Unifying fuzzy controller for IEQ: implementation in a *Raspberry Pi*

Miguel Molina-Solana[1] **Maria Ros**[1] **Miguel Delgado**[1]

[1]Dept. Computer Science and AI. Universidad de Granada

Abstract

Recent developments in computing and hardware manufacturing are allowing many pieces of software to be embedded in general-purpose, low-cost devices. This work presents and tests the feasibility and performance of implementing, in such a device, a previously developed unified fuzzy controller for managing the different aspects involved in Indoor Environment Quality (IEQ). This unifying controller overcomes the potentially inefficient interactions between several traditional controllers, and its implementation is able to accommodate the higher computational needs of the fuzzy formalism.

Keywords: Indoor Environmental Quality, Fuzzy Logic Controller, Raspberry Pi

1. Introduction

Indoor Environmental Quality (IEQ) aims at optimizing traditional energy control inside a building by taking into account users' comfort. In this context, reducing the energy consumption of HVAC (heating, ventilation, and air conditioning) systems, while maintaining an appropriate comfort level has gained great relevance lately due to the almost ubiquitous presence of these systems in many buildings, and their great contribution towards the energy consumed in those environments.

Conventional control of HVAC systems consists of On-Off and PID controllers (proportional-integral-derivative controllers), which try to minimize the error between the studied variable and the fixed setpoint throughout a defined mathematical model. Therefore, they do not directly address users' comfort, and do not easily accommodate further controls than just very reactive ones.

However, according to [1], HVAC systems could be considered as MIMO (Multiple Input Multiple Output) control problems, and henceforth be described as multi-criteria problems, since they analyse interrelated variables to extract values for a set of outputs. For instance, by changing the temperature in a room, the humidity level can also be affected. Additionally, they are influenced by a wide range of uncertain parameters, such as external air temperature and occupants' activities or preferences, that might potentially affect the controller normal operations [2].

In this context, traditional PIDs provide reasonable solutions, but they fail to model the inherent uncertainty of the dynamics of HVAC systems, which are more easily characterised using linguistic labels and rules [3, 4]. Fuzzy Logic Controllers (FLCs) appear as a viable alternative to conventional controllers, since they do not require a mathematical modelling [5] and they are prepared to handle different criteria. Furthermore, they represent the dynamic of the HVAC system according to the knowledge of a human expert. Finally, their efficiency and lower energy consumption (while satisfying the indoor comfort requirements) comparing to PID controllers, has been completely demonstrated [6]. These benefits, however, come at the cost of added complexity and higher computational costs [7].

In a previous work [8], we surveyed different proposals of fuzzy controllers for HVAC systems. In it, we identified that there was a lack of systems tackling the issues of IEQ and users comfort, as well as the interrelations between different dimensions. We therefore, proposed, implemented and tested an unifying fuzzy controller to address such requirements. Our devised FLC took into consideration changes in both outdoor and indoor temperature, relative humidity levels and CO_2 concentration, in addition to users' preferences in order to control the different aspects affecting the IEQ of a room by means of operating a HVAC system.

Fortunately, new proposals have recently appeared in literature, certainly demonstrating the big interest IEQ and comfort control currently have among researchers and industry. For instance, [9] presented a comparative study of different methods for building thermal control, based on artificial intelligence techniques. [10] also proposed and additional layer to provide an existing PID with the holistic knowledge on interrelations between variables.

The current work presents the implementation of the previously proposed unifying fuzzy controller in a general-purpose, low-cost device: a Raspberry Pi[1] with the goal of proving the feasibility of such implementation. The paper is therefore organized as follows. Section 2 summarizes the most relevant aspects of the previously designed Fuzzy Logic Controller, whereas Section 3 describes the *Raspberry Pi*. Section 4 presents the experimentation

[1]http://www.raspberrypi.org/

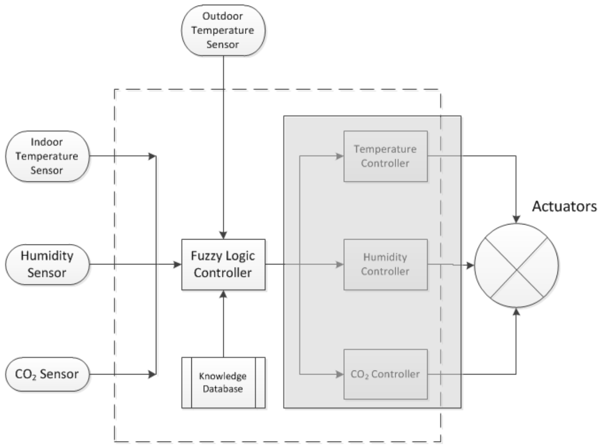


Figure 1: Architecture of original sensors and PID controllers coupled with the FLC

we have done in order to test the feasibility and performance of the proposed implementation. The work ends with some conclusions and outlining future work.

2. Unifying Fuzzy Logic Controller

In a previous work [8], we surveyed the literature on IEQ controllers and discovered that most of them (with some exceptions such as [11, 12]) were only focused on one dimension (mainly temperature) and not taking into account the potential interrelations within the different dimensions.

With the aim of filling such a gap, we then proposed a distributed control system with a Supervisory Fuzzy Logic Controller to reset and establish the set-points of the specific PID controllers with the aim of maintaining the Indoor Environment Quality (IEQ), and considering an holistic view of the environment.

In particular, the inputs to our system were five sensors (indoor and outdoor temperature, humidity, CO_2 and lighting sensors), whereas the outputs were four actuators (air conditioner program, lighting level, temperature setpoint and humidity level). All in all, the aim of our controller was to apprise the interrelations as decisions taken on one of the outputs might influence the general context, and, consequently, make changes in the others.

We based our proposal on the usage of intelligent techniques and fuzzy logic, which has proven very effective in control applications where the exact mathematical model is not known, but the behavior can be effectively defined based on the experience. We advocated for attaching our Fuzzy Control Module to the current PID controllers with the aim of improving the flexibility and adjustment of the control system (see Figure 1 for a general scheme of the system).

Our controller relies on a *Knowledge base* which collects the expert knowledge about the dynamic behaviour of the system. It is represented as a set of *IF-THEN* rules and *membership functions*. Table 1 shows a subset of the defined rules in the system to illustrate how the knowledge is encoded within this formalism (variables and their values are explained in the following paragraphs).

As said, five distributed sensors (indoor and outdoor temperature, relative humidity and CO_2 concentration sensors, lighting level) are placed in the room, monitoring it continuously. From now on, we will named sensors $S_{temp_{indoor}}$, $S_{temp_{outdoor}}$, S_{RH}, S_{CO_2} and S_{light}, respectively.

The domain of every input was described over a set of three linguistic labels: *Low*, *Medium* and *High*. The particular membership functions depend on each input, but are always in the shape of *Trapezoidal functions*, defined by equation 1. For an illustrative purpose, Figure 2 shows the different trapezoidal membership functions for the input sensor measurements. Output variables were defined in a similar manner, and will not be depicted here.

Trapezoidal functions are defined as follows:

$$\mu_L(x_i) = \begin{cases} 0 & if \quad (x_i < a) \mid (x > d) \\ \frac{x_i - a}{b - a} & if \quad a \le x_i \le b \\ 1 & if \quad b \le x_i \le c \\ \frac{d - x_i}{d - c} & if \quad c \le x_i \le d \end{cases} \tag{1}$$

where L can be *Low*, *Medium* and *High*, a and d are the end points of the trapezoidal membership function, b and c are the peak points of the trapezoidal membership function, and x_i is the ith sensor.

Regarding the outputs, we were able to manage the air conditioner program (A_{Air}), HVAC temperature setpoints ($A_{temp_{level}}$), lighting level (A_{ligth}) and humidifier functioning ($A_{h_{level}}$). These linguistic variables were defined depending on how the actuator worked:

- Air conditioner program (A_{Air}) - *hot*, *cold*, *dry*
- Temperature setpoint($A_{temp_{level}}$) - *down*, *hold*, *up*
- Lighting level (A_{ligth}) - *low*, *medium*, *high*
- Humidifier level ($A_{h_{level}}$) - *off*, *low*, *standard*,

The operation of a Fuzzy Logic Controller (FLC) is generally based on the *Inference Engine*, which is responsible of accepting the inputs after the fuzzification process, and provide the outputs values to the defuzzification module in order to obtain the output measurements, according to the rules defined in the Knowledge Base. For facilitating users' input and interpretability, we decided to used the Mamdani maxmin method [13] for the inference process (although further performance improvements could be achieved using Sugeno's sum-prod method) during the operation time:

$$\mu_{R_i}(x) = \alpha_{i1} \wedge \alpha_{i2} \wedge \alpha_{i3} \wedge \alpha_{i4} \wedge \alpha_{i5} \tag{2}$$

$$\mu_{Output_i}(x) = max\mu_{R_1}(x), \mu_{R_2}(x), ..., \mu_{R_{17}}(x) \tag{3}$$

Table 1: Example of rules in the Knowledge Database

IF $S_{temp_{indoor}}$ IS Medium AND $S_{temp_{outdoor}}$ IS High **THEN** $A_{temp_{level}}$ IS Hold AND $A_{h_{level}}$ IS Standard
IF S_{RH} IS Low AND $S_{temp_{indoor}}$ IS Low **THEN** A_{Air} IS Hot AND $A_{h_{level}}$ IS High AND $A_{temp_{level}}$ IS Up
IF S_{RH} IS Medium AND $S_{temp_{indoor}}$ IS Medium **THEN** $A_{h_{level}}$ IS Standard
IF S_{RH} IS High AND $S_{temp_{indoor}}$ IS Medium **THEN** A_{Air} IS Dry AND $A_{h_{level}}$ IS Continuous

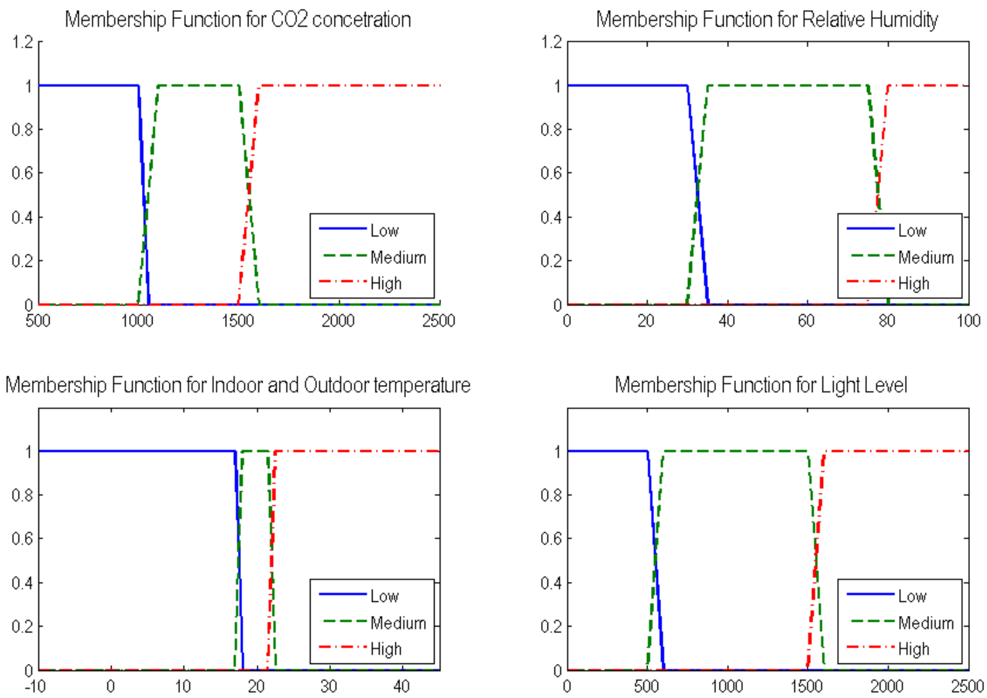


Figure 2: Membership Function of the input variables

where x is the input sensor measurements, α_i is the degree of a given input that satisfies the condition of the ith rule (R_i), and μ_{output_i} is the aggregation of fuzzy set outputs from all rules for $output_i$.

Finally, and in order to test our proposal, we firstly developed a web simulator in which users were able to define and test particular sets of rules by means of adjusting the different FLC parameters. We also reported how the unifying fuzzy controller was applied to a pilot room, during a month, to monitor and control its IEQ, comparing the performance with other PID controllers. Results were very promising.

3. Low cost device: Raspberry Pi

Low-power, data intensive computing is coming to be an area of great interest, both academically and in industry. Recent advancements in this area are allowing that different devices can run complex pieces of software in a distributed and embedded way. New applications are therefore overcoming the traditional technological limitations and achieving a state of feasibility and wide implementation.

A myriad of low-cost devices (e.g. Raspberry Pi, Arduino, BeagleBoard, Cubox, HummingBoard, and many others) are currently in the market, each one with different capabilities and aims. Among all of them, the first two ones are undoubtedly the most popular ones.

Raspberry Pi (RPi) is a credit-card-size single-board computer with and ARM processor. It is a low-cost computer capable of doing everything you'd expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games. What's more, the *Raspberry Pi* has the ability to interact with the outside world, and has been used for a wide array of projects [14]. Two models are currently available ($A+$ and $B+$), which vary in their memory and connectivity capacities and layout.

RPI provides plenty of freedom of choice regarding operating system and programming language. In particular, it supports several Linux distributions, being *Raspbian* a Debian-based one especially optimized for the *Raspberry Pi* hardware).

As said, *Raspberry Pi* can be used for many applications [14], and particularly in the Home environment: in the context of home security [15] and comfort [16] we can find several works reporting the successful usage of the *RPi* to implement controllers. Books exists describing several projects, such as the ones by Bell [17], Dennis [18] and Goodwin [19]. Like in our case, *Raspberry Pi* is often selected for its good developer community support and low cost.

For our particular implementation, we used *RPi*'s model B+, which is the most advanced model currently in the market (as for January 2015). It has a 700 MHz ARM processor and 512 MB of memory. The total cost of our implementation (including the Raspberry Pi board itself, a case, the power supply and the SD card) was less than $ 80.

As the unifying fuzzy controller was originally implemented in Java, we needed a Java Virtual Machine over the Raspberry Pi. Fortunately, there is currently an official release of Java 8 by Oracle for the *RPi*. We installed it on top of the Raspbian operating system. For comparative purposes, we also install the fuzzy unifying controller on a desktop machine running Windows 7 with a 2.1 GHz Intel Xeon E5-2620v2 processor and 3 GB of memory.

Table 2 summarizes the specs of the hardware devices and the software used for experimentation and comparison.

4. Performance of the implementation

We have deployed the Java web application implementing the FLC and the simulator on both the *Raspberry Pi* and in a Windows 7 server (see details in Table 2). In both cases, we have used version 1.8.0 of Java and Apache Tomcat server version 8.0.18.

In order to test the performance of the fuzzy controller running on the *Raspberry Pi*, we have made use of the developed simulator [8], which provide us with the capability of easily sending requests. In particular, we have made 9 different request simulating different inputs coming from sensors. We performed requests with a two-seconds delay among them.

Figure 3 shows the 9 requests using the controller on the *RPi*, while Figure 4 shows the same 9 requests to the controller on the PC. The average response time in the case of the *Raspberry Pi* is 141 ms, while it takes just 40 ms for the server. All in all, the order of those figures, compared with the big latencies that actuators (with the exception of lighting) have in the environment, is perfectly valid for the problem we have in hands.

Note here, that this delays are certainly affected by the number of rules in the Knowledge Base (17 rules in our case), and the number of antecedents (pairs variable-value separated by ANDs) in each rule. The higher the number of rules or the higher the number of antecedents, the longer it would take the FLC to process the requests and obtain an output.

We also tested the performance of the devised fuzzy controller in a real environment (the one described in our previous work [8]): a room equipped with several sensors for IEQ monitoring and control. Different fuzzy comfort rules were defined on the controller in order to manage the different actuators in the room, and the HVAC in particular.

In this real environment, data are collected from sensors every 15 minutes. Probably the granularity could increase up to, maybe, minutes, but not much further due to the great inertia of the scenario. If data were collected every second, and actions took place accordingly, it would be impossible to check if the applied measures are really working, and overlapping of instructions would occur.

From the performance results obtained in the simulating environment (with responses around 150 ms), it was already clear that the unifying fuzzy controller running in the *Raspberry Pi* would have no problem to compute the actions and send the commands within that 15-minutes interval. Those expectations were certainly confirmed by applying the *RPi* implementation to the real scenario: the unifying fuzzy controller, deployed on the *RPi*, had no problem on monitoring and controlling the IEQ of the demo room, and to perform the necessary calculations.

The performance of the controller, in terms of achieving a successful degree of control, was already reported in our previous work [8], and has been improved during the last year (although these improvements are yet to be reported, due to patent issues). However, as this fact is not relevant for the present work (which deal with the feasibility of implementing the controller in a low-cost device), we will not address it in this work. The reader interested on the performance of the controller should definitively refer to the mentioned reference.

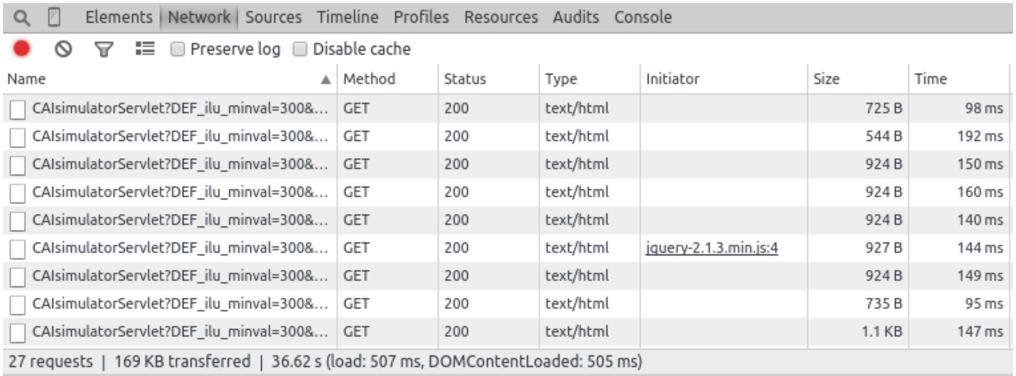
5. Conclusion

We have described in this paper the implementation in a low-cost device of a previously presented integrating fuzzy controller to manage different variables affecting the Indoor Environment Quality in a room. That fuzzy controller allowed us to unify different controllers into only one that controls several variables being fully aware of the interdependencies between variables.

The main result of this paper is that our proposal for implementing the unifying fuzzy controller in a low-cost device is feasible and achieves indeed great performance. Even though the fuzzy formalism introduces additional computations, these can be done in real-time in a device such as the *Raspberry Pi.* The results proved the potential of Raspberry Pi for the design of a compact and affordable unifying fuzzy controller for IEQ control that can be further easily deployed in multiple environments.

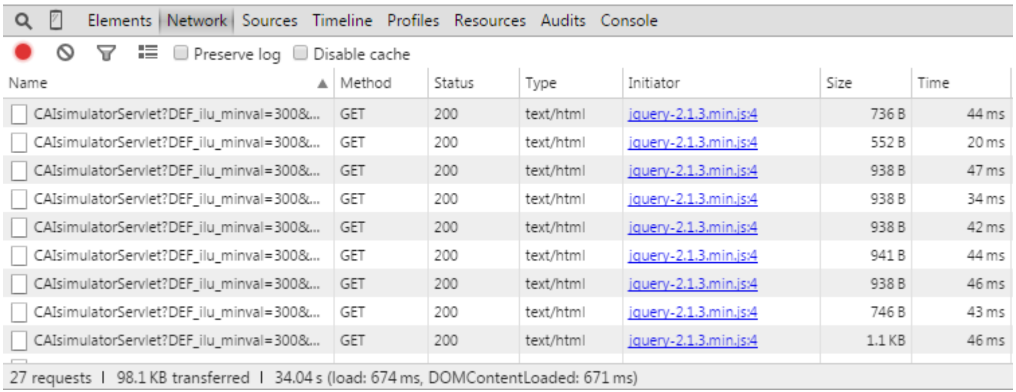
	Raspberry Pi Model B+	*desktop server*
Processor	700 MHz ARMv6	2.1 GHz Xeon E5-2620v2
RAM memory	512 MB	3 GB
Operating System	Raspbian Dec 2014	Windows 7 Enterprise
Java version	1.8.0-b132	1.8.0_31-b13
Tomcat version	8.0.18	8.0.18

Table 2: Specifications of employed hardware and software platform.



Name	Method	Status	Type	Initiator	Size	Time
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html		725 B	98 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html		544 B	192 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html		924 B	150 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html		924 B	160 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html		924 B	140 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	927 B	144 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html		924 B	149 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html		735 B	95 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html		1.1 KB	147 ms

Figure 3: Processing time for each request on *RPi*



Name	Method	Status	Type	Initiator	Size	Time
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	736 B	44 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	552 B	20 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	938 B	47 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	938 B	34 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	938 B	42 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	941 B	44 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	938 B	46 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	746 B	43 ms
CAIsimulatorServlet?DEF_ilu_minval=300&...	GET	200	text/html	jquery-2.1.3.min.js:4	1.1 KB	46 ms

Figure 4: Processing time for each request on server

We foreseen our current work as the first stages towards a broader scenario in which HVAC systems and other information sources are connected into a small, low-powered device that act as a central smart controller. *RPi* has demonstrated that it is indeed able to support a Java implementation of a fuzzy controller. We acknowledge the great potential of this trend of embedding powerful smart software controllers into smaller and cheaper devices that can be widely installed in the environment.

Acknowledgments

This work was partially funded by the Spanish Ministry of Economy (project TIN2012-30939) and the European Union (*Energy IN TIME* project, grant agreement no. 608981). The authors also thanks IFSA's reviewers for their valuable comments on earlier versions of this work.

References

[1] H. Mirinejad, K.C. Welch, and L. Spicer. A review of intelligent control techniques in HVAC systems. In *Energytech, 2012 IEEE*, pages 1–5, 2012.

[2] David Heinzerling, Stefano Schiavon, Tom Webster, and Ed Arens. Indoor environmental quality assessment models: A literature review and a proposed weighting and classification scheme. *Building and Environment*, 70:210–222, 2013.

[3] T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-15(1):116–132, 1985.

[4] Lofti Zadeh. Fuzzy sets and systems. *Int J General Syst*, 17:129–138, 1990.

[5] Servet Soyguder, Mehmet Karakose, and Hasan Alli. Design and simulation of self-tuning pid-type fuzzy adaptive control for an

expert hvac system. *Expert Systems with Applications*, 36(3, Part 1):4566–4573, 2009.

[6] D. Kolokotsa. Comparison of the performance of fuzzy controllers for the management of the indoor environment. *Building and Environment*, 38(12):1439–1450, 2003.

[7] Rafael Alcalá, Jose M. Benítez, Jorge Casillas, Oscar Cordón, and Raúl Pérez. Fuzzy control of hvac systems optimized by genetic algorithms. *Applied Intelligence*, 18(2):155–177, 2003.

[8] Miguel Molina-Solana, Maria Ros, and Miguel Delgado. Unifying fuzzy controller for indoor environment quality. In *Proc. 2013 Joint IFSA World Congress NAFIPS Annual Meeting*, pages 1080–1085, 2013.

[9] Jin Woo Moon, Sung Kwon Jung, Youngchul Kim, and Seung-Hoon Han. Comparative study of artificial intelligence-based building thermal control methods - application of fuzzy, adaptive neuro-fuzzy inference system, and artificial neural network. *Applied Thermal Engineering*, 31(14-15):2422–2429, 2011.

[10] M. Castilla, J.D. Álvarez, J.E. Normey-Rico, and F. Rodríguez. Thermal comfort control using a non-linear {MPC} strategy: A real case of study in a bioclimatic building. *Journal of Process Control*, 24(6):703–713, 2014.

[11] Jeongho Kang and Sekwang Park. Integrated comfort sensing system on indoor climate. *Sensors and Actuators A: Physical*, 82(1-3):302–307, 2000.

[12] A. Kumar and G. P. Hancke. An energy-efficient smart comfort sensing system based on the IEEE 1451 standard for green buildings. *IEEE Sensors Journal*, 14(12):4245–4252, 2003.

[13] E.H. Mamdani. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *Computers, IEEE Transactions on*, C-26(12):1182–1191, 1977.

[14] Simon Monk. *Raspberry Pi Cookbook*. O'Reilly Media, Inc., 1st edition, 2014.

[15] S. Kanaga Suba Raja, C. Viswanathan, D. Sivakumar, and M. Vivekanandan. Secured smart home energy monitoring system (sshems) using raspberry pi. *Journal of Theoretical and Applied Information Technology*, 66(1):305–314, 2014.

[16] J. Zacek and M. Janosek. Programmable control of heating for systems with long time delays. In *Procs. 15 Int. Carpathian Control Conference (ICCC)*, pages 705–709, 2014.

[17] Charles Bell. *Beginning Sensor Networks with Arduino and Raspberry Pi*. Apress, 2013.

[18] Andrew K. Dennis. *Raspberry Pi Home Automation with Arduino*. Packt Publishing Ltd, 2013.

[19] Steven Goodwin. *Smart Home Automation with Linux and Raspberry Pi*. Apress, 2013.