Thermal Prediction for Immersion Cooling Data Centers Based on Recurrent Neural Networks

Jaime Pérez[1(✉)], Sergio Pérez[1], José M. Moya[1,2], and Patricia Arroba[1,2]

[1] Integrated Systems Laboratory, Universidad Politécnica de Madrid, ETSI Telecomunicación, Avenida Complutense 30, Madrid 28040, Spain
jaime.perez.sanchez@alumnos.upm.es

[2] Center for Computational Simulation, Universidad Politécnica de Madrid, Campus de Montegancedo, Boadilla del Monte, Madrid 28660, Spain

Abstract. In the data center's scope, current cooling techniques are not very efficient both in terms of energy, consuming up to 40% of the total energy requirements, and in terms of occupied area. This is a critical problem for the development of new smart cities, which require the proliferation of numerous data centers in urban areas, to reduce latency and bandwidth of processing data analytics applications in real time. In this work, we propose a new disruptive solution developed to address this problem, submerging the computing infrastructure in a tank full of a dielectric liquid based on hydro-fluoro-ethers (HFE). Thus, we obtain a passive two phase-cooling system, achieving zero-energy cooling and reducing its area. However, to ensure the maximum heat transfer capacity of the HFE, it is necessary to ensure specific thermal conditions. Making a predictive model is crucial for any system that needs to work around the point of maximum efficiency. Therefore, this research focuses on the implementation of a predictive thermal model, accurate enough to keep the temperature of the cooling system within the maximum efficiency region, under real workload conditions. In this paper, we successfully obtained a predictive thermal model using a neural network architecture based on a Gated Recurrent Unit. This model makes accurate thermal predictions of a real system based on HFE immersion cooling, presenting an average error of 0.75 °C with a prediction window of 1 min.

Keywords: Predictive thermal modeling
Recurrent neural networks · Data center · Immersion cooling

1 Introduction

Due to the proliferation of IoT devices and smart cities, cloud computing will not be longer a valid alternative for managing the volume of data generated by

H. Yin et al. (Eds.): IDEAL 2018, LNCS 11314, pp. 491–498, 2018.
https://doi.org/10.1007/978-3-030-03493-1_51

these applications. Today there are around 23,000 million connected IoT devices, but it is estimated that by 2030 there will be more than 100,000 million [4]. The latency and the bandwidth are now critical metrics so, novel Edge data centers have arisen to bring the computing infrastructure close to the source of the data. Edge data centers, distributed in urban areas, are able to process the data, thus reducing the amount of information that reach the cloud. Traditionally, data centers' computing and cooling consumption represent about the 50% and the 40% of the energy budget [1]. Cooling infrastructures are mainly based on Computer Room Air Conditioner (CRAC) units that are highly inefficient, not only in terms of energy consumption, but also in terms of the area. As Edge data centers will be placed in urban locations, which also present power grid limitations, both the area and the power consumption are also a critical restriction.

In this work we propose an immersion cooling solution, compatible with the Edge requirements, that will help to enable future smart cities. Our approach present a passive two phase-cooling system, achieving zero-energy cooling and reducing its area when compared with traditional CRAC-based data rooms. However, the nature of the hydro-fluoro-ether (HFE) dielectric fluid used to submerge the servers has specific thermal conditions to ensure the maximum heat transfer capacity.

Therefore, this research aims at implementing an accurate predictive thermal model to provide temperature predictions well in advance to keep the temperature of the cooling system within the maximum efficiency region, under real workload conditions. Predictive models are key for any system that needs to work around the point of maximum efficiency, but making a model for complex systems that is accurate and fast can be a difficult challenge. Analytical models help us to represent a solution in a closed form. However, they require the classification of all the parameters that have an impact on the system's performance, thus understanding the complex non-linear relationships between them, which can be a very tedious task in complex systems. On the other hand, Recurrent Artificial Neural Networks, as higher level metaheuristics, have been satisfactorily applied for modeling time series [3].

The **key contribution** of our work is to provide a predictive thermal model using a recurrent neural network architecture, for a disruptive HFE-based immersion cooling solution for Edge data centers. The remainder of this paper is organized as follows: Sect. 2 gives further information on the related work on this topic. Section 3 explains the theory on artificial neural networks for thermal modeling. Our problem description is provided in Sect. 4. Section 5 describes profusely the experimental results. Finally, in Sect. 6 the main conclusions are drawn.

2 Related Work

In this section, we analyze different state-of-the-art approaches for the implementation and use of thermal models. We will also study recent research on modeling time series that will help us to find appropriate techniques for designing predictive models. Traditionally, modeling the temperature in a data center helps to improve the efficiency of the cooling subsystem by creating strategies that distribute the workload along the computing infrastructure.

Moore et al. [6], use simple heuristics to model the behavior of a data center. They study the temperature variation and tested different algorithms and experiments to gauge the inefficiencies of the system. Their algorithm was able to nearly halve cooling costs when compared to other approaches. Xu et al. [8] present an empirical cooling optimization system an m-block alternating direction method of multipliers (ADMM) algorithm. According to their study, by modeling the temperature and distributing the loads with this algorithm, they provide savings between 15% and 20% for the cooling energy and between 5% and 20% of the overall energy cost. Tang et al. [7] develop a thermal model to minimize temperature peaks in the servers' inlet. They propose a thermal-aware load placement strategy based on the estimations of a heat recirculation model. Their approach offers 30% energy savings for a small simulated data center when compared to previous models.

2.1 Recurrent Artificial Neural Networks for Modeling Time Series

According to Connor et al. [3] Recurrent Neural Networks (RNN) respond very satisfactorily to problems of temporal prediction. This is because, in the RNN, the connections between nodes form a graph directed along a sequence, which allows them to also learn dynamic temporal behavior. And unlike the rest of neural networks, some can use an internal state or memory to process the input sequences. Zaytar et al. [9] present a meteorological prediction model using recurrent neural networks of LSTM type. Using this modeling technique, they are able to predict temperature, humidity and wind speed with errors around 2%. Che et al. [2] propose a multivariable temporal prediction using recurrent neural networks of the GRU type with an average error of around 0.7 and a standard deviation of 0.02. This strategy is especially interesting when working with signals that are noisy and unstable. Research proposed by Kermanshahi [5] predicts the electric power demand of nine industrial facilities using RNN with errors between 0.53% and 2.76%.

We can conclude that modeling the temperature to vary the dynamic load is effective to improve the energy efficiency in conventional cooling systems in data centers. However, previous research works model the temperature in data centers with air-ventilated cooling infrastructures. In this paper we propose a thermal model for a two phase-immersion cooling data center based on an HFE dielectric fluid. For this purpose we use Recurrent Artificial Neural Networks, as they have been satisfactory applied for modeling time series in other research fields with high accuracy.

3 Recurrent Neural Networks for Thermal Modeling

Based on the state-of-the-art presented in Sect. 2, we decided to model our thermal predictions using Artificial Neural Networks (ANNs). These algorithms are named after the neural networks of animal's nervous systems because they try to emulate their behavior. ANNs are able to automatically learn complex patterns, correlations and behaviors, analyzing large amounts of information.

The ANNs form a system of links that interconnects the different neurons, which makes them collaborate with each other to produce exit stimuli. Each link has a numerical weight, which is recalculated during the network training to be adapted to the data input, thus being able to "learn". Thanks to the non-linear activation functions inside the neurons, the networks are able to describe any behavior of the real world, which is precisely non-linear. Some typical examples are the sigmoidal function, the normalized exponential (SoftMax), the hyperbolic tangent (tanh) or the rectified linear (ReLu) among others.

As our models need to predict the temperature over time, we need neural networks with memory, which are called Recurrent Neural Networks (RNNs). RNNs are able to work with temporary data sequences because their neurons have an internal state (memory) to process the input sequences. In this work we use GRU architectures as they have been able to reduce gradient problems and also are better suited for small datasets like the one used in this research (our dataset present five features and 2400 time steps). GRU cells, as shown in Fig. 1, have two gates called *update gate* (z_t) and *reset gate* (r_t). z_t determines the amount of information from the previous state that will affect the current state and the r_t specifies how much information from the past state is forgotten. In our thermal modeling, x_t is the current temperature, while h_{t-1} is the information about previous temperatures, h_t is the future temperature prediction passed to following cells and $\tilde{h}_t$ is the current memory content. W_z, W_r and W are the weights of the parameter matrices respectively as in Eqs. 1–4:

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \tag{1}$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{2}$$

$$\tilde{h}_t = tanh(W \cdot [r_t * h_{t-1}, x_t]) \tag{3}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \tag{4}$$

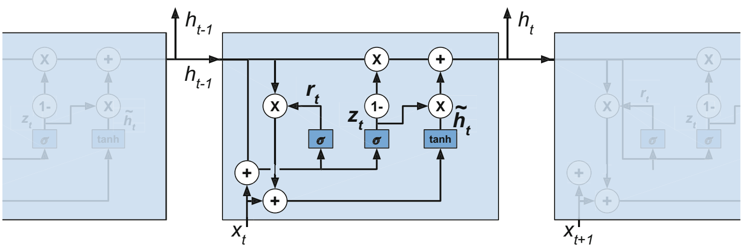

Fig. 1. Scheme of GRU neuron

4 Problem Description

Our prototype, shown in Fig. 2 is a system cooled by passive immersion that consists of a small container filled with dielectric liquid based on Hydro-Fluoro-Ethers, in which we submerge a Raspberry Pi 3 Model B+ cluster. This cluster runs a workload based on data analytics, in particular, a prediction of environmental pollution in the city of Beijing, China[1]. This has been chosen to provide a real Edge data center environment. Using a Graphite-based monitoring system[2] we have compiled a dataset, which includes the temperature ($T_{CPU,x}$), the utilization ($U_{CPU,x}$) and the working frequency ($f_{CPU,x}$) of each Raspberry Pi's CPU (x), collecting data every ten seconds. After that, we use our dataset to design and implement the predictive thermal model.

Fig. 2. HFE-based immersion cooled prototype.

To set the value of the RNN hyperparameters, which are all the tunable settings that the network allows to change, we have first defined a basic model using state-of-the-art examples. We then conduct experiments to set the optimizer (among RMSprop, Adam, Adamax and Nadam) and the best loss function (among MSE, MAE and Binary Cross Entropy) that offer the best results. We set these hyperparameters first as they are considered to be the most independent from the rest. Afterwards, we proceed to make a specific model using the following strategy. First we optimize the neural structure (number of neurons and layers) leaving the rest of the hyperparameters fixed. We perform experiments from smaller structures (1 layer with 2 neurons) to more complex structures (3 layers with 32, 16, and 8 neurons respectively). Finally, we optimize the

[1] archive.ics.uci.edu/ml/datasets/Beijing+PM2.5+Data.
[2] graphiteapp.org.

hyperparameters involved in model training. We tune the batch size (from 15 to 125) and the number of epochs (from 10 to 250) in steps of 10. Also, we optimize the learning rate (from 0.001 to 0.01), the activation function (Softmax, Relu, Elu, tanh) and the learning decay (from 0.001 to 0.01).

In our problem, the most important hyperparameter is the prediction window. A bigger window means longer reaction time to optimize our system but it also means worse predictions and longer training times. We decide to set it in one minute, because we consider that it gives the system enough time to perform direct or indirect optimization actions. To examine the performance of our prediction approach, we include results with three widely used prediction error metrics: Mean Absolute Error (MAE), Root Mean Square Deviation (RMSD) and Coefficient of determination (R^2), which can be seen in Eqs. 5 to 7 respectively, where y_n is the real measurement, $\overline{y}$ its average value, x_n is the prediction and N is the number of traces in our dataset.

$$MAE = \frac{1}{N} \sum_{n} |y_n - x_n|, \qquad 1 \leq n \leq N \tag{5}$$

$$RMSD = \sqrt{\frac{1}{N} \cdot \sum_{n} (y_n - x_n)^2} \tag{6}$$

$$R^2 = 1 - \frac{\sum_n (y_n - x_n)^2}{\sum_n (y_n - \overline{y})^2} \tag{7}$$

5 Performance Evaluation

Our dataset (explained in Sect. 4) has been split into a training and a testing set. The training stage builds the thermal model according to the hyperparameter's settings. Then, the testing stage checks the model accuracy for a workload never seen by the modeling process, which consists of the 12% of our traces. Using the methodology proposed in Sect. 4 for the optimization of the hyperparameters, we obtain an optimized neural network for the GRU recurrent unit proposed in this research. Also, we apply the same methodology using other types of recurrent unit (Fully connected and LSTM) in order to compare the prediction accuracy. Table 1 presents the characteristics of each type of RNN and the prediction accuracy results in the testing dataset. For each recurrent unit type we provide results for the best solution found for 3, 2 and 1 layers respectively, when optimizing the neural structure. Additionally, we also present the best solution found when optimizing the hyperparameters involved in model training, following our modeling methodology. Our experiments are configured using the Nadam optimizer, the tanh activation function and the MAE loss function.

Based on the results it can be observed that our GRU model has an edge over LSTM and fully connected structures. It provides a MAE±STD error of 0.75±0.59 °C when evaluating the entire testing dataset, similar to the one provided by LSTM. Our GRU model also presents an RMSD of 0.957 °C and an R^2 value of 89.539% that defines how well the observed outcomes are replicated by

Table 1. Comparison of testing prediction accuracy

Recurrent unit	Neural structure	Batch size	Epochs	MAE ± STD (°C)	RMSD (°C)	R^2 (%)
Fully Connected	(16, 32, 16)	65	80	1.11 ± 1.14	1.588	71.925
	(16, 8)	65	80	1.13 ± 1.08	1.562	72.824
	(256)	65	80	1.09 ± 0.83	1.371	79.071
	(256)	95	140	0.84 ± 0.84	1.187	83.897
LSTM	(8, 4, 2)	65	80	0.95 ± 1.13	1.475	75.782
	(1, 1)	65	80	1.02 ± 0.82	1.311	80.871
	(2)	65	80	0.91 ± 0.89	1.270	82.044
	(2)	65	210	0.71 ± 0.66	0.969	87.527
GRU	(4, 2, 1)	65	80	0.81 ± 0.83	1.157	82.211
	(4, 2)	65	80	0.85 ± 0.78	1.150	82.429
	(4)	65	80	0.74 ± 0.68	1.002	86.668
	(4)	**95**	**120**	**0.75 ± 0.59**	**0.957**	**89.539**

the prediction model, outperforming the results provided by the other recurrent units. Figure 3 shows the thermal fitting of our GRU model with a prediction window of 1 min and it can be seen that it offers a good fitting to the real measurement's curve on scenarios with workloads that vary significantly during runtime.

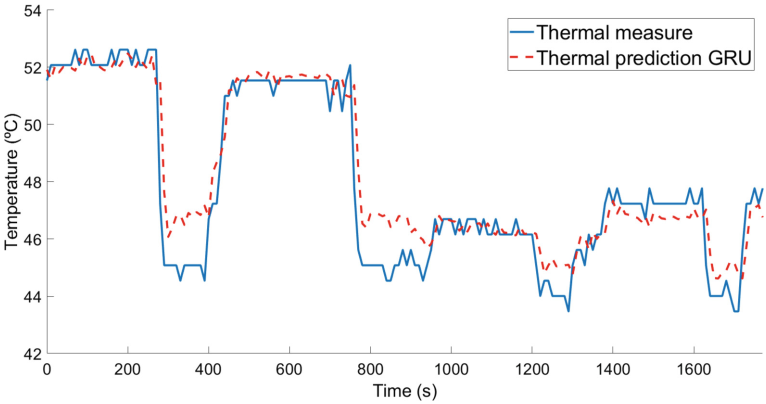

Fig. 3. Testing fitting for our RNN-based thermal predictive model.

6 Conclusions

In this research, we successfully provide a predictive model based on a recurrent artificial neural network architecture, with GRU type neurons. Our model makes accurate thermal predictions in a real system based on immersion cooling, where the computing infrastructure is submerged in a tank full of HFE-based fluid running real data analytics applications. The proposed model predicts the temperature of the computing system one minute in advance, presenting an average error of 0.75 °C, an RMSD of 0.957 °C and an R^2 value of 89.539% when compared to the real measurements provided by our monitoring system. By using this model, predictions can be made well in advance to take proactive decisions, both direct and indirect, on the system's temperature. This enables the development of novel proactive optimization strategies that provide a cooling setpoint temperature within a specific range, thus ensuring the maximum heat transfer capacity of the HFE.

Acknowledgment. This project has been partially supported by 3M, the Centre for the Development of Industrial Technology (CDTI) under contract IDI-20171194, and the Spanish Ministry of Economy and Competitiveness, under contracts TIN-2015-65277-R and AYA2015-65973-C3-3-R.

References

1. Breen, T.J., Walsh, E.J., Punch, J., Shah, A.J., Bash, C.E.: From chip to cooling tower data center modeling: part I influence of server inlet temperature and temperature rise across cabinet. In: 2010 12th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems, pp. 1–10 (June 2010)
2. Che, Z., Purushotham, S., Cho, K., Sontag, D., Liu, Y.: Recurrent neural networks for multivariate time series with missing values. CoRR abs/1606.01865 (2016)
3. Connor, J.T., Martin, R.D., Atlas, L.E.: Recurrent neural networks and robust time series prediction. IEEE Trans. Neural Netw. **5**(2), 240–254 (1994)
4. Howell, J.: Number of connected IoT devices will surge to 125 billion by 2030. IHS Markit Press Release (2017)
5. Kermanshahi, B.: Recurrent neural network for forecasting next 10 years loads of nine Japanese utilities. Neurocomputing **23**(1), 125–133 (1998)
6. Moore, J., Chase, J., Ranganathan, P., Sharma, R.: Making scheduling "cool": temperature-aware workload placement in data centers. In: Proceedings of the Annual Conference on USENIX Annual Technical Conference, ATEC 2005, p. 5. USENIX Association, Berkeley (2005)
7. Tang, Q., Gupta, S.K.S., Varsamopoulos, G.: Energy-efficient thermal-aware task scheduling for homogeneous high-performance computing data centers: a cyber-physical approach. IEEE Trans. Parallel Distrib. Syst. **19**(11), 1458–1472 (2008)
8. Xu, H., Feng, C., Li, B.: Temperature aware workload managementin geo-distributed data centers. IEEE Trans. Parallel Distrib. Syst. **26**(6), 1743–1753 (2015)
9. Zaytar, M.A., Amrani, C.E.: Sequence to sequence weather forecasting with long short-term memory recurrent neural networks. Int. J. Comput. Appl. **143**(11), 7–11 (2016)