Figura 4: Decaimento do Nível de luminosidade

Na figura 5, podemos observar o comportamento do sistema, numa simulação de decaimento de luz ambiente e, consequentemente, incremento gradual do nível de luminosidade do sistema, de acordo com os valores *desfuzzificados*.

4. CONCLUSÃO

O controlador de luminosidade implementado com a utilização da *Lógica Fuzzy* apresentou resultados satisfatórios.

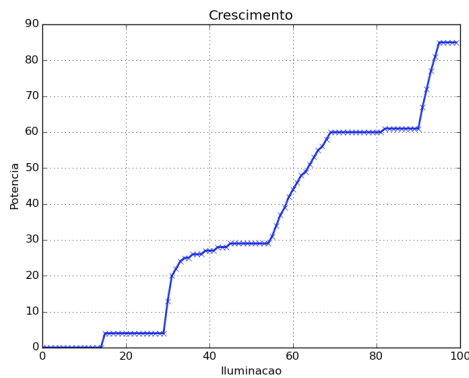


Figura 5: Incremento do nível de luminosidade

O conjunto de regras, então definido, permitiu que o sistema se comportasse de maneira que as mudanças no nível de luminosidade ocorressem gradualmente e de forma sutil, tornando-se quase imperceptível aos usuários, em consonância com a ideia de presença ubíqua dos dispositivos inteligentes.

Deve-se ressaltar ainda que a inclusão de um sensor de presença no sistema proporcionou maior economia energética, tendo em vista que caso não seja captada presença no ambiente o sistema automaticamente desliga a iluminação. Outro ponto a ser destacado é a facilidade de implantação, bem como a possibilidade de conversão em um sistema de segurança, bastando apenas realizar pequenas modificações na aplicação implementada.

Uma limitação que podemos observar refere-se ao fato dos valores *desfuzzificados* precisarem ser discretizados antes de repassar para o *REST* e, como consequência, pequenas alterações – no nível de luz ambiente – não modificam o nível de luminosidade do sistema tendo em vista que, para tanto, deveríamos repassar valores contínuos para regular a potência do *dimmer*.

Como podemos observar nos gráficos acima, esse comportamento do sistema acaba gerando repetições de valores (potência do *dimmer*), ocasionando assim um decaimento/crescimento não linear do nível de luminosidade aplicada pelo sistema.

Observamos durante o processo de experimentação que o conjunto de regras deve ser exaustivamente testado, para que o sistema atue da forma desejada. O conhecimento especialista tem uma importância fundamental nessa fase, podendo abreviar o processo de experimentação.

Outro ponto a ser destacado é que um conjunto com muitas regras nem sempre funciona da forma desejada, podendo ocorrer perda de informação resultando em um comportamento inesperado do sistema.

Sugerimos como trabalhos futuros que sejam realizados experimentos com o controlador de luminosidade em estudo, implementado-o com diferentes técnicas de inferência (lógica Bayesiana, por exemplo) comparando os resultados obtidos, a fim de verificar a mais eficiente.

Agradecimentos

Os autores agradecem à UFS pelo suporte financeiro concedido [Edital POSGRAP/COPES/UFS No 03/2014 14/2012

(HERMES), Processo 008325/14-72] e ao CNPq pela bolsa de produtividade concedida ao pesquisador Hendrik Macedo [Modalidade/Nível: DT-II, Processo 310446/2014-7].

Referências

- [1] Ibm knowledge center. <http://www-01.ibm.com/support/knowledgecenter/>. Accessed: 2015-12-11.
- [2] Whats openhab? <http://www.openhab.org/>. Accessed: 2015-12-11.
- [3] Z. L. A. Fuzzy sets. Dept. Electrical Engineering and Electronics Res. Lab., Univ. California, Berkeley, CA, 1965.
- [4] M. Athanasopoulos and K. Kontogiannis. Extracting {REST} resource models from procedure-oriented service interfaces. volume 100, pages 149 – 166, 2015.
- [5] L. Atzori, A. Iera, and G. Morabito. The internet of things: A survey. volume 54. Computer Networks, 2010.
- [6] G. P. Chenci, D. G. Rignel, and C. A. Lucas. Uma introdução à lógica fuzzy. 1(1), 2011.
- [7] H. X. d. e. O. G. H. d. C. Coelho, L. dos S. end Araújo and N. Mendes. Projeto de controlador pi nebuloso aplicado ao sistema climatização de uma edificação. Proceedings of the 10th Brazilian Congress of Thermal Sciences and Engineering – ENCIT 2004 Braz. Soc. of Mechanical Sciences and Engineering, 2004.
- [8] E. COX. The fuzzy systems handbook: a practitioner’s guide to building, using, and maintaining fuzzy systems. 1994.
- [9] E. M. S. de Paula and M. J. N. de Souza. Lógica fuzzy como técnica de apoio ao zoneamento ambiental. Florianópolis, 2007.
- [10] R. Fielding. Architectural styles and the design of network-based software architectures (ph.d. thesis). University of California, 2000.
- [11] Y. He, I. S. end Yunhuai Liu, and Y. Gu. Smart city. volume 2014. International Journal of Distributed Sensor Networks, 2014.
- [12] M. O’Grady and G. O’Hare. How smart is your city? *U.S. National Library of Medicine*, 335, 2012.
- [13] J. V. P. Sousa. Desenvolvimento de um sistema fuzzy embarcado para controle de iluminação. XX Congresso Brasileiro de Automática, 2014.