Prediction of indoor temperatures for energy optimization in buildings

Pablo Rodríguez-Mier, Marc Fresquet, Manuel Mucientes, and Alberto Bugarín

Centro Singular de Investigación en Tecnoloxías da Información (CiTIUS).
Universidade de Santiago de Compostela,
pablo.rodriguez.mier@usc.es, marc.fresquet@rai.usc.es
{manuel.mucientes, alberto.bugarin.diz}@usc.es

Abstract. The reduction of energy consumption in buildings is one of the goals to improve energy efficiency. One way to achieve energy savings in buildings is to develop intelligent control strategies for heating systems that are able to reduce power consumption without affecting the thermal comfort. An intelligent control system must be able to predict the temperature of the building in order to manage the heating system. In this paper, we present a rule-based model that is able to predict the indoor temperature for different values of k (hours ahead in time). The model has been learned with FRULER, a genetic fuzzy system that generates accurate and simple knowledge bases. Our approach has been validated with real data from a residential college.

Keywords: energy optimization, indoor temperatures prediction, TSK fuzzy rules for regression, genetic fuzzy systems

1 Introduction

Buildings account for 40% of the total energy consumption in the EU, according to European Directive 2010/31/EU on energy efficiency in buildings. Because of the expansion this sector is currently experiencing, a rise of that percentage will be inevitable. Therefore, it seems clear that the reduction of energy consumption and the use of energy from renewable sources in the building sector will play a key role in future measures to reduce emissions of greenhouse gases.

One way to achieve energy savings in buildings is by reducing the total working hours of heating systems. However, a decrease in the total usage may lead to important decreases of indoor temperatures that can affect thermal comfort. In order to prevent this, automatic heating control systems must predict the future indoor temperature for a particular control policy in order to find the best strategy that minimizes power consumptions while keeping thermal comfort.

Current methods for indoor temperature prediction [3] are mostly based on physical model simulations [13] and black-box machine learning methods [5, 14, 1, 12]. Physical models describe the building behaviour by solving theoretical equations that describe to a certain precision the different dynamics and interactions between the variables. Although these methods are very powerful to

© Ediciones Universidad de Salamanca

simulate the different dynamics of a building, especially when there is no real data available, in general these methods are: 1) very time-consuming since they require many simulation hours, which prevents their application for predicting temperatures in small temporal windows; and 2) complex to formulate, since it is very difficult to produce a detailed model of a complex building, especially when there are many unknown factors that can affect the temperature dynamics. On the other hand, machine learning models can overcome some of these limitations by learning the behaviour from real data. However, current techniques, which are mostly black-box models based on neural networks, are hard to interpret and thus the interaction of the different variables of the building remains unknown.

In this sense, the generation of accurate and interpretable models for indoor temperature prediction is fundamental for 1) modelling the energy-building behaviour and 2) discovering which are the most relevant variables that affect the indoor building temperature and are related to power consumption. Within this context, initiatives such as the EU LIFE-OPERE project [2], where this research is framed, have started. OPERE has among its goals the setting of efficient management systems in energy networks, both thermal and electrical, in existing installations with large energy consumption.

In this paper, we propose a rule-based regression model for indoor temperature prediction. To do so, we have modelled the indoor temperatures of a residential college using the FRULER Genetic Fuzzy System (GFS) [10]. The knowledge bases learned by FRULER include TSK fuzzy rules that accurately predict the temperature dynamics from a set of different predictors that can be measured both inside and outside the building.

2 FRULER: Fuzzy RUle Learning through Evolution for Regression

FRULER (Fuzzy RUle Learning through Evolution for Regression) [10] is a novel GFS that obtains accurate and simple linguistic TSK-1 fuzzy rule base models for regression problems. FRULER (Fig. 1) is composed of a new instance selection method for regression, a novel multi-granularity fuzzy discretization of the input variables, and an evolutionary algorithm that uses a fast and scalable method with Elastic Net regularization to generate accurate and simple TSK-1 fuzzy rules.

Instance selection. The objective of the instance selection module is to reduce the variance of the models, focusing the generated rules on the representative examples. The instance selection method for regression is an improvement of the CCISR (Class Conditional Instance Selection for Regression) algorithm [9], which is an adaptation for regression of the instance selection method for classification CCIS (Class Conditional Instance Selection) [4].

Multi-granularity fuzzy discretization. In a multi-granularity proposal, each granularity has a different fuzzy partition. The generation of the fuzzy

© Ediciones Universidad de Salamanca

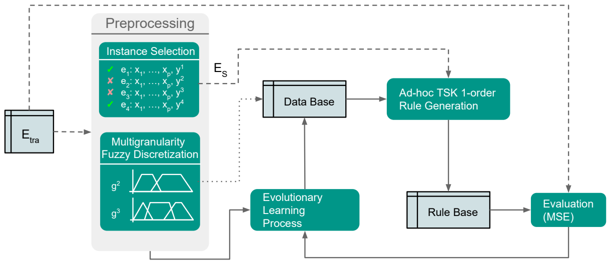


Fig. 1: FRULER architecture. Dashed lines indicate flow of datasets, dotted lines multigranularity information and solid lines represent process flow.

linguistic labels can be divided into two stages. First, the variable must be discretized to obtain a set of split points C^g for each granularity g. Then, given the split points, the fuzzy labels can be defined for each granularity. In regression problems (TSK-1 in our case), the discretization process must search for the split point that minimizes the error when a linear model is applied to each of the resulting intervals.

Evolutionary algorithm. The evolutionary algorithm learns a linguistic TSK model. The integration of the evolutionary algorithm with the preprocessing stage is as follows (Fig. 1):

- First, the instance selection process is executed over the training examples E_{tra} in order to obtain a subset of representative examples E_S.
- Then, the multi-granularity fuzzy discretization process obtains the fuzzy partitions for each input variable.
- Finally, the evolutionary algorithm searches for the best data base configuration using the obtained fuzzy partitions, generates the entire linguistic TSK rule base using E_S and evaluates the different rule bases using E_{tra}.

The chromosome is codified with a double coding scheme ($C = C_1 + C_2$). C_1 represents the granularity of each input variable. C_2 represents the lateral displacements of the split points of the input variables fuzzy partitions.

FRULER uses the Wang & Mendel algorithm to create the antecedent part of the rule base for each individual. The consequent part of the rules is learned using the Elastic Net method [15] in order to obtain the coefficients of the degree 1 polynomial for each rule. Elastic Net linearly combines the ℓ_1 (Lasso regularization) and ℓ_2 (Ridge regularization) penalties of the Lasso and Ridge methods, minimizing the following equation:

$$\hat{\beta} = \arg\min_{\beta} ||Y - X \cdot \beta||_2^2 + \lambda \cdot \alpha \cdot \|\beta\|_2^2 + \lambda \cdot (1 - \alpha) \cdot \|\beta\|_1 \qquad (1)$$

where β is the coefficients vector, Y is the outputs vector, X is the inputs matrix, λ is the regularization parameter and α represents the trade-off between ℓ_1 and ℓ_2 penalization. In order to solve the minimization problem of Elastic Net (Eq. 1), we used Stochastic Gradient Descent (SGD).

The rule base is generated using only those examples in E_s. In this manner, those examples that are not representative are not taken into account, the method avoids the generation of too specific rules, and reduces the time needed to create the rule base.

The fitness function is:

$$\mathit{fitness} = MSE(E_{tra}) = \frac{1}{2 \cdot |E|} \sum_{i=1}^{|E|} (F(x^i) - y^i)^2, \tag{2}$$

where E_{tra} is the full training dataset and $F(x^i)$ is the output obtained by the knowledge base for input x^i. Using all the examples for evaluation can be seen, in some way, as a validation process, as the rule base was constructed with a subset of them (E_S).

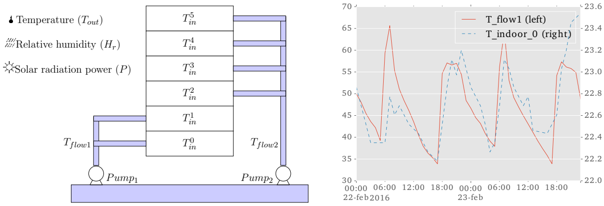
3 Indoor temperature prediction

The main goal of the OPERE project [2] is to implement efficient management systems in both thermal and electrical energy grids in existing installations with large energy consumption. To achieve this goal, in this work we propose a method that automatically learns an accurate and interpretable non-linear model using FRULER. The learned model predicts the indoor temperature dynamics of an existing building in order to find a better heating control that minimizes the energy consumption without sacrificing thermal comfort. Concretely, we focus this study on the residential facilities of Monte da Condesa, a building located at the University of Santiago de Compostela.

Monte da Condesa comprises a set of centers that act as separate buildings, but nevertheless maintain thermal interaction through their conditioning circuits connected to a common cogeneration plant. The building is about 25,000 m^2 and reached in 2013 a total power consumption of 5,747 MWh. The set of all centers is supervised by a SCADA system that has 469 variables (inputs and outputs) that are associated with signals from the primary heating circuits and power consumption. Signals are collected in two different ways: synchronous (sync) and asynchronous (async). Synchronous signals are sequentially sampled at a fixed interval of 10 s, whereas asynchronous signals are registered by detecting a change of a value above an stablished threshold. These signals include information about the indoor temperature of each floor, the outside temperature, the pumped water temperature of the heating systems, plus many other low level variables. In order to predict the indoor temperatures, we focus on the variables that may directly affect the temperature dynamics.

These variables are represented in Fig. 2a, which shows a high-level representation of the building. T_{in}^n corresponds with the indoor temperature sensors

© Ediciones Universidad de Salamanca



(a) Schema of the Monte da Condesa Residence with the related variables.

(b) T_{flow1} temperature (left) vs. T_{in}^0 indoor temperature (right).

Fig. 2: Monte da Condesa schema and sample representation of indoor and water temperatures.

of the building. In total, there are 6 different sensors ($T_{in}^0, \ldots, T_{in}^5$), one for each floor, which are the objective variables we want to predict. T_{flow1} and T_{flow2} refer to the temperature of the pumped water of the two heating systems installed in Monte da Condesa. T_{flow1} corresponds with the pumped water temperature of the heating system that feeds both floors 0 and 1, whereas T_{flow2} feeds the remaining floors. Note that, for the sake of clarity, in the following we will refer to T_{flow} instead of T_{flow1} and T_{flow2}, where $T_{flow} = T_{flow1} \forall n \in [0, 1]$ and $T_{flow} = T_{flow2} \forall n \in [2, 5]$. Fig. 2b shows an example of T_{flow} and T_{in}^0 between 22-02-2016 and 24-02-2016.

In addition to these SCADA variables, we also obtained the humidity (H_r) and solar radiation power (P) from *Santiago-EOAS*, a Meteogalicia [6] weather station situated approximately 100 meters from the reference building.

Moreover, the temperature (T_{out}^{MS}), relative humidity (H_r^{MS}) and sky state (sky^{MS}) predictions at Monte da Condesa are obtained from *MeteoSIX* [7], a galician numerical weather service that provides hourly predictions from the current day to four days in ahead.

Synchronous measures were downsampled to 1 h bins and asynchronous measures were converted into time series by appling linear interpolation and 1 h resampling. To summarize, the selected signals, sampled at 1 h interval (t) are:

- $T_{in}^n(t)$: indoor temperature at t of floor n (°C, async).
- $T_{out}(t)$: outside temperature at t (°C, async).
- $T_{flow1}(t)$: water temperature of the first heating system (1) at t (°C, sync).
- $T_{flow2}(t)$: water temperature of the second heating system (2) at t (°C, sync).
- $H_r(t)$: relative humidity (%, sync, Meteogalicia).
- $P(t)$: global solar radiation power (W/m^2, sync, Meteogalicia).
- $T_{out}^{MS}(t)$: outdoor temperature prediction (°C, MeteoSIX).
- $H_r^{MS}(t)$: relative humidity prediction (%, MeteoSIX).
- $sky^{MS}(t)$: sky state prediction (MeteoSIX).

© Ediciones Universidad de Salamanca

Only $T_{in}^n(t)$ and $T_{out}(t)$ are directly used into the model as predictor variables at t. The rest are used to predict related variables at $t+k$, as the predictions of MeteoSIX are usually biased:

- $\hat{T}_{out}(t+k)$: A correction is performed over the predicted outdoor temperature $T_{out}^{MS}(t+k)$ in order to approximate these values to the real ones. So that, the real outdoor temperature $T_{out}(t)$ is taken into account to make this adjustment.

$$\hat{T}_{out}(t+k) = T_{out}^{MS}(t+k) + (T_{out}^{MS}(t) - T_{out}(t))$$

- $\hat{H}_r(t+k)$: In the same way that $\hat{T}_{out}(t+k)$ is calculated, an adjustment is performed to calculate the predicted relative humidity.

$$\hat{H}_r(t+k) = H_r^{MS}(t+k) + (H_r^{MS}(t) - H_r(t))$$

- $\hat{P}(t+k)$: The radiation is predicted using a model with the real radiation values P in the last twelve hours -enough information to describe its behaviour- until t and the sky state prediction sky^{MS} at $t+k$. The sky state returns a categorical value that will be converted from 0 -sunny- to 1 -completely cloudy-. At night, it is set to 1. This model was learned with Random Forest, as it contains both numerical and categorical variables.

$$\hat{P}(t+k) = f(P(t-12), \ldots, P(t), sky^{MS}(t+k))$$

- $\%r(t+k)$: This variable represents the boiler operating percentage in a time interval. It is the system control variable, since the boiler operation can be adjusted in order to satisfy the comfort temperature.

We constructed a rule-based regression model F with FRULER to predict each variable response $\hat{T}_{in}^n(t+k), n \in [0,5]$ for different values of k (hours ahead in time), where $\hat{T}_{in}^n$ is the predicted indoor temperature on floor n at instant $t+k$. As k might be large (up to 96 h), those variables that have to be known in a future time where predicted at 1 h intervals and averaged in different time windows. Thus instead of using as features $\hat{T}_{out}(t+k)$, $\hat{H}_r(t+k)$, $\hat{P}(t+k)$, and $\%r(t+k)$, we defined variables $\{\bar{T}_{out}^s, \bar{H}_r^s, \bar{P}^s, \%r^s\}$. Algorithm 1 shows how these features are calculated. $\bar{X}^s$ is any of variables in $\{\bar{T}_{out}^s, \bar{H}_r^s, \bar{P}^s, \%r^s\}$.

Algorithm 1 Definition of the predicted features given a future time k.

if $k < 4$ **then**
  $\alpha = 1;\ \beta = k$
else
  $\alpha = k/4;\ \beta = 4$
end if
$s = \{0, \ldots, \beta - 1\}$
$\bar{X}^s = \frac{1}{\alpha}\sum_{i=1}^{\alpha} \hat{X}(t + s \cdot \alpha + i)$

© Ediciones Universidad de Salamanca

In order to train the models, several values of k could be set. In this case, $k = \{1, 2, 4, 8, 16, 24\}$h are proposed. To calculate the indoor temperature for another k, a combination of the previous models can be carried out. Then, the predicted indoor temperature is:

$$\hat{T}_{in}^{n}(t+k) = F[T_{in}^{n}(t), T_{out}(t), \bar{T}_{out}^{0}, \cdots, \bar{T}_{out}^{\beta-1}, \bar{H}_{r}^{0}, \cdots, \bar{H}_{r}^{\beta-1}, \bar{P}^{0}, \cdots, \bar{P}^{\beta-1},$$
$$\%r^{0}, \cdots, \%r^{\beta-1}]$$

At $k = 1$ and $k = 2$ we use 6 and 10 predictor variables respectively. For $k >= 4$, the total predictor variables remains equal to 18. These variables are represented in Figure 3.

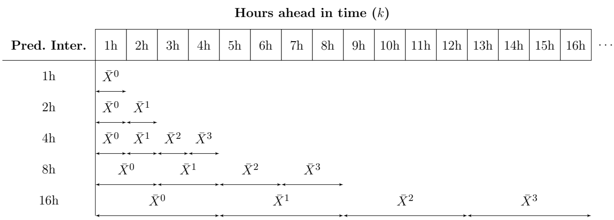


Fig. 3: Example of a predicted feature for different future times.

4 Experiments and results

4.1 Experimental setup

FRULER was designed to keep the number of parameters as low as possible. For the instance selection technique, no parameters are needed. In the multi-granularity fuzzy discretization, the fuzziness parameter used for the generation of the fuzzy intervals from the split points was 1, i.e., the highest fuzziness value. For the evolutionary algorithm, the values of the parameters were: population size = 61, maximum number of evaluations = 100,000, $p_{cross} = 1.0$, $p_{mut} = 0.2$, and $n_{ls} = 5$. For the generation of the TSK fuzzy rule bases, the weight of the tradeoff between ℓ_1 and ℓ_2 regularizations on the Elastic Net is $\alpha = 0.95$, and the regularization parameter λ was obtained from a grid search in the interval $[1, 1E - 10]$. η^0 was obtained halving the initial value (0.1) until the result worsens.

We present the results of the second floor (P2) with a a 5-fold cross-validation. Moreover, 6 trials (with different seeds for the random number generation) of

FRULER were executed for each 5-fold cross validation. Thus, a total of 30 runs were obtained for prediction hour in this floor. For the experiments in the remaining floors we just performed 3 trials without cross-validation.

The results shown in the next section are the mean values over all the runs. Data was recorded from 27-02-2016 to 14-06-2016 (2,483 h). Note that variable H_r^{MS} was not recorded until 23-07-2016 and consequently, H_r is used instead of $\hat{H_r}$. Nevertheless, it may be used in future as a predictor variable.

4.2 Results

In order to evaluate the performance of FRULER, we did a comparison with *ElasticNet* and *Random Forest Regressor*, both implemented in the *scikit-learn* package [8]. Table 1a shows the average test error in °C of the three approaches for the indoor temperature prediction on the second floor (P2) at several prediction intervals. For each algorithm and interval, the table displays the test error measured in °C. This indicator allows to compare the accuracy of the algorithms. The values with the best accuracy —lowest error— in Table 1a are marked in bold.

Pred. Interval	FRULER	ElasticNet	Random Forest
1h	**0.129**	0.177	0.197
2h	**0.222**	0.317	0.346
4h	**0.329**	0.464	0.479
8h	**0.434**	0.640	0.561
16h	**0.532**	0.824	0.662
24h	**0.558**	0.872	0.660

(a) Average test error in °C for the compared algorithms.

Algorithm	Ranking
FRULER	3.50
Random Forest	11.00
ElasticNet	14.00
p-value	0.012

(b) Aligned Friedman Test.

Table 1: Comparison results of the three algorithms for the indoor temperature prediction on the second floor (P2) at several prediction intervals.

FRULER gets the best accuracy for all the experiments. In order to check whether there are significant differences among the algorithms, we applied the Aligned Friedman statistical test, that computes the ranking of the results of the algorithms. The application of the test, using the STAC platform [11], rejects the null hypothesis, which states that the results of all the algorithms are equivalent with a given confidence -significance level ($\alpha = 0.05$)-. Table 1b shows the ranking for the test error and the p-value of the test, which indicates that the differences among the algorithms are statistically significant and that FRULER ranks first.

In Table 2a, the average test error in °C for the indoor temperature prediction on the second floor (P2) is displayed for several prediction intervals. Note that

© Ediciones Universidad de Salamanca

for the prediction intervals $\in$ {1h, 2h, 4h, 8h, 16h, 24h}, the learned models are applied whereas for the remaining prediction intervals -they have been chosen arbitrarily-, a concatenation of the previous models is performed. This technique lets us to predict the indoor temperature for any prediction interval from 1h to 96h.

As depicted in Table 2a, the test error is higher for the larger prediction intervals. The results are what could be expected, i.e., it is more accurate to predict the indoor temperature for the next hour rather than four days ahead.

1h	2h	4h	6h
0.129	0.222	0.329	0.335
8h	12h	16h	20h
0.434	0.504	0.532	0.617
24h	48h	72h	96h
0.558	0.845	0.954	0.279

(a) Average test error in °C on the second floor (P2).

Floor	1h	2h	4h	8h	16h	24h
P0	0.123	0.216	0.378	0.462	0.534	0.493
P1	0.109	0.207	0.336	0.492	0.473	0.518
P3	0.219	0.332	0.335	0.549	0.703	0.613
P4	0.102	0.191	0.311	0.386	0.409	0.467
P5	0.154	0.102	0.204	0.220	0.283	0.351

(b) Average test error in °C for the remaining floors at several prediction intervals.

Table 2: Test error on the second floor by concatenating the learned models (a) and test error for the remaining floors (b).

Finally, Table 2b presents the average test error in °C for the remaining floors at several prediction intervals. As we concluded before, the test error tends to increase as the prediction interval does.

5 Conclusions

In this paper we presented a model for indoor temperature prediction using the FRULER Genetic Fuzzy System to generate the knowledge base, made up of TSK fuzzy rules. The model has been learned from data recorded at Monte da Condesa Residential College during 2,483 hours and from several sensors. The model can predict the future indoor temperature for each floor of the building with an average error in the range 0.10-0.22 °C at t+1 and in the range 0.35-0.61 °C at t+24. The learned model will be used in the near future in the LIFE-Opere EU project [2] for planning efficient heating control strategies, in order to guarantee that the global power consumption of the heating system is reduced without sacrificing thermal comfort.

© Ediciones Universidad de Salamanca

Acknowledgement

This research was supported by the European Union LIFE programme (grant LIFE12 ENV/ES/001173), the Spanish Ministry of Economy and Competitiveness (grant TIN2014-56633-C3-1-R) and the Galician Ministry of Education (grants CN2012/151 and GRC2014/030). All grants were co-funded by the European Regional Development Fund (FEDER program).

References

1. Prediction of building's temperature using neural networks models. *Energy and Buildings*, 38(6):682–694, 2006.
2. Life-OPERE web page, http://www.life-opere.org/, Last visited May 30th 2016.
3. Aurélie Foucquier, Sylvain Robert, Frédéric Suard, Louis Stéphan, and Arnaud Jay. State of the art in building modelling and energy performances prediction: A review. *Renewable and Sustainable Energy Reviews*, 23:272–288, 2013.
4. Elena Marchiori. Class conditional nearest neighbor for large margin instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(2):364–370, 2010.
5. A. Mechaqrane and M. Zouak. A comparison of linear and neural network ARX models applied to a prediction of the indoor temperature of a building. *Neural Computing and Applications*, 13(1):32–37, 2004.
6. Meteogalicia. Galician meteorological web page, http://www.meteogalicia.es/, Last visited May 30th 2016.
7. Meteogalicia - MeteoSIX. Galician numerical weather prediction service, http://servizos.meteogalicia.gal/api_manual/gl/, Last visited May 30th 2016.
8. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
9. I. Rodríguez-Fdez, M. Mucientes, and A. Bugarín. An instance selection algorithm for regression and its application in variance reduction. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8, 2013.
10. I. Rodríguez-Fdez, M. Mucientes, and A. Bugarín. FRULER: Fuzzy rule learning through evolution for regression. *Information Sciences*, 354:1–18, 2016.
11. Ismael Rodríguez-Fdez, Adrián Canosa, Manuel Mucientes, and Alberto Bugarín. STAC: a web platform for the comparison of algorithms using statistical tests. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8, 2015.
12. Tao Lu and M Viljanen. Prediction of indoor temperature and relative humidity using neural network models: model comparison. *Neural Computing Applications*, 18(4):345–57, 2009.
13. Catalin Teodosiu, Raluca Hohota, Gilles Rusaouën, and Monika Woloszyn. Numerical prediction of indoor air humidity and its effect on indoor environment. *Building and Environment*, 38(5):655–664, 2003.
14. Bertil Thomas and Mohsen Soleimani-Mohseni. Artificial neural network models for indoor temperature prediction: investigations in two buildings. *Neural Computing and Applications*, 16(1):81–89, 2006.
15. Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society*, 67(2):301–320, 2005.

© Ediciones Universidad de Salamanca