Energy savings in data centers: A framework for modelling and control of servers' cooling ⋆

Riccardo Lucchese * **Jesper Olsson** * **Anna-Lena Ljung** ** **Winston Garcia-Gabin** *** **Damiano Varagnolo** *

* *Department of Computer Science, Electrical and Space engineering, Luleå University of Technology, Sweden.*
** *Department of Engineering Sciences and Mathematics, Luleå University of Technology, Sweden.*
*** *ABB Corporate Research, Västerås, Sweden.*

Abstract Aiming at improving the energy efficiency of air cooled servers in data centers, we devise a novel control oriented, nonlinear, thermal model of the servers that accounts explicitly for both direct and recirculating convective air flows. Instrumental to the optimal co-design of both geometries and cooling policies we propose an identification methodology based on Computational Fluid Dynamics (CFD) for a generic thermal network of m fans and n electronic components. The performance of the proposed modelling framework is validated against CFD measurements with promising results. We formalize the minimum cooling cost control problem as a polynomially constrained Receding Horizon Control (RHC) and show, in-silico, that the resulting policy is able to efficiently modulate the cooling resources in spite of the unknown future computational and electrical power loads.

Keywords: Server fan control, Energy saving, Thermal network modelling, Air cooling, Data center.

1. INTRODUCTION

The increasing societal requirements for data storage and analysis are served by an increasing number of datacenters, and this induces the need of operating these notoriously energy-hungry systems in energetically friendly ways. Only in Europe, in fact, datacenters consumed on average approximately 11.8GW in 2013, i.e., about 3% of the total electricity produced across the continent, and these figures have been steadily growing at a yearly rate of 4%, with an associated generation of CO_2 comparable to the aviation industry [1].

Aiming at making datacenters more energy efficient, here we focus on how to improve the efficiency of their thermal cooling systems (known to account for up to 40% of the total energy consumption in data centers [2]).

To define more precisely the problem that we consider, we note that the primary control objective of a datacenter is to provide a given set of Information Technology (IT) services with high overall energy efficiency. The aim is then to maintain specified Quality of Service (QoS) levels while decreasing the total electrical power consumption and/or reusing (after an opportune harvesting) the heat produced by the IT elements.

A data center has three main infrastructures: the IT system, the Cooling Technology (CT) system and the power distribution system. From an organizational point of view, instead, the operations happen on three different layers: single server level, rack level and datacenter level. At all levels, electrical power is converted into IT services and heat that has to be rejected by the CT (consuming more power) [3]. Incrementing the energy efficiency of the system can be done at any one of these three levels; nonetheless here we focus on the lowest one, and thus consider the problem of minimizing the energy consumption in a single server.

Literature review: the existing strategies for diminishing the energy usage in a server while maintaining a specified QoS can be divided into:

- reducing the power consumption by scheduling opportunely the network components, e.g., by turning them on only when needed [4,5];
- regulating the power dissipated by the components through Dynamic Voltage and Frequency Scaling (DVFS) techniques, i.e., diminishing the chip's frequency or voltage when it is not fully utilized and increasing it when the demand is higher [6,7];
- optimizing the power consumption of the server fans while respecting the thermal constraints on the internal components (CPUs, memory banks, etc.).

In this paper we focus on the last strategy and aim at finding the fans control law that provides the minimum cost air flow satisfying the thermal constraints on the server components. We are indeed interested in this problem since both the state of the art and (specially) the state of the practice in server fan control often results in

⋆ This work is supported by the Celtic Plus project *SENDATE-Extend* (C2015/3-3), the LTU Smart Machines and Materials project *Control Oriented Computational Fluid Dynamics*, and the Swedish research council Norrbottens Forskningsråd project *DISTRACT*.

over-provisioning the air flows, and thus in a higher energy consumption than necessary.

The state of the practice indeed employs basic fan control schemes such as simple on-off strategies according to some thresholds, and tend to disregard variable fan speed strategies.

A big part of the literature dedicated to the servers' fans control focuses on Proportional Integral Derivative (PID) strategies: e.g., [8] proposes a control strategy based on an adaptive PID controller, while [9] considers a PID controller that is trained on-line by a neural network that assumes a linear lumped parameter model of the thermal network inside the server. We also mention [10], that proposes to solve the thermal control problem by means of cascades of PI controllers. PID controllers improve the power consumption with respect to the state of the practice, but still lead to over-provisions since the set-points of the air flows have to be over-estimated to satisfy the thermal constraints on-line.

Other approaches consider instead either Multiple Input Multiple Output (MIMO) minimum cost control strategies, as in [11], bang-bang schemes, as in [12], and finite state machines deciding to actuate jointly or separately various actuators in a MIMO fashion [13,14].

All these techniques nonetheless disregard: 1) modelling the effects of other components on the board due to warm flow recirculation, 2) accounting for potentially available IT load forecasts, 3) considering the unavoidable uncertainties in the models of the controlled physical system.

Statement of contributions: Model Predictive Control (MPC) is now considered a standard tool for solving model-based, complex multivariable optimal control problems where the dynamics is subject to state and actuator constraints. Here we thus tailor this control strategy to the specific needs of datacenters by:

- proposing a novel model of the thermal dynamics at the server level that accounts for nonlinear direct and recirculation flows inside a generic server's enclosure;
- describing how to identify the model above by leveraging CFD tools;
- validating the predictive capabilities of the so-identified thermal model by means of opportune experiments;
- deriving and testing a dedicated deterministic MPC strategy based on the model above and accounting for forecasts of future IT and thermal loads.

In this way we address problems 1) and 2) above, while keeping 3) as a future research issue.

Structure of the manuscript: Section 2 develops a novel parametric, nonlinear, modelling framework aimed at the thermal dynamics of server enclosures. Section 3 details what are the issues in identifying the parameters of the proposed dynamics and develops a least square approach that exploits steady state measurements from CFD trials. Section 4 formalizes our MPC strategy in terms of a minimum cost optimization problem with polynomial constraints. In Section 5 we show the performance of our modelling framework using validation data from a CFD campaign and demonstrate sample in-silico trajectories of a controlled server. Finally, Section 6 collects final thoughts and future directions.

Symbol	*Description*	*Dim.*	*Type*
t	continuous time	$\mathbb{R}$	index
k	discrete time	$\mathbb{N}$	index
n	number of IT components	$\mathbb{N}$	index
m	number of fans	$\mathbb{N}$	index
$\boldsymbol{x}^c$	temperatures of the n IT components	$\mathbb{R}^n$	signal
$\boldsymbol{x}^f$	temperatures of the flows crossing the n IT components	$\mathbb{R}^n$	signal
x^i	temperature of the server's inlet air	$\mathbb{R}$	signal
$\boldsymbol{u}$	set-points for the mass flows from the m fans	$\mathbb{R}^m$	signal
$\boldsymbol{p}$	power consumption of the n IT components	$\mathbb{R}^n$	signal
$\boldsymbol{f}$	total flows crossing the n IT components	$\mathbb{R}^n$	signal
Δ	sampling period	$\mathbb{R}$	parameter
$\boldsymbol{b}$	electrical power to heat conversion rates	$\mathbb{R}^n$	parameter
$\boldsymbol{h}$	heat exchange rates of the n IT components	$\mathbb{R}^n$	parameter
R	matrix of the conduction rates among the various n components	$\mathbb{R}^{n\times n}$	parameter
ρ	vector of the parasitic losses thermal resistances of the n IT components	$\mathbb{R}^n$	parameter
Λ^d	direct air mass flows maps (i.e., $\Lambda^d_{(i,j)}$ considers the contribution of the direct flow from fan j onto the IT component i)	$\mathbb{R}^{n\times m}$	map
Λ^r	recirculation air mass flows maps (i.e., $\Lambda^r_{(i,j)}$ considers the contribution of the recirculation flow from IT component j onto the IT component i)	$\mathbb{R}^{n\times n}$	map
$\star_{(j)}$	j-th coordinate of the vector $\star$ or j-th row of the matrix $\star$		
$\star_{(i,j)}$	i,j-th element of the matrix $\star$		

Table 1. Notation and most important formulas used throughout the manuscript.

2. THERMODYNAMICS INSIDE THE SERVER'S ENCLOSURE

High-end servers such as the one in Figure 1 are packaged in standardized enclosures and stacked within racks to facilitate their access to networking, power and cooling resources. A single rack can host from tens up to hundreds of server enclosures. Groups of racks are further organized in rows (within the computer hall) to form cold and hot aisles: cool air from the cold aisle is supplied at the front face of the rack, travels through the servers' enclosures and is eventually rejected at a higher temperature through the back side to the hot aisle.

At the single enclosure (i.e., server) level a convective flow is forced using local cooling fans: the cool air that enters the server's inlet at a temperature x^i (for all practical purposes constant along the air inlet) travels then the enclosure, absorbs the heat energy dissipated by the electronics, and then exits from the server's outlet.

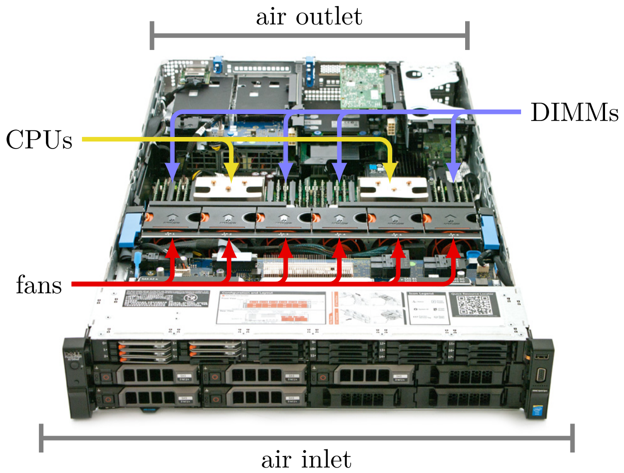

Figure 1. A Dell R730xd Power-Edge server. This platform develops on a rectangular base that is 44.40cm wide, 68.40cm long and 8.73cm high, contains six main cooling fans, 2 Central Processing Units (CPUs), 16 banks of Dual In-line Memory Modules (DIMMs) organized in four groups, companion chips and connectivity support electronics, a power supply unit and up to 12 mass storage docks

In this work we propose a novel generic modelling framework to describe the temperature dynamics of the electrical components inside the server, and specifically model the nonlinear convective effects of direct and recirculation air flows. Due to format constraints, we tailor our discussion to standard rack-unit enclosures disregarding other enclosure types such as blades, even if the thermal behavior of these IT devices can be captured within the same framework. Our eventual aim is to use the proposed model of the thermodynamics to regulate the temperatures of the electronics within safe limits while minimizing the cost of operating the fans and in spite of varying environmental conditions (e.g., temperature and humidity in the computer room) and computational load (and thus the total heat load inside the enclosure).

In our control oriented model of a server, the state is given by the temperatures of those electrical components within the enclosure that dissipate the largest amount of power, namely the CPUs and the DIMMs. The manipulable variables are the mass flows produced by the fans, affecting the amount and the distribution of the cooling resources within the enclosure. Moreover, in our model of a server we consider two exogenous inputs: 1) the inlet air temperature x^i, and 2) the electrical power consumption of the components. In practice, the inlet temperature x^i is slowly varying in time and can be measured; We thus assume that x^i is a known deterministic quantity that remains constant on a sufficiently short prediction horizon. We moreover assume that the power consumption of the electronics is indirectly set by the job scheduling controllers that allocate the IT requests. Since these schedulers operate at a datacenter level, in our framework we can consider their output as an external reference imposed on the single server.

Our approach is then to consider a minimum cost MPC strategy that computes the air flows that should be provided by the servers' fans, and, instrumentally to it, we introduce a novel modelling framework for the enclosure's thermal dynamics. The functional structure of the latter dynamics builds on top of the following ingredients:

1) A static flow model that describes how the cool air moves from the fans onto the components and how the warm air recirculates inside the enclosure (Section 2.1);
2) A dynamic thermal model that describes how the temperature of a single component depends on the intensity and temperature of the flows crossing it and on the electrical power that is currently consuming (Section 2.3).

The models defining the behavior of our generic network of m-fans and n-components will then include the following variables:

- as *states*, the n temperatures of the heat-dissipating IT components $\boldsymbol{x}^c = \left[x_1^c \ \ldots \ x_n^c\right]^T$ and the n temperatures of the air flows through each of these components $\boldsymbol{x}^f = \left[x_1^f \ \ldots \ x_n^f\right]^T$;
- as *exogenous inputs*, the n unknown (average) powers dissipated by each component during the sampling interval $\boldsymbol{p} = \left[p_1 \ \ldots \ p_n\right]^T$ and the temperature of the air inlet x^i;
- as *manipulable inputs*, the m air mass flows produced by the m fans within the server $\boldsymbol{u} = \left[u_1 \ \ldots \ u_m\right]^T$, that in their turn determine the total cooling air flows through the n components $\boldsymbol{f} = \left[f_1 \ \ldots f_n\right]^T$.

2.1 A control-oriented static air flow model

Minimizing the power used for rejecting the heat produced by the CPU and the DIMMs modules requires an accurate but control-oriented model of how the cool air moves from the fans onto the components and how the warm air recirculates inside the enclosure. We thus need a model that is simple enough to lead to a numerically tractable MPC problem but at the same time captures with a sufficient degree of approximation the essential physical behavior of the system.

Consider then that fluids, especially compressible ones as gases, follow dynamical laws; nonetheless in our servers-cooling framework we assume a static model, with the rationale behind this simplification being that the dynamics of the air flows inside the servers' enclosures are much faster than the thermal dynamics that we want to control. We thus safely exploit separation of time scales considerations and approximate the faster manifold corresponding to the fluid dynamics as always in steady state.

We then model the air flux blowing on the n components in the server to be the sum of an opportune polynomial transformation of $\boldsymbol{u}$ to account for the *direct* fluxes from

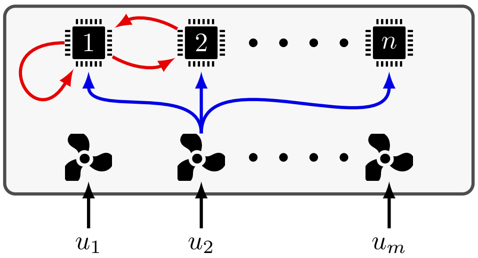

Figure 2. The flow model proposed in this manuscript accounts for: 1) direct and cold air flows (in blue) that are pushed from the fans onto the components, and 2) recirculating warm air flows (in red) that affect the temperature of neighboring components and, in this way, couple the thermal dynamics of the various individual elements.

the m fans to the n components, plus a polynomial transformation of the air flows $\boldsymbol{f}$ to account for *recirculation* effects within the server; graphically, thus, we consider the situation depicted in Figure 2. Compactly, we write the total flow $\boldsymbol{f}$ crossing the n components as the sum of the two previous contributions, i.e., as

$$\boldsymbol{f} = \Lambda^d[\boldsymbol{u}] + \Lambda^r[\boldsymbol{f}], \tag{1}$$

where Λ^d and Λ^r in (1) are opportune polynomial functions (described in more detail in Appendix A). We stress that the flow from the fans is "cold" while the flow due to recirculation is "warm" and acts as a disturbance coupling the individual dynamics.

With this model the total flow $\boldsymbol{f}$ can then be computed as a solution to the $2n$ polynomial constraints

$$(I - \Lambda^r)[\boldsymbol{f}] = \Lambda^d[\boldsymbol{u}], \quad \boldsymbol{f} \succeq 0, \tag{2}$$

where I denotes the identity operator, i.e., $I[\boldsymbol{f}] = \boldsymbol{f}$.

Model (2) generalizes different existing models by capturing:

i) those models that assume no *warm* recirculation flows, obtainable when taking $\Lambda^r = 0$;
ii) those models that preserve total mass-flows, obtainable when Λ^d and Λ^r are both row-stochastic matrices;
iii) those models that do not preserve total mass-flows, in the remaining settings.

Interestingly, when Λ^r is a linear operator (i.e., a matrix), then the total flow $\boldsymbol{f}$ is just a polynomial transformation of the fans operating conditions $\boldsymbol{u}$, since in this case (for relevant sufficient conditions on the non-singularity of Λ^r see Appendix 5 in [3]).

$$\boldsymbol{f} = (I - \Lambda^r)^{-1} \Lambda^d[\boldsymbol{u}]. \tag{3}$$

2.2 Modelling the dynamics of the temperature x_j^f of the flow crossing the single thermal component j

As done for the flow model (1), we approximate the temperature of the flow crossing the j-th component, $x_j^f(\boldsymbol{u}, \boldsymbol{x}^c, x_i)$, using a static mapping. Instrumental for our purposes, it is convenient to re-write the generic j-th coordinate of the recirculation operator Λ^r in (1) as the sum of n scalar polynomial functions, i.e., as

$$\Lambda^r_{(j)}[\boldsymbol{f}] \doteq \sum_{h=1}^{n} \Lambda^r_{(j,h)}[\boldsymbol{f}]. \tag{4}$$

With the above definition, we model $x_j^f(\boldsymbol{u}, \boldsymbol{x}^c, x_i)$ as the weighted average of the temperatures of the direct and recirculation flows under the simplifying assumptions of perfect flow mixing and heat energy conservation, i.e., as

$$x_j^f = \frac{\Lambda^d_{(j)}[\boldsymbol{u}] \cdot x^i + \sum_{h=1}^{n} \Lambda^r_{(j,h)}[\boldsymbol{f}] \cdot x_h^c}{f_j}, \quad j = 1, \ldots, n. \tag{5}$$

The flow temperature model (5) accounts for

(1) the temperature and intensity of the direct flow through the product $\Lambda^d_{(j)}[\boldsymbol{u}] \cdot x^i$;
(2) the temperature and intensity of the recirculation flows from all the other components through the sum $\sum_{h=1}^{n} \Lambda^r_{(j,h)}[\boldsymbol{f}] \cdot x_h^c$.

Using (1) together with (5) corresponds then to approximating the (average) convective cooling effect on the j-th component through the total flow f_j and the energy and mass conservation assumptions leading to the flow temperature model (5).

2.3 Modelling the dynamics of the temperature x_j^c of the single thermal component j

We model the dynamics of the generic j-th component through three main contributions:

$$\dot{x}_j^c = \underbrace{-h_j f_j(\boldsymbol{u}) \left(x_j^c - x_j^f(\boldsymbol{u}, \boldsymbol{x}^c, x^i) \right)}_{\text{convection}} + \ldots$$
$$+ \underbrace{\begin{bmatrix} R_{(j)} & \rho_j \end{bmatrix} \begin{bmatrix} \boldsymbol{x}^c \\ x^i \end{bmatrix}}_{\text{conduction}} + \underbrace{b_j p_j}_{\text{el. power}} \qquad j = 1, \ldots, n. \tag{6}$$

Specifically, the rate of change of temperature in (6) depends on:

i) a convection term that expresses the rate at which heat is transferred between the electronic component and the air flow f_j crossing it. h_j here is a lumped parameter describing the average heat transfer coefficient in Newton's law of cooling. The rate of this heat exchange is moreover modeled to be proportional to the equivalent mass flow f_j crossing the component; we then recall that the latter is in its turn a function of the controls $\boldsymbol{u}$, i.e., $f_j \doteq f_j(\boldsymbol{u})$ (c.f. (2));
ii) a conduction term that expresses the rate at which heat is exchanged among neighboring components through the mechanism of conduction. Parasitic losses to the environment are taken into account by introducing the thermal resistance ρ_j between the component and a fictitious environmental node;
iii) a self-heating term that expresses the rate at which electrical power flowing through the electrical component is converted into heat.

Specializing (6) with (5) we eventually get, for $j = 1, \ldots, n$,

$$
\begin{aligned}
\dot{x}_j^c &= -h_j f_j x_j^c + h_j \sum_{h=1}^{n} \Lambda^r_{(j,h)}[\boldsymbol{f}] \cdot x_h^c + R_{(j)} \boldsymbol{x}^c + \dots \\
&\quad + h_j \Lambda^d_{(j)}[\boldsymbol{u}] \cdot x^i + \rho_j x^i + b_j p_j \\
&= \Psi_j \left(x_j^c(k), \boldsymbol{u}, \boldsymbol{p}, x^i \right).
\end{aligned} \tag{7}
$$

Discretizing (7) using Euler's rule (we assume that the power consumption of the components is piecewise constant and that the controls are zero-order held), with a sample interval of length Δ, yields

$$
\begin{aligned}
x_j^c(k+1) &= x_j^c(k) + \Delta \cdot \Psi_j(x_j^c(k), \boldsymbol{u}, \boldsymbol{p}, x^i) \\
&=: \Psi_j^\Delta(x_j^c(k), \boldsymbol{u}, \boldsymbol{p}, x^i).
\end{aligned} \tag{8}
$$

3. ESTIMATION OF THE PARAMETERS OF THE AIR FLOW MODEL (1) FROM CFD TRIALS

In our control-oriented framework we seek also to estimate the model's parameters through opportune system identification schemes. Unfortunately, measuring $\boldsymbol{x}^c$, $\boldsymbol{x}^f$, $\boldsymbol{p}$, $\boldsymbol{u}$ and $\boldsymbol{f}$ in a real-world system is quite complicated, since it needs expensive and bulky measurement systems (that, moreover, perturb the measured dynamics with their presence).

The design of a server's enclosures, nonetheless, is always performed using opportune CFD tools that simulate accurately the distribution of the temperatures and of the air flows within the server. The intuition is thus that it is possible to simultaneously co-design the structure of the servers' enclosures and the fans control algorithm through the very same CFD model.

We thus assume that we are endowed with a CFD simulator that allows us to [1] [2]:

(1) specify the boundary conditions at the air inlet (i.e., the inlet air temperature x^i and the air mass flows $\boldsymbol{u}$ produced by the m fans);
(2) specify the vector $\boldsymbol{p}$ of the electrical power supplied to the n electronic components;
(3) run the CFD simulation and consequently retrieve the temperatures $\boldsymbol{x}^c$.

Having then collected K sets of initial conditions and temperature measurements from K different CFD runs one can estimate both the coefficients in the maps Λ^d, Λ^r in (1) and the remaining parameters in (7).

Importantly, we notice that the identification problem is not in a standard form; indeed the measurements refer to steady-state CFD simulations, so that all the measurements come from a system that is always in some equilibrium. Recalling (8), formulating the estimation problem in a Least Squares (LS) fashion thus leads to

[1] In this manuscript, for simplicity, we ignore the problem of designing the experiments in the maximally informative way, and leave it for future research issues. In this work we considered a Latin hypercube sampling strategy to generate the sets of initial conditions.

[2] We also notice that throughout each CFD simulation-run, the variables $\boldsymbol{u}$ and $\boldsymbol{p}$ are maintained constant so that the measured variables correspond to a server at steady state. This is compliant with the scope of this manuscript, where we neglect the dynamics of the air flows as stated in Section 2.1.

$$
\begin{aligned}
&\min_{\boldsymbol{\theta}} \sum_{j=1}^{n} \sum_{t=0}^{N-2} \Psi_j \left(x_j^c(t), \boldsymbol{u}(t), x^i \right)^2 \\
&\text{subject to} \quad (I - \Lambda^r)[\boldsymbol{f}(t)] = \Lambda^d[\boldsymbol{u}(t)] \quad 0 \leq t \leq N-2
\end{aligned} \tag{9}
$$

where $\boldsymbol{\theta} \doteq \left[\Lambda^d, \Lambda^r, R, \boldsymbol{h}, \boldsymbol{b}, \rho, \boldsymbol{f}(0), \dots, \boldsymbol{f}(N-2) \right]$ (for notational simplicity we omit in (9) stating obvious non-negativity constraints for some components in $\boldsymbol{\theta}$ like $\boldsymbol{f}(t) \succeq \boldsymbol{0}$).

The structure of (9) nonetheless leads to two separate statistical identifiability issues and two separate numerical issues:

i) considering (7), the dynamics $\dot{x}_j^c = \alpha \Psi$ with $\alpha \neq 0$ has the same steady-state behavior of the dynamics $\dot{x}_j^c = \Psi$; this implies that choosing $\alpha \left[R, \boldsymbol{h}, \boldsymbol{b}, \rho \right]$ instead of $[R, \boldsymbol{h}, \boldsymbol{b}, \rho]$ in (8) leads to the same cost;
ii) (9) is non-convex when the order of the polynomials in Λ^d, Λ^r is higher than 1. This raises the issue of numerical problems in solving the optimization problem when the number of involved parameters is high.

To cope with these issues we:

- address *i)* by letting $\boldsymbol{b} = \boldsymbol{h} = [1, \dots, 1]$, i.e., normalizing the electrical power to heat conversion rates plus absorbing the heat exchange rates in Λ^d and Λ^r;
- address *ii)* by restricting our recirculation model to be at most linear, as in (3). With this hypothesis

$$
\Lambda^r_{(j,h)}[\boldsymbol{f}] = \left[\Lambda^r \left(I - \Lambda^r \right)^{-1} \Lambda^d[\boldsymbol{u}] \right]_{(j,h)} \doteq \Lambda^r_{(j,h)}[\boldsymbol{u}]. \tag{10}
$$

We thus highlight the fact that with linear recirculation models the flows can be determined explicitly as a polynomial mapping of merely the controls $\boldsymbol{u}$ rather then as solutions to the constrained problem (2) (notice that we override the symbol Λ^r for notational simplicity; nonetheless we stress that $\Lambda^r_{(j,h)}[\boldsymbol{f}] \neq \Lambda^r_{(j,h)}[\boldsymbol{u}]$, to avoid confusion).

Exploiting the two strategies above the estimation problem (9) thus reduces to

$$
\begin{aligned}
\min_{\boldsymbol{\theta}} \sum_{j=1}^{n} \sum_{t=0}^{N-2} \Bigg(&\sum_{h=1}^{n} \Lambda^r_{(j,h)}[\boldsymbol{u}] \cdot x_h^c + R_{(j)} \boldsymbol{x}^c + \\
&+ \Lambda^d_{(j)}[\boldsymbol{u}] \cdot x^i + \rho_j x^i + p_j \Bigg)^2
\end{aligned} \tag{11}
$$

subject to the non-negativity constraints on $\boldsymbol{\theta}$.

Our last step is then to rewrite problem (11) as a classical linearly constrained least squares problem by rewriting the polynomial functions $\Lambda^r[\cdot]$ and $\Lambda^d[\cdot]$ as scalar products using (A.2) in Appendix A, In this way every summand in (11) can be rewritten in the separable form

$$
\begin{aligned}
&\boldsymbol{a}^T_{\Lambda^d_{(j)}} \boldsymbol{\pi}_{\Lambda^d_{(j)}} \left[\boldsymbol{u}(k) \right] \left(x^i - x_j^c(k) \right) + \\
&+ \sum_{h=1}^{n} \left(\boldsymbol{a}^T_{\Lambda^r_{(j,h)}} \boldsymbol{\pi}_{\Lambda^r_{(j,h)}} \left[\boldsymbol{u}(k) \right] \left(x_h^c - x_j^c \right) \right) + \\
&+ R_{(j)} \boldsymbol{x}^c(k) + \rho_j x^i + p_j(k).
\end{aligned} \tag{12}
$$

Paraphrasing, thus, for each j-th IT component, solving (11) means solving the least squares problem

$$\Phi^j \boldsymbol{\theta}^j = -\boldsymbol{p}^j, \tag{13}$$

where the hyper-parameter $\boldsymbol{\theta}^j$ and the vector of powers $\boldsymbol{p}^j$ are defined through

$$\boldsymbol{\theta}^j \doteq \left[\boldsymbol{a}^T_{\Lambda^d_{(j)}} \ \boldsymbol{a}^T_{\Lambda^r_{(j,1)}} \ \dots \ \boldsymbol{a}^T_{\Lambda^r_{(j,n)}} \ R_{(j)} \ \rho_j \right]^T, \tag{14}$$

$$\boldsymbol{p}^j \doteq \left[p_j(1) \ p_j(2) \ \dots \ p_j(N) \right]^T, \tag{15}$$

and where the regressor Φ^j is obtained by reorganizing and stacking accordingly the coefficients of (12). For instance, the k-th row of the j-th regressor Φ^j is

$$\Phi^j_k \doteq \Big[\boldsymbol{\pi}^T_{\Lambda^d_{(j)}} [\boldsymbol{u}(k)] \left(x^i - x^c_j(k) \right), \quad \boldsymbol{\pi}^T_{\Lambda^r_{(j,1)}} [\boldsymbol{u}(k)] \left(x^c_1 - x^c_j \right), \quad \vdots \quad \boldsymbol{\pi}^T_{\Lambda^r_{(j,n)}} [\boldsymbol{u}(k)] \left(x^c_n - x^c_j \right), \quad (\boldsymbol{x}^c(k))^T, \quad x^i \Big]. \tag{16}$$

4. MINIMUM COST FAN CONTROL

To design a fans control policy we notice that:

- the rotational speeds of the fans are limited, so that the control values $\boldsymbol{u}$ are constrained in the hyper-rectangle defined by the extreme points $\boldsymbol{u}_{\min}, \boldsymbol{u}_{\max} \in \mathbb{R}^m_{\geq 0}$;
- the temperatures of the IT components shall be kept below some specified safe limits, that implies that there exist state constraints of the kind $\boldsymbol{x}^c \preceq \boldsymbol{x}^c_{\max} \in \mathbb{R}^n_{\geq 0}$;
- the concept of minimizing the cooling provisioning can be translated into minimizing the sum of the power consumption of the individual fans while guaranteeing the temperatures of the IT components to be within their limits. In first approximation, then, the power necessary to produce a given air mass flow is proportional to the product of the generated pressure drop and the mass flow itself. For the generic h-th fan at time t, the latter product is then proportional to the cubic power of the control value $u_h(t)$.

Given the above remarks, we propose to control the fans by solving on-line the following RHC problem:

$$\begin{aligned} &\min_{\boldsymbol{u}(0),\dots,\boldsymbol{u}(H-1)} \sum_{t=0}^{H-1} \sum_{h=1}^{m} (u_h(t))^3 \\ &\text{subject to (for } 0 \leq t \leq H-1, 1 \leq j \leq n): \\ &\quad \boldsymbol{x}^c(0) = \boldsymbol{x}^c_0 \\ &\quad \boldsymbol{u}_{\min} \preceq \boldsymbol{u}(t) \preceq \boldsymbol{u}_{\max} \\ &\quad \boldsymbol{x}^c(t+1) \preceq \boldsymbol{x}^c_{\max} \\ &\quad x^c_j(t+1) = \Psi^{\Delta}_j \left(x^c_j(t), \boldsymbol{u}(t), \boldsymbol{p}(t), x^i \right) \end{aligned} \tag{17}$$

Notice that the electrical power consumption of the n IT components over the future horizon is unknown, i.e., at each time t the values of $\boldsymbol{p}(t), \boldsymbol{p}(t+1), \dots, \boldsymbol{p}(t+H-1)$ are unknown. Since the investigation of opportune predictors of the power consumption is outside of the scope of this work, here we adapt the following heuristic: We consider that the worst case temperature and cost scenario in (17) occurs when the server runs at its full computational load, i.e., when $\boldsymbol{p}(t) = \boldsymbol{p}_{\max}, t = 0, 1, \dots, H-1$. In this manuscript we then use as a forecast of the future power consumption $\boldsymbol{p}(t), \boldsymbol{p}(t+1), \dots$, the worst-case power consumption. We stress that less conservative and more realistic forecast can be (if available) directly plugged into the proposed MPC strategy (17). The degree of conservativeness of the resulting feedback law is demonstrated in Section 5.

5. NUMERICAL EXPERIMENTS

We divide this section in two separate parts: Section 5.1, analyzing the predictive capabilities of model (1), and Section 5.2, analyzing the performance of the control strategy (17).

5.1 Assessment of the identification methodology

We demonstrate the predictive capabilities of model (1) through simulations on a CFD model of a Dell R730xd Power-Edge server like in Figure 1. The CFD model is based on the true size specifications of the server and contains 6 fans, 4 groups of DIMMs and two CPUs with heat-sinks mounted on their top. The simulations were run using the double precision steady state solver in the CFD software ANSYS CFX.

We thus generated a training dataset

$$\mathcal{D}_{\text{train}} \doteq \left\{ \boldsymbol{x}^c_\kappa, \boldsymbol{p}_\kappa, \boldsymbol{u}_\kappa, x^i_\kappa \right\}_{\kappa=1}^{K}$$

by feeding the CFD simulator with K different [3] boundary conditions $\boldsymbol{p}_\kappa, \boldsymbol{u}_\kappa, x^i_\kappa$ and collecting the corresponding $\boldsymbol{x}^c_\kappa$.

We then used this dataset to solve the estimation problem (11) [4] and find an opportune estimate $\widehat{\boldsymbol{\theta}}$.

To asses the predictive capability of $\widehat{\boldsymbol{\theta}}$ we then generated a second dataset

$$\mathcal{D}_{\text{test}} \doteq \left\{ \boldsymbol{x}^c_\kappa, \boldsymbol{p}_\kappa, \boldsymbol{u}_\kappa, x^i_\kappa \right\}_{\kappa=1}^{K'}$$

by feeding the CFD simulator with other K' boundary conditions $\boldsymbol{p}_\kappa, \boldsymbol{u}_\kappa, x^i_\kappa$ that have not been included in the training set $\mathcal{D}_{\text{train}}$, then both collecting again the corresponding $\boldsymbol{x}^c_\kappa$ from the CFD and also forecasting these values through (7) (where we implicitly compute $\boldsymbol{x}^c$ as the equilibrium of the thermal dynamics defined by the estimated $\widehat{\boldsymbol{\theta}}$).

The results are graphically plotted in Figure 5, where the x-axis represents the simulation index, the left y-axis indicates the temperature measured by the CFD simulator,

[3] For simplicity in this work we generated both the training and test sets by means of Latin Hypercube Sampling strategy.

[4] We notice that increasing the number of terms in the flow model (1) may lead to both overfitting and increasing computational requirements. To mitigate these problems in our estimation problem we actually regularized the LS problem (11) by means of a lasso penalty with regularization parameter automatically chosen by means of a leave-one-out cross validation, and by choosing the maximal polynomial order by means of an Akaike Information Criterion (AIC)-type model order selection strategy.

and the right y-axis indicates the error committed by our trained model. As it can be seen, the maximal error is about 2 degrees but usually stays well below 1 degree, indicating overall a good prediction capability at least in the considered scenario.

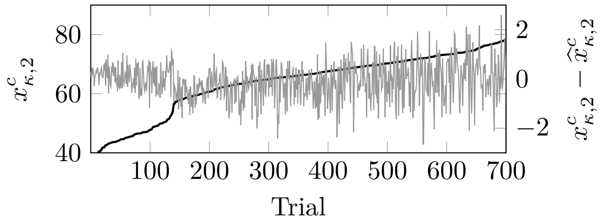

Figure 3. Reconstruction of the steady state temperatures of the first CPU (black line) over 700 validation trials. The reconstruction error is shown in gray.

5.2 *Assessment of the control methodology*

In this section we test the capabilities of our control scheme in minimizing the power consumption used by the fans while constraining the thermal dynamics of the IT components to be within their thermal comfort ranges. Our assessments are not based on a real server, but rather on the model identified in Section 5.1.

We consider two different heat load scenarios:

1) the true power consumption of the components at time t, $\boldsymbol{p}(t)$, is set to be the worst-case power consumption. That is, in this scenario we set $\boldsymbol{p}(t) = \boldsymbol{p}_{\max}$ for all t;
2) the true power consumption of the components is set to be a realization of the stochastic process with seasonal trends described in [15], simulating a medium to high load scenario.

For both scenarios we set the inlet temperature to be $x^i = 20\,^\circ C$, a prediction horizon of length $H = 4$ and a sampling time of $\Delta = 1$ seconds. Iteratively solving the minimum cost problem (17) in a receding horizon fashions yields then the temperature and control trajectories in Figure 4 and Figure 5 for scenarios *i*) and *ii*) respectively.

We observe that in scenario *i*) the full utilization of the two CPUs leads to an over provisioning of the cooling resources for the DIMMs: Due to the flow mixing inside the enclosure, indeed, this is the minimum cost control that can dissipate the heat produced by the two CPUs while satisfying all the temperature constraints of all the IT components. In scenario *ii*), instead, the minimum cost controller is able to regulate all temperatures in an efficient manner near to the upper comfort limits.

Finally, we notice that in both scenarios the control policy (17) is able to operate the simulated server with no constraint violations in spite of the unknown future power loads.

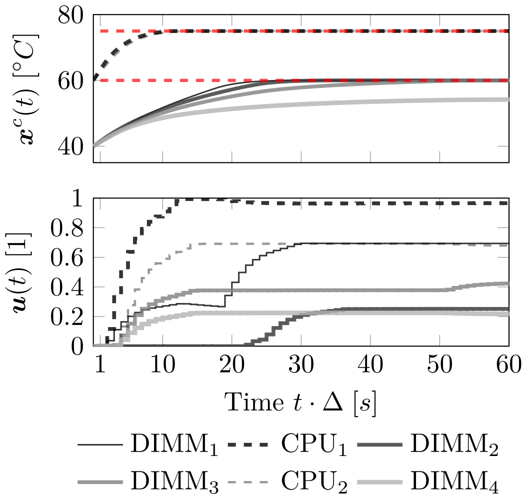

Figure 4. Controlled cooling trajectories of a server under full computational and power loads.

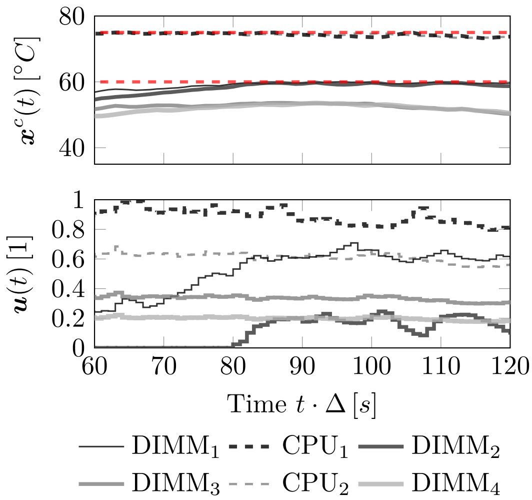

Figure 5. Controlled cooling trajectories of a server under medium to high computational loads.

6. CONCLUSIONS

Developing control oriented models is the most time-consuming task in the implementation of model-based controllers. Here, we proposed a novel modelling and optimal control framework tailored for air cooled servers within data centers. Our dynamics shows a promising ability at capturing the complex convective cooling effects inside the server's enclosure; its functional structure moreover is open to other cooling applications.

In the light of these new and fast methodologies to approximate the network's thermal dynamics and to compute the control costs, this work hints at enabling the co-design of the geometries within the server enclosure and that of the cooling control policies. Towards this eventual goal, we detailed an identification strategy that is based on steady state measurements performed by a CFD tool. We showed that our modelling strategy is effective at capturing the steady state behavior of the server's thermal state and demonstrated minimum cost in-silico sample trajectories of an air cooled, controlled, server.

For the future we envision the study of generic thermal networks with an arbitrary number of m supply nodes that distribute the cooling resources and an arbitrary number of n heating nodes that necessitate cooling.

ACKNOWLEDGEMENTS

We kindly thank the research team at SICS North [16] for the discussions and support.

REFERENCES

[1] The Boston Consulting Group, "Digital Infrastructure and Economic Development: An impact assessment of Facebook's data center in Northern Sweden," The Boston Consulting Group, Tech. Rep., 2014.

[2] A. Capozzoli and G. Primiceri, "Cooling Systems in Data Centers: State of Art and Emerging Technologies," *Energy Procedia*, 2015.

[3] L. Parolini, "Models and Control Strategies for Data Center Energy Efficiency," Ph.D. dissertation, Carnegie Mellon University, 2012.

[4] H. Zhu, X. Liao, C. de Laat, and P. Grosso, "Joint flow routing-scheduling for energy efficient software defined data center networks," *Journal of Network and Computer Applications*, 2016.

[5] H.-P. Jiang, D. Chuck, and W.-M. Chen, "Energy-Aware Data Center Networks," *Journal of Network and Computer Applications*, 2016.

[6] V. K. M. Raj and R. Shriram, "Power management in virtualized datacenter – a survey," *Journal of Network and Computer Applications*, 2016.

[7] S. Mittal, "Power Management Techniques for Data Centers: A Survey," *arXiv preprint arXiv:1404.6681*, 2014.

[8] J. Kim, M. M. S. Aly, D. Atienza Alonso, K. Vaidyanathan, and K. Gross, "Global Fan Speed Control Considering Non-Ideal Temperature Measurements in Enterprise Servers," 2014.

[9] C. Lee and R. Chen, "Optimal self-tuning PID controller based on low power consumption for a server fan cooling system," *Sensors (Switzerland)*, 2015.

[10] Y. Fu, N. Kottenstette, Y. Chen, C. Lu, X. D. Koutsoukos, and H. Wang, "Feedback Thermal Control for Real-time Systems," in *2010 16th IEEE Real-Time and Embedded Technology and Applications Symposium*, 2010.

[11] Z. Wang, C. Bash, N. Tolia, M. Marwah, X. Zhu, and P. Ranganathan, "Optimal fan speed control for thermal management of servers," 2009.

[12] M. Zapater, J. L. Ayala, J. M. Moya, K. Vaidyanathan, K. Gross, and A. K. Coskun, "Leakage and Temperature Aware Server Control for Improving Energy Efficiency in Data Centers," in *Design, Automation & Test in Europe Conference & Exhibition*, 2013.

[13] R. Ayoub, R. Nath, and T. Rosing, "JETC: Joint energy thermal and cooling management for memory and CPU subsystems in servers," in *Proceedings - International Symposium on High-Performance Computer Architecture*, 2012.

[14] C. S. Chan, Y. Jin, Y.-K. Wu, K. Gross, K. Vaidyanathan, and T. Rosing, "Fan-speed-aware scheduling of data intensive jobs," in *International symposium on Low power electronics and design*, 2012.

[15] A. M. De Livera, R. J. Hyndman, and R. D. Snyder, "Forecasting Time Series With Complex Seasonal Patterns Using Exponential Smoothing," *Journal of the American Statistical Association*, 2011.

[16] "Swedish Institute for Computer Science North (Luleå)."

Appendix A. POLYNOMIAL FUNCTIONS

Definition 1. (Polynomial function). We call $\boldsymbol{u} \mapsto \Lambda(\boldsymbol{u}) : \mathbb{R}^m \to \mathbb{R}^n$ a *polynomial function* if each of its coordinates Λ_i, $i = 1, \ldots, n$, is a polynomial mapping of $\boldsymbol{u} \doteq \begin{bmatrix} u_1 & u_2 & \ldots & u_m \end{bmatrix}$. The degree of Λ is the highest *monomial degree* among all the monomials of $\Lambda_1, \ldots, \Lambda_n$.

Example 2. The functions $\Lambda_1 : \mathbb{R}^2 \to \mathbb{R}$ and $\Lambda_2 : \mathbb{R}^2 \to \mathbb{R}^2$ defined below are polynomial with degree 2 and 3 respectively:

$$\Lambda_1(u_1, u_2) \doteq u_1 u_2, \quad \Lambda_2(u_1, u_2, u_3) \doteq \begin{bmatrix} u_1 u_2 \\ u_1^3 \end{bmatrix}.$$

Example 3. A generic real-valued polynomial function with degree 3 is written

$$\Lambda(u_1, \ldots, u_m) = \sum_{1 \le i \le j \le k \le m} a_{ijk} u_i u_j u_k + \ldots + \sum_{1 \le i \le j \le m} a_{ij} u_i u_j + \sum_{1 \le i \le m} a_i u_i + a_0.$$

More in general, it can be shown (e.g., using a stars and bars combinatoric argument) that the number of coefficients in a scalar polynomial mapping with m variables and degree d is given by counter

$$\sigma(m, d) \doteq \binom{m+d}{d}.$$

In this manuscript we prefer the notation $\Lambda[\mathbf{u}]$ to $\Lambda(u_1, \ldots, u_m)$ and write a generic scalar polynomial of degree d and m variables as

$$\Lambda[\boldsymbol{u}] = \sum_{i=1}^{\sigma(m,d)} a_i \boldsymbol{u}^{\boldsymbol{\alpha}_i}, \tag{A.1}$$

where $a_i \in \mathbb{R}$, the generic $\boldsymbol{\alpha}_i = \begin{bmatrix} \alpha_{i1} & \alpha_{i2} & \ldots & \alpha_{im} \end{bmatrix}$ is an m-tuple in $\{0, 1, \ldots, d\}^m$ such that $\sum_{j=1}^{m} \alpha_{ij} \le d$ and the notation $\boldsymbol{u}^{\boldsymbol{\alpha}_i}$ is taken to mean $\prod_{j=1}^{m} u_j^{\alpha_{ij}}$. We order the finite families $\{\boldsymbol{\alpha}_i\}$ in a lexicographical manner, i.e., so that $\boldsymbol{\alpha}_1 = (0, 0, \ldots, 0)$ precedes $\boldsymbol{\alpha}_2 = (0, 0, \ldots, 1)$, the latter precedes $\boldsymbol{\alpha}_{d+2} = (0, \ldots, 1, 0)$ and so on up to $\boldsymbol{\alpha}_{\sigma(m,d)} = (d, 0, \ldots, 0)$.

Given the ordering above we can always rewrite (A.1) as the scalar product

$$\Lambda[\boldsymbol{u}] = \boldsymbol{a}_\Lambda^T \boldsymbol{\pi}_\Lambda[\boldsymbol{u}], \tag{A.2}$$

where the monomial coefficients and the powers of $\boldsymbol{u}$ have been organized in two vectors $\boldsymbol{a}_\Lambda, \boldsymbol{\pi}_\Lambda \in \mathbb{R}^{\sigma(m,d)}$ defined through

$$\begin{aligned} \boldsymbol{a}_\Lambda &\doteq \begin{bmatrix} a_1 & a_2 & \ldots & a_{\sigma(m,d)} \end{bmatrix}^T, \\ \boldsymbol{\pi}_\Lambda[\boldsymbol{u}] &\doteq \begin{bmatrix} \boldsymbol{u}^{\boldsymbol{\alpha}_1} & \boldsymbol{u}^{\boldsymbol{\alpha}_2} & \ldots & \boldsymbol{u}^{\boldsymbol{\alpha}_{\sigma(m,d)}} \end{bmatrix}^T. \end{aligned} \tag{A.3}$$