A Transient Model for Data Center Thermal Prediction

Michael Jonas, Rose Robin Gilbert, Joshua Ferguson, Georgios Varsamopoulos, Sandeep K. S. Gupta
Impact Lab, http://impact.asu.edu/
School of Computing, Informatics and Decision Systems Engineering, ASU, Tempe, Arizona
{mjonas, roserobin.gilbert, jwfergus, georgios.varsamopoulos, sandeep.gupta}@asu.edu

***Abstract*—Fast thermal maps are a crucial component for many green data center design techniques. However, most state of the art work on thermal mapping ignores critical temporal aspects of thermal behavior and relies on modeling assumptions, such as the steady state assumption, that can reduce their accuracy and cause heat-induced performance throttling when used for task scheduling. These problems have the potential to affect the energy savings projected by such models. This paper introduces a fast thermal modeling technique that captures the transient behavior necessary to improve thermal prediction capability while retaining the fast thermal model required for many green data center design paradigms. This model is compared against Computational Fluid Dynamics (CFD) simulations (using OpenFOAM) and an open platform framework, namely BlueTool.**

I. Introduction

Energy-aware computing has grown dramatically in the recent years. From longer cell phone battery lifetimes to low-power screens and power-scaling CPU's, the net effect has been an increase in the operating efficiency of computing devices that has led to greener devices with improved functionality. Due to their $7.4 billion in annual electricity use [1], data centers are an important topic in green computing research. In many data centers, the majority of their energy is spent on support infrastructure rather than on the computing equipment itself. Researchers have developed numerous energy aware approaches to reduce these costs [2]–[12]. Many of these improvements rely on a thermal model to predict the temperature at places of interest throughout data center facilities. These temperature predictions can be used in making proper management decisions (e.g., which server to assign an incoming workload). However, the thermal models used in these approaches only predict steady state temperatures, ignoring critical temporal aspects of thermal behavior. This not only diminishes effectiveness but also can cause service-level agreement (SLA) violations and equipment damage through unforeseen temperature spikes.

Moreover, data center technology shifts to a higher-density deployments, e.g., *containerized* or *contained-aisle* data centers, which typically exhibit lower air-to-equipment mass ratios compared to traditional big-room data centers. This lower ratio means that transient conditions tend to rise faster and be more extreme. The most prominent conditions that warrant the use of transient models are:

- The *cooling delay* problem [13], [14]: High density data centers can increase localized temperatures much faster than conventional data centers. If the cooling equipment (*chiller*) is slow at detecting and responding to the temperature rise, the servers may quickly reach high temperatures before the chiller catches up.
- *Oscillating* behavior of cooling. Most conventional vapor-compression chillers have few discrete compression states (i.e., modes), e.g., off (no compression), low (one activated compressor), and high (two activated compressors or one stronger activated compressor). With such configuration the chiller can remove heat at almost fixed rates; thus, if the data center produces heat at some rate between two cooling rates, the chiller will have to oscillate between those two rates. Since oscillations cannot be captured by steady-state models, transient models are needed for such cases.

In the aforementioned examples, transient modeling allows us to predict temperature peaks as well as to predict the portion of the time the data center spends in high and low temperatures.

Most transient cyber-physical modeling of data centers is based on CFD models. Transient CFD simulations take a considerable amount of time, typically greater than steady-state CFD simulations. Such long stretches of simulation time (which can take several hours) are not suitable for online model-based decision making. Therefore, there is a need for a lightweight model that can describe transient thermal behavior of data centers.

This paper introduces a transient thermal model that captures the temporal behavior of data centers. The model predicts three aspects of thermal behavior: *division (spatial distribution)*: how the heat produced by each computing server is split and circulated into each other server or to each chiller; *temporal distribution*: how the heat portion of one server to another is distributed over time, which also captures the *hysteresis*: how long the heat takes to travel from one unit to another.

The temporal nature of this model inherently provides a more accurate characterization of temperature dynamics for use in energy aware scheduling techniques. This transient

Developed as a part of the NSF CRI project #0855277 "BlueTool" (http://impact.asu.edu/BlueTool/). Work in this paper was also funded in part by NSF grant #0834797.

Corresponding author: Sandeep K. S. Gupta. School of Computing, Informatics and Decision Systems Engineering. Arizona State University. Tempe, AZ 85287-8809.

thermal model can be used to improve the thermal safety of the system by providing the data necessary to anticipate and predict thermal hot spots before they arise [15]. The model we have developed is proven to retain the computational speed required by many online energy aware methods.

A. Overview of the paper and of contributions

This paper is organized as follows:

Section II covers the preliminary material on the physical design of data centers on which data centers are based (§II-A), reviews the state of the art of thermal modeling (§II-B) and establishes a critical drawback of assuming steady state when using discontinuous cooling models (§II-C).

Section III formally defines our transient heat circulation model (§III-A), which states that the heat observed at a moment at the air inlet of some equipment is the cumulative integral of past produced heat as it was spread over time (Eq. 5). Then a symmetry property is presented which establishes that the curve $c(t)$ that describes how heat accumulates to a moment from all the past is a reflection of the curve $\check{c}(t)$that describe how a heat produced at a moment spreads over time toward a destination air inlet (Eq. 6) (§III-B). The section closes with a demonstration of how to combine the new transient heat circulation model with heat generation and cooling models (§III-C).

Section IV provides a methodology for calculating the parameters of our model for a given physical or modeled data center. The methodology involves inducing heat spikes at the air outlets and then observing the temperature effect at the air inlets of the servers. The methodology is applied to a CFD model of BlueCenter, the data center of the BlueTool project (http://impact.asu.edu/BlueTool/). The results show that $\check{c}(t)$ curves resemble the inverse gaussian and gamma functions.

Section V presents a validation of the model through: (i) a discrete-time simulator and a comparison of the results it produces with CFD results from [16], and (ii) a comparison of the prediction of the nre transient model with measurements from BlueCenter V-C.

II. Preliminaries

A. Data center operation

A typical data center is organized as follows: all servers are situated on a raised floor plenum. These servers are organized into *rows* that separate *aisles*. These aisles alternate between cold air intake aisles and hot air exhaust aisles. Cold air from the *computer room air conditioner* (CRAC) is supplied to the servers through perforated tiles in the raised floor. This cold air is then passed through the servers, where it absorbs heat and is ejected into the hot air aisle. Finally the hot air is returned to the AC through a ceiling intake forming the cycle.

Cooling inefficiency in this system has a number of sources, but for this paper we focus on *heat recirculation*, i.e., the return of hot air to the air inlets of computing equipment) and the notion of *hot spots*, i.e., localized increased temperatures due to heat recirculation. CRACs are incapable of provisioning varying amounts of cool air to specific, dynamic locations in the data center. As a result, much of the equipment is cooled far below red line temperatures in order to ensure that all equipment is below this threshold. This is due to hot spots, or locations where inlet temperatures are dramatically higher than the rest. Hot spots force the CRAC to over-cool the rest of the data center while trying to bring the hot spots to normal temperature. This negatively impacts CRAC efficiency not only through higher utilization, but by reducing the CRAC's performance according to the Coefficient of Performance (CoP), defined and exemplified in Figure 1.

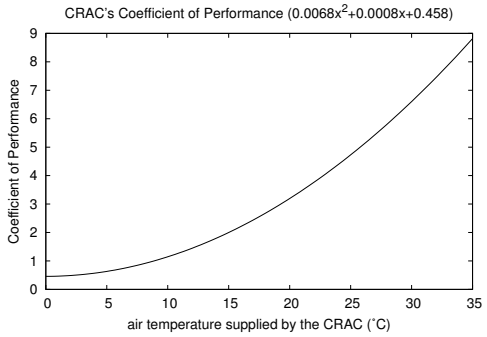

Fig. 1. Coefficient of performance (CoP) of a typical CRAC unit. The COP of the CRAC is the unit amount of heat removed for each unit of energy consumed. Formally defined: $COP = \frac{\text{Heat Removed}}{\text{Work Consumed By CRAC}}$. With a COP of 1, fifty percent of the energy expended by the data center will be the cost of cooling [18].

Knowledge of the heat recirculation, CRAC efficiency and the magnitude of over-cooling can be used to save energy in a data center by predicting how much energy would be spent for upcoming scenarios. By placing load intelligently with respect to its thermal impacts, thermal aware scheduling can allow the CRAC to act with higher efficiency by increasing the required supply temperature. To the degree that the temperature of a data center can be accurately predicted for any load, the amount of over-cooling shrinks and efficiency is gained. This highlights the potential of having a better *thermal model*, a mapping of how load effects the temperature throughout the data center. As long as the inlet of all equipment is kept below its manufacturer specified red-line temperature and system performance is not overly affected, this efficiency gain causes no adverse effects on the performance, e.g., no SLA violations in Internet applications [17].

B. Existing thermal modeling techniques (steady-state)

While the previous section motivates the need for an accurate thermal model. This section will briefly review existing methods used for thermal modeling, compare it against empirical behavior, and motivate the need for a transient model.

Thermal modeling of data centers requires three components (see Figure 2):

1) **Heat generation**: Heat generation describes how power consumed by a server is emitted as heat into the data center. A common and practical assumption is that mechanical (sonic) and electromagnetic energy produced

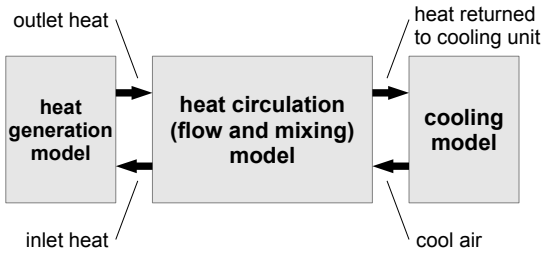

Fig. 2. Thermal modeling of a data center requires three components: a *heat generation* model, which describes how power consumed is introduced as heat into the data center, a *heat circulation* (heat flow and mixing) model, which describes how heat is distributed among the data center equipment, and a *cooling* model, which describes how cooling is introduced into the data center.

is negligible compared to the heat produced, thus:

$$\dot{Q}_{\text{computing}} \approx P_{\text{computing}} = P.$$

In a steady state, the heat rate that is output at a server's air outlet equals to the heat at the server's air inlet plus the heat produced:

$$\dot{Q}_{\text{outlet}} = \dot{Q}_{\text{inlet}} + P,$$

which can be translated into the temperature domain as:

$$T_{-} = T_{+} + \frac{P}{c_{\text{p}} f} \text{ (outlet temp = inlet temp + temp rise),}$$

where c_{p} is the specific heat of air and f is the air flow rate through the server. In computational fluid dynamics (CFD) simulation software such as the commercially available FloVENT or the open source OpenFOAM [16], heat generation is modeled as (forced) convection through the servers.

2) **Heat circulation (heat flow and mixing)**: heat circulation describes how heat is transferred around the room by the air. In CFD software, this is done by modeling the room as a 3-D grid of Navier-Stokes equations that express physics laws such as conservation of energy, momentum, mass and pressure in a defined geometry. Although CFD simulations can predict the temperature at all points of the grid, it is of practical interest to predict the temperature at a few points only, specifically at the equipment's air inlets and outlets. Fast approximate solvers such as used in [10], [18]–[20] all rely on the following *steady state assumption*: **for any constant input of heat the data center will reach a steady thermal state such that the temperature anywhere in the data center becomes invariant to time**. These approaches frequently used FloVENT for model building but once a model is constructed they heuristically approximate temperatures. For example, the work in [10], [20] constructs a *heat recirculation matrix* $\boldsymbol{D}$ that is used to predict the steady-state temperature at the air inlets of servers given the power consumption p_i of each server, plus the supplied cool temperature T_{out} from the CRAC:

$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_n \end{bmatrix} = T_{\text{out}} + \boldsymbol{D} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}. \quad (1)$$

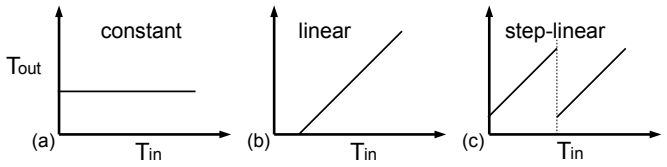

Fig. 3. Various cooling models assumed in the literature. (a) the *constant* model assumes a constant temperature of supplied cool air, regardless of the input temperature; (b) the *linear* model assumes a linear reduction of temperature with a constant shift; (c) the *step-linear* model is a set of discontinuous linear functions that best reflects empirically observed behavior [3].

3) **Cooling**: Cooling models describe how heat is removed from the air. It is analogous to heat generation. In Figure 3(a), the *constant* cooling model the outlet of the CRAC supplies cool air of a constant temperature, regardless of the input air temperature. The power removed by this model scales linearly with inlet temperature. Previous research routinely assumes a constant cooling model because given this model the steady state assumption is guaranteed to hold [19], [20].

C. Shortcomings of steady-state cooling assumptions

The constant cooling model behavior is only true for certain types of CRAC units. The most crucial difference with the research models is that the step-linear behavior does not allow for steady state solutions to occur for a majority of scheduling patterns. To understand this first consider that data centers are designed to be closed systems such that the only points of energy exchange in a system are known and considered. This is known as the *thermal isolation assumption* [21]. Figure 3(b) shows the linear cooling model which is simply a function of the form $y = mx + b$. With $m = 1$ the power removed by the CRAC is constant. As m approaches 0 this model converges to the constant model.

Consider how the total energy of a system evolves when the heat input to the system from chassis is near-constant and the heat removed from the system follows the empirical step-linear cooling model in Figure 4.

From this perspective, **in order for a data center to reach a steady thermal state the incoming energy must equal the outgoing energy**. Here we show that this is not commonly the case. The incoming energy can be derived using the sum of the power of all heat sources

$$P_{\text{in}} = \sum_{i=0}^{n} p_i.$$

P_{in} is assumed to be constant for a particular schedule. The outgoing energy can be inferred using the temperature drop

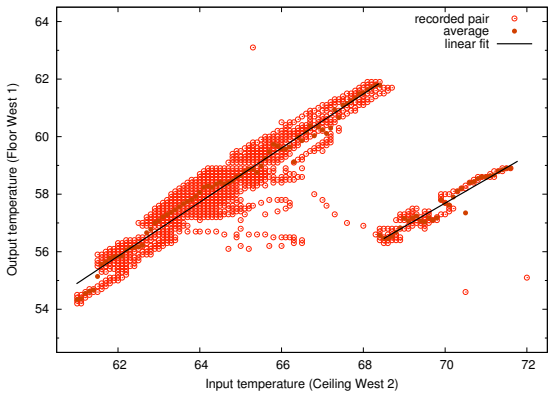

Fig. 4. Temperature sensor measurements at the CRAC inlet and outlet of a real data center over 2 weeks time [3].

TABLE I
NOMENCLATURE

Symbol	Definition
n	Number of points of interest: chassis, CRAC inlets etc
t	Time
$T_{i+}(t)$	Entering (sink) temperatures of point i at time t
$T_{i-}(t)$	Exiting (source) temperatures of point i at time t
$T_{i\infty}$	Starting steady state temperature of point i
$c_{ij}(t)$	Temporal contribution of i's temperature upon j
$\check{c}_{ij}(t)$	Temporal distribution of i's temperature upon j
w_{ij}	Heat weighting factor (at sink)
u_{ij}	Heat division factor (at source)
$\bar{Q}_{ij}$	Cumulative contributed heat from point i to j at time t
$\bar{T}_{ij}$	Cumulative contributed temperature from i to j at time t
f_i	Air flow rate through a point (server or CRAC) i
$p_i(t)$	Power at server i
c_p	specific heat of air

across the CRAC so long as the airflow mass is known. Thus an equilibrium temperature exists:

$$T_{AC} = \frac{\frac{P_\mathrm{in}}{c_\mathrm{p} f_\mathrm{AC}} + b}{1-m}, \tag{2}$$

where c_p is the specific heat of air and f_AC is the airflow rate of the AC.

With a linear cooling model it should be clear that there is a contiguous *band* of P_in for which equilibrium can be reached while keeping all equipment within their safe operating temperatures. The closer the value of m is to 1 the narrower this band becomes and the bigger the flaw of thermal models based on a steady state assumption becomes. In the case for which $m = 1$, the data center can only reach equilibrium at exactly one value of $P_\mathrm{in} = -bc_\mathrm{p} f_\mathrm{AC}$ (from Eq. 2). Since typically the slope of the step-linear cooling model is near 1 we can only conclude that **in the majority of cases, no steady state is possible**. Hence any answer that steady state thermal models provide will be false and generally speaking also not conservative. This leads to possible critical situations where red-line temperatures are exceeded as temperature fluctuates.

This is a problem for fast approximate solvers that do not support *step-linear* cooling models. This is the empirical behavior for many types of CRACs. It extracts a near constant amount of heat while the inlet temperature is within each temperature range.

In summary, steady state solvers cannot provide accurate temperature estimates in situations where thermal steady states do not exist [22]. Where inaccuracies occur steady state solvers are not conservative in their estimates [15]. When heat spikes occur on a regular basis they can cause heat induced performance throttling, equipment shutdown, and equipment damage over time. These are all costly consequences that a transient thermal model can prevent.

III. A New Transient Model For Heat Circulation

This section will provide both an intuitive and a formal definition of the transient thermal model motivated above. Intuitively, we propose that a useful estimate of the temperature at a point of interest can be provided by creating a weighted average of contributing temperatures of each source, where each contribution temperature is a weighted average of its own outlet temperatures over time. Essentially, heat added to a room is at different places at different times and as air mixes the resulting temperature averages according to the relative air masses. If we monitor the temperature at every point where it is modified the temperature everywhere else in the room can only be an averaging of measured temperatures at various times past. Therefore this model requires the thermal isolation assumption. This is primarily to avoid the need for a bias factor and so that we can constrain the model such that the sum of all weights is 1.

The other key assumption of this model is that the airflows in the data center are controlled and dominated by the fans such that the portion of air arriving everywhere over time is invariant to temperature. This ignores changes in airflow due to pressure changes caused by temperature such as hotter air rising faster (a likely cause for slight inaccuracy in our model).

A. Model definition

Our model, formally defined, uses the symbols of Table I, where source temperatures are initialized as steady state temperatures. The heart of the model is a collection of $n \times n$ ***temporal contribution curves***, $c_{ij}(t)$, each one denoting how heat arrives to the air inlet of a *sink* (*receiving*) server j from the air outlet of a *source* server i. (For notational convenience, the CRAC also is included in the ij enumeration). A c_{ij} curve conforms to the following:

$$c_{ij}(t) : (-\infty, 0] \to [0,1], \text{ with } \int_{-\infty}^{0} c_{ij}(t)dt = 1. \tag{3}$$

A key concept of the model is the ***contributing temperature*** $\bar{T}_{ij}(t)$ of a source server i to a sink server j, which is the temperature rise as affected by the history of the source server i's outlet temperature $T_{i-}(t)$ through the spreading of c_{ij}:

$$\bar{T}_{ij}(t) = \int_{-\infty}^{0} c_{ij}(\tau) T_{i-}(t+\tau) d\tau. \tag{4}$$

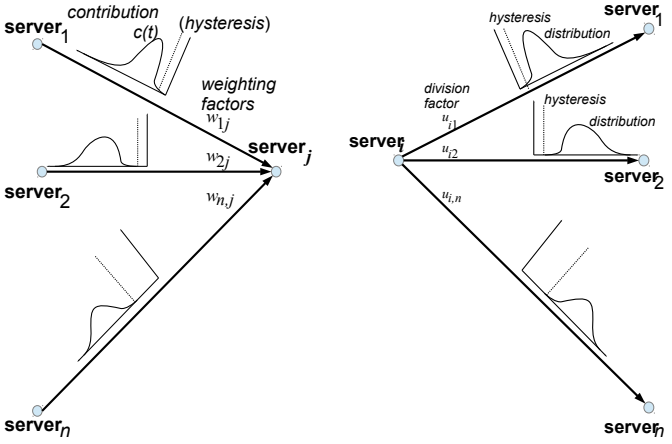

Fig. 5. (a) Conceptual depiction of the proposed transient heat circulation model. Heat at time t at a server j is a weighted sum of integrals of accumulated heat from the past. (b) The distribution function $\check{c}_{ij}(t)$ of the heat from a source server to a sink server mirrors the $c_{ij}(t)$ function of the model (Eq. 6).

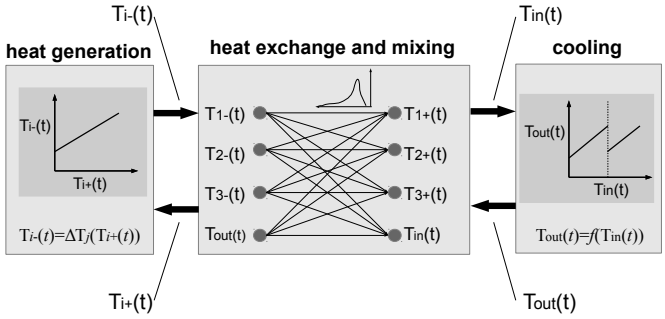

Fig. 6. An example combination of the transient heat circulation model with *linear* temperature rise (i.e., a constant heat generation) and step-linear cooling models (i.e. a two-rate heat removal).

In addition to the c_{ij} curves, the model assumes $n \times n$ ***weighting factors*** w_{ij}, which define how each contributing temperature $\bar{T}_{ij}(t)$ factors into the actual temperature $T_{j+}(t)$ at the air inlet of server j. The w_{ij} can be organized into a matrix W:

$$W = \begin{bmatrix} w_{11} & w_{21} & \cdots & w_{n1} \\ w_{12} & w_{22} & \cdots & w_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ w_{1m} & w_{2m} & \cdots & w_{nm} \end{bmatrix}.$$

The **core hypothesis** of the model is that the temperature at the air inlet of a server j is the convex weighted sum of the temperature contributions of all servers (and of the CRAC) as they accumulate over time from the past:

$$T_{j+}(t) = \sum_{i=1}^{n} w_{ij}\bar{T}_{ij}(t) = \sum_{i=1}^{n} w_{ij}\int_{-\infty}^{0} c_{ij}(\tau)T_{i-}(t+\tau)d\tau, \qquad (5)$$

where $\sum_{i=1}^{n} w_{ij} = 1, \forall j$, because of the thermal isolation assumption. Also, since a contribution temperature intuitively represents the temperature of air from a source weighted over some past time interval, it should never be hotter or colder than it has ever been:

$$\min(T_{i-}) \leqslant \bar{T}_{ij} \leqslant \max(T_{i-}).$$

B. The symmetry property

The proposed circulation model, as defined, suggests that between a contributing server and a receiving server, at any point in time t, there is heat from the contributing server that arrives from any point in the past $(-\infty, t-d)$ (see Figure 5(a)). The ***hysteresis*** parameter d expresses the delay it takes for heat to start arriving.

Since the above is true for any point in time, it is also true, in a converse manner, that heat $Q_{i-}(t)$ produced at any moment t will divide according to some ***division factors*** u_{ij} ($\sum_{j=1}^{n} u_{i,j} = 1, \forall i$) and will start arriving after time d and will continue contributing some portion of itself until $+\infty$, i.e., it will contribute in the future interval $(t+d, +\infty)$ (see Figure 5(b)). The insight of the model is that air and heat that leaves a server outlet spreads itself out in arriving at a new location. Assume that the ***heat distribution function*** is $\check{c}_{ij}(\tau)$ (this function incorporates the hysteresis d as well).

Assuming that heat is distributed according to $\check{c}_{ij}(\tau)$, irrespective of when it is generated, then the contributed heat at time t at the receiving server is:

$$\bar{Q}_{ij}(t) = \int_{-\infty}^{t} \check{c}_{ij}(-\tau)u_{ij}Q(t+\tau)d\tau.$$

Therefore, if we can easily calculate the function $c_{ij}(t)$ as:

$$c_{ij}(t) = \check{c}_{ij}(-t). \qquad (6)$$

Also, we can establish a relationship between the heat division factors u_{ij} and the weighting factors w_{ij} as follows:

$$\begin{aligned} T_{j+}(t) &= c_{\mathrm{p}} f_j Q_{j+}(t) = c_{\mathrm{p}} f_j \sum_{i=1}^{n} \bar{Q}_{ij}(t) \\ &= c_{\mathrm{p}} f_j \sum_{i=1}^{n} u_{ij} \int_{-\infty}^{0} Q_{i-}(t+\tau)c_{ij}(\tau)d\tau \\ &= \sum_{i=1}^{n} \frac{u_{ij}c_{\mathrm{p}}f_j}{c_{\mathrm{p}}f_i} \int_{-\infty}^{0} c_{\mathrm{p}} f_i Q_{i-}(t+\tau)c_{ij}(\tau)d\tau \\ &= \sum_{i=1}^{n} u_{ij}\frac{f_j}{f_i} \int_{-\infty}^{0} T_{i-}(t+\tau)c_{ij}(\tau)d\tau \\ &\Rightarrow w_{ij} = u_{ij}\frac{f_j}{f_i}. \end{aligned} \qquad (7)$$

Equations 6 and 7 define a symmetry relationship between heat distribution as it is generated at the sources, and heat contribution as it arrives at the sinks.

It should be noted that the matrix $U = \{u_{ij}\}$ of the division factors is closely related to the *heat recirculation matrix* $\boldsymbol{D}$ (Eq. 1). Matrix $\boldsymbol{D}$ describes, in a steady-state notion, how output heat from a server is split among all servers (and to the CRAC as well) and also does a conversion from power to temperature [20].

C. Combining the transient heat circulation model with heat generation and cooling

In order to use the proposed transient model on a data center, it is necessary to choose the heat generation and cooling models.

For *heat generation*, it is easy to use any model that conforms to the generic form that defines a ΔT relationship between the air inlet and the air outlet of a server: $T_{j-}(t) = \Delta T_j(T_{j+}(t))$.

For *cooling*, it is possible to use any model of cooling that defines a transfer function between T_{in} and T_{out}, including the step-linear model: $T_{out} = f(T_{in})$.

Example: Figure 6 demonstrates an example combination of the transient model with a constant heat generation model and a step-linear cooling model. The equations are:

$$T_{i-}(t) = T_{i+}(t) + a,\ i = 1 \ldots n-1, \quad \text{(constant rate)}$$

$$T_{n-}(t) = \begin{cases} T_{n+}(t) - b_1, & \text{if } T_{n+}(t) < T_{\text{thres}} \\ T_{n+}(t) - b_2, & \text{if } T_{n+}(t) \geqslant T_{\text{thres}} \end{cases} \quad \text{(step-linear)}$$

$$T_{i+}(t) = \sum_{i=1}^{n} w_{ij} \bar{T}_{ij}(t),\ \forall i. \quad \text{(Eq. 5)}$$

Our convention is to use the n^{th} index to denote the CRAC, i.e., $T_{\text{in}}(t) \equiv T_{n+}(t)$ and $T_{\text{out}}(t) \equiv T_{n-}(t)$.

The above example assumes *instantaneous* heat generation and cooling models, i.e., hot air that enters the CRAC instantly exits cooled at the outlet. Integration of our transient heat circulation model with non-instantaneous transient models is left as future work.

The next sections answer the fundamental questions of: (i) how to derive the c_{ij} curves and the weighting factors w_{ij} (§IV), and (ii) how accurate and useful the model is (§V).

IV. Model Learning

In this section, we present a methodology on how to derive the weights w and c_{ij} curves for a given data center. The methodology leverages the symmetry relationship to first derive the $\check{c}_{ij}$ curves and u_{ij} factors from CFD simulations, and then compute the $\check{c}(t)$ and w_{ij}.

A. Methodology for yielding w_{ij} and c_{ij}

A series of CFD simulations with high temporal resolution is required (in this case, we used the OpenFOAM CFD solver). *Once the model is learned, the CFD solver would no longer be required*. Furthermore, a CFD solver can be done away with entirely if the data can be obtained from sensors in the data center.

To obtain the $\check{c}(t)$ curves we first generate n CFD simulations which start with a steady state of T_{const} temperature and then having a single location at a time produce a considerable *temperature spike*. The simulation should include a parameter specifying that the outlet of all servers (other than the previously specified short period of extreme heat) should act as "heat sinks", that is emit a constant temperature $T_{i-}(t) = T_{\text{const}}$, regardless of their inlet temperatures $T_{i-}(t)$. Heat sinking ensures that eventually the heat introduced by the temperature spike will dissipate and the system will revert to the pre-spike state. This procedure is performed in n separate experiments, one for each server and CRAC, to obtain their respective curves.

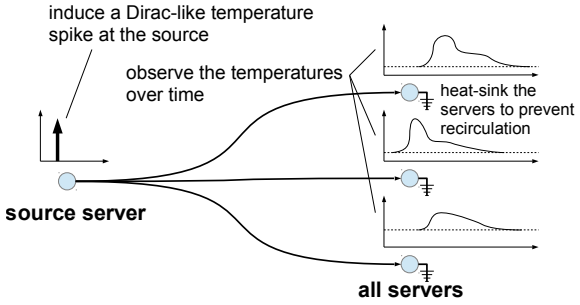

Fig. 7. The methodology of yielding the $\check{c}_{ij}$ curves involves inducing a Dirac-like heat spike at the source server i, one source i at a time, and then observing the temperatures over time at the sink servers. The u_{ij} factors are computed by dividing the areas of each curve by the sum of all the areas at a sink.

The temperature curves $T_{j+}(t)$ that are observed at the air inlets are used to derive the $c_{ij}(t)$ and w_{ij} as follows:

1) Remove the bias temperature of the pre-spike steady state for the curves.
2) (Calculate u_{ij}) For each experiment i, divide the area of each temperature curve by the sum of the areas of the curves:
$$u_{ij} = \frac{\int_0^\infty T_{j+}(\tau)d\tau}{\sum_{j=1}^{n} \int_0^\infty T_{j+}(\tau)d\tau}, \text{ for each experiment } i.$$
3) Calculcate w_{ij} using Eq. 7.
4) (Calculate $\check{c}_{ij}(t)$) For each experiment i, normalize each curve to a unit area:
$$\check{c}_{ij}(t) = \frac{T_{j+}(t)}{\int_0^\infty T_{j+}(\tau)d\tau}.$$
5) Calculate $c_{ij}(t)$ using Eq. 6.

B. Example CFD experiment with preliminary results

We provide an example application of the methodology on the CFD model of our BlueCenter (8), a data center with 288 servers organized in two rows of 4 racks each. Each rack contains 36 1U rack-mount servers. To keep a relatively low length of the experiment, we bundle the servers to *chassis*, such that each chassis contains 12 servers. Although this bundling has an effect on the resolution granularity of the results, it does not effect the demonstration of the methodology. In this application of our model derivation we have modified our CFD model of BlueCenter to contain an alternating pattern of empty chassis. These empty chassis allow air to flow freely between the cold and hot aisles. Thus, each of the 8 racks contain 3 chassis with servers and 4 empty chassis.

Figures 9 show preliminary obtained results for $\check{c}(t)$ curves and w weights respectively. Verification of the results is ongoing. The results show that heat does indeed spread out over time, and that the spreading follows curves that seem to be combinations of one or more Inverse Gaussian (or Gamma) functions. This can be attributed to the heat following multiple turbulent paths with different lengths each. Additionally, the

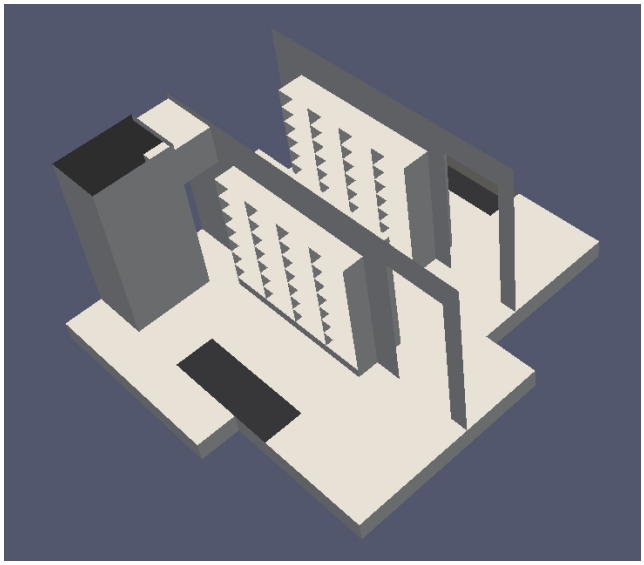

Fig. 8. 3-D rendering of the physical layout of the model data center using OpenFOAM. This CFD model is used to apply the methodology in §IV-B to yield $c_{ij}(t)$ and w_{ij}.

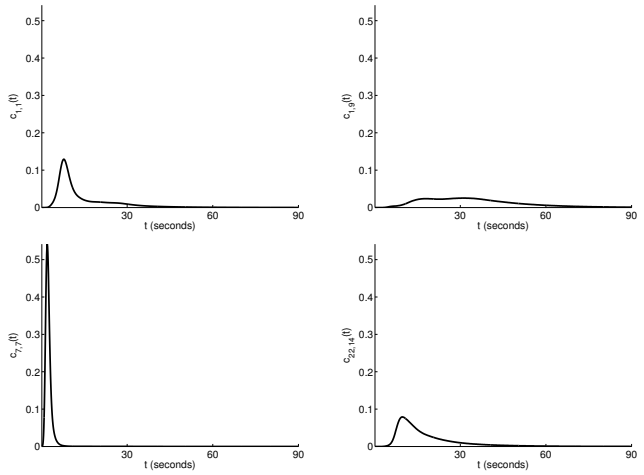

Fig. 9. Sample c_{ij} curves as produced from the CFD experiments in §IV-B.

quick and high peak of $c_{7,7}(t)$ in Fig. 9 shows how effectively $\check{c}(t)$ curves can display thermal inefficiencies such as the quick heat recirculation allowed by keeping empty chassis between hot and cold aisles.

V. Validation

We have implemented this model and created a discrete time, step-wise simulator. In Section V-A we show an intuitive case to check the correctness of the simulator. In Section V-B we compare the results of our simulator to those of a modified CFD solver. These results validate our simulator. In Section V-C we simulate a larger, more complex data center using the transient model.

A. Empirical check for correctness of the new transient model

As a simple proof of correctness of the simulator implementing our model consider the connected system shown in

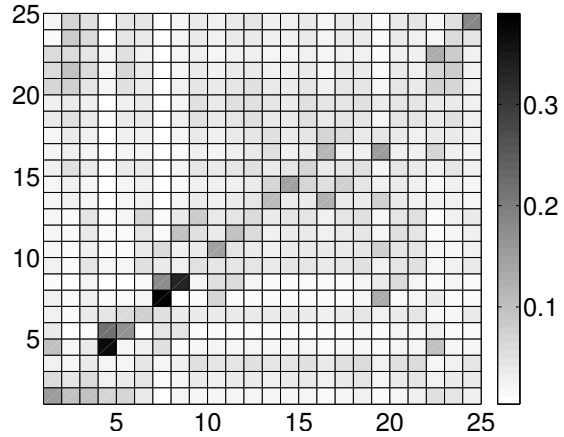

Fig. 10. The matrix of w_{ij} weighting factors as produced from the CFD experiments in §IV-B.

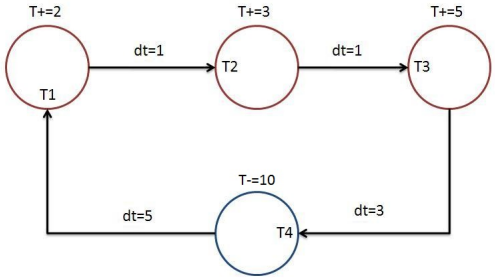

Fig. 11. Basic simulation setup for the discrete-time simulator in §V-A. Every airflow arrives at exactly one other place. Air passing through each node changes by the amount listed above it. Air leaving a node takes dt time to arrive at the next node.

Figure 11. For Figure 13 and Figure 14 a simple unit square c_{ij} curve is used that is 1 between $-dt-1$ and $-dt$ such as the curve in Figure 12. Deriving c_{12} from Figure 11, $c_{12}=1$ from -2 to -1 and 0 everywhere else. The full matrix derived from the layout is shown in Figure 12.

The weights are similarly derived from the layout. The $T_{-\infty}$ are arbitrarily set to 0.

$$W=\begin{bmatrix}0&0&0&1\\1&0&0&0\\0&1&0&0\\0&0&1&0\end{bmatrix},\quad T_{-\infty}=\begin{bmatrix}0\\0\\0\\0\end{bmatrix}.$$

Intuitively this system must behave periodically at first. The period duration depends heavily on this matrix and the time step. As a minimum bound on the period consider that the minimum circulation time is 10. As a maximum bound, consider that the last of the air gets through each location 1 time step later because of the shape of the c_{ij} curve used. Thus the maximum frequency is 15, which is the minimum of 10 plus 4 for the span of all c_{ij} curves in sequence plus 1 time step for measurement. In Figure 13 for which $dt=1$ the total period is 15. This reaches the maximum because of the sampling time is aligned with the c_{ij} curve shapes. With an

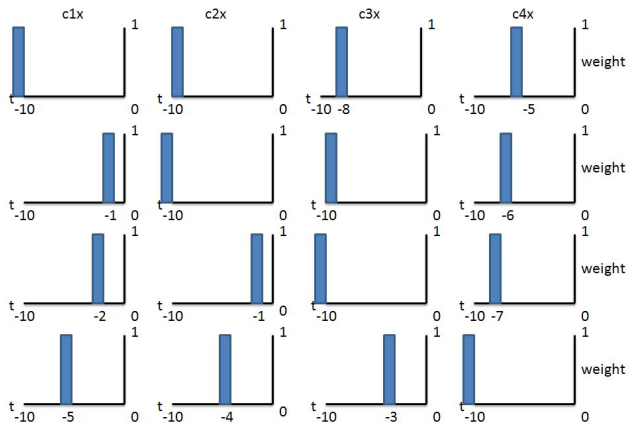

Fig. 12. Temporal Contribution Curves for the simple data center.

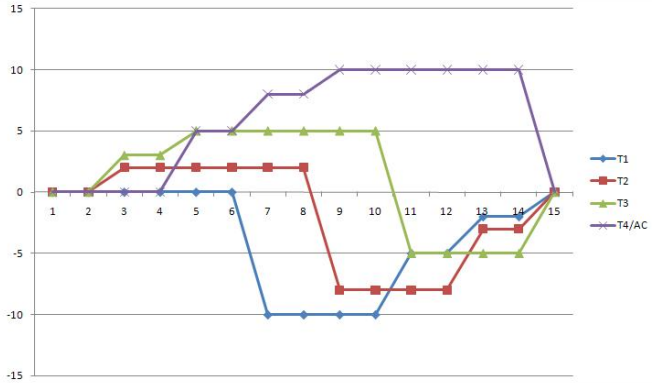

Fig. 13. The first period of the simple data center shown in Figure 11 with time step 1 and unit square $c(t)$ curve.

infinitesimal c_{ij} curve and a continuously evaluated simulation the period would approach 10.

Next consider whether this system will converge. At the next iteration, the system will no longer be acting on its $T_{-\infty}$ temperatures as it was last cycle. Rather the outlet of each point is 0 at time 15 because they each heat or cool by an amount equal to the inverse of their current temperature. So whether the system converges is a function of whether these outlet temperatures are maintained for long enough to match the original $T_{-\infty}$ signature during the the span of the c_{ij} curve. For this reason when $dt = 1$ the system does not converge. For any $dt < 1$ the system will experience some heat mixing causing the outlet temperatures to maintain the $T_{-\infty}$ signature for some amount of time less than the span of the c_{ij} curves. Figure 14 confirms this behavior. As predicted, as the timestep value is lowered the period shrinks and the system begins to converge.

Figure 15 shows the effect of various c_{ij} curve shapes. Note that while increasing the span of the c_{ij} curve increases the period of oscillation it also causes faster convergence. This is due to air mixing more in each period because the $c(t)$ curves overlap more from various points in time, i.e. the unit square $c(t)$ curve (Figure 15(a)) overlaps and mixes with hit produced

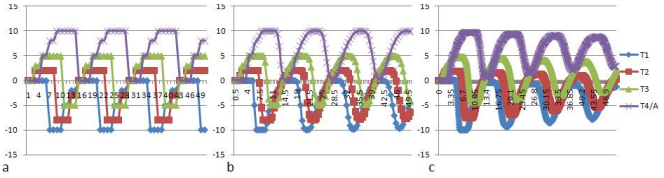

Fig. 14. Effects of time step on simulation result. (a) $dt = 1$ (b) $dt = 0.5$ (c)$dt = 0.05$.

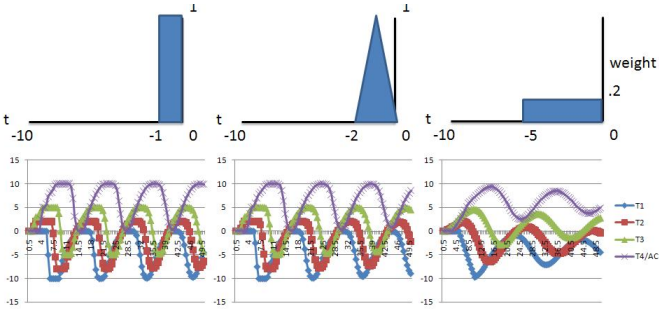

Fig. 15. Effects of $c_{ij}(t)$ curve shape on simulation result. (a) unit square (b) triangle, (c) rectangle.

over a one time unit range whereas the rectangle $c(t)$ curve of 5 time units (Figure 15(b)) overlaps and mixes with heat produced over five time units.

B. Comparison against CFD simulation

To validate these results we have implemented the setup of Section V-A in a modified version of OpenFOAM. The physical setup is a 10 meter long rectangle of a .1 meter by .1 meter square vent with the 3 chassis and 1 CRAC inserted at distances apart from each other proportional to the initial delay indicated in Figure 11. Our simulation used a $dt = 0.1$. The results of the CFD simulator strongly agree with our simulator (see Figure 16). Between the models, there is a median temperature difference of 0.3505°C and average difference of 0.8106°C. We attribute this difference to c_{ij} curve selection.

The primary advantage of our simulator over the CFD simulator is that our results can be obtained in less than 1 second as opposed to the approximately 1 hour solution time of the CFD simulation. This enables our model to be used as a thermal map by data center administrators provided that the parameters of our model are learned through the procedure described in Section IV.

C. Simulation on a large data center

In the transient model simulation, we use our model to predict CRAC inlet temperature as utilization is varied for all servers. Beginning at time t=0, all 144 servers are running at idle power, producing enough heat to stabilize the CRACs inlet temperature. At t=264 seconds, all servers are ramped up to 100% utilization. As this occurs, it is visible that the CRACs inlet temperature spikes to 48°C. The step-linear cooling model of the CRAC is apparent at t=300 seconds as the CRAC switches modes and the inlet temperature cools

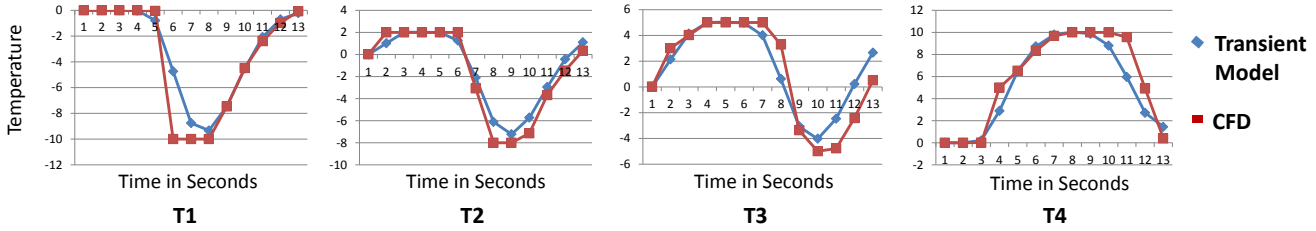

Fig. 16. Temperature over time results compared for each node between Transient Model and CFD calculations.

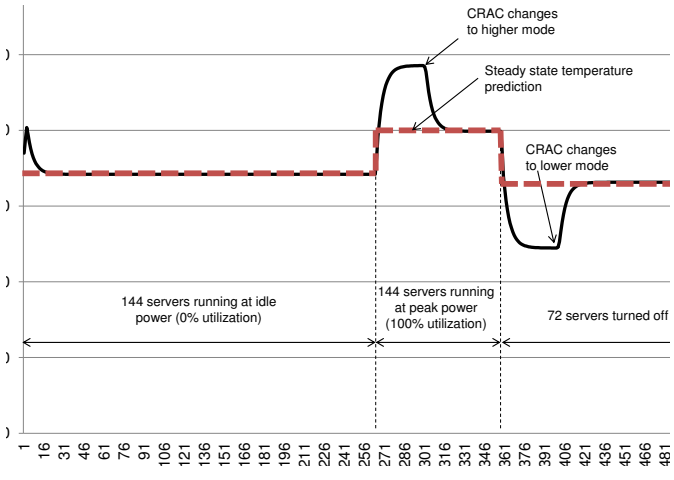

Fig. 17. Simulation of Transient Model's predicted CRAC inlet temperature, versus the prediction of the system's steady state model.

to 40 °C. At t=358 seconds, half of the servers switch off, causing inlet temperatures to drop. Finally, at t=400 seconds the CRAC reverts modes and brings the system back to a converged temperature of approximately 33 °C.

Figure 17 highlights the deficiencies of steady state thermal models. From t=264 through 358 the converged state of the system is 40 °C, and is thus what a steady state model would predict. This prediction misses the near 10 °C heat spike that occurs during the interim as the CRAC switches cooling settings according to its cooling model. This spike shows the temporal deficiency of steady state thermal models and the necessity, while using them, to make conservative estimates of temperatures.

VI. Conclusions

In this paper we motivated the need for a light-weight transient heat circulation model to be used for quick evaluation of workload management decisions. The proposed model (§III-A) accurately characterizes thermal behavior as it varies over time. The addition of a time dimension allows data center operators to model the periodic temperature fluctuations implied by discontinuous, step-linear cooling models. While it maintains a good level of accuracy, its speed is rather attractive: discrete-time simulations take only a few seconds to complete, as opposed to several hours of CFD computations (§V-A).

The core hypothesis of the model is that heat, as it is produced at the computing equipment, is spread out as it follows the air flow in the data center, and it arrives with a certain distribution at the air inlets of either other computing equipment or of the CRAC. Preliminary simulations in §IV show that the shape of the distribution curves resembles a combination of inverse-Gaussian functions. The splitting of the produced heat among the equipment is captured by a matrix of division factors u_{ij}, which is closely related to the *heat recirculation matrix* [20].

Research on this model is congruent with the objectives of the BlueTool project (http://impact.asu.edu/BlueTool), whose purpose is to develop and provide evaluation and validation tools for green data center design. Operating fast yet accurate models to produce results is a cornerstone factor for the success of the project.

References

[1] U.S. Environment Protection Agency, "Report to congress on server and data center energy efficiency," 2007. [Online]. Available: http://www.energystar.gov/ia/partners/prod_development/downloads/EPA_Datacenter_Report_Congress_Final1.pdf

[2] Z. Abbasi, G. Varsamopoulos, and S. K. S. Gupta., "Thermal aware server provisioning and workload distribution for internet data centers," in *ACM International Symposium on High Performance Distributed Computing (HPDC10)*, June 2010, pp. 130–141.

[3] A. Banerjee, T. Mukherjee, G. Varsamopoulos, and S. K. S. Gupta, "Cooling-aware and thermal-aware workload placement for green HPC data centers," in *International Conference on Green Computing Conference (IGCC2010)*, Aug. 2010, pp. 245–256.

[4] T. D. Boucher, D. M. Auslander, C. E. Bash, C. C. Federspiel, and C. D. Patel, "Viability of dynamic cooling control in a data center environment," in *(IEEE) Inter Society Conference on Thermal Phenomena*, 2004, pp. 593–600.

[5] J. Donald and M. Martonosi, "Techniques for multicore thermal management: Classification and new exploration," *SIGARCH Comput. Archit. News*, vol. 34, no. 2, pp. 78–88, 2006.

[6] M. Fontecchio, "Companies reuse data center waste heat to improve energy efficiency," May 2008. [Online]. Available: http://searchdatacenter.techtarget.com/news/article/0,289142,sid80_gci1314324,00.html

[7] T. Heath, B. Diniz, E. V. Carrera, W. M. Jr., , and R. Bianchini, "Energy conservation in heterogeneous server clusters," in *In Proceedings of the Symposium on Principles and Practice of Parallel Programming (PPoPP)*, Chicago, IL, USA, June 2005. [Online]. Available: http://doi.acm.org/10.1145/1065944.1065969

[8] T. D. Boucher, D. M. Auslander, C. E. Bash, C. C. Federspiel, and C. D. Patel, "Viability of dynamic cooling control in a data center environment," *Journal of Electronic Packaging*, vol. 128, no. 2, pp. 137–144, 2006. [Online]. Available: http://link.aip.org/link/?JEP/128/137/1

[9] Q. Tang, S. K. S. Gupta, D. Stanzione, and P. Cayton, "Thermal-aware task scheduling to minimize energy USAge for blade servers," in *2nd IEEE Int'l Dependable, Autonomic, and Secure Computing (DASC'06)*, September 2006, pp. 195–202.

[10] Q. Tang, S. K. S. Gupta, and G. Varsamopoulos, "Thermal-aware task scheduling for data centers through minimizing heat recirculation," in *IEEE Cluster*, Sept. 2007, pp. 129–138.

[11] T. Mukherjee, A. Banerjee, G. Varsamopoulos, and S. K. S. Gupta, "Model-driven co-ordinated management of data centers," *(Elsevier) Computer Networks, Special Issue on Managing Emerging Computing Environments*, vol. 54, no. 16, pp. 2869–2886, Nov. 2010.

[12] G. Varsamopoulos, A. Banerjee, and S. K. S. Gupta, "Energy efficiency of thermal-aware job scheduling algorithms under various cooling models," in *International Conference on Contemporary Computing IC^3*, Noida, India, Aug. 2009, pp. 568–580. [Online]. Available: http://impact.asu.edu/thesis/Varsamopoulos2009-cooling-models.pdf

[13] S. Gupta, G. Varsamopoulos, A. Haywood, P. Phelan, and T. Mukherjee, "Bluetool: Using a computing systems research infrastructure tool to design and test green and sustainable data centers," in *Handbook of Energy-Aware and Green Computing*, 1st ed., I. Ahmad and S. Ranka, Eds. Chapman & Hall/CRC, 2012.

[14] E. K. Lee, I. Kulkarni, D. Pompili, and M. Parashar, "Proactive thermal management in green datacenters," *The Journal of Supercomputing*, vol. 60, no. 2, pp. 165–195, 2012.

[15] L. Marshall, "Transient CRAC failure analysis," Applied Math Modeling Inc., 2010. [Online]. Available: http://www.coolsimsoftware.com/wwwroot/LinkClick.aspx?fileticket=ap5wEZJbecQ%3D&tabid=189

[16] Z. T. Hrvoje Jasak, Aleksandar Jemcov, "OpenFOAM: A C++ library for complex physics simulations," in *International Workshop on Couplted Methods in Numerical Dynamics IUC*, September 2007, pp. 47–66. [Online]. Available: http://powerlab.fsb.hr/ped/kturbo/openfoam/papers/CMND2007.pdf

[17] T. Mukherjee, A. Banerjee, G. Varsamopoulos, S. K. S. Gupta, and S. Rungta, "Spatio-temporal thermal-aware job scheduling to minimize energy consumption in virtualized heterogeneous data centers," *Computer Networks*, vol. 53, no. 17, pp. 2888–2904, December 2009.

[18] J. Moore, J. Chase, and P. Ranganathan, "Making scheduling "cool": temperature-aware workload placement in data centers," in *Proceedings of the annual conference on USENIX Annual Technical Conference*, ser. ATEC '05. Berkeley, CA, USA: USENIX Association, 2005, pp. 5–5. [Online]. Available: http://dl.acm.org/citation.cfm?id=1247360.1247365

[19] ——, "Weatherman: Automated, online and predictive thermal mapping and management for data centers," in *Proceedings of the 2006 IEEE International Conference on Autonomic Computing*, ser. ICAC '06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 155–164. [Online]. Available: http://dx.doi.org/10.1109/ICAC.2006.1662394

[20] Q. Tang, S. K. S. Gupta, and G. Varsamopoulos, "Energy-efficient thermal-aware task scheduling for homogeneous high-performance computing data centers: A cyber-physical approach," *IEEE Transactions on Parallel and Distributed Systems, Special Issue on Power-Aware Parallel and Distributed Systems,* conditionally accepted, vol. 19, no. 11, pp. 1458–1472, Nov. 2008. [Online]. Available: http://dx.doi.org/10.1109/TPDS.2008.111

[21] J. Moore, R. Sharma, R. Shih, J. Chase, C. Patel, and P. Ranganathan, "Going beyond CPUs: The potential of temperature-aware data center architectures," in *First Workshop on Temperature-Aware Computer Systems*, June 2004. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.87.6147

[22] A. Kubilay, S. Zimmermann, I. Zinovik, B. Michel, and D. Poulikakos, "Compact thermal model for the transient temperature prediction of a water-cooled microchip module in low carbon emission computing," *Numerical Heat Transfer*, vol. A, no. 59, pp. 815–835, 2011.