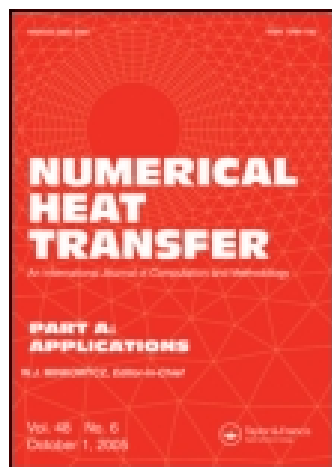


This article was downloaded by: [Georgia Tech Library]

On: 30 August 2015, At: 14:28

Publisher: Taylor & Francis

Informa Ltd Registered in England and Wales Registered Number: 1072954 Registered office: 5 Howick Place, London, SW1P 1WG



Numerical Heat Transfer, Part A: Applications: An International Journal of Computation and Methodology

Publication details, including instructions for authors and
subscription information:

<http://www.tandfonline.com/loi/unht20>

Rapid Temperature Predictions in Data Centers using Multi-Parameter Proper Orthogonal Decomposition

Rajat Ghosh ^a & Yogendra Joshi ^a

^a The George W. Woodruff School of Mechanical Engineering ,
Georgia Institute of Technology , Atlanta , Georgia , USA
Published online: 03 Apr 2014.



[Click for updates](#)

To cite this article: Rajat Ghosh & Yogendra Joshi (2014) Rapid Temperature Predictions in Data Centers using Multi-Parameter Proper Orthogonal Decomposition, Numerical Heat Transfer, Part A: Applications: An International Journal of Computation and Methodology, 66:1, 41-63, DOI: [10.1080/10407782.2013.869090](https://doi.org/10.1080/10407782.2013.869090)

To link to this article: <http://dx.doi.org/10.1080/10407782.2013.869090>

PLEASE SCROLL DOWN FOR ARTICLE

Taylor & Francis makes every effort to ensure the accuracy of all the information (the "Content") contained in the publications on our platform. However, Taylor & Francis, our agents, and our licensors make no representations or warranties whatsoever as to the accuracy, completeness, or suitability for any purpose of the Content. Any opinions and views expressed in this publication are the opinions and views of the authors, and are not the views of or endorsed by Taylor & Francis. The accuracy of the Content should not be relied upon and should be independently verified with primary sources of information. Taylor and Francis shall not be liable for any losses, actions, claims, proceedings, demands, costs, expenses, damages, and other liabilities whatsoever or howsoever caused arising directly or indirectly in connection with, in relation to or arising out of the use of the Content.

This article may be used for research, teaching, and private study purposes. Any substantial or systematic reproduction, redistribution, reselling, loan, sub-licensing, systematic supply, or distribution in any form to anyone is expressly forbidden. Terms &

RAPID TEMPERATURE PREDICTIONS IN DATA CENTERS USING MULTI-PARAMETER PROPER ORTHOGONAL DECOMPOSITION

Rajat Ghosh and Yogendra Joshi

The George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, Georgia, USA

A proper orthogonal decomposition (POD)-based multi-parameter, reduced-order modeling framework that rapidly predicts air temperatures in an air-cooled data center is developed. The modeling parameters are heat load and time. The framework is applied on initial temperature snapshots acquired near a server simulator rack by measurements at discrete time instants and selected rack heat loads. To estimate the accuracy of the modeling framework, the predicted temperature data are compared with corresponding experimental observations. The proposed algorithm is demonstrated to be effective and efficient for full-factorial parametric temperature characterization.

1. INTRODUCTION

With the proliferation of cloud-computing-based information technology (IT) and telecommunication services, internet data centers (DCs) have become an indispensable cyber-physical system of daily life in many areas including online merchandise, communications, financial services, satellite navigation, and entertainment. With DCs consuming as much as 2.2% of the total electricity produced in the United States in 2012, maintaining their sustainable growth is a critical engineering infrastructure design problem [1]. Benchmarking studies reveal that the DC cooling infrastructure consumes as much as 30%–50% of the total facility energy usage, and the life-cycle cost of cooling is becoming comparable to that of IT hardware for commodity computing [2]. DC energy efficiency has been traditionally improved by implementing accepted best practices [3], infrastructure monitoring and management capability [4].

Since DC energy usage has a strong correlation with DC air temperatures [5], a detailed temperature prediction framework is essential for DC energy usage monitoring and power management strategy. To create such a framework, we must

Received 10 September 2013; accepted 7 November 2013.

Address correspondence to Yogendra Joshi, 771 Ferst Drive, J. Erskine Love Building, Room 338, The George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0245, USA. E-mail: yogendra.joshi@me.gatech.edu

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/unht.

NOMENCLATURE

(x, y, z)	spatial co-ordinates, (m)	k_s	optimal number of principle components in the secondary procedure
t_{en}	ensemble time, (s)	\otimes	tensor product
Q_{en}	ensemble heat load, (W)	\exists	such that
t_{int}	interrogation time, (s)	\cup	union
Q_{int}	interrogation heat load, (W)	$\langle \rangle$	spatial average
\bar{t}	normalized time	λ	eigenvalue
\bar{Q}	normal heat load		
k_p	optimal number of principle components in the primary procedure		

characterize the transient convective air temperature in response to various dynamic events resulting from time-varying IT heat loads or fluctuating allocation of cooling resources [6]. Typically, the login rates escalate during early afternoon and go down after midnight. In addition, DCs observe flash crowds, during which large numbers of users login in a short period of time. This stochastic load-profile of DCs makes optimal cooling resource utilization particularly challenging. To implement an optimal resource utilization scheme, data center designers must avoid resource over or under-provisioning and must minimize systemic latencies to allocate cooling resources dynamically [7].

Current DCs are typically air-cooled using an array of computer room air conditioning (CRAC) units, which operate air-liquid heat exchangers and air handlers. The liquid side of the heat exchanger in a CRAC includes the evaporator of a vapor compression refrigeration cycle, integrated with the building chiller, the chilled water distribution pump, and the compressor. Typically, CRACs are return or supply air temperature controlled. Based on these temperature set-points, the micro-processor-based control hardware modulates cooling liquid flow rate, which is the major variable cooling cost in any DC. One strategy to reduce cooling cost is to raise the temperature set-points for CRACs. This requires a real-time thermal feedback control system that can rapidly assess cooling demand and allocate optimal coolant flow rate. A major obstacle in developing such a control system is the lack of a rapid high-resolution temperature characterization capability inside the DCs. The efficiency of the characterization framework must be near real time so that it can be used as a feedback signal generator in air-conditioning control units.

For DC air temperature characterization, computational fluid dynamics/heat transfer (CFD/HT)-based numerical simulations have been widely used by several researchers, including Kang et al. [8], Patel et al. [9], and Rambo and Joshi [10]. These simulations model DCs as turbulent forced-convective thermal systems, which have been used for the design optimization of DCs with respect to parameters such as plenum depth, facility ceiling height, cold aisle spacing, CRAC flow rates, rack flow rate, and power dissipation. However, these techniques are based on iterative solutions of discretized conservation equations [11]. The computational time for such schemes scales as mn^3 , where m is the number of time steps and n is the number of spatial grid points. This higher-order dependence makes such schemes impracticable for real-time DC temperature modeling because the grid-size (n) is often in the order

of several million. Furthermore, the stochastic nature of the DC's thermal boundary conditions makes the computational requirements even more challenging.

A potential alternative to CFD/HT simulation is measurement-based dynamic characterization of air temperatures via the deployment of sensor networks [12]. The possible temperature sensors could be thermocouples, RTDs, and thermistors. The time scale for typical air temperature change is in the order of a few seconds following a change in rack heat load. Therefore, a desirable sampling time interval for temperature measurement should be ~ 1 s. On the other hand, the average scale for the change in heat load for a typical IT rack is on the order of few hundred Watts. Therefore, a desirable sampling interval of heat load for rack inlet air temperature control should be ~ 100 W. Any dynamic characterization framework for such a high-resolution parametric space is currently cost-prohibitive because it requires expensive distributed sensor deployment. A potential strategy to circumvent this challenge involves developing a measurement-based temperature response prediction framework that takes experimental temperature data with coarse parametric resolution as input. Essentially, a response surface is developed and is best-fitted with the predictors or input measurement data. The dimension of the best-fit subspace is equal to the size of the input/predictor space. Prevalent techniques for computing the basis function for the best-fit subspace are Galerkin-POD [13, 14] or snapshot-POD [15]—the latter being often the preferred technique for experimental data analysis [16]. The POD is a spectral procedure for extracting orthogonal basis functions from an ensemble of signals, here DC air temperatures. The attractive feature of POD basis functions (POD modes) lies in their optimality in the sense that N POD modes convey more information about the response surface than any other basis functions generated by comparable decompositions such as fast Fourier transform (FFT). The fast energy convergence of POD decomposition makes it a preferred simulation technique for dynamic optimization problems [17, 18]. The computational time of POD is on the order of $\log(n)$, whereas that of FFT is on the order of $n\log(n)$, where n is the rank of the input space. POD optimally identifies the coherent structure of the response variable [14]. The POD output lies within the subspace of the coherent structures of the corresponding eigenspace. Such a physics-based feature improves the fidelity of the empirical model.

This article presents a measurement-based algorithm to predict DC air temperatures rapidly and accurately. The proposed model is particularly suitable for parametric optimization of cooling resources. Section 2 describes this two-parameter POD modeling algorithm with heat load and time as the parameters. Section 3 provides the experimental details for the air temperature measurements. Section 4 presents a case study to demonstrate the utility of the proposed modeling framework. Subsequent sections include pertinent results and discussion (section 5), and summary and conclusions (section 6).

2. METHODOLOGY

A data-driven algorithm is developed to obtain fine-grained air temperature predictions using heat load of the test rack and time as the parameters. Figure 1 shows the flowchart of the algorithm. Experimental temperature data at a few

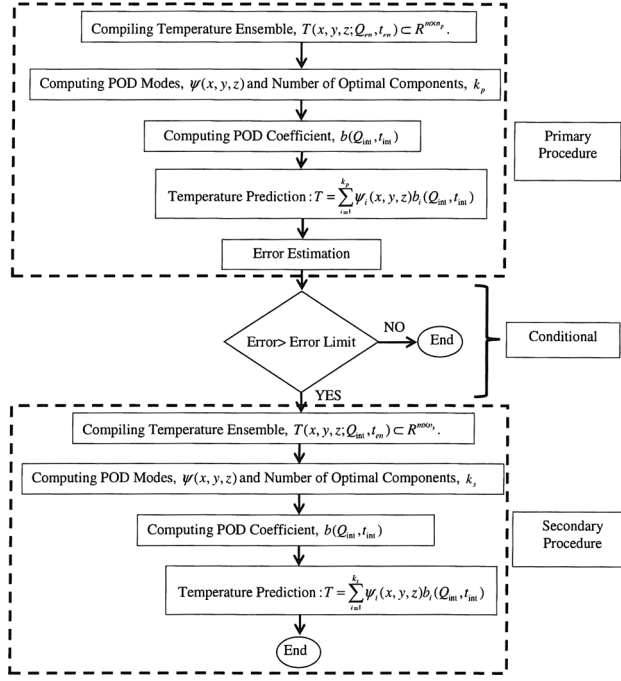


Figure 1. The POD-based temperature prediction algorithm is comprised of two numerical procedures, both based on the computation of optimal basis functions by POD. The primary procedure involves regression analyses both in heat load and time, whereas the secondary procedure involves regression analyses in time and is executed only when the condition block is satisfied.

selected heat loads and time instants constitute the model input space as a two-dimensional data matrix. The independent variable is the spatial locations (x, y, z) , which represents the row index of the data matrix. The row dimension of the data matrix is equal to the number of deployed sensors. Temperature data are collected at selected test rack heat loads, Q_{en} and time instants, t_{en} ; the data are stored in different columns of the data matrix, with time as the inner variable and heat load as the outer variable. Essentially, time index and heat load index are condensed into the column index, j :

$$j = i_t + (i_h - 1) N_t \quad (1)$$

Where i_t is the temporal index, i_h is the heat load index, and N_t is the number of time-varying inputs for a given heat load.

The number of columns represent the overall size of the parametric input space. The intrinsic assumption for the algorithm is that temperature predictions at the interrogation point (Q_{int}, t_{int}) lie in the column space for the data matrix; and hence, the predictions can be expressed as a linear combination of suitable basis functions. Using POD, the optimal basis functions (POD modes) for a given data matrix are determined. The optimality of the POD modes is quantified by the number of eigenvalues, which capture dominant components of the temperature

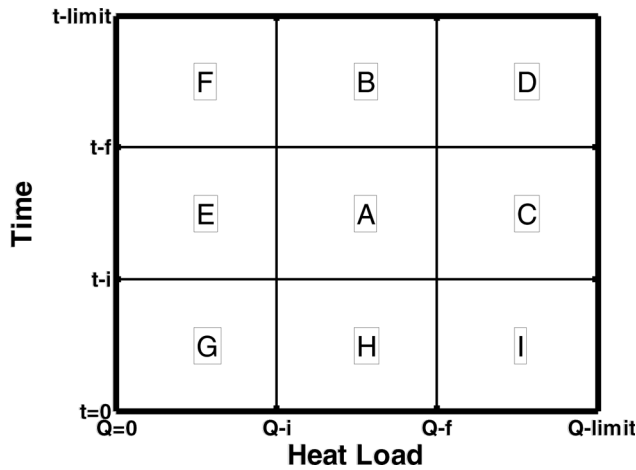


Figure 2. Interrogation space is comprised of all possible heat loads and time instants.

field. The corresponding weighting scalars (POD coefficients) for basis functions are parameter-dependent and are determined by a suitable regression operation. The location of an interrogation point with respect to (Q_{en}, t_{en}) determines the type of regression operation (interpolation or extrapolation) required. Figure 2 identifies different prediction zones in the parametric space. Table 1 specifies the regression operations needed to determine the POD coefficients at the interrogation point. POD is a model order reduction technique. The prediction vector at an interrogation point (Q_{int}, t_{int}) is expressed as the product of POD modes, ψ and POD coefficients b [19]:

$$T(x, y, z; Q_{int}, t_{int}) = \sum_{i=1}^k \psi_i(x, y, z) b_i(Q_{int}, t_{int}) \quad (2)$$

Where k is the number of principal components that capture the dominant characteristics of the interrogation vector.

The numerical algorithm for computing POD modes involves the following steps.

Table 1. Specification of different parametric zones in terms of different regression operations

Zone	Heat load	Time
A	Interpolation	Interpolation
B	Interpolation	Extrapolation
C	Extrapolation	Interpolation
D	Extrapolation	Extrapolation
E	Extrapolation	Interpolation
F	Extrapolation	Extrapolation
G	Extrapolation	Extrapolation
H	Interpolation	Extrapolation
I	Extrapolation	Extrapolation

- Determine the covariance matrix of the data matrix.

$R = \frac{T_d^* \otimes T_d}{n}$, where $T_d \in \mathbb{R}^{m \times n}$: Data Matrix. $n = n_1 \times n_2$: $Q_{en} \in \mathbb{R}^{n_1 \times 1}$, $t_{en} \in \mathbb{R}^{n_2 \times 1}$.

- Determine the eigenvector of R by the power method: $u^{k+1} = \frac{Ru^k}{\|Ru^k\|}$.
- Compute the POD mode as the product of T_d and u : $\psi = T_d \otimes u$, $\psi \in \mathbb{R}^{m \times n}$.

The numerical algorithm for computing POD coefficients at the interrogation parametric point is described as follows.

1. Compute the complete coefficient matrix: $B(Q_{en}, t_{en}) = \psi^+ \otimes (T(x, y, z; Q_{en}, t_{en}))$, $B \in \mathbb{R}^{n \times n}$.
2. Determine the POD coefficient, $b(Q_{int}, t_{int}) \in \mathbb{R}^{n \times 1}$ by applying multidimensional interpolation of B .

The optimality of POD modes is qualified by the number of optimal basis (k), which is defined as the minimum number of basis functions that stratify the following inequality:

$$\left(1 - \frac{\sum_{i=1}^k \sigma_i}{\sum_{i=1}^n \sigma_i} \right) < tol \quad (3)$$

where σ_i is i th singular value of the data matrix, and tol is an operator-defined number estimating the error incurred for the model order reduction.

Ultimately, POD-based modeling decomposes the data matrix into a low-rank matrix multiplied with a suitable coefficient matrix. Based on the optimality criterion in Eq. (3), the model order is reduced, and the corresponding truncated POD-coefficient vector is determined. These two components are multiplied together to predict temperature at an interrogation point defined by (Q_{int}, t_{int}) . However, due to the data driven nature of the algorithm, the accuracy of the model needs to be controlled by a pre-defined error limit. As shown in Figure 1, if the prediction error is higher than the error limit, a secondary procedure is executed, which starts with a data matrix comprised of transient temperature snapshots for various interrogation heat loads. POD-based temperature modeling with time as the parameter is well documented in the literature [19]. The number of times the secondary procedure is invoked is quantified by a scalar named Count. Count has significant ramifications on the data compression of the proposed reduced-order model. Together with proposed error limit, Count damps out error incurred in the primary procedure. The efficiency of any dynamic model is determined by the time complexity of the underlying algorithm. Essentially, the proposed POD-based modeling algorithm relies on the Power method [20] for the iterative computation of POD modes. The convergence criteria and the time complexity of the Power method are described in the Appendix.

3. EXPERIMENTAL DETAILS

In this article, the POD-based algorithm is applied to the observation data generated experimentally. The next sections include descriptions of the experimental facility and the temperature measurement system.

3.1. Experimental Facility

Figure 3a shows the experimental setup. It is a raised-floor DC facility (located in Atlanta, GA at elevation $\sim 1,027'$ (313m)) with three computer room air

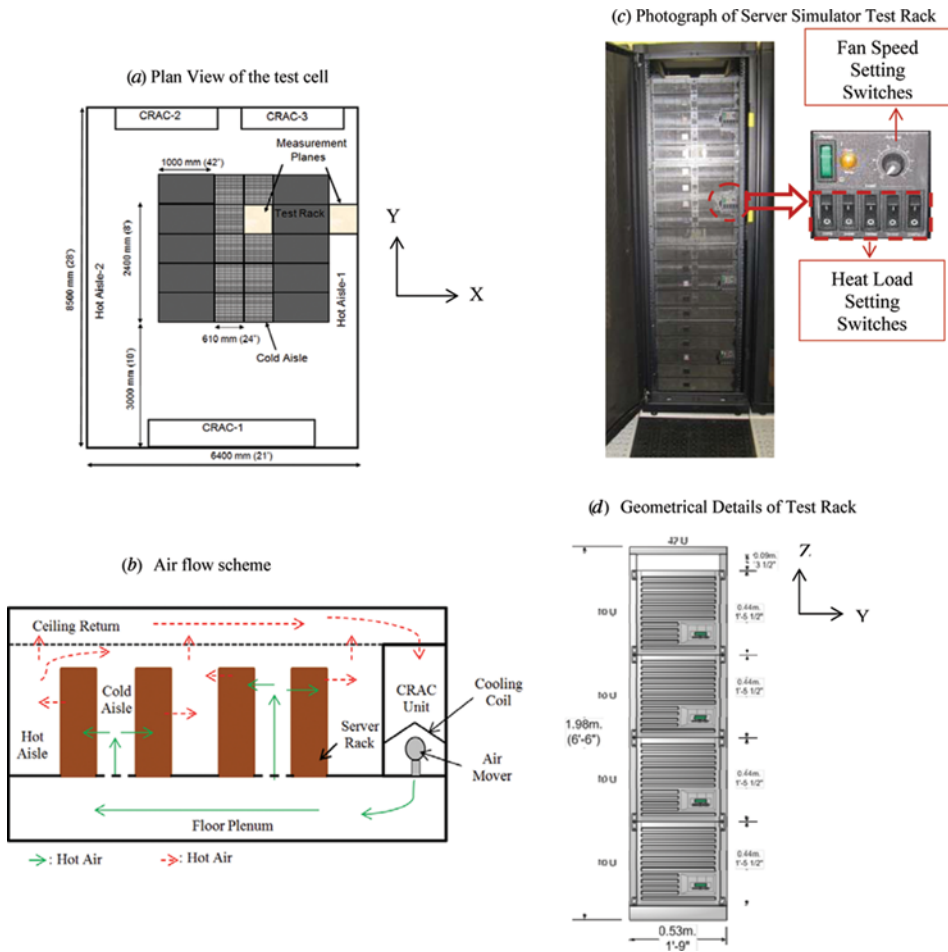


Figure 3. Details of the experimental facility. (a) plan view of the 672sq. ft. experimental facility, which has 10 racks arranged in a 5 x 2 architecture with alternating cold/hot aisles; (b) air flow scheme; (c) photograph of the test rack which is a server simulator rack, showing fan speed setting dial and heat load control switches; and (d) geometrical details of the test rack, which contains four server simulators. Each server simulator is a 10-U structure or 444.5mm (1'5") in height and 533mm (1'9") in width. The rack cabinet is a 42-U standard sized cabinet: height: 1981mm (6'6"), width: 600mm (1'11"), and depth: 1067mm (3'6") (taken from reference [25]).

conditioning (CRAC) units. Each CRAC is essentially a liquid-to-air heat exchanger controlled by the return air temperature. For this study, only CRAC-1 (FH740 Liebert® downflow) was operational in a raised-floor plenum supply and overhead plenum return airflow scheme (shown in Figure 3*b*). CRAC-1 is oriented in the 1-R configuration with the cold aisle. It discharges $6.7 \text{ m}^3/\text{s}$ (14,200 CFM) cooling air with a return air temperature control set point of 23.9°C (75°F). Figure 3*a* shows the facility. It includes 10 racks arranged in a 5×2 alternating cold/hot aisle arrangement: the server inlets face a cold aisle and the outlets a hot aisle. The CRAC unit supplies pressurized cold air to the under-floor plenum; this cold air is supplied to the cold aisle via perforated tiles. Measurements were performed in the vicinity of a server simulator rack, shown by solid squares in Figure 3*a*. Figure 3*c* shows the test rack, which has four identical vertically-stacked server simulators. Each has five heaters of capacities: 250 W, 500 W, 1,000 W, 1,000 W, and 2,000 W. The fan speed of the server simulators can be modulated from 0% – 100% at steps of 10%. At 100% capacity, the server simulator fan discharges $0.32 \text{ m}^3/\text{s}$ (678 CFM) of air [21]. Geometrically, a server simulator is a 10-U structure ($1 - \text{U} = 1.75'' = 44.5 \text{ mm}$), shown in Figure 3*d*.

3.2. Temperature Measurement

To characterize the air temperatures in the vicinity of the test rack, measurements are conducted in two planes, shown by the solid squares in Figure 3*a*: one is located in the hot aisle and another in the cold aisle. The height of the measurement plane extends to the top of the test rack, which is 2,000 mm tall. Figure 4*a* outlines the temperature measurement chain. The acquired analog thermocouple data are processed by data acquisition modules, and are transmitted to a LabVIEW™-based output interface via a Cisco® WET200 Wireless-G Business Ethernet Bridge router. At the beginning of a measurement, the typical latency period for the digital system is 13 s. Except for this initial offset, the data acquisition frequency is 1 Hz. The thermocouple-based data acquisition system has an uncertainty of $\pm 0.5^\circ\text{C}$, determined by an Omega® CL122 thermocouple calibrator, which utilizes a NIST traceable calibrated thermometer. Both systematic and random uncertainties for the temperature measurement are characterized. This systematic uncertainty is caused by the cold junction compensation and measurement inaccuracies incorporated by the data acquisition modules [22].

The measuring thermocouples are deployed in a grid-based mobile aluminum structure, shown in Figures 4*b* and *c*. Figure 4*b* shows the plan view of the thermocouple grid, and Figure 4*c* the side view. The grid is based on a $600 \text{ mm} \times 600 \text{ mm}$ ($\sim 2' \times 2'$) square planar geometry, designed for measuring temperature data parallel to the perforated tile. Every square grid includes 21 copper-constantan (Type T) thermocouples made from 28 gauge wire (0.9 mm diameter). Each thermocouple provides a response time of around 200 ms, which is suitable for rack-level air temperature measurements (sampling frequency 1 Hz.). Based on the square symmetry, two neighboring in-plane thermocouples are at 150 mm distance. For both cold and hot aisles, six thermocouple grids are deployed at various heights: (220, 576, 932, 1288, 1644, and 2000) mm. Figure 4*d* shows the orientation of six different thermocouple grids in the hot aisle and the global coordinate system. Overall, the temperature measurement system scans the rack-level temperature field

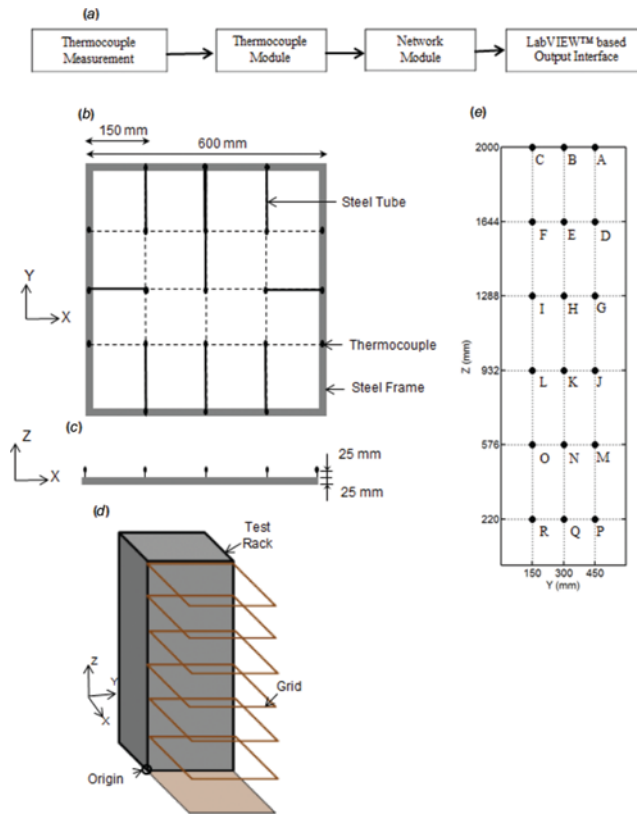


Figure 4. Details of the air temperature measurements. (a) Measurement chain, (b) plan view of grid-based thermocouple measurement unit, (c) side view of the measurement unit, (d) orientation of the thermo-couple grids at the exhaust of the test rack, and (e) data visualization plane located at the exhaust of the test rack. Filled circles indicate the location of thermocouples, where A is the top-right sensor located at (450, 2000)mm and R is the left-bottom sensor located at (150, 220)mm.

as a function of discretized spatial locations, (x, y, z) . The z -coordinate specifies the vertical height from the floor, varying from 220 mm – 2000 mm. Each ordered pair (x, y) identifies the planar location for a given height. For the square thermocouple grid, both x and y vary from 0 – 600 mm at a step of 150 mm. Overall, the 3-D temperature field in a given aisle is scanned by 126 ($= 21 \times 6$) thermocouples. Figure 4e shows the rack exhaust plane ($x=0$) with 18 thermocouples, taken as visualization and analysis plane for the rest of this article. Experimental data are collected by 18 sensors, shown by filled circular markers arranged in the form of a 3×6 grid. Located at the exhaust of the test rack, the interrogation region is 300 mm (150 mm – 450 mm) in width and 1780 mm (220 mm – 2000 mm) in height (Figure 4e).

4. CASE STUDY

To illustrate the present approach, the heat load of the test rack is varied parametrically to 11 different levels: $Q=[4,6,8,10,12,14,16,18,20,22, \text{ and } 23]\text{kW}$. For

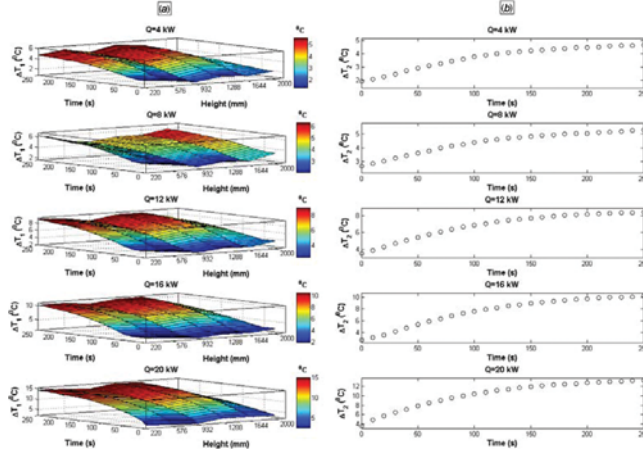


Figure 5. Impulse response of air temperature after turning on of the server simulator heat loads. Five different values of rack heat loads are used: $Q_i = [4, 8, 12, 16, 20]$ kW. (a) For a given value of the heat load Q_i , the corresponding surface plot shows the variation of ΔT_1 as a function of height from the floor and time. The data sampling set corresponds to six different heights from the floor: $h = [220, 576, 932, 1,288, 1,644, 2,000]$ mm. and 26 different time instants: $t = 0-250$ s at $\Delta t = 10$ s. (b) For a given value of the heat load Q_i , the plot shows the time series of ΔT_2 . The data sampling set includes 26 different time instants: $t = 0-250$ s at $\Delta t = 10$ s.

each heat load, the ensuing transient is observed by temperature measurements in the vicinity of the test rack at a sampling frequency of 1 Hz. It is characterized by two parameters: ΔT_1 and ΔT_2 . ΔT_1 is defined as the air temperature difference between the rack exhaust and the spatially-averaged incoming cooling air from the perforated tile located at the foot of the test rack.

$$\Delta T_1(x, y, z, Q, t) = T_1(x, y, z, Q, t) - T_{supply}(Q, t) \quad (4)$$

For a given heat load, ΔT_1 is a function of sensor height and time. In the exhaust plane, 18 sensors are deployed; therefore, ΔT_1 is an 18-dimensional vector. Figure 5a shows the variation of ΔT_1 as a function of time and height from the floor. As Q increases, the variation in ΔT_1 increases: for $Q = 4$ kW, the variation of ΔT_1 is within a range of $[2-5]^\circ\text{C}$; that for $Q = 20$ kW is $[5-15]^\circ\text{C}$. Also, ΔT_1 increases with time and gradually achieves steady-state at 200 s. Therefore, the transient measurements are stopped at 250 s. On the other hand, ΔT_2 is defined as the transient difference of the spatially-averaged temperatures in the hot and cold aisles.

$$\Delta T_2(Q, t) = \langle T_{hot}(x, y, z, Q, t) \rangle - \langle T_{cold}(x, y, z, Q, t) \rangle \quad (5)$$

Figure 5b shows time series of ΔT_2 for different values of $Q = [4, 8, 12, 16, \text{ and } 20]$ kW. As expected, the rise in ΔT_2 increases with Q : for $Q = 4$ kW, the rise is 2.7°C ; for $Q = 20$ kW, it is 9.6°C . Also, for all values of Q , ΔT_2 reaches a steady state before the arbitrarily determined transient measurement window of 250 s.

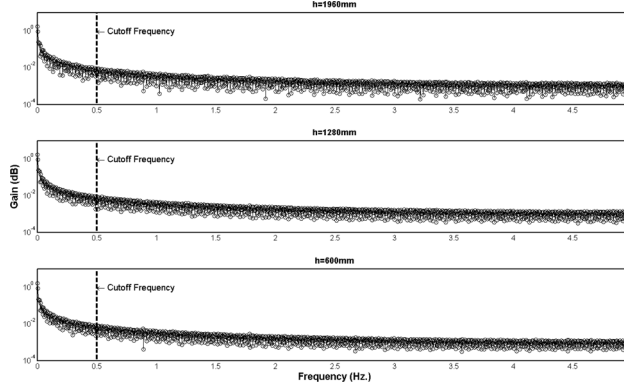


Figure 6. Frequency response of the hot aisle temperature field at three different heights in the rack exhaust plane.

Figure 6 shows frequency responses for air temperatures collected at three different heights, $h = [1960, 1280, \text{ and } 600]$ mm for the maximum rack heat load of 23 kW. The Y -axis shows the gain (in dB) of rack exhaust temperature scaled by the CRAC return air temperature set point ($22.9^{\circ}\text{C} \sim 75^{\circ}\text{F}$). The gain is computed by the discretized Fourier transform (DFT) using a MATLAB-based fast Fourier transform (FFT) algorithm [20, 23]. As shown in Figure 6, the scaled amplitude significantly flattens out after 0.5 Hz, implying that the minimum sampling rate to characterize the transient temperature must be at least 1 Hz, via the Nyquist-Shannon sampling theorem.

As the first step of the POD-based framework, the ensemble set of the parameters needs to be identified. Without losing generality, it is chosen as a subset of the experimental heat load values.

$$Q_{en} = [4, 8, 12, 16, 20] \text{ kW} \quad (6)$$

This heat load input space defines a representative sample space of commonly-occurring dynamic heat load patterns [6] in a typical data center rack. Also, the ensemble heat load dimension reduces the input size from 11 to 5 (54% input data compression with respect to the measurement set). On the other hand, the sampling frequency for the time ensemble is reduced to 0.1 Hz.

$$t_{en} = 10 - 200 \text{ s at } \Delta t = 10 \text{ s} \quad (7)$$

Such a down-sampling reduces the ensemble size by 90% with respect to the measured transient data. Overall, the data for the primary procedure is organized in a 124×95 matrix, where 124 is the number of thermocouples deployed in the hot aisle. The column index of the data matrix is defined by (1). For the given problem, $N_t = 19$, $i_t = \{1 - 19\}$, and $i_h = \{1 - 5\}$. The size of the input space is equal to $95 (= 19 \times 5)$ whereas that of the primitive observations is equal to 2,101 ($= 191 \times 11$). That amounts to 95.5% data compression. However, if the secondary procedure is invoked, the data compression in heat load dimension is completely nullified for

the sake of improving prediction fidelity. In the secondary procedure, the problem becomes single-parameter with time as the parameter. The secondary procedure amounts to 90% data compression.

5. RESULTS AND DISCUSSION

The temperature ensemble is generated by applying the POD-based algorithm on distributed temperature data collected at (Q_{en}, t_{en}) . As shown in Figure 2, an interrogation point (Q_{int}, t_{int}) can lie in nine possible regions in the $Q - t$ parametric plane. This classification is based on the position of an interrogation point with respect to (Q_{en}, t_{en}) and the associated regression operation. For example, the point (10kW, 120s) lies in the region A, which requires interpolation both in heat load and time to compute POD-based temperature predictions. Detailed specifications of the different regions are documented in Table 1.

For the sake of scalable parametric modeling, heat load is normalized as follows.

$$\bar{Q} = \frac{Q}{Q_{base}}, \quad Q_{base} = 4 \text{ kW} \quad (8)$$

Based on this normalized definition of heat load, the set of normalized heat load snapshots is $\bar{Q}_{en} = [1, 2, 3, 4, 5]$.

This suggests that any interrogation heat load for which $\bar{Q}_{int} \in (1, 5)$ is within the heat load ensemble set, \bar{Q}_{en} . Similarly, time is normalized as follows.

$$\bar{t} = \frac{t - t_{initial}}{t_{final} - t_{initial}}, \quad t_{initial} = 10 \text{ s}, t_{final} = 200 \text{ s} \quad (9)$$

Based on this normalized definition of time, $\bar{t}_{en} = [0, (10/190), (20/190), \dots, (170/190), (180/190), 1]$. This normalization suggests that interrogation time, for which $\bar{t}_{int} \in (0, 1)$, is within the time ensemble set, \bar{t}_{en} . The normalization scheme is chosen for the sake of compactness and scalability of the parametric analysis. Based on required regression operations, there are four ($=2^2$) possible combinations: interpolation both in heat load and time (zone A), interpolation in heat load and extrapolation in time (zones B and H), extrapolation in heat load and interpolation in time (zones C and E), and extrapolation both in heat load and time (zones D, F, G, and I). The zones corresponding to $(\bar{t} < 0 \text{ or } \bar{Q} < 1)$ are ignored for its least practical significance in DC characterization. Hence, this study focuses on model predictions in zones A, B, C, and D.

5.1. Model Prediction in Zone A

For this case study, zone A is the parametric subspace, spanned by $(Q_{int}, t_{int}) \in \{(1 < \bar{Q}_{int} < 5) \cup (0 < \bar{t}_{int} < 1)\}$. In this zone, an arbitrary parametric point is chosen as (14kW, 124s); the corresponding normalized interrogation point is (3.5, 0.6). For this interrogation point, Figure 7 demonstrates the fidelity of POD-based temperature predictions in the spatial domain located at the rack

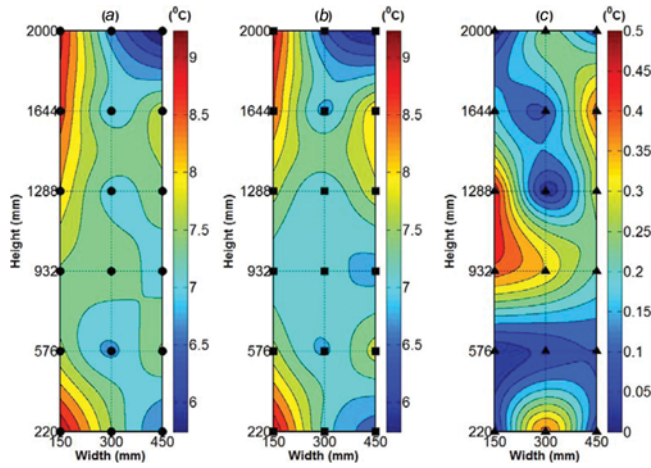


Figure 7. Temperature prediction capability of the POD-based framework in zone A. The interrogation point is: $[\bar{Q} = 3.5, \bar{t} = 0.6]$. (a) Mapping of experimentally-acquired ΔT_1 . Circular markers represent the location of temperature sensors. (b) Mapping of POD-predicted ΔT_1 . Square markers represent the prediction points. (c) Mapping of the deviations between experimental data and POD predictions. Triangular markers show absolute deviation data. The contour mapping from the discrete data points is performed via the Delaunay triangulation. For the conditional step, the error limit assigned is 0.5°C . The number of times the second level computation is invoked or Count is observed to be equal to 4.

exhaust plane (as shown in Figure 4e). Figure 7a shows ΔT_1 computed from the temperature measurement data and its mapping at the test rack exhaust. The POD-based algorithm is applied on the measurement data. The absolute deviations between experimental data and POD predictions are also computed. From these discrete measurements/predictions, the contour surfaces are generated by Delaunay triangulation [24]. Figure 7b shows ΔT_1 computed from the POD-based predictions.

The computational time for the POD-based prediction is on the order of 10 s (on an Intel® Core™2.Duo CPU of specification E8200 @ 2.66 GHz supported by 4 GB RAM). Figure 7c shows the absolute deviation between the experimentally acquired ΔT_1 and the corresponding POD-predicted ΔT_1 . For zone A, the error limit for the conditional block is set equal to the calibration error (0.5°C). The value of Count, which is defined as the number of times the secondary procedure is executed, is observed to be 4. The sensors for which the absolute error exceeds the error limit are C, F, O, and R. Interestingly, all these sensors lie along the line, $y = 150\text{ mm}$, and Figure 7a suggests that these locations are hotspots. The overall root mean square (RMS) deviation is observed to be equal to 0.25°C .

Due to the semi-empirical nature of the algorithm, modeling fidelity needs to be verified for various interrogation points. Table 2 shows the prediction errors for various interrogation points defined by first two columns \bar{Q} and \bar{t} . For every combination of (\bar{Q}, \bar{t}) , the third column of the error table presents the maximum error, defined as the maximum of the absolute deviations between measurement data and POD predictions. The fourth column tabulates the RMS of the relative errors. It is clear from Table 2 that the RMS errors vary within a range of [0%–6%], with an average of 2.31% and standard deviation of 1.66%. The sampling

Table 2. Error table for POD-based predictions in zone A for different interrogation points

Normalized heat load (\bar{Q})	Normalized time (\bar{t})	Max. error ($^{\circ}\text{C}$)	RMS relative error
1.5	0.2	0.49	4%
	0.4	0.40	2%
	0.6	0.39	1%
	0.8	0.20	1%
2.5	0.2	0.48	6%
	0.4	0.48	4%
	0.6	0.45	4%
	0.8	0.44	4%
3.5	0.2	0.12	1%
	0.4	0.49	2%
	0.6	0.46	3%
	0.8	0.45	2%
4.5	0.2	0.08	1%
	0.4	0.09	1%
	0.6	0.12	1%
	0.8	0.1	0%

time interval for constructing the temperature ensemble is arbitrarily assumed to be equal to 10s, which is ten times higher than the measurement sampling interval. Since this choice is arbitrary, the sensitivity of the sampling time interval on POD predictions is analyzed in Table 3. In general, a higher sampling time interval means improved effectiveness of data compression. Nevertheless, such data compression often comes at the cost of modeling accuracy [25]. Due to the semi-empirical nature of the proposed model, three different sampling intervals of 5s, 10s, and 15s (or 5, 10, 15 when they are scaled by the measurement sampling interval of 1s) are tested for prediction fidelity at four time instants: $\bar{t} = [0.2, 0.4, 0.6, 0.8]$. The third column tabulates the RMS values of the relative errors. A sampling interval choice of 10s yields the most accurate prediction.

Table 3. Sensitivity of model fidelity to ensemble sampling interval. The interrogation heat loads for these predictions are kept constant at 14kW. A candidate space of $\Delta t = 5\text{ s}, 10\text{ s}, 15\text{ s}$ is examined.

Normalized time	Scaled time step size	RMS relative error
0.2	5	6%
	10	1%
	15	5%
0.4	5	2%
	10	2%
	15	2%
0.6	5	3%
	10	3%
	15	2%
0.8	5	3%
	10	2%
	15	2%

5.2. Model Prediction in Zone B

For this case study, zone B is the parametric subspace spanned by $(Q_{int}, t_{int}) \in \{(1 < \bar{Q}_{int} < 5) \cup (0 < \bar{t}_{int} > 1)\}$. This predictor space demands a temperature prognosis in time. Therefore, it is particularly pertinent to the analysis of various thermal runaway phenomena during commonly-occurring emergencies, such as grid power outages and cooling equipment (e.g., chiller pump) failures. In this zone, an arbitrary parametric point is chosen as (14kW, 209s); the corresponding normalized interrogation point is (3.5, 199/190). Figure 8 demonstrates the fidelity of POD-based temperature predictions in the spatial domain located at the rack exhaust plane, shown in Figure 4e. Figure 8a shows ΔT_1 computed from the temperature measurement data and its mapping at the test rack exhaust. Figure 8b shows ΔT_1 computed from the POD-based predictions. The computational time for the POD-based prediction is on the order of 10s (on an *Intel®Core™2.Duo* CPU of specification E8200 @ 2.66GHz supported by 4 GB RAM). Figure 8c shows the absolute deviation between the experimentally acquired ΔT_1 and the corresponding POD-predicted ΔT_1 . For zone B, the error limit for the conditional block is set equal to the calibration error (0.5°C). The value of Count is equal to 15. The RMS deviation is equal to 0.6°C.

The secondary procedure in the proposed algorithm is designed for improving prediction accuracy. However, for extrapolation, after some critical time window even the proposed secondary procedure fails to damp out the error. Therefore, it

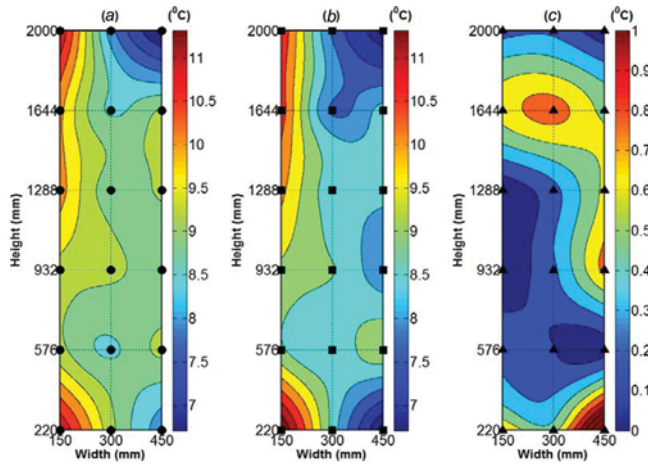


Figure 8. Ability of the POD-based framework to predict new temperature data in zone B. The interrogation point is arbitrarily assigned: $[\bar{Q} = 3.5, \bar{t} = 199/190]$. Experimental data are collected by 18 sensors shown by black circular markers arranged in the form of a 3×6 grid. Located at the exhaust of the test rack, the interrogation region is 300 mm (150 mm – 450 mm) in width and 2000 mm (0 mm – 2000 mm) in height. The POD-based algorithm is applied on the measurement data. The deviations between experimental data and POD predictions are noted. (a) Mapping of experimentally-acquired temperature data; (b) mapping of POD-predictions; and (c) Mapping of the deviations between experimental data and POD predictions. The mapping from the data points is done via the Delaunay triangulation. The intermediate error limit assigned is 0.5 °C. The number of times the second level computation is invoked or Count is equal to 15.

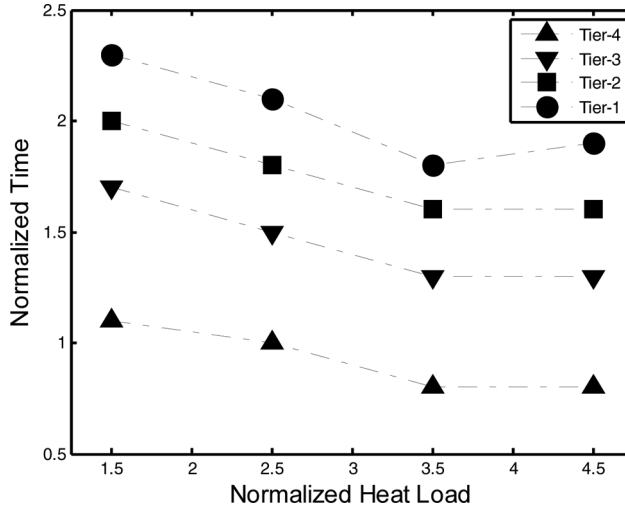


Figure 9. For four different DC Tiers, the normalized extrapolation time limit ($\bar{t}_{exp} = \frac{t_{exp} - t_{final}}{\Delta t} : t_{final} = 200 \text{ s}, \Delta t = 1 \text{ s}$) versus interrogation heat load. DC Tiers are classified by the error factor, which is defined as the ratio of the tolerable RMS error to the calibration error (0.5°C).

is imperative to estimate an acceptable extrapolation window. The fault-tolerance depends on the design redundancy or the tier-status [26] of a DC. A tier-4 DC is least fault-tolerant. On the other hand, a tier-1 DC is most fault-tolerant. For this case study, the design redundancies are quantified as the error factor which is defined as the ratio of acceptable root mean square error to the calibration error (0.5°C). A tier-4 facility has an error factor numerically equal to 1, and that for a tier-1 facility is 4. For tiers-2 and -3, the error factors are 3 and 2, respectively. Therefore, an error factor equal to 3 means the tolerable RMS error limit is equal to 1.5°C . The reliable extrapolation time window is defined as the normalized extrapolation time interval: ($\bar{t}_{exp} = \frac{t_{exp} - t_{final}}{\Delta t} : t_{final} = 200 \text{ s}, \Delta t = 10 \text{ s}$), for which the proposed fidelity criterion is satisfied. Figure 9 shows reliable extrapolation time bounds as a function of heat load for four different DC tiers. For different error factors, the reliable extrapolation window is computed for various values of interrogation heat load: $\{\bar{Q} = 1.5, 2.5, 3.5, 4.5\}$. Figure 9 indicates that with increase in heat load, the reliable prediction window decreases—predictive framework in zone B. This facilitates prognostic-based thermal reliability modeling of a DC.

5.3. Model Prediction in Zone C

For this case study, zone C is the parametric subspace spanned by $(Q_{int}, t_{int}) \exists \{(\bar{Q}_{int} > 5) \cup (0 < \bar{t}_{int} < 1)\}$. This predictor space demands a temperature prognosis in heat load and is particularly useful for the analysis of thermal spikes resulting from rapidly escalating DC IT loads induced by the flash-crowd effect [27]. To verify the prediction fidelity, an arbitrary parametric point is chosen as (22kW, 124s); the corresponding normalized interrogation point is

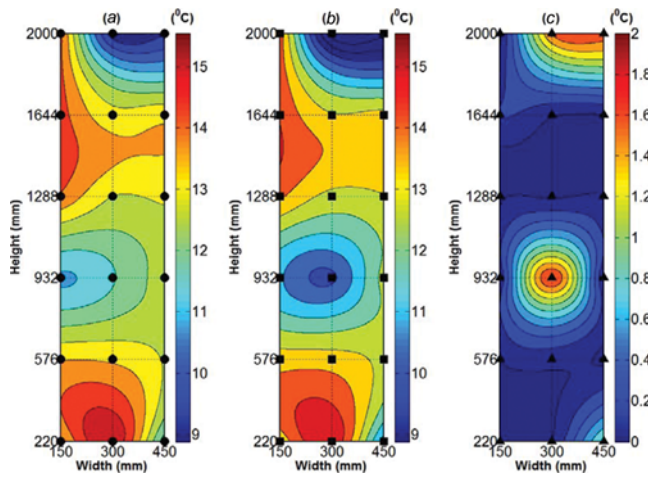


Figure 10. Ability of the POD-based framework to predict new temperature data in zone C. The interrogation point is arbitrarily assigned: $[\bar{Q} = 5.5, \bar{t} = 0.6]$. Experimental data are collected by 18 sensors shown by black circular markers arranged in the form of a 3×6 grid. Located at the exhaust of the test rack, the interrogation region is 300 mm (150 mm – 450 mm) in width and 2000 mm (0 mm – 2000 mm) in height. The POD-based algorithm is applied on the measurement data. The deviations between experimental data and POD predictions are noted. (a) Mapping of experimentally-acquired temperature data; (b) mapping of POD-predictions; and (c) mapping of the deviations between experimental data and POD predictions. The mapping from the data points is done via the Delaunay triangulation. The intermediate error limit assigned is 2°C. The number of times the second level computation is invoked or Count is equal to 14.

(5.5, 0.6). Figure 10 demonstrates the fidelity of POD-based temperature predictions in the spatial domain located at the rack exhaust plane, shown in Figure 4e. Figure 10a shows ΔT_1 computed from the temperature measurement data and its mapping at the test rack exhaust. Figure 10b shows ΔT_1 computed from the POD-based predictions. The computational time for the POD-based prediction is on the order of 10s (on an Intel® Core™2. Duo CPU of specification E8200 @ 2.66GHz supported by 4 GB RAM). Figure 10c shows the absolute deviation between experimentally acquired values of ΔT_1 and the corresponding POD-predicted values of ΔT_1 . For zone C, the error limit for the conditional block is set equal to four times the calibration error (0.5°C). Often during thermal spikes, rapid thermal characterization is desirable even at the cost of some prediction accuracy. The value of Count, which is defined as the period of time over which the secondary procedure is executed, was observed to be equal to 14. The RMS deviation is equal to 0.7°C.

Table 4 shows the prediction errors for various interrogation points defined by the first two columns \bar{Q} and \bar{t} . For every combination of (\bar{Q}, \bar{t}) , the third column of the error table tabulates the maximum error, defined as the maximum of the absolute deviations between measurement data and POD predictions. The fourth column tabulates the root mean square value of the relative errors. It is clear from Table 4 that the relative RMS error varies within a range of [5%–12%], with an average of 8.1% and a standard deviation of 2.4%.

Table 4. Prediction performance of the POD-based framework in zone C

Normalized heat load (\bar{Q})	Normalized time (\bar{t})	Max. error ($^{\circ}\text{C}$)	Relative RMS error
5.5	0.2	1.86	12%
	0.4	1.98	9%
	0.6	1.75	7%
	0.8	1.97	6%
5.75	0.2	1.62	8%
	0.4	1.91	11%
	0.6	1.99	7%
	0.8	1.74	5%

5.4. Model Prediction in Zone D

For this case study, zone D is the parametric subspace spanned by $(\bar{Q}_{int}, t_{int}) \ni \{(\bar{Q}_{int} > 5) \cup (\bar{t}_{int} > 1)\}$. Such a parametric space is particularly pertinent to the worst case analysis useful for the preliminary design of a data center. In this zone, an arbitrary interrogation point is chosen as (22 kW, 209 s); the corresponding normalized interrogation point is (5.5, 199/190). Figure 11 demonstrates the fidelity of POD-based temperature predictions in the spatial domain located at the rack exhaust plane, shown in Figure 4e. Figure 11a shows ΔT_1 computed from the

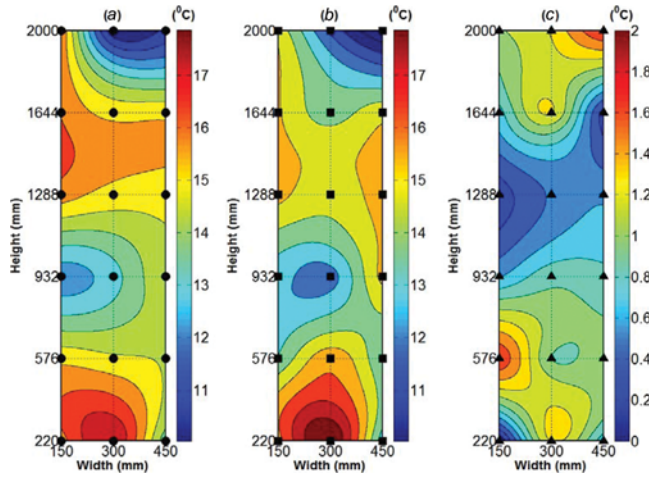


Figure 11. Ability of the POD-based framework to predict new temperature data in zone D. The interrogation point is arbitrarily assigned: $[\bar{Q} = 5.5, \bar{t} = 199/190]$. Experimental data are collected by 18 sensors shown by black circular markers arranged in the form of a 3×6 grid. Located at the exhaust of the test rack, the interrogation region is 300 mm (150 mm – 450 mm) in width and 2000 mm (0 mm – 2000 mm) in height. The POD-based algorithm is applied on the measurement data. The deviations between experimental data and POD predictions are noted. (a) Mapping of experimentally-acquired temperature data; (b) mapping of POD-predictions, and (c) Mapping of the deviations between experimental data and POD predictions. The mapping from the data points is done via the Delaunay triangulation. The intermediate error limit assigned is 2°C . The number of times the second level computation is invoked or Count is observed to be equal to 14.

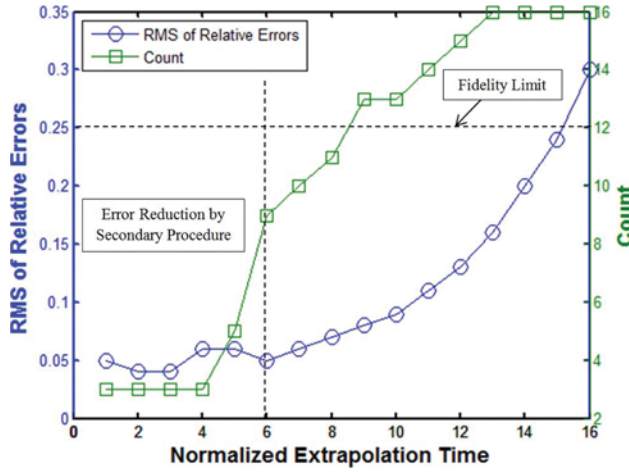


Figure 12. Root mean square of relative errors versus normalized extrapolation time for $\bar{Q} = 5.5$. All data points lie in zone D. The second Y-axis shows the variation of Count with extrapolation time.

temperature measurement data and its mapping at the test rack exhaust. Figure 11b shows ΔT_1 computed from the POD-based predictions. The computational time for the POD-based prediction is on the order of 10s (on an Intel®Core™2. Duo CPU of specification E8200 @ 2.66GHz supported by 4 GB RAM). Figure 11c shows the absolute deviation between experimentally acquired ΔT_1 and the corresponding PODpredicted ΔT_1 . For zone D, the error limit for the conditional block is set to be equal to four times the calibration error (0.5°C). The value of Count is observed to be equal to 13. The RMS deviation is equal to 1.02°C .

POD coefficient computation in zone D requires extrapolation-based regression operations. Therefore, it is imperative that the model fidelity criterion be identified. The cut-off criterion is postulated as: the RMS value of relative errors is equal to 25%. Figure 12 shows the root mean variation of relative error and Count with normalized extrapolation time, which is defined as: $\left(\bar{t}_{exp} = \frac{t_{exp} - t_{final}}{\Delta t} : t_{final} = 200 \text{ s}, \Delta t = 1 \text{ s}\right)$. It can be observed that for $\bar{t}_{exp} \leq 6$, the root mean square of relative error is substantially damped by the secondary procedure. For $\bar{t}_{exp} > 7$, the deviations increase rapidly to the extent that the secondary procedure fails to damp it. This is reflected in an exponential increase in root mean square values of relative errors and Count. Finally, the fidelity limit (25%) is reached at 15s.

5.5. Coherent Structure of the Experimental Data

While the prediction fidelity of the proposed approach is established, it is worthwhile to inspect the efficiency of the algorithm. For efficient compact modeling, POD-based algorithms rely on model order reduction. For the present case study, a tolerance limit (as defined by Eq. (2)) is set at 5%. Based on the proposed tolerance limit, Eq. (3) determines the number of POD modes required for capturing dominant

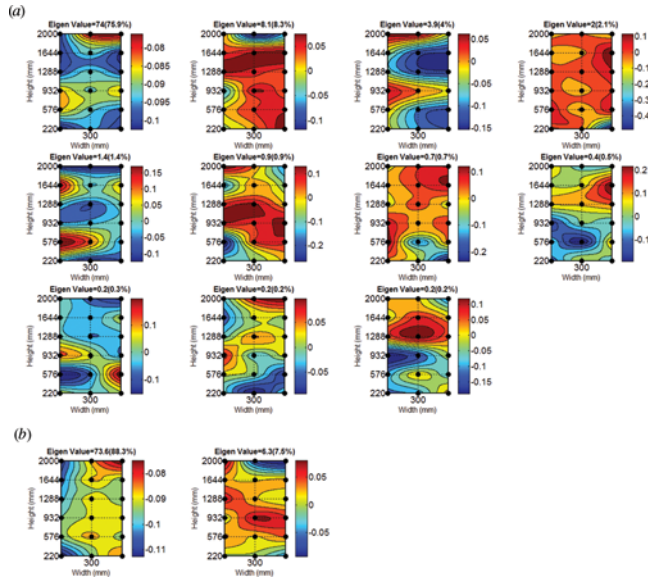


Figure 13. POD-based mode decomposition for the given data matrix. The tolerance level is set equal to 95%. For the primary procedure (a), it takes 11 out of 95 POD modes to reach the 5% tolerance limit. On the other hand, the secondary procedure (b) takes 2 out of 19 POD modes to reach the 5% tolerance limit.

components of the temperature field or coherent structures. For the present two-tier algorithm, two separate POD-based spectral analyses have been conducted. For an arbitrarily assigned 5% tolerance limit, the principal component number for the primary procedure is 11 (which means $(1-11/95) = 88.5\%$ order reduction), and that for the secondary procedure is 2 (which means $(1-2/19) = 89.5\%$ order reduction). Figure 13 shows the POD-based mode decomposition spectrum. Figure 13a shows 11 optimal (dominant) POD modes for the primary procedure in the proposed algorithm. Expectedly, the first eigenvalue is the spectral radius or the maximum eigenvalue of magnitude 74. It captures as much as 75.9% of the energy of the parametric temperature field. Figure 13b shows 2 optimal (dominant) POD modes for the secondary procedure of the proposed algorithm. In this case, the first eigenvalue captures 88.3% of the energy of the parametric temperature field. The POD modes essentially recognize the pattern of the dynamic temperature. Following this basic pattern, actual temperature responses are modulated by the parametric position on the interrogation point in the predictor space.

6. SUMMARY AND CONCLUSION

A data-driven temperature modeling algorithm is developed that enables rapid DC air temperature characterization in a two-dimensional parametric space comprising rack heat load and time. The algorithm is enabled by a two-tier spectral method-based numerical procedure. The secondary procedure improves the modeling fidelity in case the primary procedure breaches certain error limit. The

fidelity of the model is verified by comparing its predictions with corresponding data. A full-factorial parametric analysis is conducted. Recognizing the empirical nature of the proposed model, the prediction fidelity is assessed at various interrogation points in the parametric space. The comparison between POD-based local air temperature predictions and corresponding data indicates that root mean square prediction uncertainty is equal to 4.7%. The underlying coherent structures for both the primary and the secondary procedures are noted, and their optimal eigenvalues are determined.

The proposed algorithm enables us to develop an efficient and full-factorial parametric design tool. The algorithm displays a logarithmic scalability with the size of the parametric space. Therefore, the proposed algorithm is particularly useful as a rapid signal generator in feedback control systems in DC heating, ventilation, and air-conditioning (HVAC) systems. As the model offers significant amount of data compression, the static data storage requirement is significantly low. The required number of computational operations is on the order of the logarithm of the input space; this property can be leveraged for the low-cost CPU design of HVAC control units in DCs.

FUNDING

The authors acknowledge the support from IBM via a department of energy (DoE) award, monitored by Drs. Hendrik Hamann and Levente Klein.

REFERENCES

1. J. Koomey, *Growth in Data Center Electricity use 2005 to 2010*, A Report by Analytics Press Completed at the Request of The New York Times, Oakland, CA, 2011.
2. J. G. Koomey, C. Belady, M. Patterson, A. Santos, and K. D. Lange, *Assessing Trends Over Time in Performance, Costs, and Energy Use for Servers*, Final Report to Microsoft Corporation and Intel Corporation, Tech. Rep., 2009.
3. S. Greenberg, E. Mills, B. Tschudi, P. Rumsey, and B. Myatt, Best Practices for Data Centers: Lessons Learned from Benchmarking 22 Data Centers, *Proc. of the ACEEE Summer Study on Energy Efficiency in Buildings*, Asilomar, CA, pp. 76–77, ACEEE, 2006.
4. T. K. Nielsen, and D. Bouley, *How Data Center Infrastructure Management Software Improves Planning and Cuts Operational Costs*, APC White Paper, 2012.
5. M. K. Patterson, The Effect of Data Center Temperature on Energy Efficiency, *11th Intersociety Conf. on Thermal and Thermomechanical Phenomena in Electronic Systems*, Orlando, FL, pp. 1167–1174, IThERM, 2008.
6. M. K. Patterson, and D. Fenwick, *The State of Datacenter Cooling*, Intel Corporation White Paper, 2008.
7. M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, and I. Stoica, A View of Cloud Computing, *Comm. of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.
8. S. Kang, R. R. Schmidt, K. M. Kelkar, A. Radmehr, and S. V. Patankar, A Methodology for the Design of Perforated Tiles in Raised Floor Data Centers using Computational Flow Analysis, *IEEE Trans. on Components and Packaging Technologies*, vol. 24, no. 2, 177–183, 2001.

9. C. D. Patel, C. E. Bash, C. Belady, L. Stahl, and D. Sullivan, Computational Fluid Dynamics Modeling of High Compute Density Data Centers to Assure System Inlet Air Specifications, *ASME Inter. Electronic Packaging Technical Conf. and Exhibition*, Kauai, Hawaii, IPACK2001-15622, 2001.
10. J. Rambo, and Y. Joshi, Modeling of Data Center Airflow and Heat Transfer: State of the Art and Future Trends, *Distributed and Parallel Databases*, vol. 21, no. 2, pp. 193–225, 2007.
11. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Taylor & Francis Group, New York, 1980.
12. H. F. Hamann, J. A. Lacey, M. O'Boyle, R. R. Schmidt, and M. Iyengar, Rapid Three-Dimensional Thermal Characterization of Large-Scale Computing Facilities, *IEEE Trans. on Components and Packaging Technologies*, vol. 31, no. 2, pp. 444–448, 2008.
13. K. Kunisch and S. Volkwein, Galerkin Proper Orthogonal Decomposition Methods for a General Equation in Fluid Dynamics, *SIAM J. Numer. Analysis*, vol. 40, no. 2, pp. 492–515, 2002.
14. G. Berkooz, P. Holmes, and J. L. Lumley, The Proper Orthogonal Decomposition in the Analysis of Turbulent Flows, *Annu. Review of Fluid Mechanics*, vol. 25, no. 1, pp. 539–575, 1993.
15. L. Sirovich, Turbulence and the Dynamics of Coherent Structures, I—Coherent Structures, II—Symmetries and Transformations, III—Dynamics and Scaling, *Quarterly of Applied Mathematics*, vol. 45, pp. 561–571, 1987.
16. S. Bernero and H. Fiedler, Application of Particle Image Velocimetry and Proper Orthogonal Decomposition to the Study of a Jet in a Counterflow, *Experiments in Fluids*, vol. 29, no. 1, pp. S274–S281, 2000.
17. D. Chambers, R. Adrian, P. Moin, D. Stewart, and H. Sung, Karhunen–Loève Expansion of Burgers' Model of Turbulence, *Physics of Fluids*, vol. 31, pp. 2573–2582, 1988.
18. S. Ravindran, Reduced-Order Adaptive Controllers for Fluid Flows using POD, *J. of Scientific Computing*, vol. 15, no. 4, pp. 457–478, 2000.
19. R. Ghosh and Y. Joshi, Error Estimation in POD-Based Dynamic Reduced-Order Thermal Modeling of Data Centers, *Int. J. of Heat and Mass Transfer*, vol. 57, no. 2, pp. 698–707, 2013.
20. G. Strang, *Introduction to Linear Algebra*, Wellesley Cambridge Press, 2003.
21. G. M. Nelson, Development of an Experimentally-Validated Compact Model of a Server Rack, Masters Thesis, Georgia Institute of Technology, Atlanta, Georgia, 2007.
22. P. R. N. Childs, *Practical Temperature Measurement*, Butterworth-Heinemann, 2001.
23. M. Frigo and S. G. Johnson, FFTW: An Adaptive Software Architecture for the FFT, *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, vol. 3, pp. 1381–1384, Seattle, WA, 1998.
24. D. T. Lee and B. J. Schachter, Two Algorithms for Constructing a Delaunay Triangulation, *Int. J. of Parallel Programming*, vol. 9, no. 3, pp. 219–242, 1980.
25. H. MacMahon, K. Doi, S. Sanada, S. Montner, M. Giger, C. E. Metz, N. Nakamori, F. Yin, X. Xu, and H. Yonekawa, Data Compression: Effect on Diagnostic Accuracy in Digital Chest Radiography, *Radiology*, vol. 178, no. 1, pp. 175–179, 1991.
26. W. P. Turner, IV, J. H. Seader, V. Renaud, and K. G. Brill, Tier Classifications Define Site Infrastructure Performance, White Paper, Uptime Institute, 2006.
27. Q. Zhang, L. Cheng, and R. Boutaba, Cloud Computing: State-of-the-Art and Research Challenges, *J. of Internet Services and Applications*, vol. 1, no. 1, pp. 7–18, 2010.

APPENDIX

A decomposition, $A = \sum_{i=1}^n \sigma_i U_i V_i^T$ is called the proper orthogonal decomposition if the sequence of σ_i is non-increasing, and the sets of $\{U_i\}$, $\{V_i\}$ are orthonormal. The POD computation is usually carried out via the power method.

- Compute the normalized inner product of A : $P = (A^T A / \text{rank}(A^T A))$.
- Assign a random unit vector, x .
- Iterate until it reaches convergence: $x := \frac{Px}{\|Px\|}$.

The subsequent discussion pertains to the convergence of the Power method.

Let $\{v_i\}$ be the eigenvectors of P and let $\{\lambda_i\}$ be the corresponding eigenvalues. Let, x^k be the unit vector obtained after k th iteration. Since $\{v_i\}$ are orthonormal,

$$\|Px^k\|^2 = \frac{\sum_i \lambda_i^{2k+2}}{\sum_i \lambda_i^{2k}}.$$

Now, by Holder's inequality,

$$\sum_i \lambda_i^{2k} \leq \left(\sum_i \lambda_i^{2(k+1)} \right)^{\frac{k}{k+1}} n^{\frac{1}{k+1}}, \text{ where } n \text{ is the rank of the eigenspace.}$$

Rearrangement of the resulting inequality yields the following.

$$\begin{aligned} \frac{\sum_i \lambda_i^{2k+2}}{\sum_i \lambda_i^{2k}} &\geq \frac{1}{n^{1/k}} \left[\sum_i \lambda_i^{2k} \right]^{1/k} \geq \frac{\lambda_1^2}{n^{1/k}}. \\ \Rightarrow \|Px^k\|^2 &\geq \frac{\lambda_1^2}{n^{1/k}}. \end{aligned}$$

On the other hand, since $\{v_i\}$ are orthonormal: $\frac{\lambda_1^2}{n^{1/k}} \leq \|Px^k\|^2 \leq \lambda_1^2$.

This bound shows that $\|Px^k\|^2$ asymptotically converges to λ_1^2 .

The left inequality suggests the minimum number of computational steps required for reaching a converged solution. At the k th iteration the ratio of the iterative solution to the converged solution is equal to $1/n^{1/k}$. A convergence criterion is chosen as 2^{-r} such that,

$$1/n^{1/k} \sim 2^{-p} \Rightarrow \frac{\log_2(n)}{k} \sim p \Rightarrow k \sim \frac{\log_2(n)}{p}$$

Since p is a machine dependence parameter, the time complexity of the Power method is on the order of $\log(n)$. This is an important property for any reduced-order modeling algorithm; it indicates that the computation time is marginally affected even if the rank of the data matrix is quite high. Such scalability is a desirable feature for a model order reduction problem.