On developing a fast, cost-effective and non-invasive method to derive data center thermal maps

Michael Jonas, Georgios Varsamopoulos, Sandeep K. S. Gupta
IMPACT Laboratory, School of Computing & Informatics
Arizona State University
Tempe, AZ 85287, USA
http://impact.asu.edu/

I. Introduction

Ongoing research has demonstrated the potential benefits of thermal-aware load placement in data centers to both reduce cooling costs and component failure rates. However, thermal-aware load placement techniques do not witness a wide deployment in existing data centers, mainly because they rely on a thermal map or profile of the data center, the derivation of which is an interruptive process to the data center operation. We propose a noninvasive solution of producing a thermal map; it consists of training a neural network with observed data from actual data center operation. Our results show that gathering the data and selecting a training set is a fast process, while the neural network with no hidden layers achieves the lowest mean squared error.

II. Motivation & Objectives

Lowering the total costs of ownership of data centers in various ways has emerged as a leading research topic over the past several years. One of the primary factors within these studies has been to address the cooling infrastructure required to maintain an acceptable operating environment. Previous research in this field has come up with several very effective thermal-aware task scheduling algorithms to help reduce the cooling costs [1], [2], [3], [4], [5], [6], [7]. Most of the results show a reduction in the range of a 25-50% in cooling costs for a data center in typical operation. Nevertheless, most data centers throughout the world do not take advantage of these results despite the real cost savings that they provide.

One of the reasons of low application of thermal-aware solutions is the high difficulty in implementing these approaches, specifically, on predicting the *thermal map* of the data center under any configuration. Conceptually, a thermal map depicts the temperature at any physical point in the data center. Most of these approaches rely on predicting the thermal map under any scenario; however, current methods of documenting thermal maps require either exclusive access and extensive sensing equipment to test and document the data center's thermal map under all utilization loads, or detailed physical measurements and CFD simulation, which exhibits extensive runtime that can push deployment times for a solution into the time frame of months. Further exacerbating this problem is that the process needs to be repeated every time a change is made to the data center for the model to stay accurate.

Considering the practical need of data centers to frequently adjust their hardware and the shortcomings of the previous research, we are looking into developing a fast, cost-effective and non-invasive technique to derive a thermal map. Our proposed approach consists of combining the following ideas: a) reducing the concept of thermal map to the air inlet points of each computer equipment, b) using only monitored data from on-board and built-in sensors to train a neural network; this neural network takes CPU utilization rates as the input and outputs (predicts) temperatures at the equipment inlets.

III. Methodology & Preliminary Results

Training the neural network requires gathering preparing the monitored data (utilization and temperatures) as training samples. The process of this approach is shown in Figure 1. The first step in the creation of the neural network is to gather data to train with. The gathered data may come from various sources, so the data "streams" must be synchronized before they can be fused together. An example of the resulting stream of utilization and temperature can be seen in Figure 2 from data gathered from the ASU Fulton HPC Center.

After a unified, synchronized data stream has been established, a set of samples must be selected from the gathered data to train the neural network. While we are looking into creating an automated selection method, in the example data we kept all data instances where all the servers maintained a near constant utilization for a period of 16 minutes.

Finally, once a sample set has been chosen, it must be formatted to train the neural network. For instance the data must be broken into input (utilization) and output (temperature) tuples and written to a file. Another concern in this step is normalizing the data to the (0,1) scale. In the example stream, the (0, 1) interval linearly corresponds to the (5, 40) °C range.

The training process typically runs in just a few minutes. The example neural network was configured with two hidden layers and had a single input for each chassis' utilization and a single output for each chassis' inlet temperature. Table I shows the results of the neural network for several cases trained on the same data stream (negative values in the output mean predicted temperatures below 5 °C).

While our current results can already be found useful for determining trends, they do not show realistic results for all possible utilization patterns. However, we believe that

TABLE I

SAMPLE PREDICTION RUNS OF THE NEURAL NETWORK (2 LAYERS, STEEPNESS 0.5)

	all servers idle		server 11 max, all others idle			server 13 max, all others idle			server 14 max, all others idle			server 19 max, all others idle		
chassis	input	output	input	output	baseline difference	input	output	baseline difference	input	output	baseline difference	input	output	baseline difference
1	0	0.2148	0	0.1899	-0.0250	0	0.2425	0.0277	0	0.2037	-0.0112	0	0.1997	-0.0151
2	0	0.1965	0	0.1854	-0.0111	0	0.2272	0.0307	0	0.1726	-0.0239	0	0.1838	-0.0126
3	0	0.1231	0	0.1165	-0.0066	0	0.1410	0.0179	0	0.1109	-0.0122	0	0.1034	-0.0197
4	0	0.1265	0	0.1183	-0.0082	0	0.1496	0.0230	0	0.1119	-0.0147	0	0.1036	-0.0229
5	0	0.1420	0	0.1370	-0.0050	0	0.1525	0.0104	0	0.1232	-0.0188	0	0.1264	-0.0156
6	0	0.2491	0	0.2154	-0.0337	0	0.2789	0.0298	0	0.2369	-0.0122	0	0.2429	-0.0062
7	0	0.2172	0	0.1951	-0.0221	0	0.2439	0.0268	0	0.2022	-0.0150	0	0.1950	-0.0222
8	0	0.1237	0	0.1189	-0.0048	0	0.1462	0.0225	0	0.1053	-0.0184	0	0.1140	-0.0097
9	0	0.1728	0	0.1628	-0.0100	0	0.1946	0.0218	0	0.1451	-0.0277	0	0.1585	-0.0144
10	0	0.2849	0	0.2331	-0.0518	0	0.3314	0.0466	0	0.2824	-0.0025	0	0.2452	-0.0397
11	0	0.2505	1	0.2037	-0.0468	0	0.2850	0.0345	0	0.2392	-0.0113	0	0.2225	-0.0280
12	0	0.1494	0	0.1158	-0.0336	0	0.1675	0.0180	0	0.1339	-0.0156	0	0.1323	-0.0171
13	0	0.2095	0	0.1891	-0.0204	1	0.2231	0.0136	0	0.1942	-0.0153	0	0.2058	-0.0037
14	0	0.3957	0	0.3598	-0.0359	0	0.4709	0.0752	1	0.3221	-0.0736	0	0.4151	0.0194
15	0	0.1880	0	0.1677	-0.0203	0	0.2083	0.0203	0	0.1663	-0.0217	0	0.1935	0.0055
16	0	0.1506	0	0.1393	-0.0113	0	0.1627	0.0122	0	0.1378	-0.0127	0	0.1437	-0.0069
17	0	0.1474	0	0.1333	-0.0141	0	0.1670	0.0195	0	0.1290	-0.0184	0	0.1300	-0.0174
18	0	0.1870	0	0.1684	-0.0186	0	0.2021	0.0151	0	0.1717	-0.0154	0	0.1728	-0.0142
19	0	0.3658	0	0.3137	-0.0520	0	0.4050	0.0393	0	0.3453	-0.0205	1	0.3631	-0.0027
20	0	0.1826	0	0.1579	-0.0246	0	0.2111	0.0285	0	0.1641	-0.0185	0	0.1671	-0.0155
21	0	0.2001	0	0.1871	-0.0130	0	0.2331	0.0330	0	0.1801	-0.0199	0	0.1817	-0.0183
22	0	0.1480	0	0.1471	-0.0009	0	0.1766	0.0285	0	0.1269	-0.0212	0	0.1355	-0.0126
23	0	0.1412	0	0.1406	-0.0006	0	0.1607	0.0195	0	0.1221	-0.0190	0	0.1330	-0.0082
24	0	0.2685	0	0.2531	-0.0154	0	0.2937	0.0252	0	0.2365	-0.0320	0	0.2598	-0.0087
25	0	0.2308	0	0.1936	-0.0372	0	0.2662	0.0354	0	0.2208	-0.0100	0	0.2147	-0.0160
26	0	0.1049	0	0.0974	-0.0075	0	0.1204	0.0155	0	0.0875	-0.0174	0	0.0932	-0.0117
27	0	0.3300	0	0.2714	-0.0586	0	0.3582	0.0282	0	0.3193	-0.0107	0	0.3140	-0.0160

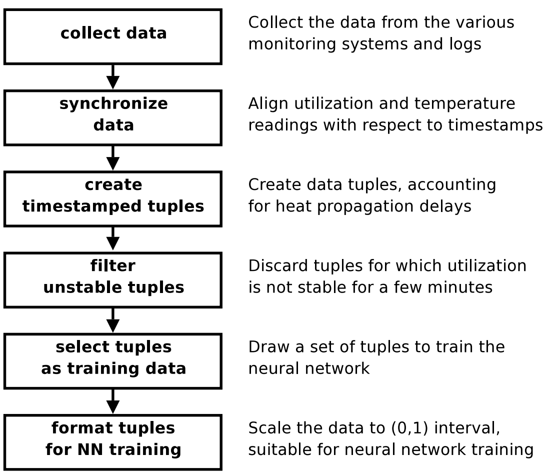

Fig. 1. Training data selection process

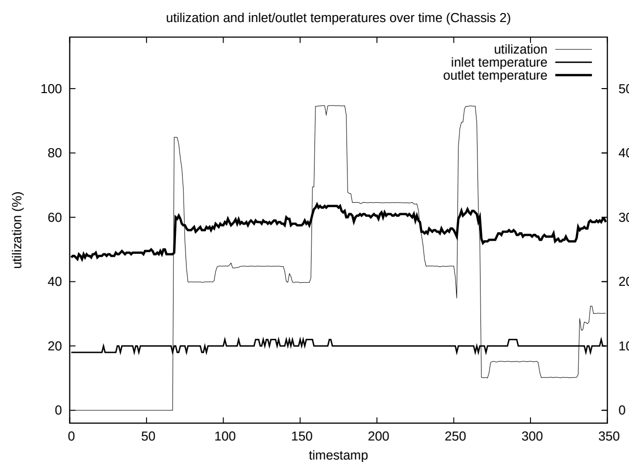

Fig. 2. Example utilization, inlet and outlet temperatures over time

by making just a few changes to our existing scheme, we can largely reduce the error in the predictions. In such a case, it would be possible to provide a reasonably accurate thermal map at a minimal cost. If this research work produces positive results, it will make energy-saving approaches based on thermal maps more attractive for application.

IV. ACKNOWLEDGMENTS

We would like to thank Dan Stanzione for granting access to the ASU Fulton HPC Facility. Parts of this work have been funded by SFAz, Intel Corporation and NSF (CNS#0649868).

REFERENCES

[1] J. Moore, J. Chase, and P. Ranganathan, "Weatherman: Automated, online, and predictive thermal mapping and management for data centers," in *IEEE International Conference on Autonomic Computing (ICAC)*, June 2006, pp. 155–164.

[2] J. Moore, J. Chase, P. Ranganathan, and R. Sharma, "Making scheduling 'cool': Temperature-aware workload placement in data centers," in *2005 USENIX Annual Technical Conference*, Apr. 2005, pp. 61–74.

[3] J. Moore, R. Sharma, R. Shih, J. Chase, C. Patel, and P. Ranganathan, "Going beyond cpus: The potential of temperature-aware solutions for the data center," in *First Workshop on Temperature-Aware Computer Systems (TACS-1)*, June 2004.

[4] Y. Chen, A. Das, W. Qin, A. Sivasubramaniam, Q. Wang, and N. Gautam, "Managing server energy and operational costs in hosting centers," in *SIGMETRICS '05: Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*. New York, NY, USA: ACM Press, 2005, pp. 303–314.

[5] T. Mukherjee, Q. Tang, and S. K. S. Gupta, "Software architecture for dynamic thermal management in datacenters," in *Int'l Conf. on Communication System Software and Middleware (COMSWARE)*, Jan 2007.

[6] Q. Tang, S. K. S. Gupta, D. Stanzione, and P. Cayton, "Thermal-aware task scheduling to minimize energy usage of blade server based datacenters," in *IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC'06)*, Oct 2006.

[7] Q. Tang, T. Mukherjee, S. K. S. Gupta, and P. Cayton, "Sensor-based fast thermal evaluation model for energy efcient high-performance datacenters," in *Int'l Conf. Intelligent Sensing & Info. Proc. (ICISIP2006)*, Dec 2006.