Thermal-aware job scheduling in data centers

van Damme, Tobias

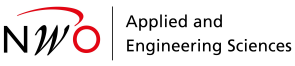
Thermal-aware job scheduling in data centers
An optimization approach

Tobias Van Damme

university of groningen

This research has been carried out at the Faculty of Science and Engineering, University of Groningen, The Netherlands as part of the Smart Manufacturing Systems - Cyber-Physical Systems (SMS-CPS) research group within the ENgineering and TEchnology Institute Groningen (ENTEG).

This dissertation has been completed in partial fulfillment of the requirements of the dutch institute of systems and control (disc) for graduate study.



Applied and Engineering Sciences

This project is funded by the Netherlands Organization for Scientific Research, branch Applied and Engineering Sciences, formerly Stichting voor de Technische Wetenschappen (STW). This project is part of the Cooperative Networked Systems project which is a subproject of the Perspectief programme Robust Design of Cyber-Physical Systems (CPS) with project number 12696. The work has been done in collaboration with industrial partners Better.Be and Target Holding.

Cover image: © Cybrain - stock.adobe.com
Printed by: *Studio
250 copies printed

ISBN 978-94-034-1790-5 (electronic version)
ISBN 978-94-034-1791-2 (printed version)

rijksuniversiteit
groningen

Thermal-aware job scheduling in data centers
An optimization approach

Proefschrift

ter verkrijging van de graad van doctor aan de
Rijksuniversiteit Groningen
op gezag van de
rector magnificus prof. dr. E. Sterken
en volgens besluit van het College voor Promoties.

De openbare verdediging zal plaatsvinden op

Vrijdag 12 juli 2019 om 11:00 uur

door

Tobias Van Damme

geboren op 23 augustus 1988
te Gent, België

Promotor
Prof. dr. C. De Persis

Copromotor
Dr. P. Tesi

Beoordelingscommissie
Prof. dr. A.J. van der Schaft
Prof. dr. D. Varagnolo
Prof. dr. L. Zaccarian

Acknowledgments

Writing a thesis is the culmination of 4 years of work. While the work itself is mostly done alone, it wouldn't be possible to survive these 4 years on your own. Throughout the years I have been lucky to have met many new people, made many new friends, and have had a lot of great experiences. I would like to thank the many people who have helped me on this journey.

First of all I would like to thank my supervisors Claudio and Pietro. Claudio, thank you for all the help. Without your sharp mind and many insightful suggestions I would not have made it this far. I will always remember the surprise birthday celebrations I organized for you and hope there will come a time that you will start celebrating your birthday by bringing your favorite birthday snacks to the office. Pietro, it is a shame that you were forced to move back to Italy because I have always enjoyed your calm and relaxed spirit. You always had time to talk about extracurricular activities and things other than research. I have enjoyed the lively discussions between you and Claudio a lot and I still must say that being supervised by two Italians is a treat. Thank you both for the nice 4 years and for welcoming me in your research group. (P.S. sorry for not making you famous, Claudio, and sorry for not making you rich, Pietro).

Many thanks to all my SMS colleagues Matin, Hongkeun, Nima, Sebastian, Tjardo, Erik, Shuai, Danial, Mingming, Mark, Henk, Tjerk, and more recently Hongyu, Monica, Meichen, Alessandro, Mehran, and Guopin. Thanks for all the fun discussions, beer drinking, (not) talking about research, lunches, building the smart grid game, doing the keuzecollege and much more. Also thanks to the many DTPA and JBI colleagues, Xiaodong, Filip, Pooya, Yuzhen, James, Jing, Ning, Martijn, Zaki, Agung, Michele, Carlo, Alain, Rodolfo, Pablo, Hector, Jesus, Hildeberto, Hadi, Yu, Mauricio, Eduardo, Krishna, Yuri, Marco, Xiaoshan, Nelson, and all the others colleagues I have forgotten to mention here. Thanks for the lunch times, benelux meetings, table soccer times, movies, the group meetings, group outings, and all the other fun times. It was inspiring to get to know you and learn about

all the different cultures and countries you come from. Sietse, Martin, and Simon, many thanks for your support on my more practical ventures. You were always there when something went haywire or I needed something else. Lastly, successfully completing a PhD is not only doing research, but there is a lot of organization and bureaucracy involved. Due to the very competent secretaries and supportive staff, Frederika, Johanna, Angela, Karen, this was a very light burden on my part. Many thanks for the support and the many nice talks about everything. I will miss you a lot in my future ventures. It is the people that make the workplace amazing and you all made my time as a PhD student truly enjoyable.

Furthermore I would like to thank Tjerk for our very productive collaboration. You helped me with a topic I have been stuck with for a very long time. Our discussions taught me a lot, and your sharp intellect always made for very inspiring research sessions. Henk, thank you for teaching me about system identification methods and the combined supervision of our bachelor students. Thanks to you I have been able to complete my thesis with some very nice practical results. Lastly many thanks to Andy, Jun, Boudewijn, and Björn, for the many years of collaboration in our STW project. Björn, thank you in particular for our nice collaboration. It was a big change to work together with somebody outside of the control field and although we had some struggles in the beginning, we achieved a nice result we can be proud of. Thanks for the nice times, and fruitful and interesting discussions.

Verder zijn er veel (nieuwe) vrienden die me geholpen hebben om door deze mooie tijd te komen en voor genoeg afleiding te zorgen om te kunnen ontsnappen aan het onderzoek wanneer dat nodig was. Bedankt aan iedereen voor de mooie feesten, maaltijden, reizen, bordspellen, goede gesprekken, koffiemomentjes, gaming en optimalisatiesessies. Ik hoop dat we nog veel mooie momenten zullen blijven beleven. Ook de Apihanen kan ik niet vergeten, bedankt voor alle gezellige bridgeavonden, ik zal ze zeker missen. Jelle, bedankt voor het geduld en de wijze bridgelessen. In het bijzonder wil ik Tjardo en Jasper bedanken als mijn paranimfen. Er moet veel geregeld worden voor een verdediging en ik ben blij dat ik die last met jullie kan delen.

Dan veel dank aan mijn familie, Lut, Peter, Isa, Lieselot, Johannes, en

Florian, Marilyn, Anthony, voor de eeuwige steun en wijze raad. Ik ben blij dat ik altijd bij jullie terecht kan als dat nodig is. Ook wil ik mijn schoonfamilie bedanken. Jullie hebben mij opgenomen in jullie familie alsof ik daar altijd thuis hoorde. Ik vind het elke keer weer fijn om naar het zuiden af te reizen om bij jullie op bezoek te komen.

Daarnaast wil ik nog even stilstaan bij mijn stiefvader, Klaas-Gert. Ik weet dat je trots op me bent met de voltooing van mijn proefschrift, zeker aangezien het onderwerp te maken heeft met energie(reductie), iets wat jou altijd ter harte ging. Je bent er helaas niet meer, maar je blijft voor de rest van mijn leven in mijn gedachten.

Uiteraard zijn er veel meer mensen die direct of indirect hebben geholpen bij de totstandkoming van dit proefschrift. Ook aan jullie, bedankt.

Als laatste wil ik mijn lieve vriendin Veerle bedanken. Ik mag mezelf gelukkig prijzen dat ik jou heb mogen ontmoeten tijdens mijn promotietraject. Je bent lief, begripvol, Belg, houdt van bordspelletjes, en geeft me de ruimte om mezelf te zijn wanneer ik dat nodig heb. Je maakt me ontzettend gelukkig en met ons aankomende kleintje hoop ik dat dit geluk nog lang mag blijven.

Tobias Van Damme
Utrecht
16th of May, 2019

Contents

CHAPTER 1

Introduction

In the year 2013 worldwide energy consumption of data centers reached 350 billion kWh of energy, or 1.73% of the global electricity consumption (Blatch, 2014; Enerdata, 2016). Data centers are facilities that contain large amounts of computers and play an important role in current day digital affairs. For example, all cloud-based services, such as e-commerce, social networks, entertainment, and financial services, find their basis operation at data centers. Not only these consumer-based products but also an ever-growing share of industrial and organizational processes, such as smart industry or the digital governmental environment, take place in large computational clusters.

Data centers became common-place in the last two decades together with the rise of the internet because they allow operators to fully utilize the economy of scale when operating and maintaining these computational beasts. At first the focus was mainly on performance, however as technology and demand continued to advance, data centers quickly grew larger and larger. As such the importance of carefully designing data centers became increasingly apparent.

The Berkeley National Laboratory did a study in 2016 on the energy consumption of United States data centers, (Shebabi et al., 2016). In Figure 1.1 the energy consumption of US data center is shown. The current trends in the Figure show the historical energy consumption until 2014, while from 2015 to 2020 a projection, based on the trends at that time, is shown. The Figure also shows a scenario of what would have happened if the energy saving efforts were halted in 2010. It is projected that by 2014 the energy consumption would be 60% higher than the historical power consumption,

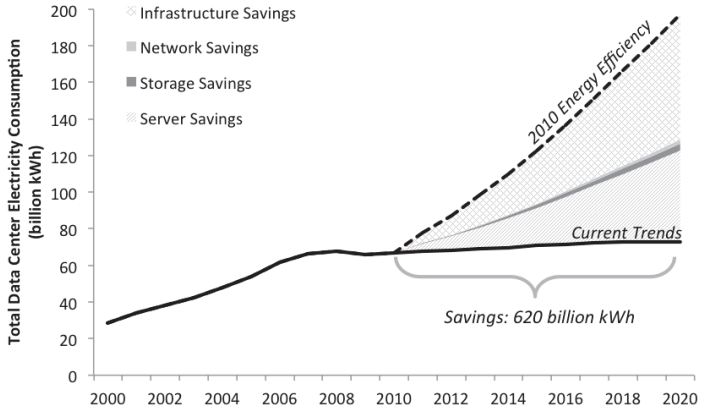


FIGURE 1.1: Energy consumption of United States data centers (Shebabi et al., 2016). Until 2014 the data is historical, from 2015-2020 is a projection based on trends up to 2014. The figure shows the estimated energy consumption if data center energy efficiency improvements would have stopped in 2010.

and by 2020 the energy consumption would be 170% higher than the estimated energy consumption following the 2014 trends. In total the energy savings will have amounted to 620 billion kWh. This shows the tremendous possibilities of energy-efficiency improvements.

From Figure 1.1 we see that the annual energy consumption of US data centers has been relatively the same from 2006-2014. According to (Shebabi et al., 2016) this stabilization is attributed to three main energy-efficiency improvements: (1) advanced cooling strategies, (2) power proportionality, and (3) server consolidation.

Advanced cooling strategies focus on techniques that improve the cooling efficiency in the data center, e.g. cold-hot aisle configuration, economizers, and liquid cooling. Power proportionality attempts to scale power consumption directly to utilization, i.e. a server running at 10% of its capacity uses 10% of its maximum power consumption. Power proportionality can be achieved by upgrading hardware and implementing better power management software. Lastly, server consolidation aims at running the same

load on as little servers as possible, such that data centers need less equipment and servers run at higher average utilization levels.

While the energy problem is a strong motivator for data center owners to save on their total cost of ownership by saving on their energy bill, data centers also provide an interesting topic from a scientific perspective. Data centers are an excellent example of cyber-physical systems (CPS). A CPS is a system in which there is a close connection between the physical world and the digital world. The physical world is measured by sensors, while the digital part will control the physical world with actuators. The data center is a system where the physical world, e.g. thermodynamics and power consumption, and the digital world, e.g. load balancing and network infrastructure, mix in an interesting way. Already many results have been developed in the last decade as computer scientists and control engineers have made efforts to devising methods to reduce the energy consumption of data centers (Hameed et al., 2014).

1.1 Advanced cooling strategies

Although much progress has been made, there are still several challenges in ensuring efficient operation of the cooling equipment. Due to bad design or unawareness for the thermal properties of the data center, local thermal hotspots can arise. This causes the cooling equipment to overreact to ensure that the temperature of the equipment stays below the safe thermal threshold. These peaks cause the cooling equipment to consume more energy then would be necessary if these hotspots were avoided. Therefore having a good understanding of the thermodynamics involved is vital to increasing the cooling efficiency of the data center.

To tackle these challenges researchers and engineers have studied both software and hardware solutions to this problem. Examples of hardware solutions are isolating cold or hot areas in the data center, or building data centers in cold regions on the planet where cold outside air can be utilized. Software solutions on the other hand focus on strategies which use the knowledge of the thermal properties of the data center to make more intelligent choices how to schedule incoming jobs. Although the two types of

solutions are equally important to study, software solutions allow data center operators to implement improvements very fast for very little costs, i.e. implementing new software is less costly than rebuilding a full data center.

Software solutions can be designed via heuristic methods or along a control theoretical direction. Heuristics have already shown to yield good results. In the work of (Moore et al., 2005) and (Tang, Gupta, and Varsamopoulos, 2008), energy consumption reductions of up to 30% are achieved after implementing smart thermal-aware job schedulers. However, heuristics might not be optimal, or might not be able to respond dynamically to the changing operating conditions. As such, researchers have also turned to control theory to understand data centers from a more fundamental point of view.

For example, (Vasic, Scherer, and Schott, 2010) have proposed a control algorithm that tries to maintain the temperature of the equipment around a target value. In (Yin and Sinopoli, 2014) it is proposed to implement a two-step algorithm that first minimizes the energy consumption by estimating the required amount of servers to handle the expected workload. In the second step the algorithm maximizes the response time given a number of servers at its disposal. In an attempt to address scalability, a distributed approach has been studied in (Doyle et al., 2013). Another distributed control approach in a hybrid systems setting is proposed in (Albea, Seuret, and Zaccarian, 2014). The hybrid controller tries to evenly divide the total load among the agents in the network in a distributed fashion.

1.2 Contributions

The contribution of this thesis to the state-of-the-art is the development of a theoretical framework that can be used to study and understand the thermodynamic behavior of the heat flows in a data center. Much of the prior work done in this field focuses on heuristic approaches that use metrics that try to approximate optimality. We contribute by providing a study that characterizes energy optimality exactly. This provides data center operators with a great opportunity of understanding what the optimal operating point looks like in their data center context. The model presented in this thesis is data

center independent, although the model is mostly usable by data centers that handle workload streams, such as HTTP requests or Google searches. High performance clusters usually run non-stop at full capacity, which reduces control opportunities via the job scheduling techniques described in this work.

Based on this model, controllers and an extension to those controllers are introduced that allow control in most common operating conditions. The integral controllers designed in chapter 3 work in most current day setups, whereas the work in chapter 4 shows how the controllers can be adapted to work in all operating conditions. chapter 5 applies the controllers in a futuristic scenario, where the data center is equipped with servers that can efficiently and safely switch power states.

The key part of the thermodynamical model is the recirculation of the heat flows in the data center. Both the model and the controllers depend on these parameters. To complete the results of this thesis, we studied subspace identification techniques with which these parameters can be identified. It is possible to design experiments that can readily be run in any data center setting, to determine the parameters for that specific data center layout.

All in all this thesis contributes to the state-of-art by supplying a complete set of results that can be applied in any data center context in any current day setting, while also providing flexibility to adapt to upcoming technological advances.

1.3 Outline

The work of this thesis is presented in five chapters, chapter 2-6, and is finalized with some conclusions and future outlook, chapter 7. The thermodynamical model and initial control design form the heart of the thesis, afterwards each chapter focuses on an extension of the main work.

In chapter 2 we design the thermodynamical model of the data center. First the different parts of the data center equipment are introduced and it is explained how each part fits into the model. By considering heat recirculation flows we can model how each computing node thermodynamically affects its neighboring nodes. Furthermore it is possible to determine the

energy consumption of the cooling equipment based on the thermodynamics of the computing equipment.

Having determined the energy consumption of the cooling equipment, we proceed to study ways to reduce data center energy consumption in chapter 3. We apply optimization theory to characterize an optimal operating point at which the data center consumes the minimal amount of energy. Although the initial problem is non-convex, and therefore difficult to study, we rewrite the problem in linear form and show that it is possible to characterize the optimal operating point analytically in different operating conditions. The chapter concludes with the design of simple integral controllers that can steer the operating point of the data center to the optimal operating point for most standard current-day operating conditions.

In chapter 4 an extension to the integral controllers designed in chapter 3 is studied such that the controllers also work in edge cases. In this chapter we design primal-dual dynamics that converge under non-strict convex cost functions, such as the linear optimization problem designed in this thesis. We show that the interconnection between the primal-dual algorithm and the integral controllers is stable for our data center context, implying that the interconnection between the primal-dual dynamics and the integral controllers indeed allow for correct control in all operating conditions.

Reducing the energy consumption of the cooling equipment is not the only way that data center energy reductions are achieved. Power management strategies aim at reducing the power consumption by reducing the amount of necessary computational equipment. In chapter 5 we combine the cooling strategies suggested in this thesis with power management strategies designed at the University of Twente. We show that by combining both approaches, further energy consumption reduction can be achieved.

All the results so far depend on knowing the thermodynamical recirculation parameters of the data center. In chapter 6 we study a possible way in which we can identify these recirculation parameters for any given data center context. Following results of subspace identification, it is possible to design simple experiments and suitable algorithms that identify the recirculation parameters with great accuracy.

1.4 List of publications

[1] T. Van Damme, C. De Persis, and P. Tesi (2018). "Optimized Thermal-Aware Job Scheduling and Control of Data Centers". In: *IEEE Transactions on Control Systems Technology*, pp. 1–12 (chapter 2-3)

[2] T. Van Damme, C. De Persis, and P. Tesi (2017). "Optimized Thermal-Aware Job Scheduling and Control of Data Centers". In: *Proceedings of the IFAC World Congress*

[3] T. W. Stegink, T. Van Damme, and C. De Persis (2018). "Convergence of projected primal–dual dynamics with applications in data centers". In: *7th IFAC Workshop on Distributed Estimation and Control in Networked Systems* (chapter 4)

[4] B. F. Postema, T. Van Damme, C. De Persis, P. Tesi, and B. R. Haverkort (2018). "Combining Energy Saving Techniques in Data Centres using Model-Based Analysis". In: *Companion of the 2018 ACM/SPEC International Conference on Performance Engineering*. ACM, pp. 67–72 (chapter 5)

1.5 Notation

We denote by $\mathbb{R}$ and $\mathbb{R}_{\geq 0}$ the set of real numbers and non-negative real numbers respectively. Vectors and matrices are denoted by $x \in \mathbb{R}^n$ and $A \in \mathbb{R}^{m\times n}$ respectively. The transpose is denoted by x^T, the inverse of a matrix is denoted by A^{-1}, and the Moore-Penrose inverse of a matrix is denoted by $A^\dagger$. If the entries of x are functions of time then the element-wise time derivative is denoted by $\dot{x}(t) \triangleq \frac{d}{dt}x(t)$. An optimal solution to an optimal problem is denoted by $\bar{x}$.

By x_i we denote the i-th element of x and by a_{ij} we denote the ij-th element of A. If a variable already has another subscript then we switch to superscripts to denote individual elements, i.e. T^i_{out} and C_3^{ij}. We construct the diagonal matrix from the elements of vector x as $\text{diag}\{x_1, \quad x_2, \quad \cdots, \quad x_n\}$. The identity matrix of dimension n is denoted by I_n, the vector of all ones

by $\mathbb{1} \in \mathbb{R}^n$ and the vector of all zeros by $\mathbb{0} \in \mathbb{R}^n$. Furthermore the vector comparison $x \preccurlyeq y$ is defined as the element-wise comparison $x_i \leq y_i$ for all elements in x and y.

For $A \in \mathbb{R}^{m \times n}$, we let $\|A\|$ denote the induced 2-norm. Given $v \in \mathbb{R}^n$ and positive definite matrix $A \in \mathbb{R}^{n \times n}$, we write $\|v\|_A \triangleq \sqrt{v^T A v}$. For vectors $u, v \in \mathbb{R}^n$ we write $u \perp v$ if $u^T v = 0$. We use the compact notational form $\mathbb{0} \preccurlyeq u \perp v \preccurlyeq \mathbb{0}$ to denote the complementarity conditions $u \succcurlyeq \mathbb{0}, v \succcurlyeq \mathbb{0}$, and $u \perp v$.

1.6 Preliminaries

In this section we state some preliminaries on dynamical systems and convex optimization that are used as a basis of obtaining some of the results in this thesis.

1.6.1 Lyapunov stability

Consider the system

$$\dot{x} = f(x), \tag{1.1}$$

with $x \in \mathbb{R}^n$ and locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}^n$. An equilibrium $\bar{x}$ is considered to be a solution to the system such that $f(\bar{x}) = \mathbb{0}$. Stability of such an equilibrium is often studied using Lyapunov functions.

Definition 1.1 (Lyapunov stability)**.** An equilibrium $\bar{x}$ of system (1.1) is called Lyapunov stable, if for any $\epsilon > 0$ there exists a $\delta > 0$ such that given a solution $x(t)$ to the system, $\|x(0) - \bar{x}\| < \delta$ implies that $\|x(t) - \bar{x}\| < \epsilon$ for all $t \geq 0$.

Definition 1.2 ((local) Lyapunov function)**.** A smooth function $V : \mathcal{D} \to \mathbb{R}$, on the domain $\mathcal{D} \subset \mathbb{R}^n$, $\{\mathbb{0}\} \in \mathcal{D}$, is a local Lyapunov function for system (1.1) if

1. $V(x) \geq 0$ for all $x \in \mathcal{D}$ and $V(x) = 0$ if and only if $x = \mathbb{0}$.
2. $\dot{V}(x) = (\nabla V(x))^T f(x) \leq 0$ for all $x \in \mathcal{D}$.

If $\mathcal{D} = \mathbb{R}^n$, and V is radially unbounded, then V is called a (global) Lyapunov function. If $\dot{V}(x) < 0$ for all $x \in \mathcal{D}, x \neq \mathbb{0}$ then V is called a strict (local) Lyapunov function.

Theorem 1.1 (Lyapunov stability theorem (Khalil, 2002)). *Let $\bar{x} = \mathbb{0}$ be an equilibrium of* (1.1) *and let V be a Lyapunov function with domain $\mathcal{D} \subset \mathbb{R}^n$, such that $\{\mathbb{0}\} \in \mathcal{D}$. Then $\bar{x} = \mathbb{0}$ is stable. Moreover if V is a strict Lyapunov function, then $\bar{x}$ is (locally) asymptotically stable.*

It is not always straightforward to construct a suitable strict Lyapunov function. In some of these cases the Lyapunov stability theorem can be extended using LaSalle's invariance principle in order to still draw conclusions on the asymptotic behavior of the system.

Lemma 1.1 (LaSalle's invariance principle (Sepulchre, Jankovic, and Kokotovic, 1997)). *Let Ω be a positively invariant set of* (1.1)*, i.e. $x(0) \in \Omega$ implies $x(t) \in \Omega$ for all $t \geq 0$. Suppose that all solutions of* (1.1) *converge to a subset $S \subseteq \Omega$, and let M be the largest positively invariant subset of S under* (1.1)*. Then, every bounded solution of* (1.1) *starting in Ω converges to M as $t \to \infty$.*

Lemma 1.2 ((Pointwise) asymptotic convergence (Haddad and Chellaboina, 2008)). *Let $\bar{\mathcal{X}} = f^{-1}(\mathbb{0}) \ni \mathbb{0}$ be the set of equilibria of* (1.1) *and suppose it admits a local Lyapunov function V with domain $\mathcal{D} \ni \{\mathbb{0}\}$. Suppose furthermore that there exists a sublevel set $\Omega = \{x : V(x) \leq c \in \mathbb{R}_{>0}\} \subset \mathcal{D}$ of V around the origin. Then each trajectory of* (1.1) *initialized in Ω converges to the largest invariant set M contained in*

$$S \triangleq \{x \in \Omega \mid \dot{V}(x) = 0\}.$$

If furthermore each point in M is Lyapunov stable, then this trajectory converges to a point in M.

1.6.2 Convex optimization

A general optimization problem can be formulated as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & x \in \mathcal{X} \end{aligned} \tag{1.2}$$

where $f : \mathcal{X} \to \mathbb{R}$ is the objective function, $\mathcal{X}$ is the feasibility set, i.e. set of feasible solutions, and x is often called the primal variable. The aim of this optimization problem is to find $\bar{x} \in \mathcal{X}$ that minimizes the objective function f, i.e. $f(\bar{x}) \leq f(x)$, for all $x \in \mathcal{X}$. In this thesis we assume that $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{X}$ is a closed convex set, and that f is a continuously differentiable convex function. Since f is a convex function and $\mathcal{X}$ convex, we call (1.2) a convex optimization problem. Very often the feasibility set can be characterized explicitly by

$$\mathcal{X} = \{x \in \mathbb{R}^n \mid Ax = b, g_i(x) \leq 0, i = 1, \ldots, q\},$$

where $A = \mathbb{R}^{m \times n}$, $b = \mathbb{R}^m$, and $g_i : \mathbb{R}^n \to \mathbb{R}$, for $i = 1, \ldots, q$ are continuously differentiable convex functions. Without loss of generality we assume that the equality constraints formed by $Ax = b$ are linearly independent. Rewriting (1.2) using this explicit characterization we get

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) && \text{(1.3a)} \\ \text{subject to} \quad & Ax = b && \text{(1.3b)} \\ & g(x) \preccurlyeq \mathbb{0} && \text{(1.3c)} \end{aligned}$$

where we have collected the inequality constraints in one vector. This optimization problem is referred to as the primal problem, and associated to this problem one can formulate a dual problem with corresponding dual variables. These dual variables are often introduced via the Lagrangian function.

Definition 1.3 (Lagrangian function)**.** The Lagrangian function of (1.3) is given by

$$L(x, \lambda, \mu) = f(x) + \lambda^T(Ax - b) + \mu^T g(x), \tag{1.4}$$

where λ, and μ are called the dual variables, or Lagrange multipliers, of (1.3).

The dual problem is formulated using this Lagrangian.

Definition 1.4 (Dual problem)**.** The dual problem of (1.3) is given by

$$\underset{(\lambda,\mu)}{\text{maximize}} \quad g(\lambda,\mu) \tag{1.5a}$$

$$\text{subject to} \quad \mu \succcurlyeq \mathbb{0} \tag{1.5b}$$

where $g(\lambda,\mu)$ is the dual function:

$$g(\lambda,\mu) = \inf_x L(x,\lambda,\mu) = \inf_x \left(f(x) + \lambda^T(Ax-b) + \mu^T g(x)\right). \tag{1.6}$$

Definition 1.5 (Primal-dual optimizer)**.** A triplet $(\bar{x},\bar{\lambda},\bar{\mu})$ is a primal-dual optimizer if $\bar{x}$ is an optimizer for the primal problem (1.3), and $(\bar{\lambda},\bar{\mu})$ is an optimizer of the dual problem (1.5).

It is a standard result that for any primal-dual optimizer $(\bar{x},\bar{\lambda},\bar{\mu})$ we have $g(\bar{\lambda},\bar{\mu}) \leq f(\bar{x})$ (Boyd and Vandenberghe, 2004). This is often referred to as weak duality. In some cases equality holds, and then this condition is referred to as strong duality. Multiple constraint qualifications exist under which strong duality is guaranteed. Slater's condition is one of those

Definition 1.6 (Slater's conditions)**.** There exists $x \in \mathbb{R}^n$ such that

$$\begin{aligned} Ax &= b \\ g_i(x) &\leq 0 \quad \text{if } g_i(.) \text{ is an affine function} \\ g_i(x) &< 0 \quad \text{if } g_i(.) \text{ is not an affine function} \end{aligned}$$

Proposition 1.1 (Strong duality)**.** *Strong duality holds if Slater's condition is satisfied.*

When strong duality holds, the optimality of both the primal and dual problem can be verified by the first-order optimality conditions, called the Karush-Kuhn-Tucker (KKT) conditions

Lemma 1.3 (KKT optimality conditions (Boyd and Vandenberghe, 2004))**.** *Suppose that Slater's condition holds. Then* $(\bar{x}, \bar{\lambda}, \bar{\mu})$ *is a primal-dual optimizer if and only if it satisfies the KKT optimality conditions*

$$\begin{aligned} \nabla f(\bar{x}) + A^T\lambda + (\nabla g(\bar{x}))^T\bar{\mu} &= \mathbb{0} \\ A\bar{x} &= b \\ \mathbb{0} \preccurlyeq g(\bar{x}) \perp \bar{\mu} &\succcurlyeq \mathbb{0}. \end{aligned}$$

CHAPTER 2

Thermodynamic modeling of heat flows in data centers

ABSTRACT

Analyzing thermodynamics in data centers is a step in the good direction in order to reduce energy consumption of data centers. Constructing a thermodynamical model allows for understanding the heat flows between the cooling infrastructure and the computing infrastructure of the data center. In this chapter we model the temperature changes in the computing equipment as a result of different choices in workload division and cooling efforts. This allows us to set the basis for a framework that can be used to minimize energy consumption through thermal-aware controllers in next chapters.

2.1 Introduction

Ever since the internet was picked up by the general public in the late 1990's, more and more aspects of our societal and business life exist in the digital world. In order to reduce costs of maintaining and operating the digital backbone of our society, companies have turned to data centers to organize their digital infrastructure. A data center is an overarching term for (a large scale) digital infrastructure consisting of computer, server, and networking systems and components. Typically the digital infrastructure is used for storing, processing, and serving large amounts of data to agents interacting with the data center. Data centers offer the benefit of economy of scale by scaling up the amount of equipment such that operational costs can be reduced

greatly. Furthermore, improvements in technology have allowed for increasingly compact equipment, increasing the computational capacity per unit area, therefore increasing the utility of data centers.

One of the largest costs in maintaining a data center is the energy bill of all the equipment housed. Data center power consumption can be split up into three parts: cooling energy consumption, server energy consumption, and support infrastructure energy consumption. How much each of these parts make up of the total energy consumption will vary from data center to data center, but different characterizations can be found in (Emerson Network Power, 2009; Dayarathna, Wen, and Fan, 2016). As the energy bill of a data center is a big part of the operational budget of a data center, a lot of effort is done in finding ways to reduce the total energy consumption of said data centers. In particular the energy spent by the cooling equipment is often a large chunk of the total energy consumption.

Furthermore, as the computational density is increased there is an increasing challenge to maintain the temperature of the data center equipment (Heath et al., 2006). One of the important factors in maintaining a data center is ensuring that the operating temperature of the equipment is within the recommended operating range. Operation above this recommended range increases equipment failure rates and increases power consumption (ASHRAE, 2011). Due to the compactness of the equipment, greater amounts of heat are generated by the equipment, which have to be countered by the appropriate cooling measures.

Therefore we will look into understanding the relation between the temperature of the computing equipment and the heat flows of the cooling installations, and the energy consumption of the cooling system of the data center. In this chapter we will introduce a model that describes the thermodynamics in relation to workload assignment and cooling effort done by the cooling equipment. Also we will introduce a metric to derive the energy consumption of the cooling equipment based on the measured temperature in the data center.

In section 2.2 we will describe a data center in detail, in section 2.3 we will describe what a job is and how to model the power consumption of the server equipment, next in section 2.4 we will derive a model for the temperature changes of the server equipment based on workload division and

cooling set points, and lastly in section 2.5 we will derive a metric for determining the power consumption of the computer room airconditioning unit (CRAC) from the modeled temperatures of the server equipment.

2.2 Data center layout

The main hall of a data center consists of aisles of racks which house the server equipment, the main body of the data center. The physical size of data center equipment is measured in rack units [U], where one rack unit is defined as a component height of 44.50 mm (note that width and depth of the equipment part is ignored). The typical size of one rack in a data center is 42U-48U, or 42-48 rack units, which makes a typical data center rack between 1.80 m and 2.20 m tall. These racks are filled with subunits, or simply units, that can have various sizes such as 1U, 2U, 4U, or 7U. Depending on the size of the unit, it will house an increasing amount of servers.

The cooling of data centers is usually done by air conditioning where cold air is supplied by computer room air conditioning (CRAC) units. The cold air is blown in front of the racks, and fans mounted on the front of the server rack push the cold air to the back of the rack. While passing through the racks, the cold air absorbs the heat produced by the servers. After the air exits the servers, it is extracted and send back to the CRAC units where it is cooled down to the desired supply temperature. To improve the efficiency of the cooling, the racks are organized in aisles which alternate between cold and hot aisles, where the front of the racks is always in the cold aisle, and the back of the rack is always in the hot aisle. Cold aisles denote the aisles where the cold air is entering the data center, and hot aisles denote the aisles where hot air is extracted from the racks. By this separation of hot and cold air, data center operators make sure that the cold air remains as cold as possible before it is blown through the racks. In Figure 2.1 a schematic overview of a data center layout is shown depicting the hot and cold aisles in the data center.

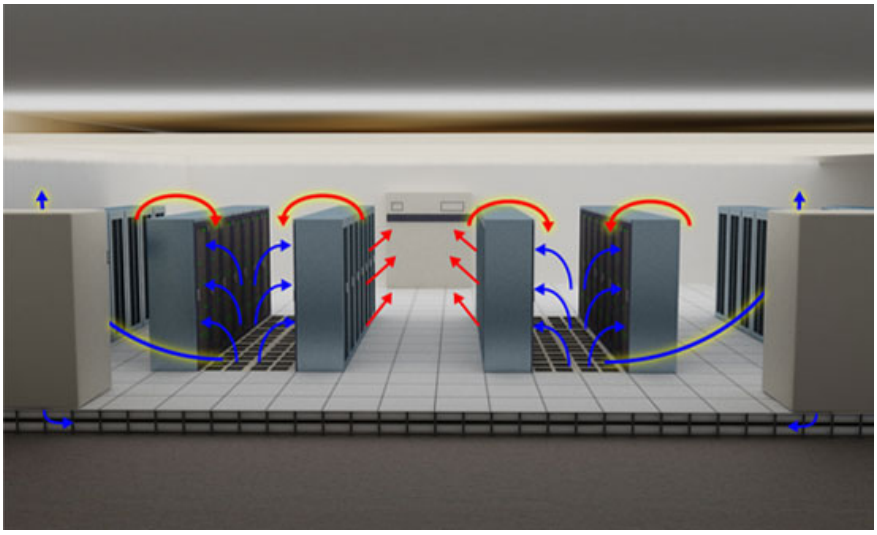


FIGURE 2.1: Schematic layout of a data center where servers are oriented in hot and cold aisles. The cold air (blue arrows) enters the data center in front of the servers, while the hot air (red arrows) exits from the back. Hot air leaks into the cold aisle (red-yellow arrows), creating inefficiencies in the cooling process.

2.2.1 Recirculation flows

Ideally the temperature of the air at the inlet of the racks is equal for all racks in the cold aisle, and is equal to the temperature of the air delivered by the CRAC unit. However due to the complex nature of air flows, variation in inlet air temperature occur (Schmidt, 2004). For example, the cold air enters the cold aisle via perforated tiles. The width of the perforations and the velocity at which the air flows through these perforations have a direct effect on the local rack inlet temperature (Boucher et al., 2006). Secondly so-called recirculated air raises the temperature of the air in the cold aisle, i.e. some of the air from the hot aisle is leaked into the cold aisle (Mukherjee et al., 2007; Tang et al., 2006a).

Every server needs to be cooled below a certain temperature threshold, therefore these temperature variations at the rack inlets cause over-cooling by the CRAC unit. The cooling unit will lower its target supply temperature to make sure that the hottest server will stay below it temperature threshold. The standard CRAC unit however operates at lower efficiencies as discussed in (Moore et al., 2005), and as a result will have a higher energy consumption.

In section 2.4 we will integrate these temperature variations due to re-circulation flows in a thermodynamical model that models the temperature at each cluster of servers in the racks.

2.2.2 Support equipment

Although the most important infrastructural part in the data center is the server equipment, many other components are required to keep the servers running non-stop. For example think about lighting, uninterruptible power supplies (UPS), transformers, switches. In our models we don't include these components as it has been proposed that the power consumption of these components is either fixed, or linearly dependent on the power consumption of the server equipment (Emerson Network Power, 2009).

2.2.3 Computational load

Computational load, or workload, is the general term to denote the work that a data center handles. This work can have different characteristics, i.e.

it could be very computationally demanding, like a large-scale simulation, or it could be large quantities of very small requests, like Google search requests or banking transactions. A different kind of job is a virtual machine, where a client is assigned some network bandwidth and computational capacity which can be used for hosting a website, running servers and services to which a lot of people have to connect, or as a cloud computer.

When a request or job enters the data center, a scheduler automatically assigns the job to a corresponding physical server. This scheduling is done via some scheduling policy, decided by the data center operator. Possible scheduling policies are round robin, each server is given a job in turn, shortest queue, the server with the shortest waiting queue is given the next job, or some more complex decision policies such as thermal-aware strategies. Examples can be found in (Postema and Haverkort, 2018; Hameed et al., 2014). After the server has finished processing the task, the response (if any) is communicated back to the client.

2.2.4 Modeling blocks

Since data centers are very modular in nature, there is a lot of freedom in selecting how to model a data center. A schematic overview of different abstraction layers is given in Figure 2.2.

Depending on how accurate one can, or wants to, measure the temperature of the data center equipment, one can select an abstraction level on which to model the temperature dynamics. As heat flows are involved at a higher abstraction level, it is natural to model the thermodynamics at the rack level or the unit level. However to allow for additional heat variations and heat exchange within the rack itself, we choose to model the thermodynamics at the unit level.

2.3 Server power consumption

The first part we model is the power consumption of the units. Different ways to model the power consumption exist (Dayarathna, Wen, and Fan, 2016), with the main difference being the scope and focus of the models. Some models try to go as close to the CPU level as possible by modeling

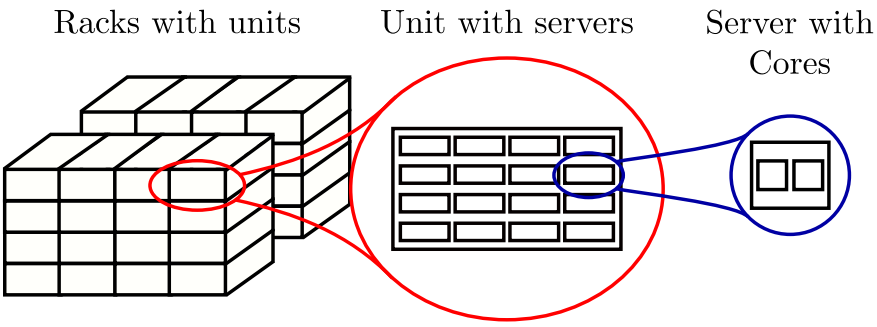


FIGURE 2.2: Schematic overview of different abstraction levels in a data center. Racks consist of several blocks, or units. Units consist of individual servers. Lastly a server can have multiple computing cores.

the power consumption as a function of the CPU clock frequency. While other models aim at modeling the system on a higher level and capture the power consumption of the CPU as a function of the workload applied to the server. The models trade between complexity and detail, where the CPU frequency model captures more details, but results in a non-linear model, and the workload model results in a linear model which operates on a higher level. Before we explain our choice of server power consumption model, we will first explain the notion of a job.

2.3.1 Computational jobs

Requests arriving at the data center are collected by a scheduler which then decides according to some policy how to divide this work among the available units. We assume that each job has an accompanying tag which denotes the time and the number of computing units (CPU) it requires for execution. Let J denote the integer number of jobs that the scheduler has to schedule in the data center at time t. Then $\mathcal{J}(t) = \{1, \cdots, J\}$ denotes the set of jobs to be scheduled at time t. Furthermore let λ_j be the number of CPU's that job j requires at time t. Then the total number of CPU's, D^*, that the scheduler

has to divide over the units at time t is given by

$$D^*(t) = \sum_{j=1}^{\mathcal{J}(t)} \lambda_j. \tag{2.1}$$

We denote by $D_i(t)$ the number of CPU's the schedulers assigns to unit i at time t. These variables are collected in the vector

$$D(t) \triangleq: \begin{pmatrix} D_1(t) & D_2(t) & \cdots & D_n(t) \end{pmatrix}^T.$$

2.3.2 Power consumption of units

Because in this work we abstract away from the inner workings of server, we choose a model on a higher operating level in the data center environment. In our case the linear model fits much better to our situation. This model has been studied many times before and the accuracy loss is small, as it has been found that these models are about 95% accurate (Gao et al., 2013; Li et al., 2012; Dayarathna, Wen, and Fan, 2016; Fan, Weber, and Barroso, 2007; Lauri Minas, 2009; Gupta, Nathuji, and Schwan, 2011; Tang et al., 2006a; Heath et al., 2006; Ranganathan et al., 2006).

Let $P_i(t)$ denote the power consumption of unit i at time t. We model $P_i(t)$ to consist of a load-independent part, e.g. the server consumes a constant amount of power, and a load-dependent part, e.g. the number of CPU's that are actively processing jobs

$$P_i(t) = v_i + w_i D_i(t), \tag{2.2}$$

where v_i [Watts] is the power consumption for the unit being powered on, w_i [Watts CPU^{-1}] is the power consumption per CPU in use. The variables are collected in the vectors

$$P(t) \triangleq: \begin{pmatrix} P_1(t) & P_2(t) & \cdots & P_n(t) \end{pmatrix}^T,$$
$$V \triangleq: \begin{pmatrix} v_1 & v_2 & \cdots & v_n \end{pmatrix}^T,$$

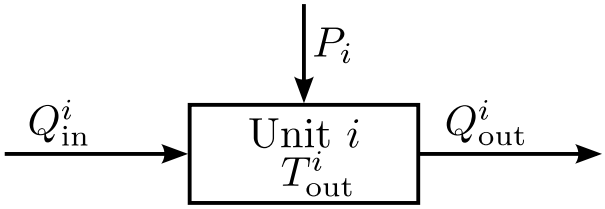


FIGURE 2.3: Heat model of an individual unit. T^i_{out} is the current exhaust air temperature of the unit, Q^i_{in} is the heat entering the unit, Q^i_{out} is the heat exiting the unit and P_i is the power consumption of the unit.

and

$$W \stackrel{\Delta}{=}: \text{diag}\{w_1, \quad w_2, \quad \cdots, \quad w_n\},$$

so that

$$P(t) = V + WD(t). \tag{2.3}$$

2.4 Thermodynamical model

In order to understand how scheduling decisions affect the temperature of the server equipment, and how much cooling we should apply to the data center, we model the temperature dynamics of each individual unit, following similar arguments as in (Vasic, Scherer, and Schott, 2010) and (Tang et al., 2006a). For our model we focus on the temperature of the exhaust air of the units as we study the thermodynamical coupling between the workload that is processed by the servers and the energy efficiency of the cooling equipment. As we will show below there is a direct coupling between the output temperature of the units and both these elements. Furthermore by thermodynamical principles almost all of the energy consumed during computational efforts is dissipated as heat in the unit.

In Figure 2.3 a schematic representation of the heat flows involved is given. The change of temperature of a unit is given by the difference in heat

entering and exiting the unit,

$$m_i c_p \frac{d}{dt} T^i_{\text{out}}(t) = Q^i_{\text{in}}(t) - Q^i_{\text{out}}(t) + P_i(t). \tag{2.4}$$

Here T^i_{out} [°C] is the temperature of the exhaust air at unit i, c_p [J °C^{-1} kg^{-1}] is the specific heat capacity of air, m_i [kg] is the mass of the air inside the unit, Q^i_{in} [Watts] and Q^i_{out} [Watts] are the heat entering and exiting the unit respectively. The heat that enters a unit consists of two parts due to the complex air flows in the data center, i.e. the recirculated air originating from the other units and the cooled air supplied by the CRAC

$$Q^i_{\text{in}}(t) = \sum_{j=1}^{n} \gamma_{ji} Q^j_{\text{out}}(t) + Q^i_{\text{sup}}(t). \tag{2.5}$$

Here Q^i_{sup} [Watts] is the heat supplied by the CRAC to unit i, and $\gamma_{ji} \in [0, 1)$ is the percentage of the flow which recirculates from unit j to unit i. Using thermodynamical principles we find the relation between heat and temperature for each flow

$$Q^i_{\text{in}}(t) = \rho c_p f^i_{\text{in}} T^i_{\text{in}}(t), \tag{2.6}$$

$$Q^i_{\text{out}}(t) = \rho c_p f^i_{\text{out}} T^i_{\text{out}}(t), \tag{2.7}$$

$$Q^i_{\text{sup}}(t) = \rho c_p f^i_{\text{sup}} T_{\text{sup}}(t), \tag{2.8}$$

where ρ [kg m^{-3}] is the density of the air and f^i_{in}, f^i_{out}, f^i_{sup} [m^3 s^{-1}] are the flow rates of the air entering a unit, exiting a unit, and flow rate of the air going from the CRAC to unit i respectively, and T^i_{in}, and T_{sup} [°C] are the temperature of the air at the inlet of a unit, and the supply temperature of the returned air of the CRAC respectively. Note that $f^i_{\text{in}} = f^i_{\text{out}} = f_i$ as we have conservation of mass and we assume that the air entering a unit can only exit at the exhaust of the unit.

Lastly the air flow in a unit is constructed from two parts: the recirculated air from all the units present in the data center, and the air going from

the CRAC to the unit

$$f_i = \sum_{j=1}^{n} \gamma_{ji} f_j + f_{\text{sup}}^i. \tag{2.9}$$

Combining (2.5)-(2.9) with (2.4) yields

$$\begin{aligned} \frac{d}{dt} T_{\text{out}}^i(t) = \frac{\rho}{m_i} \left(\sum_{j=1}^{n} \gamma_{ji} f_j T_{\text{out}}^j(t) - f_i T_{\text{out}}^i(t) \right) \\ + \frac{\rho}{m_i} \left(f_i - \sum_{j=1}^{n} \gamma_{ji} f_j \right) T_{\text{sup}}(t) + \frac{1}{m_i c_p} P_i(t). \end{aligned} \tag{2.10}$$

Rewriting the above relation in matrix form, i.e. combining the temperature changes of all units in one equation, results in

$$\frac{d}{dt} T_{\text{out}}(t) = A(T_{\text{out}}(t) - \mathbb{1} T_{\text{sup}}(t)) + M^{-1} P(t). \tag{2.11}$$

Here

$$T_{\text{out}}(t) \triangleq \left(T_{\text{out}}^1(t) \quad T_{\text{out}}^2(t) \quad \cdots \quad T_{\text{out}}^n(t) \right)^T,$$

and

$$\begin{aligned} A &\triangleq \rho c_p M^{-1} (\Gamma^T - I_n) F, \\ F &\triangleq \text{diag}\{f_1, \quad f_2, \quad \cdots, \quad f_n\}, \\ M &\triangleq \text{diag}\{c_p m_1, \quad c_p m_2, \quad \cdots, \quad c_p m_n\}, \\ \Gamma &\triangleq [\gamma_{ij}]_{n \times n}. \end{aligned}$$

Remark 2.1. It is assumed here that the flow rates remain constant. This assumption allows for modeling the thermodynamical system with a static mapping for the recirculation parameters. Experimental validation of this model can be found in (Tang et al., 2006a). Allowing varying flow rates converts the system to a bilinear system which increases the difficulty of the

theoretical analysis. While this is an interesting extension, this is left for future work.

Property 2.1. Matrix A is Hurwitz.

Proof. As defined above, matrix A is given by

$$A = \rho c_p M^{-1}(\Gamma^T - I_n)F. \tag{2.12}$$

Writing the matrix out in full gives

$$A = \rho \begin{pmatrix} \frac{\gamma_{11}-1}{m_1} f_1 & \frac{\gamma_{21}}{m_1} f_2 & \cdots & \frac{\gamma_{n1}}{m_1} f_n \\ \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{\gamma_{1n}}{m_n} f_1 & \frac{\gamma_{2n}}{m_n} f_2 & \cdots & \frac{\gamma_{nn}-1}{m_n} f_n \end{pmatrix}. \tag{2.13}$$

If we can show that matrix A is strictly diagonal dominant and that the diagonal elements are negative then by the Gerschgorin circle theorem we have shown that matrix A is Hurwitz.

First we will prove strict diagonal dominance of matrix A. Starting from (2.9), and extracting the self-recirculation of a unit from the summation we have

$$f_i = \gamma_{ii} f_i + \sum_{j=1, j\neq i}^{n} \gamma_{ji} f_j + f_{\text{sup}}^i.$$

Hence,

$$\begin{aligned} (\gamma_{ii} - 1) f_i &= - \sum_{j=1, j\neq i}^{n} \gamma_{ji} f_j - f_{\text{sup}}^i \\ &< - \sum_{j=1, j\neq i}^{n} \gamma_{ji} f_j, \end{aligned} \tag{2.14}$$

from which

$$|(\gamma_{ii}-1)f_i| > \left|-\sum_{j=1,j\neq i}^{n}\gamma_{ji}f_j\right| = \sum_{j=1,j\neq i}^{n}\gamma_{ji}f_j, \tag{2.15}$$

because all $\gamma_{ij} \in [0,1)$. Comparing (2.15) with (2.13) and ignoring the mass, as the same mass appears in every row i, we see that matrix A is strictly diagonal dominant.

Furthermore as $\gamma_{ii} \in [0,1)$, we have that all the diagonal elements of A are strictly negative. By Gerschgorin circle theorem, all the eigenvalues of matrix A are strictly negative and therefore the matrix is Hurwitz. □

2.5 Power consumption of CRAC

Having completed the thermodynamical model, we can now model the power consumption of the CRAC. This power consumption depends on the amount of heat that needs to be extracted from the air. This in turn is dependent on the temperature of the air which is returned to the CRAC and the supply temperature it has to provide. The air flow which is returned from unit i to the CRAC is given by

$$f_{\text{sup},i}^{\text{ret}} = \left(1-\sum_{j=1}^{n}\gamma_{ij}\right)f_i. \tag{2.16}$$

Following the same thermodynamical principles as in (2.6)-(2.8), it follows that the heat returned from all the units to the CRAC is

$$Q_{\text{ret}}(t) = \rho c_p \sum_{i=1}^{n}\left(1-\sum_{j=1}^{n}\gamma_{ij}\right)f_i T_{\text{out}}^{i}(t). \tag{2.17}$$

The heat the CRAC sends back to the data center is given by $Q_{\text{sup}}(t) = \rho c_p f_{\text{sup}} T_{\text{sup}}(t)$, where $f_{\text{sup}} = \sum_{i=1}^{n} f_{\text{sup}}^{i}$, and f_{sup}^{i} is obtained from (2.9). With this, the heat the CRAC has to remove from the air, $Q_{\text{rem}}(t)$, is given

by

$$\begin{aligned} Q_{\text{rem}}(t) &= Q_{\text{ret}}(t) - Q_{\text{sup}}(t) \\ &= \rho c_p \sum_{i=1}^{n} \left[\left(1 - \sum_{j=1}^{n} \gamma_{ij} \right) f_i (T^i_{\text{out}}(t) - T_{\text{sup}}(t)) \right] \\ &= -\mathbb{1}^T M A (T_{\text{out}}(t) - \mathbb{1} T_{\text{sup}}(t)). \end{aligned} \tag{2.18}$$

To determine the amount of work the CRAC has to do to remove a certain amount of heat, (Moore et al., 2005) introduced the Coefficient of Performance, $\text{COP}(T_{\text{sup}}(t))$, to indicate the efficiency of the CRAC as a function of the target supply temperature. They found that CRAC units work more efficiently when the target supply temperature is higher. The COP represents the ratio of heat removed to the amount of work necessary to remove that heat. For a water-chilled CRAC unit in the HP Utility Data Center they found that the COP is a quadratic, increasing function. In a general sense the COP can be any monotonically increasing function. The power consumption of the CRAC units can then be given by

$$P_{AC}(T_{\text{out}}(t), T_{\text{sup}}(t)) = \frac{Q_{\text{rem}}(t)}{\text{COP}(T_{\text{sup}}(t))}. \tag{2.19}$$

Assumption 2.1. The function $\text{COP}(T_{\text{sup}})$ of the CRAC unit considered in this work, is monotonically increasing in the range of operation for T_{sup}. □

Example 2.1. Let us consider a small example to illustrate the influence of a small difference in supply temperature on the power consumption of the CRAC. Consider the quadratic $\text{COP}(T_{\text{sup}}(t))$ found by (Moore et al., 2005), and two cases where the returned air has to be cooled down by 5 °C, in the first case from 25 °C to 20 °C and in the second case from 30°C to 25°C. Assume that the energy contained in 5 °C temperature difference of air is 100 Watts. In the first case $\text{COP}(20) = 3.19$ and in the second case $\text{COP}(25) = 4.73$. By (2.19), the energy consumed by the CRAC to cool down the returned air to the required temperature is

$$P_{AC,1} = \frac{100}{3.19} = 31.34 \text{ W}, \quad P_{AC,2} = \frac{100}{4.73} = 21.14 \text{ W}.$$

Here it seen that if the temperature of the returned air increases by 5 °C the power consumption of the CRAC unit decreases by 30%. □

2.6 Conclusions

The cooling infrastructure in data centers account for a large part of the energy consumption of data centers. Improvements in the cooling efficiency of data centers therefore result in big financial gains for data center operators. In this chapter we set up a thermodynamical model that can model temperature changes of the computing infrastructure as a result of different choices in workload division and CRAC supply temperature set points. Furthermore we have given a metric for calculating CRAC energy consumption based on the modeled temperature profile of the computing infrastructure.

The key of the model is the recirculation airflow, that is the leakages which occur when extracting the hot air from the data center back to the CRAC. The heat output of each server affects the temperature of its surrounding servers and as such this has to be taken into account when distributing workload among the servers. In the next chapter we will use the temperature model to set up an optimization problem in order to find the optimal workload division and supply temperature setpoint, and in effect characterizing the thermodynamical inefficiencies of each computing unit.

CHAPTER 3

Asymptotic convergence to optimal interior point using integral control action

ABSTRACT

A general optimization problem is set up to study energy consumption minimization in data centers. An optimal operating point, i.e. optimal job distribution and CRAC cooling set point, is characterized under different loading conditions. Furthermore, under mild assumptions we design controllers that regulate the system to the optimal state without knowledge of the current total workload to be handled by the data center. The response of our controller is validated by simulations and convergence to the optimal set points is achieved under varying workload conditions.

3.1 Introduction

Different studies have been done on energy minimization in data centers based on thermodynamics, some with a more theoretic approach (Vasic, Scherer, and Schott, 2010; Li et al., 2012; Parolini et al., 2012), and others with a heuristic approach (Moore et al., 2005; Tang, Gupta, and Varsamopoulos, 2008; Mukherjee et al., 2009; Banerjee et al., 2011). Other studies focus on energy minimization based on power management strategies (Gaggero and Caviglione, 2014; Postema and Haverkort, 2015; Dai, Wang, and Bensaou, 2016), covering mainly how different scheduling strategies minimize the energy consumption of the server equipment. However a framework which allows both the design of control theory based controllers and an understanding of energy minimal operating conditions seems missing.

In section 3.2 the problem formulation is stated. Following from the problem statement we set up an optimization problem aimed at minimizing energy consumption of the data center in section 3.3. Since the optimization problem is non-convex, the problem is linearized in section 3.4, and its solutions are characterized analytically in section 3.5. Based on the solutions, we design suitable controllers in section 3.6 that steer the system to the energy optimal operating point. Lastly in section 3.7, we simulate the controllers in a real-life data center context obtained from a testbed located at IBM Zurich.

3.2 Problem formulation

The thermodynamical model that has been established can be used to model the temperature changes of the server equipment and model the effect of different choices of workload division and cooling setpoints on the power consumption of the CRAC unit. Now we can set up a framework that can achieve two things: first we can use it to find the optimal operating point for the data center, secondly we can use it to design controllers which ensure convergence of the system to the optimal operating point. The optimal operating point is defined as the optimal workload division, and supply temperature setpoint such that all the incoming workload is processed, the total energy consumption is minimized, and the temperature stays below the safe temperature threshold. Hence the control problem is defined as follows:

Problem 3.1. For system (2.11) design controllers for the workload distribution $D(t)$ and supply temperature $T_{\text{sup}}(t)$ such that, given an unmeasured total load $D^*(t)$, any solution of the closed-loop system is bounded and satisfies

$$\lim_{t\to\infty} (T_{\text{out}}(t) - \bar{T}_{\text{out}}) = 0, \tag{3.1}$$

$$\lim_{t\to\infty} (T_{\text{sup}}(t) - \bar{T}_{\text{sup}}) = 0, \tag{3.2}$$

$$\lim_{t\to\infty} (D(t) - \bar{D}) = 0, \tag{3.3}$$

where $\bar{T}_{\text{out}}$, $\bar{T}_{\text{sup}}$ and $\bar{D}$ are the optimal setpoint values for the temperature distribution, supply temperature and the workload distribution, i.e. power consumption, respectively, which are defined in section 3.3. □

From this point on we will implicitly assume the dependence of the variables on time and only denote it when confusion might arise otherwise.

3.3 General optimization problem

To optimize over the power consumption of the vital infrastructure of the data center, we combine the power consumption of the server equipment, (2.3), and the CRAC unit, (2.19), in a non-convex cost function

$$\mathcal{C}(T_{\text{out}}, T_{\text{sup}}, D) = \frac{Q_{\text{rem}}}{\text{COP}(T_{\text{sup}})} + \mathbb{1}^T P(D). \tag{3.4}$$

We formulate an optimization problem to minimize the power consumption while taking into account the physical constraints of the equipment, i.e the servers only have finite computational capacity and the temperature of the servers cannot exceed a certain threshold. The power consumption of the data center can be written as a combination of two parts, the power consumption of the cooling equipment and the power consumption of the racks.

A reasonable way (Li et al., 2012; Yin and Sinopoli, 2014) to formulate the optimization problem is

$$\begin{aligned}
\min_{T_{\text{out}}, T_{\text{sup}}, D} \quad & \frac{Q_{\text{rem}}}{\text{COP}(T_{\text{sup}})} + \mathbb{1}^T P(D) && \text{(3.5a)}\\
s.t. \quad & D^* = \mathbb{1}^T D && \text{(3.5b)}\\
& \mathbb{0} \preccurlyeq D \preccurlyeq D_{\text{max}} && \text{(3.5c)}\\
& \mathbb{0} = A(T_{\text{out}} - \mathbb{1}T_{\text{sup}}) + M^{-1}P(D) && \text{(3.5d)}\\
& T_{\text{out}} \preccurlyeq T_{\text{safe}}. && \text{(3.5e)}
\end{aligned}$$

Equation (3.5b) ensures that all the available work is divided among the racks, (3.5c) encompasses the computational capacity of the rack, i.e. rack i has D^i_{max} CPU's available at most. The system dynamics should be at steady

state once the optimal point has been reached, see (3.5d), and finally (3.5e) enforces that the temperature of the racks is below the given safe threshold, $T_{\text{safe}} \in \mathbb{R}^n$.

3.4 Equivalent optimization problem for homogeneous data centers

Due to the non-linear nature of how the COP affects the power consumption it is not trivial to analyze the general optimization problem. Although (3.5) is a difficult problem to solve analytically, it is possible to reduce the optimization problem to a simpler equivalent problem for a specific important case. In many of the larger real-life data centers most of the equipment is identical, i.e. the power consumption characteristics of the computational equipment is identical, that is $v_i = v$ and $w_i = w$ for all i in (2.2). It is desirable for data centers to employ identical equipment because this allows for decreased maintenance complexity and allows for bulk purchases of the equipment which reduce operational costs. In this case the data center is said to be composed of homogeneous racks or, more simply, the data center is homogeneous.

In case of a homogeneous data center the power consumption is given by $P(D) = v\mathbb{1} + wD$ and the total computational power consumption is given by

$$\mathbb{1}^T P(D) = nv + w\mathbb{1}^T D = nv + wD^*. \tag{3.6}$$

For this case, the computational power consumption no longer depends on the way the jobs are distributed but only depends on the total workload. This property simplifies the cost function defined in (3.4) considerably.

Theorem 3.1. *Let the data center consist of homogeneous racks, i.e.* $v_i = v$, *and* $w_i = w$ *for all* i *in* (2.2) *and assume constraint* (3.5d) *is satisfied. Then*

problem (3.5) *is equivalent to*

$$\max_{T_{out}} \quad C_1^T T_{out} \tag{3.7a}$$

$$s.t. \quad \mathbb{0} \preccurlyeq C_3 T_{out} + C_4(D^*) \preccurlyeq D_{max} \tag{3.7b}$$

$$T_{out} \preccurlyeq T_{safe}, \tag{3.7c}$$

for suitable C_1, C_3, *and* C_4. □

Before we prove this theorem, we need to introduce some notation and extra preparatory results. In these preparatory results (Lemma 3.1-3.3 below), the homogeneity condition is not required, and statements are given in terms of the power consumption vector P defined as in (2.3).

Lemma 3.1. *Equation* (3.5d) *implies that the following relation holds*

$$\mathbb{1}^T P(D) = -\mathbb{1}^T MA(T_{out} - \mathbb{1}T_{sup}) = Q_{rem},$$

with Q_{rem} *defined in* (2.18). *This reduces cost function* (3.4) *to*

$$\mathcal{C}(T_{out}, T_{sup}, D) = \left(1 + \frac{1}{\mathrm{COP}(T_{sup})}\right) \mathbb{1}^T P(D). \tag{3.8}$$

Proof. By pre-multiplying (3.5d) *by* $\mathbb{1}^T M$ *and solving for* $\mathbb{1}^T P(D)$ *we obtain above result.* □

Lemma 3.2. *If* (3.5b) *and* (3.5d) *are satisfied, then*

$$T_{sup} = C_1^T T_{out} + C_2(D^*), \tag{3.9}$$

$$C_1^T \triangleq \frac{\mathbb{1}^T W^{-1} MA}{\mathbb{1}^T W^{-1} MA\mathbb{1}},$$

$$C_2(D^*) \triangleq \frac{D^* + \mathbb{1}^T W^{-1} V}{\mathbb{1}^T W^{-1} MA\mathbb{1}}.$$

Proof. After pre-multiplying (3.5d) *by* $\mathbb{1}^T W^{-1} M$, *combining with* (3.5b) *and some basic matrix manipulations, the result is obtained.* □

Lemma 3.3. *If* (3.5b) *and* (3.5d) *are satisfied, then*

$$D = C_3 T_{out} + C_4(D^*), \tag{3.10}$$
$$C_3 \triangleq: -W^{-1} M A (I_n - \mathbb{1} C_1^T),$$
$$C_4(D^*) \triangleq: W^{-1} M A \mathbb{1} C_2(D^*) - W^{-1} V.$$

Proof. *Substituting the result of Lemma 3.2 in* (3.5d)*, pre-multiplying* (3.5d) *by* $W^{-1}M$*, and solving for* D *yields the result.* □

Remark 3.1. The dimensions of the constants from above Lemmas are $C_1 \in \mathbb{R}^n$, $C_2 \in \mathbb{R}$, $C_3 \in \mathbb{R}^{n \times n}$ and $C_4 \in \mathbb{R}^n$. The following identities for the constants C_1, C_3 and C_4 are observed

$$C_1^T \mathbb{1} = 1, \quad \mathbb{1}^T C_3 = \mathbb{0}^T, \quad C_3 \mathbb{1} = \mathbb{0}, \quad \mathbb{1}^T C_4 = D^*. \tag{3.11}$$

An important consequence worth to note is that the constant $\mathbb{1}^T D$, with D defined as in (3.10), satisfies the identity $\mathbb{1}^T D = D^*$. □

Lemma 3.2 and Lemma 3.3 show that at the steady state the supply temperature, T_{sup}, and workload distribution vector, D, are uniquely defined by the total workload, D^*, and the temperature distribution, T_{out}. With these properties in mind we are ready to prove Theorem 3.1.

Proof. [Proof of Theorem 3.1] Assume that problem (3.5) has a solution. By Lemma 3.1, the cost function reduces to (3.8). By the homogeneity assumption, (3.6) holds, which shows that the cost function (3.8) is independent of the distribution D and depends only on T_{sup}. Hence, in view of Assumption 2.1 (monotonicity of the function $\text{COP}(T_{\text{sup}})$) a solution to problem (3.5) is the one that maximizes T_{sup}. By (3.9) in Lemma 3.2, this solution must maximize the cost function in (3.7a). The constraints in (3.5) and Lemma 3.3 imply the constraints in (3.7), showing that a solution to (3.5) must be also a solution to (3.7).

Conversely, if a solution to (3.7) exists, define D as in (3.10), and notice that (3.5b) is satisfied, as it is promptly verified using the identities (3.11). Then by the homogeneity assumption, (3.5d), Lemma 3.1, and Lemma 3.2,

maximizing the cost function in (3.7a) implies minimizing the cost function in (3.5a). Moreover, the definition of D and the constraint (3.7b) implies (3.5c). Constraint (3.5e) trivially holds because of (3.7c). This ends the proof. □

3.5 Characterization of the optimal solution

In the previous section we have showed the possibility to reduce the optimization problem to a simpler form. In this section we show that using Karush-Kuhn-Tucker (KKT) optimality conditions it is possible to further characterize the optimal point.

3.5.1 KKT optimality conditions

Because the optimization problem (3.7) is convex and all inequality constraints are linear functions we have that Slater's condition holds. Therefore it follows that $\bar{T}_{\text{out}}$ is an optimal solution to (3.7) if and only if there exists $\bar{\mu}, \bar{\mu}_+, \bar{\mu}_- \in \mathbb{R}^n_{\geq 0}$ such that the following set of relations is satisfied (Boyd and Vandenberghe, 2004):

$$-C_1 + \bar{\mu} + C_3^T(\bar{\mu}_+ - \bar{\mu}_-) = \mathbb{0}, \tag{3.12a}$$

$$\mathbb{0} \preccurlyeq C_3\bar{T}_{\text{out}} + C_4(D^*) \preccurlyeq D_{\max}, \tag{3.12b}$$

$$\bar{T}_{\text{out}} \preccurlyeq T_{\text{safe}}, \tag{3.12c}$$

$$\bar{\mu}_+^T(C_3\bar{T}_{\text{out}} + C_4(D^*) - D_{\max}) = 0, \tag{3.12d}$$

$$\bar{\mu}_-^T(-C_3\bar{T}_{\text{out}} - C_4(D^*)) = 0, \tag{3.12e}$$

$$\bar{\mu}^T(\bar{T}_{\text{out}} - T_{\text{safe}}) = 0, \tag{3.12f}$$

$$\bar{\mu}, \bar{\mu}_+, \bar{\mu}_- \succcurlyeq \mathbb{0}. \tag{3.12g}$$

The Lagrangian corresponding to the optimal problem is given by:

$$\begin{aligned}\mathcal{L}(\mu, \mu_+, \mu_-, T_{\text{out}}) = &- C_1^T T_{\text{out}} + \mu^T(T_{\text{out}} - T_{\text{safe}}) \\ &+ \mu_-^T(-C_3T_{\text{out}} - C_4(D^*)) \\ &+ \mu_+^T(C_3T_{\text{out}} + C_4(D^*) - D_{\max}).\end{aligned} \tag{3.13}$$

3.5.2 Characterization of optimal temperature profile

By studying the KKT optimality conditions we can characterize the optimal solution in different cases.

- *Inactive workload constraints:* Every rack is processing some work but not all the processors of each rack are utilized:

$$0 < (C_3\bar{T}_{\text{out}} + C_4(D^*))_i < D^i_{\text{max}} \quad \forall i \in \{1, \cdots, n\}.$$

- *Partially active workload constraints:* In k racks, all processors are processing jobs. The other $n-k$ racks are processing some work but still have processors available:

$$\begin{aligned}(C_3\bar{T}_{\text{out}} + C_4(D^*))_i &= D^i_{\text{max}} \quad \forall i \in \{1, \cdots, k\},\\ 0 < (C_3\bar{T}_{\text{out}} + C_4(D^*))_i &< D^i_{\text{max}} \quad \forall i \in \{k+1, \cdots, n\}.\end{aligned}$$

The optimal temperature profile corresponding to these two cases is summarized in the following two theorems.

Theorem 3.2. *Assume the case that none of the workload constraints are active, i.e.*

$$0 < (C_3\bar{T}_{out} + C_4(D^*))_i < D^i_{max} \quad \forall i \in \{1, \cdots, n\}.$$

The solution to (3.12) *and the optimal solution for the optimization problem* (3.7) *is then given by*

$$\bar{\mu}_+ = \bar{\mu}_- = \mathbb{0}, \quad \bar{\mu} = C_1 \succ \mathbb{0}, \quad \bar{T}_{out} = T_{safe}. \tag{3.14}$$

Proof. Because all the inequality constraints regarding the workload are inactive we have that both $C_3\bar{T}_{\text{out}} + C_4(D^*) - D_{\text{max}} \prec \mathbb{0}$, and $-C_3\bar{T}_{\text{out}} - C_4(D^*) \prec \mathbb{0}$. Then from (3.12d) and (3.12e) we have that $\bar{\mu}_+ = \bar{\mu}_- = \mathbb{0}$. From (3.12a) it follows that $\bar{\mu} = C_1 \succ \mathbb{0}$ such that from (3.12f) we conclude that $\bar{T}_{\text{out}} = T_{\text{safe}}$. □

Theorem 3.3. *In the case that a part of the workload constraints are active, i.e.*

$$\begin{aligned}(C_3\bar{T}_{out} + C_4(D^*))_i &= D^i_{max} \quad \forall i \in \{1, \cdots, k\},\\ 0 < (C_3\bar{T}_{out} + C_4(D^*))_i &< D^i_{max} \quad \forall i \in \{k+1, \cdots, n\},\end{aligned}$$

the solution of (3.12) *is as follows:*

(i) For the racks at the constraint boundary, $i \in \{1, \cdots, k\}$*:*

$$\bar{\mu}^i_- = 0, \quad \frac{C^i_1 + \sum_{j=1, j\neq i}^{k} \bar{\mu}^j_+ \left|C^{ji}_3\right|}{C^{ii}_3} \geq \bar{\mu}^i_+ \geq 0, \tag{3.15}$$

$$\bar{\mu}^i = C^i_1 + \sum_{j=1, j\neq i}^{k} \bar{\mu}^j_+ \left|C^{ji}_3\right| - \bar{\mu}^i_+ C^{ii}_3 \geq 0, \tag{3.16}$$

$$\begin{aligned}\bar{T}^i_{out} = &\frac{D^i_{max} - C^i_4(D^*)}{C^{ii}_3} + \sum_{j=k+1}^{n} \frac{\left|C^{ij}_3\right|}{C^{ii}_3} T^j_{safe}\\ &+ \sum_{j=1, j\neq i}^{k} \frac{\left|C^{ij}_3\right|}{C^{ii}_3} \bar{T}^j_{out}\\ &\leq T^i_{safe}.\end{aligned} \tag{3.17}$$

(ii) For the racks that are not at the constraint boundary, $i \in \{k+1, \cdots, n\}$*:*

$$\bar{\mu}^i_- = \bar{\mu}^i_+ = 0, \tag{3.18}$$

$$\bar{\mu}^i = C^i_1 + \sum_{j=1}^{k} \bar{\mu}^j_+ \left|C^{ji}_3\right| > 0, \tag{3.19}$$

$$\bar{T}^i_{out} = T^i_{safe}. \tag{3.20}$$

□

Before we can prove Theorem 3.3 we need to know more about the structure of C_3.

Property 3.1. Consider C_3 as defined in Lemma 3.3. The off-diagonal terms of this matrix are strictly negative and the diagonal terms are strictly positive.

Proof. The proof of this property can be found in section 3.9 □

Proof of Theorem 3.3. Because part of the workload constraints are at the constraint boundary, the analysis following from the Lagrange multipliers is more involved. First we can say that

$$\begin{aligned}
&\bar{\mu}_-^i = 0 \quad \forall i, \\
&\bar{\mu}_+^i = 0 \quad \forall i \in \{k+1, \cdots, n\}, \\
&\bar{\mu}_+^i \geq 0 \quad \forall i \in \{1, \cdots, k\}.
\end{aligned}$$

Then from (3.12a)

$$\bar{\mu}^i = C_1^i - \sum_{j=1}^{k} \bar{\mu}_+^j C_3^{ji} \quad \forall i \in \{1, \cdots, n\}. \tag{3.21}$$

From Property 3.1 we have that the off-diagonal elements of C_3 are strictly negative. For racks $i \in \{k+1, \cdots, n\}$ we have that the C_3^{ji} elements in (3.21) will always be off-diagonal elements. Therefore rewriting (3.21) gives

$$\bar{\mu}^i = C_1^i + \sum_{j=1}^{k} \bar{\mu}_+^j \left| C_3^{ji} \right| > 0 \quad \forall i \in \{k+1, \cdots, n\}, \tag{3.22}$$

then from (3.12f) it holds that

$$\bar{T}_{\text{out}}^i = T_{\text{safe}}^i \quad \forall i \in \{k+1, \cdots, n\}. \tag{3.23}$$

For racks $i \in \{1, \cdots, k\}$ (3.21) is given by

$$\bar{\mu}^i = C_1^i + \sum_{j=1, j\neq i}^{k} \bar{\mu}_+^j \left| C_3^{ji} \right| - \bar{\mu}_+^i C_3^{ii} \geq 0. \tag{3.24}$$

For (3.24) to hold, it should hold that

$$\frac{C_1^i + \sum_{j=1,j\neq i}^{k} \bar{\mu}_+^j \left|C_3^{ji}\right|}{C_3^{ii}} \geq \bar{\mu}_+^i \quad \forall i \in \{1, \cdots, k\}. \tag{3.25}$$

As the left hand side of (3.25) is strictly positive for all $i \in \{1, \cdots, k\}$, it is possible to find feasible $\bar{\mu}_+^i \geq 0$ such that $\bar{\mu}^i \geq 0$ for all i. It can be shown that $\bar{T}_{\text{out}}^i$ for all $i \in \{1, \cdots k\}$ is given as

$$\begin{aligned}\bar{T}_{\text{out}}^i =& \frac{D_{\text{max}}^i - C_4^i(D^*)}{C_3^{ii}} + \sum_{j=k+1}^{n} \frac{\left|C_3^{ij}\right|}{C_3^{ii}} T_{\text{safe}}^j \\ &+ \sum_{j=1,j\neq i}^{k} \frac{\left|C_3^{ij}\right|}{C_3^{ii}} \bar{T}_{\text{out}}^j \\ &\leq T_{\text{safe}}^i. \end{aligned} \tag{3.26}$$

□

Remark 3.2. One cannot freely choose the k racks for which $D_i = D_{\text{max}}^i$. Whether or not a rack is processing its maximum capacity depends on the data center parameters, i.e. small amount of recirculated air at the input of the rack and low power consumption of the computational equipment. For these racks it holds that

$$\bar{T}_{\text{out}}^i \leq T_{\text{safe}}^i \quad \forall i \in \{1, \cdots, k\}.$$

3.6 Temperature based job scheduling control

As established in section 3.5 it is possible to calculate the optimal solution under the assumption that the total workload at time t, D^*, is known. However it might not always be possible to obtain this quantity. For example when jobs arrive in the data center in some cases it might be hard to assess how much resources these jobs need. Consider the case where a virtual machine is requested by a user. Usually a certain amount of resources are allocated

to this virtual machine, however the user need not use all the available resources all the time. In those situation it is hard to obtain the real workload. In this section we design a controller that is still able to achieve the control goals defined in (3.1)-(3.3) under the assumption that $\mathbb{0} \prec D \prec D_{\max}$. From Theorem 3.2 we see that in this case the optimal solution is always $\bar{T}_{\text{out}} = T_{\text{safe}}$, independent of the way the jobs are distributed. Since most data centers are designed to have overcapacity, usually the computational bounds of the racks will not be reached and this assumption is valid in those setups.

3.6.1 Controller design

We will now design the control inputs for the workload distribution, D, and the supply temperature of the CRAC unit, T_{sup}, while the total workload D^* is unknown. Furthermore the controllers only have access to the measurement of the output temperature of the air at the outlet of each rack, T_{out}. In other words we design temperature feedback algorithms to dynamically adjust D and T_{sup} such that control objectives (3.1)-(3.3) are achieved. The proposed controllers for the supply temperature and the workload distribution are given by

$$\dot{T}_{\text{sup}} = \mathbb{1}^T A^T Z (T_{\text{out}} - T_{\text{safe}}), \tag{3.27}$$

$$\dot{D} = (\frac{\mathbb{1}\mathbb{1}^T}{n} - I_n) B^T Z (T_{\text{out}} - T_{\text{safe}}), \tag{3.28}$$

where A is Hurwitz. Since A is Hurwitz we can find a positive definite matrix Z such that

$$A^T Z + Z A = -2 I_n, \tag{3.29}$$

and B is defined as

$$B = M^{-1} W,$$

where W is defined section 2.3, and A and M are defined in section 2.4. The controllers (3.27) and (3.28) depend only on the output temperature and the system parameters and will continue to vary until the output temperature reaches the safe temperature, which is in line with the control objectives. The

workload controller contains the all-to-all matrix which allows for shifting workload from one server unit to any other server unit without preference. As such, the workload controller will shift jobs between racks based on the temperature deviation until the data center has reached the optimal state. In the results below we discuss the convergence behavior of the controllers in a time frame where the total workload, D^*, is assumed to be constant.

Theorem 3.4. *Let the data center consist of homogeneous racks, i.e.* $v_i = v$, *and* $w_i = w$ *for all* i *in* (2.2), *and assume* $\mathbb{1}^T D(0) = D^*$ *and* D^* *constant. Then the solution of system* (2.11) *with controllers* (3.27) *and* (3.28) *is bounded and converges to the optimal solution of the optimal problem defined in* (3.5) *and therefore satisfies control objectives* (3.1)-(3.3).

Proof. For ease of notation we introduce incremental variables to denote deviations from optimal values

$$\begin{aligned} \tilde{T}_{\text{out}} &= T_{\text{out}} - \bar{T}_{\text{out}}, \\ \tilde{T}_{\text{sup}} &= T_{\text{sup}} - \bar{T}_{\text{sup}}, \\ \tilde{D} &= D - \bar{D}, \end{aligned}$$

where $\bar{T}_{\text{out}} = T_{\text{safe}}$, $\bar{T}_{\text{sup}}$ as in (3.9), and $\bar{D}$ defined as the right-hand side of (3.10). With these definitions, system (2.11) can be rewritten as

$$\dot{\tilde{T}}_{\text{out}} = A\tilde{T}_{\text{out}} - A\mathbb{1}\tilde{T}_{\text{sup}} + B\tilde{D}, \tag{3.30}$$

where A and B are as before

$$\begin{aligned} A &= \rho c_p M^{-1}(\Gamma^T - I_n)F, \\ B &= M^{-1}W. \end{aligned}$$

Define the incremental storage functions as

$$\Xi_1(\tilde{T}_{\text{sup}}) = \frac{1}{2}\left\|\tilde{T}_{\text{sup}}\right\|^2, \tag{3.31}$$

$$\Xi_2(\tilde{D}) = \frac{1}{2}\left\|\tilde{D}\right\|^2. \tag{3.32}$$

The storage functions satisfy

$$\begin{aligned}\dot{\Xi}_1(\tilde{T}_{\text{sup}}, \tilde{T}_{\text{out}}) &= \tilde{T}_{\text{sup}}^T \dot{T}_{\text{sup}} \\ &= \tilde{T}_{\text{sup}}^T \mathbb{1}^T A^T Z \tilde{T}_{\text{out}}, \end{aligned} \tag{3.33}$$

and

$$\begin{aligned}\dot{\Xi}_2(\tilde{D}, \tilde{T}_{\text{out}}) &= \tilde{D}^T \dot{D} \\ &= \tilde{D}^T (\frac{\mathbb{1}\mathbb{1}^T}{n} - I_n) B^T Z \tilde{T}_{\text{out}} \qquad (3.34) \\ &= \tilde{D}^T \frac{\mathbb{1}\mathbb{1}^T}{n} B^T Z \tilde{T}_{\text{out}} - \tilde{D}^T B^T Z \tilde{T}_{\text{out}}. \qquad (3.35)\end{aligned}$$

Note that $\mathbb{1}^T D(t) = D^*$ is satisfied for all $t \geq 0$. In fact, first we notice that $\mathbb{1}^T \dot{D} = 0$ at all times $t \geq 0$. Hence if $\mathbb{1}^T D(0) = D^*$ then $\mathbb{1}^T D(t) = D^*$ for all $t \geq 0$. With this we see that $\tilde{D}^T \mathbb{1} = (D - \bar{D})^T \mathbb{1} = D^* - D^* = 0$ such that (3.35) is reduced to

$$\dot{\Xi}_2(\tilde{D}, \tilde{T}_{\text{out}}) = -\tilde{D}^T B^T Z \tilde{T}_{\text{out}}. \tag{3.36}$$

Now consider the following Lyapunov function $V(\tilde{T}_{\text{out}}) = \frac{1}{2} \tilde{T}_{\text{out}}^T Z \tilde{T}_{\text{out}}$, where Z is defined in (3.29). Then $V(\tilde{T}_{\text{out}})$ satisfies

$$\dot{V}(\tilde{T}_{\text{out}}) = -\left\| \tilde{T}_{\text{out}} \right\|^2 - \tilde{T}_{\text{sup}}^T \mathbb{1}^T A^T Z \tilde{T}_{\text{out}} + \tilde{D}^T B^T Z \tilde{T}_{\text{out}}. \tag{3.37}$$

If we combine the two storage functions with $V(\tilde{T}_{\text{out}})$, then the total Lyapunov function $V_{\text{tot}} = V + \Xi_1 + \Xi_2$ satisfies

$$V_{\text{tot}} = \dot{V}(\tilde{T}_{\text{out}}) + \dot{\Xi}_1(\tilde{T}_{\text{sup}}, \tilde{T}_{\text{out}}) + \dot{\Xi}_2(\tilde{D}, \tilde{T}_{\text{out}}) = -\left\| \tilde{T}_{\text{out}} \right\|^2 \leq 0. \tag{3.38}$$

Since V_{tot} is radially unbounded, (3.38) implies boundedness of the solutions. Using LaSalle's invariance principle this result implies that every solution to the closed loop system initialized as $\mathbb{1}^T D(0) = D^*$ converges to the largest invariant set where $\tilde{T}_{\text{out}} = 0$. Next we show that $\tilde{D}$ and $\tilde{T}_{\text{sup}}$ are zero

on this invariant set. Because $\tilde{T}_{\text{out}}$ is zero, (3.30) reduces to

$$\mathbb{0} = -A\mathbb{1}\tilde{T}_{\text{sup}} + B\tilde{D}. \tag{3.39}$$

Pre-multiplying this by $\mathbb{1}^T B^{-1}$ we get

$$-\mathbb{1}^T\tilde{D} = 0 = -\mathbb{1}^T B^{-1} A\mathbb{1}\tilde{T}_{\text{sup}}, \tag{3.40}$$

and since

$$-\mathbb{1}^T B^{-1} A\mathbb{1} > 0, \tag{3.41}$$

we obtain that $\tilde{T}_{\text{sup}} = 0$. To understand why (3.41) holds true, observe that $A\mathbb{1}$ has all entries strictly negative, as it is immediately deduced from (2.13) and (2.14) in the proof of Property 2.1. Now the inequality easily follows.

With $\tilde{T}_{\text{sup}} = 0$ and with B non-singular it follows from (3.39) that $\tilde{D} = \mathbb{0}$. Hence, the largest invariant set to which the solutions converge is the singleton $(\tilde{T}_{\text{out}}, \tilde{T}_{\text{sup}}, \tilde{D}) = (\mathbb{0}, 0, \mathbb{0})$. Therefore we conclude that system (3.30) with controllers (3.27) and (3.28) satisfies control objectives (3.1)-(3.3), and the state and the inputs of the system converge to the optimal solution. □

The proposed controller for the workload rebalances the workload currently present in the data center. The initial scheduling is assumed to be taken care of by an external entity over which we have no control. This approach is most applicable in cases where the initial scheduling is done in a non-controllable way, e.g. when the scheduling is hard-coded and incoming jobs are scheduled by means of chassis numbers. In these situations the only option available is to move jobs around to drive the data center to the optimal state.

The above result shows guaranteed asymptotic tracking of constant reference signals. However in practice, the controller can handle variations in setpoints, provided that the setpoints change sufficiently slow. In the next section we will study the behavior of our controller under varying setpoints in a real data center context.

3.7 Case study

To evaluate the performance of the proposed controller, we use Matlab to simulate the closed loop system with a synthetic workload trace. For both the data center parameters and the workload trace we use the data presented in (Vasic, Scherer, and Schott, 2010). The data center parameters were obtained from measurements by Vasic et al. at the IBM Zurich Research Laboratory. This data is to our best knowledge the most extensive characterization of the heat recirculation parameters of a data center.

3.7.1 Data center parameters

The simulated data center consists of 30 homogeneous server racks, i.e. the power consumption characteristics, the safe temperature threshold and physical parameters are identical for all 30 racks. The rack model is a Dell PowerEdge 1855, with 10 dual-processor blade servers, i.e. a total of 20 CPU units per rack. The power consumption of the racks is modeled by $P_i(t) = 1728 + 145.5D_i(t)$ (Tang et al., 2006b). The safe threshold temperature is set at 30°C. We supply a synthetic workload trace to the data center, see Figure 3.1. The workload trace is constructed by varying the total workload by $\pm 10\%$ about two nominal values, 40% and 60% of the total data center capacity, representing nighttime and daytime operation levels respectively. The total workload is a piecewise constant function which changes value every 7.5 minutes. Each time the total workload changes new work is added by or released to an external entity over which we assume to have no control. After this update has taken place we observe the change in temperature from the desired temperature profile. When $(T_{\text{out}} - \bar{T}_{\text{out}})$ starts deviating from $\mathbb{0}$ the controllers will act to respond to the changing conditions.

In Figure 3.2, Figure 3.3 and Figure 3.4 the responses of $(T_{\text{out}} - \bar{T}_{\text{out}})$, $(T_{\text{sup}} - \bar{T}_{\text{sup}})$, and $(D - \bar{D})$ respectively for 4 selected racks are shown. To investigate the performance of the controllers we calculated the optimal values for the variables offline and used those to plot the incremental variables. The initial overshoots the Figures depend on the change in total workload between intervals. The larger the change, the larger this initial overshoot

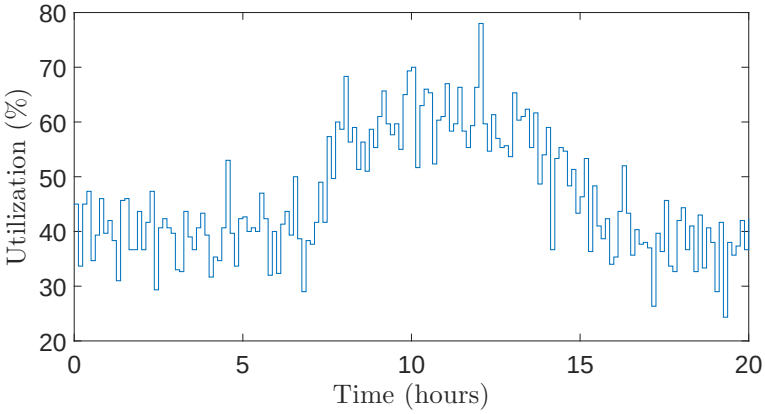


FIGURE 3.1: Synthetic workload trace supplied to data center. The workload varies $\pm 10\%$ about two nominal values, representing nighttime and daytime operation levels. The total workload changes every 7.5 minutes during which the workload is assumed to be constant.

will be. We observe different behavior for the two controllers. The controller for the supply temperature results in very oscillatory behavior for the supply temperature which in turn results in a fluctuating output temperature profile. The controller for the workload division however shows a much smoother response and more gradually steers the workload distribution to the optimal distribution. Every time the workload changes the controllers drive the system back to the optimal value in approximately 0.01 hour = 36 seconds.

In Figure 3.5 the response of $(T_{\text{out}} - \bar{T}_{\text{out}})$ is shown for a larger time interval. In this time interval the total workload changes multiple times and it is seen how, after a very short transient, the controllers steer the temperature of the servers back to the optimal value. This shows that our controllers can cope with variations in total workload.

Although this is a very quick response it is not likely that this convergence time will be attained in practice. In the simulation the cooled air of the CRAC instantly reaches the racks, whereas in a real data center it will

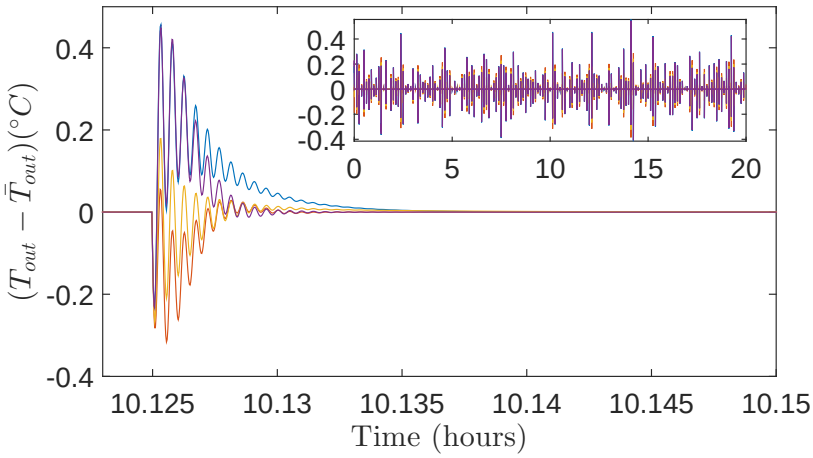


FIGURE 3.2: Plot of the response of $(T_{\text{out}} - \bar{T}_{\text{out}})$ during the simulation for 4 selected racks. The full simulation is shown in the inset and the main plot is a magnification of the response after a change in total workload around $t = 10$ hours. Each time the total workload changes, the temperature of the racks start to deviate from the optimal value and the controllers drive the data center to the new optimal solution, $(T_{\text{out}} - \bar{T}_{\text{out}}) = \mathbb{0}$ again. The oscillatory response of the output temperature coincides with the response of the supply temperature controller. Over the whole simulation the temperature is kept in a bandwidth of ± 0.5 °C around the target temperature distribution.

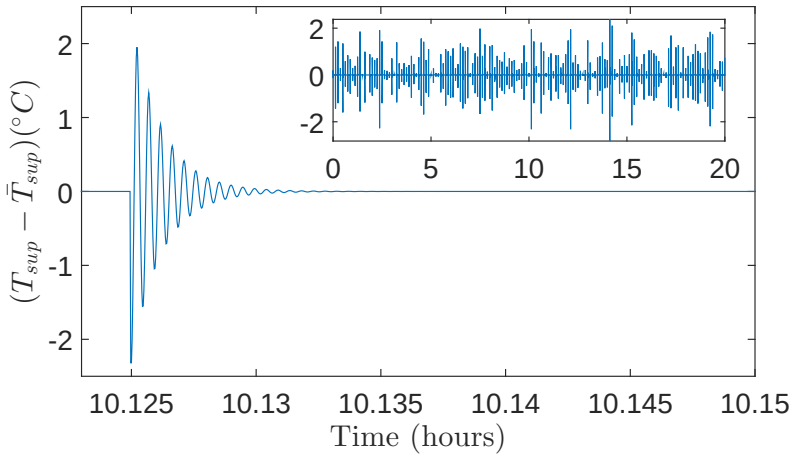


FIGURE 3.3: Plot of the response of $(T_{\text{sup}} - \bar{T}_{\text{sup}})$ during the simulation for 4 selected racks. The full simulation is shown in the inset and the main plot is a magnification of the response after a change in total workload around $t = 10$ hours. The controller successfully drives the system to the new optimal value under varying total workload. The initial overshoot depends on the change of the total workload, i.e. the difference between the optimal supply temperatures in the two intervals. The oscillatory response results in an oscillatory fluctuation in the output temperature profile.

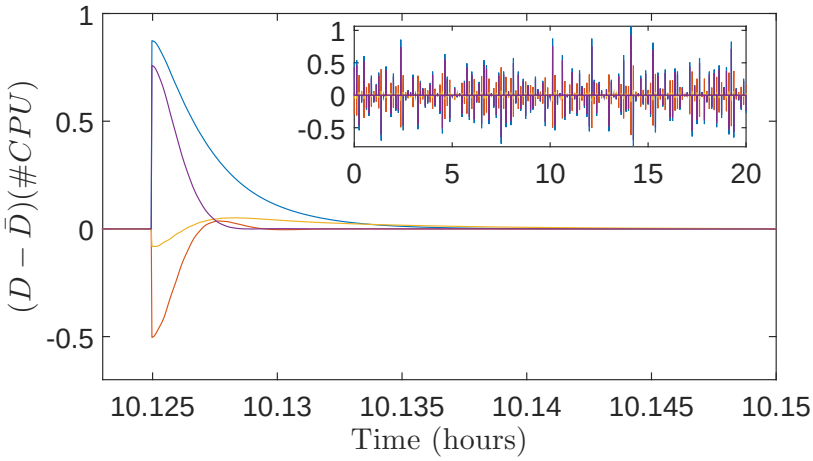


FIGURE 3.4: Plot of the response of $(D-\bar{D})$ during the simulation for 4 selected racks. The full simulation is shown in the inset and the main plot is a magnification of the response after a change in total workload around $t = 10$ hours. The controller drives the system to the optimal value each time the total workload changes. When the total workload changes, an external entity adds or subtracts work from the racks in a non-optimal way which causes an initial overshoot. The controller redistributes the work again to the new optimal workload distribution.

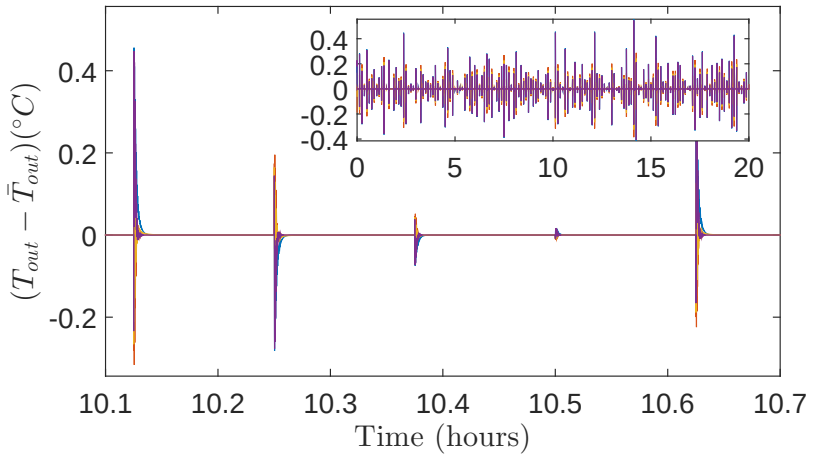


FIGURE 3.5: Plot of the response of $(T_{\text{out}} - \bar{T}_{\text{out}})$ during the simulation for 4 selected racks. The full simulation is shown in the inset and the main plot shows the temperature response over a larger time interval which covers multiple changes in total workload. The fast response of the controllers is clearly visible here and we see that, after a very short transient, the controllers steer the temperature of the servers back to the optimal value.

take some time for the air to travel from the CRAC to the racks. On the contrary the workload division happens on a much shorter timescale, therefore we expect that in practice the output temperature will first increase, as new work is assigned to the rack, and after a certain delay the cooling will start to kick in to drive the temperature profile back to the setpoint.

The supplied workload simulated a day and night cycle to study the response of the controller under large varying loads. From the results we see no difficulty for the controller to handle these different conditions. We conclude that the controller is able to keep the temperature of the racks around the target setpoint under all load conditions.

3.8 Conclusions

Many papers on thermal-aware job scheduling have studied the topic from a practical perspective, however a theoretical analysis has less often been done. In this work we describe data centers and corresponding thermodynamics in a control theoretical fashion combining optimization theory with controller design.

We have studied the minimization of energy consumption in a data center where recirculation of airflow is present, i.e. inefficiencies in cooling of the racks, through thermal-aware job scheduling and cooling control. We have set up an optimization problem and characterized the optimal workload distribution and cooling temperature to achieve minimum energy consumption while ensuring job processing and thermal threshold satisfaction. In addition we have presented controllers that track a reference signal and are able to drive the control and state variables to the optimal values. Furthermore simulations show that the controllers can work with varying workload conditions as the convergence time of the controllers is significantly faster than the frequency of the workload variation.

We have shown that it is possible to uniquely determine the optimal cooling supply temperature and workload distribution as a function of the total workload and desired temperature distribution of the racks in the data center. Furthermore we have shown that the optimal temperature distribution can be analytically calculated and that this distribution is independent

of the workload distribution if none of the racks reaches its computational capacity.

With the assumption that none of the racks is at its computational capacity we have designed controllers that control the supply temperature and workload distribution to drive the data center to the optimal state.

3.9 Proofs

Proof of Property 3.1. From Lemma 3.3 we have that

$$C_3 = -W^{-1}MA(I_n - \mathbb{1}C_1^T),$$

where

$$C_1^T = \frac{\mathbb{1}^T W^{-1} MA}{\mathbb{1}^T W^{-1} MA\mathbb{1}}.$$

Defining a temporary variable $\alpha = W^{-1}MA$ we can write C_3 as

$$C_3 = -\alpha + \frac{1}{\mathbb{1}^T\alpha\mathbb{1}}\alpha\mathbb{1}\mathbb{1}^T\alpha.$$

The ij-th component of C_3 is then given by

$$C_3^{ij} = -\alpha_{ij} + \frac{\sum_{l=1}^n \alpha_{il} \sum_{k=1}^n \alpha_{kj}}{\sum_{l=1}^n \sum_{k=1}^n \alpha_{lk}}. \tag{3.42}$$

From the definition of α we find that the ij-th component of α is given by

$$\alpha_{ij} = c_p\rho\frac{1}{w_i}(\gamma_{ji} - \delta_{ji})f_j, \tag{3.43}$$

where δ_{ji} is the Kronecker delta, which is 1 if $i = j$ and 0 otherwise. To simplify the mathematics a little from now on, we assume that the data center consists of homogeneous racks, see (3.6). Combining (3.43) with (3.42) we have

$$C_3^{ij} = -c_p\rho\frac{1}{w}\Bigg((\gamma_{ji}-\delta_{ji})f_j + \frac{(f_i-\sum_{l=1}^{n}\gamma_{li}f_l)\,(f_j-\sum_{k=1}^{n}\gamma_{jk}f_j)}{\sum_{l=1}^{n}(f_l-\sum_{k=1}^{n}\gamma_{kl}f_k)}\Bigg). \tag{3.44}$$

Although the big fraction in (3.44) looks a bit daunting it is actually easy to conceptually understand it. The airflow at the inlet of the rack consists of two parts, air coming from the CRAC unit and air recirculating from other racks to the rack in question. At the outlet of the rack the airflow is composed of the air going back to the CRAC unit and the air recirculating from the rack in question to all the other racks. Looking closer at the numerator of (3.44) we see that the first half is the air flowing from the CRAC unit to rack i, and the second half is the air flowing from rack j to the CRAC unit. The denominator is the sum of the airflow each rack receives from the CRAC unit which is equal to the supplied airflow, f_{sup}. In this way we can simplify (3.44) to

$$C_3^{ij} = -c_p\rho\frac{1}{w}\left((\gamma_{ji}-\delta_{ji})f_j + \frac{f_{\text{(CRAC to i)}}f_{\text{(j to CRAC)}}}{f_{\text{sup}}}\right). \tag{3.45}$$

Now in the case that $i \neq j$, (3.45) is reduced to

$$C_3^{ij} = \underbrace{-c_p\rho\frac{1}{w}}_{<0}\left(\underbrace{\gamma_{ji}f_j}_{>0} + \underbrace{\frac{f_{\text{(CRAC to i)}}f_{\text{(j to CRAC)}}}{f_{\text{sup}}}}_{>0}\right) < 0. \tag{3.46}$$

Here we see that the off-diagonal terms of C_3 are strictly negative.

As for the diagonal terms, $i = j$, we have

$$C_3^{ii} = c_p\rho\frac{1}{w}\left((1-\gamma_{ii})f_i - \frac{f_{\text{(CRAC to i)}}f_{\text{(i to CRAC)}}}{f_{\text{sup}}}\right). \tag{3.47}$$

Since

$$(1-\gamma_{ii})f_i = \underbrace{f_i - \sum_{l=1}^{n}\gamma_{li}f_l}_{f_{\text{(CRAC to i)}}} + \sum_{l=1,l\neq i}^{n}\gamma_{li}f_l, \tag{3.48}$$

we have that

$$C_3^{ii} = \underbrace{c_p\rho\frac{1}{w}}_{>0}\Bigg(\underbrace{\sum_{l=1,l\neq i}^{n}\gamma_{li}f_l}_{>0} + \underbrace{f_{\text{(CRAC to i)}}}_{>0}\Bigg(\underbrace{1-\frac{f_{\text{(i to CRAC)}}}{f_{\text{sup}}}}_{>0}\Bigg)\Bigg) > 0. \tag{3.49}$$

In (3.49) we see that the diagonal terms of C_3 are strictly positive. This concludes the proof. □

CHAPTER 4

Solving linear constrained optimization problems under hard constraints using projected dynamical systems

ABSTRACT

This chapter studies the convergence of projected primal-dual dynamics under mild conditions on the (general) optimization problem. In particular, we do not require strict convexity of the objective function nor uniqueness of the optimizer. By regarding inequality constraints as hard constraints, we construct a suitable primal-dual dynamics in the complementarity formalism. We establish pointwise asymptotic stability of the set primal-dual optimizers by a suitable invariance principle involving two different Lyapunov functions. In addition, we show how these results can be applied for online optimization in data centers.

4.1 Introduction

The focus of the research up to this point was designing controllers that react dynamically to continuously varying data center conditions. The controllers designed in chapter 3 work in a large operating range by having a fixed thermal setpoint. However outside this operating range the fixed setpoint is no longer valid and the controllers give faulty results.

In order to extend the usability of the controllers, it is a natural step to investigate a way to dynamically adjust the controller setpoint in reaction to changing operating conditions. As the setpoint follows from the solution of

the optimization problem described in section 3.3, this can be achieved by dynamically solving this optimization problem.

Besides variable operating conditions, it is also possible to have power state switching of the computational equipment. Power state switching has the potential for large reductions in energy consumption, see chapter 5, so incorporating this behavior in our control algorithms increases their utility greatly. Power state switching alters the constraints to the optimization problem, which in turn affects the solution of the optimization problem. Dynamically solving for the desired setpoint, allows the algorithms to cope with variable system parameters as well.

The (constrained) primal-dual dynamics is a well-known continuous-time algorithm for determining the primal-dual optimizers of a constrained convex optimization problem. The research on these dynamics has a rich history starting from the classical work of (Arrow et al., 1958) and has regained interest in the last decade, see for example (Jokić, Lazar, and Bosch, 2009; Cherukuri, Mallada, and Cortés, 2016; Goebel, 2017; Feijer and Paganini, 2010). In particular, the passivity property that the primal-dual dynamics naturally admits (Stegink, De Persis, and Van Der Schaft, 2015) has been exploited in numerous applications including network flow control (Wen and Arcak, 2004), power networks (Stegink, De Persis, and Van Der Schaft, 2017), data centers (Van Damme, De Persis, and Tesi, 2018) and energy efficient buildings (Hatanaka et al., 2017).

However, throughout the literature several assumptions on the underlying optimization problem are typically made. Firstly, most works consider *soft* constraints meaning that the constraints may be violated throughout execution of the algorithm. However, this may not be feasible when considering for example (input) saturation or non-negativity constraints. In addition, in the previous mentioned references strict convexity of the objective function is required for the stability analysis. An exception is the work of (Richert and Cortés, 2015), but here (i) linear programs are considered which (ii) are in standard form.

In this chapter we relax some of the commonly made assumptions in the literature while retaining the asymptotic stability properties of the primal-dual dynamics. More specifically, the contributions are summarized as follows.

1. We consider *hard* inequality constraints, that is, constraints that may not be violated throughout execution of the algorithm.

2. A general form of the (in)equality constraints is considered, not only (decoupled) box constraints.

3. Only convexity is required for the objective function, capturing also the special case of linear programs.

4. We do not assume uniqueness of the optimal point. Instead, we establish convergence

   (a) to the set of primal-dual optimizers.

   (b) to a point within this set.

5. We show the results can be used for online thermal-aware job scheduling in data centers.

In the problem setup, we consider a general constrained convex optimization problem and write the associated primal-dual dynamics as a complementarity system. By implicitly using the equivalence with evolutionary variational inequalities and projected dynamical systems as shown in (Brogliato et al., 2006), we can show that there exists a unique (slow) solution of the primal-dual dynamics, which in addition is continuous with respect to the initial condition. These properties of the dynamics are exploited to establish pointwise asymptotic stability of the set of primal-dual optimizers.

In the last part of this chapter, we apply the suggested primal-dual algorithm to find the setpoint for the thermal-aware controller designed in chapter 3 and simulate the interconnection of the primal-dual algorithm and the integral controllers in the data center framework considered in this thesis.

4.2 Convergence of projected primal-dual dynamics

We consider a convex optimization problem of the form

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\ f(x) \tag{4.1a}$$

$$\text{subject to}\ \ Ax = b \tag{4.1b}$$

$$g(x) \preccurlyeq 0, \tag{4.1c}$$

with $g(.) = \begin{bmatrix} g_1(.) & \dots & g_q(.) \end{bmatrix}$, $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$. The inequality (4.1c) holds element-wise. For problem (4.1) we assume the following.

Assumption 4.1 (Convexity and Slater's condition)**.**
$f, g_1, \dots, g_q : \mathbb{R}^n \to \mathbb{R}$ are continuously differentiable convex functions and there exists an $x \in \mathbb{R}^n$ such that $Ax = b$, and $g_i(x) < 0, \forall i = 1, \dots, q$. □

In particular, Assumption 4.1 ensures that strong duality of problem (4.1) holds, see (Boyd and Vandenberghe, 2004). As a result $\bar{x} \in \mathbb{R}^n$ is an optimizer of (4.1) if and only if there exists $\bar{\lambda} \in \mathbb{R}^m, \bar{\mu} \in \mathbb{R}^q$ such that the Karush–Kuhn–Tucker (KKT) optimality conditions of (4.1), which are given by

$$\begin{aligned} \mathbb{0} &= \nabla f(\bar{x}) + A^T\bar{\lambda} + \nabla g(\bar{x})\bar{\mu}, \\ \mathbb{0} &= A\bar{x} - b, \\ \mathbb{0} &\succcurlyeq g(\bar{x}) \perp \bar{\mu} \succcurlyeq \mathbb{0}, \end{aligned} \tag{4.2}$$

are satisfied. Here $\nabla g(.) = \begin{bmatrix} \nabla g_1(.) & \dots & \nabla g_q(.) \end{bmatrix}$. It will be convenient later to define the set of optimal points by

$$\Omega = \{(\bar{x}, \bar{\lambda}) : \exists \bar{\mu} \in \mathbb{R}^q \text{ such that (4.2) holds}\} \subset \mathbb{R}^{n+m}.$$

In the sequel, we assume that there exists at least one primal-dual triple $(\bar{x}, \bar{\lambda}, \bar{\mu})$ satisfying (4.2), i.e., $\Omega \neq \emptyset$. Based on the KKT conditions (4.2), we propose the following projected primal-dual dynamics[1] to deal with the

[1] Although (4.3) is represented in the complementarity formalism, we will refer to it as a projected system for notational convenience and its equivalence with a projected dynamical system is shown later.

hard constraints (4.1c).

$$\dot{x} \overset{\text{a.e.}}{=} -\nabla f(x) - A^T\lambda - \nabla g(x)\mu - A^T\Xi(Ax-b), \tag{4.3a}$$

$$\dot{\lambda} \overset{\text{a.e.}}{=} Ax - b, \tag{4.3b}$$

$$\mathbb{0} \overset{\text{a.e.}}{\preccurlyeq} g(x) \perp \mu \overset{\text{a.e.}}{\succcurlyeq} \mathbb{0}. \tag{4.3c}$$

Here 'a.e.' stands for almost everywhere, and $\Xi \in \mathbb{R}^{m\times m}$ is a positive definite matrix. Note that the last term of (4.3a) does not alter the equilibria of (4.3). Moreover, this *augmented* term improves the convergence rate of the dynamics (see e.g. Simpson-Porco, 2016) and allows for weaker assumptions on the objective function for the convergence as we will show later. The state variables x, λ are denoted compactly as $\mathtt{x} := (x, \lambda) \in \mathbb{R}^{\mathtt{n}}$, with $\mathtt{n} = m + n$. As observed from (4.3), the (x, λ)-dynamics are projected on the closed convex set $K = \{\mathtt{x} \in \mathbb{R}^{\mathtt{n}} \,:\, g(x) \preccurlyeq \mathbb{0}\}$. Furthermore note that the set of equilibria of (4.3) is identical to $\Omega \subset K$.

The following result guarantees the existence and uniqueness of a solution $\mathtt{x}(t, t_0, \mathtt{x}_0)$ of (4.3) for $t \geq t_0$ and $\mathtt{x}_0 \in K$. Moreover, the unique solution can be proven to be *slow*, that is, $\dot{\mathtt{x}}(t)$ is of minimal norm in the set it belongs to:[2]

$$\begin{aligned} \dot{x} &= -\nabla f(x) - A^T\lambda - A^T\Xi(Ax-b) - \nabla g(x)\mu, \\ \dot{\lambda} &= Ax - b, \end{aligned} \tag{4.4}$$

$$\mu \in \underset{\substack{\hat{\mu}_i \geq 0, i\in I(x)\\ \hat{\mu}_i = 0, i\notin I(x)}}{\arg\min} \left\| \nabla f(x) + A^T\lambda + A^T\Xi(Ax-b) + \nabla g(x)\hat{\mu} \right\| \tag{4.5}$$

with $I(x) := \{i \,:\, g_i(x) = 0\}$ and $\dot{\mathtt{x}} \equiv \dot{\mathtt{x}}(t; t_0, \mathtt{x}_0), \mu \equiv \mu(t)$.

Proposition 4.1 (Existence and uniqueness of solutions)**.**
Let Assumption 4.1 hold. Then for each $\mathtt{x}_0 \in K$, there exists a unique solution $\mathtt{x}(t; t_0, \mathtt{x}_0) \in C^0([t_0, \infty); \mathbb{R}^{\mathtt{n}})$ of (4.3)*, which is slow and right-differentiable on $[t_0, \infty)$.*

[2] Note that by exploiting closedness and convexity of (4.5), at each time t there is a unique $\dot{x}(t)$ (and $\dot{\lambda}(t)$) of minimal norm.

Proof. Let the function F be defined by

$$F(\mathtt{x}) = F(x, \lambda) = \begin{bmatrix} \nabla f(x) + A^T\lambda + A^T\Xi(Ax - b) \\ -(Ax - b) \end{bmatrix}.$$

We observe that F is *hypomonotone*, see (Brogliato and Goeleven, 2005, Remark 3). Then the existence and uniqueness of solutions of the system (4.3) is guaranteed by (Brogliato et al., 2006, Theorem 1) as K is closed and convex, F is a hypomonotone operator and the fact that the constraint qualifications are guaranteed by Slater's condition (Assumption 4.1). □

In addition, the solutions of (4.3) are continuous with respect to the initial condition, which is crucial for showing that the *limit set* $\Lambda(\mathtt{x}_0)$ defined by (4.6) is invariant

$$\Lambda(\mathtt{x}_0) := \{z : \exists\{\tau_i\} \subset [t_0, \infty); \tau_i \to \infty, \mathtt{x}(\tau_i; t_0, \mathtt{x}_0) \to z\}. \tag{4.6}$$

Proposition 4.2 (Continuity w.r.t. the initial condition)**.**
Consider the system (4.2) *and suppose Assumption 4.1 holds. Let* $t \geq t_0$ *be fixed. Then the function*

$$\mathtt{x}(t; t_0, .) : K \to \mathbb{R}^n, \qquad \mathtt{x}_0 \mapsto \mathtt{x}(t; t_0, \mathtt{x}_0), \tag{4.7}$$

is continuous.

Proof. The claim follows from the fact that F is monotone, the equivalence between complementarity systems and evolutionary variational inequalities, and (Brogliato and Goeleven, 2005, Theorem 2). □

Now we come to the main result, which establishes pointwise asymptotic stability of (4.3).

Theorem 4.1 (Convergence of primal-dual dynamics (4.3))**.**
Consider system (4.3) *and let Assumption 4.1 hold. The set of optimizers* Ω *is asymptotically stable. Moreover, the convergence of each trajectory* $\mathtt{x}(t, t_0, \mathtt{x}_0)$ *of* (4.3) *with* $\mathtt{x}_0 \in K$ *is to a point in* Ω*.*

Remark 4.1 (Structure of the proof)**.** The proof of Theorem 4.1 consists of two parts. Firstly, we invoke the usual arguments of the invariance principle

along the lines of (Brogliato and Goeleven, 2005) to show convergence to the nonempty limit set. Here we exploit the properties of the complementarity system which allows for a more convenient and shorter proof. For completeness, we include the full proof of this result. In the second part of the proof we use ideas from (Arsie and Ebenbauer, 2010) to further characterize the limit set and to show that it is contained in the set of equilibria. We finalize the proof by showing that the convergence is to a point.

Proof. Let $\bar{\mathtt{x}} := (\bar{x}, \bar{\lambda}) \in \Omega$ and let $\mathtt{x}_0 := (x_0, \lambda_0) \in K$ be given. We show first that limit set $\Lambda(\mathtt{x}_0)$ is invariant.

Invariance of $\Lambda(\mathtt{x}_0)$: Let $z \in \Lambda(\mathtt{x}_0)$ be given. Then there exists a time sequence $\tau_i, i = 1, 2, \ldots$ with $\tau_i \to \infty$ as $i \to \infty$ such that $\lim_{i\to\infty} \mathtt{x}(\tau_i; t_0, \mathtt{x}_0) = z$. Let $\tau \geq t_0$ be given. By continuity w.r.t. the initial conditions (Proposition 4.2) we have $\lim_{i\to\infty} \mathtt{x}(t; t_0, \mathtt{x}(\tau_i; t_0, \mathtt{x}_0)) = \mathtt{x}(t; t_0, z)$. Then by the uniqueness of solutions (Proposition 4.1) we have $\mathtt{x}(\tau; t_0, \mathtt{x}(\tau_i; t_0, \mathtt{x}_0)) = \mathtt{x}(\tau - t_0 + \tau_i; t_0, \mathtt{x}_0)$ and therefore $\lim_{i\to\infty} \mathtt{x}(\tau - t_0 + \tau_i; t_0, \mathtt{x}_0) = \mathtt{x}(\tau, t_0, z)$. Setting $w_i = \tau - t_0 + \tau_i$ we see that $w_i \geq t_0, w_i \to \infty$ and $\mathtt{x}(w_i; t_0, \mathtt{x}_0) \to \mathtt{x}(\tau; t_0, z)$. Thus $\mathtt{x}(\tau; t_0, z) \in \Lambda(\mathtt{x}_0)$.

Limit points correspond to sublevel set of V: Consider the function $V(\mathtt{x}) = V(x, \lambda) = \frac{1}{2} \|x - \bar{x}\|^2 + \frac{1}{2} \left\|\lambda - \bar{\lambda}\right\|^2$, then there exists a compact sublevel set Ψ of V such that $\mathtt{x}_0 \in \Psi$ since V is radially unbounded. We claim that

$$V(y) = k, \quad \forall y \in \Lambda(\mathtt{x}_0). \tag{4.8}$$

Let $T > 0$ be given. Let us define the mapping $V^* : [t_0, \infty) \to \mathbb{R}$ by $V^*(t) = V(\mathtt{x}(t; t_0, \mathtt{x}_0))$. The function $\mathtt{x}(.) \equiv \mathtt{x}(.; t_0, \mathtt{x}_0)$ is absolutely continuous on $[t_0, t_0 + T]$ and thus V^* is a.e. strongly differentiable on

$[t_0, t_0 + T]$. Specifically, by writing $x = x(t), \lambda = \lambda(t)$, we have

$$\begin{aligned}
&\frac{dV^*}{dt}(t) = \langle \nabla V(\mathtt{x}(t)), \frac{d\mathtt{x}}{dt}(t) \rangle = -(x - \bar{x})^T (\nabla f(x) + A^T \lambda) \\
&-(x - \bar{x})^T (A^T \Xi (Ax - b) + \nabla g(x)\mu) + (\lambda - \bar{\lambda})^T (Ax - b) \\
&\overset{(4.2)}{=} -(x - \bar{x})^T (\nabla f(x) - \nabla f(\bar{x}) + \nabla g(x)\mu - \nabla g(\bar{x})\bar{\mu}) \\
&- \|Ax - b\|_\Xi^2 + (x - \bar{x})^T (A^T(\lambda - \bar{\lambda}) - A^T(\lambda - \bar{\lambda})) \qquad (4.9) \\
&\preccurlyeq -(x - \bar{x})^T (\nabla f(x) - \nabla f(\bar{x})) + g(x)\bar{\mu} + g(\bar{x})\mu \\
&- \|Ax - b\|_\Xi^2 \preccurlyeq \mathbb{0}, \qquad \text{a.e. } t \in [t_0, t_0 + T].
\end{aligned}$$

We have $\mathtt{x} \in C^0([t_0, t_0 + T]; \mathbb{R}^{\mathtt{n}})$, $\frac{d\mathtt{x}}{dt} \in L^\infty(t_0, t_0 + T; \mathbb{R}^{\mathtt{n}})$ and $V \in C^1(\mathbb{R}^{\mathtt{n}}; \mathbb{R})$. It follows that $V^* \in W^{1,1}(t_0, t_0 + T; \mathbb{R}^{\mathtt{n}})$ and thus V^* is non-increasing on $[t_0, t_0 + T]$. Since T has been chosen arbitrary, V^* is non-increasing on $[t_0, \infty)$. By continuity of $\mathtt{x}(t)$ it then follows that the *orbit* $\gamma(\mathtt{x}_0) := \{\mathtt{x}(\tau; t_0, \mathtt{x}_0); \tau \geq t_0\}$ satisfies $\gamma(\mathtt{x}_0) \subset \Psi \cap K$ as Ψ is a compact sublevel set of V. It results that

$$\lim_{\tau \to \infty} V(\mathtt{x}(\tau; t_0, \mathtt{x}_0)) = k,$$

for some $k \in \mathbb{R}$. Let $y \in \Lambda(\mathtt{x}_0)$. There exists $\{\tau_i\} \subset [t_0, \infty)$ such that $\tau_i \to \infty$ and $\mathtt{x}(\tau_i, t_0, \mathtt{x}_0) \to y$. By continuity,

$$\lim_{i \to \infty} V(\mathtt{x}(\tau_i; t_0, \mathtt{x}_0)) = V(y). \qquad (4.10)$$

Therefore, $V(y) = k$. Here, y has been chosen arbitrary in $\Lambda(\mathtt{x}_0)$ and thus (4.8) holds. In addition, the set $\gamma(\mathtt{x}_0)$ is bounded and thus $\Lambda(\mathtt{x}_0)$ is non-empty and

$$\lim_{\tau \to \infty} d(\mathtt{x}(\tau, t_0, \mathtt{x}_0), \Lambda(\mathtt{x}_0)) = 0. \qquad (4.11)$$

where $d(x, S)$ denotes the Euclidian distance between $x \in \mathbb{R}^n$ and the set $S \subset \mathbb{R}^n$.

Dynamics on sublevel sets of V: Let $z \in \Lambda(\mathtt{x}_0)$ be given. By the invariance of $\Lambda(\mathtt{x}_0)$ we see that $\mathtt{x}(t; t_0, z) \in \Lambda(\mathtt{x}_0), \forall t \geq t_0$ and thus

$V(\mathrm{x}(t;t_0,z)) = k,\ \forall t \geq t_0$. It results that

$$\frac{d}{dt}V(\mathrm{x}(t;t_0,z)) = 0, \qquad \text{a.e. } t \geq t_0. \tag{4.12}$$

Consequently, by (4.9) we have

$$\begin{aligned} Ax(t) &= b, \qquad & \nabla f(x(t)) &= \nabla f(\bar{x}), \\ g(x(t))\bar{\mu} &= 0, \qquad & g(\bar{x})\mu(t) &= 0, \end{aligned}$$

a.e. $t \geq t_0$ where $\mathrm{x}(t) \equiv \mathrm{x}(t;t_0,z)$. In particular, by (4.3)

$$\begin{aligned} \dot{x}(t) &= c - \nabla g(x(t))\mu(t), \\ \dot{\lambda}(t) &= \mathbb{0}, \\ \mathbb{0} &\succcurlyeq g(x(t)) \perp \mu(t) \succcurlyeq \mathbb{0}, \end{aligned} \tag{4.13}$$

a.e. $t \geq t_0$ where $c = -\nabla f(\bar{x}) - A^T\lambda(t_0)$ is constant. By Proposition 4.1, the unique solution of (4.13) is slow, i.e. at any time $t \geq t_0$, $\mu(t)$ minimizes the norm of $\dot{x}(t)$ (and $\dot{\lambda}(t)$). Therefore $\mu(t)$ is an optimizer of the following problem

$$\underset{\substack{\hat{\mu}_i \geq 0, i \in I(x(t)) \\ \hat{\mu}_i = 0, i \notin I(x(t))}}{\text{minimize}} \{\|c - \nabla g(x(t))\hat{\mu}\|\}, \tag{4.14}$$

where $I(x) := \{i \ : \ g_i(x) = 0\}$, see also (Rosen, 1965) and (Brogliato et al., 2006). For notational convenience we do not explicitly write time-dependency of the variables in the following part of the proof. Instead of considering (4.14), at each time $t \geq t_0$ we can set $\mu_i = 0$ for $i \neq I(x)$ and for $i \in I(x)$ solve the the equivalent minimization problem

$$\begin{aligned} &\underset{\mu_{I(x)}}{\text{minimize}}\ \tfrac{1}{2}\left\|c - \nabla g_{I(x)}(x)\mu_{I(x)}\right\|^2 \\ &\text{subject to } \mu_{I(x)} \succcurlyeq \mathbb{0}, \end{aligned} \tag{4.15}$$

where $g_{I(x)}(.)$ is formed similarly as $g(.)$ but by taking only the g_i's with $i \in I(x)$ and likewise $\mu_{I(x)} = \mathrm{col}_{i\in I(x)}(\mu_i)$ is defined. The Lagrangian of

(4.15) takes the form

$$L = \tfrac{1}{2}\left\|c - \nabla g_{I(x)}(x)\mu_{I(x)}\right\|^2 - \nu_{I(x)}^T \mu_{I(x)},$$

which results in the following KKT optimality conditions.

$$\nabla g_{I(x)}(x)^T (c - \nabla g_{I(x)}(x)\mu_{I(x)}) + \nu_{I(x)} = \mathbb{0}, \tag{4.16}$$

$$\mathbb{0} \preccurlyeq \nu_{I(x)} \perp \mu_{I(x)} \succcurlyeq \mathbb{0}. \tag{4.17}$$

In particular, by premultiplying (4.16) with $\mu_{I(x)}^T$ we have

$$\left\|\nabla g_{I(x)}(x)\mu_{I(x)}\right\|^2 = c^T \nabla g_{I(x)}(x)\mu_{I(x)} \text{ a.e. } t \geq t_0. \tag{4.18}$$

Nonincreasing function W: Define the map $W : \mathbb{R}^{\mathtt{n}} \to \mathbb{R}$ as $W(\mathtt{x}) = -c^T x$, then

$$\begin{aligned}
&\frac{d}{dt} W(\mathtt{x}(t; t_0, z)) = \langle \nabla W(\mathtt{x}), \dot{\mathtt{x}} \rangle = \langle \tfrac{\partial W}{\partial x}(x, \lambda), c - \nabla g(x)\mu \rangle \\
&= -c^T c + c^T \nabla g(x)\mu = -c^T c + c^T \nabla g_{I(x)}(x)\mu_{I(x)} \\
&\overset{(4.18)}{=} -c^T c + 2c^T \nabla g_{I(x)}(x)\mu_{I(x)} - \left\|\nabla g_{I(x)}(x)\mu_{I(x)}\right\|^2 \\
&= -\left\|c - \nabla g_{I(x)}(x)\mu_{I(x)}\right\|^2 \preccurlyeq \mathbb{0}, \quad \text{a.e. } t \geq t_0.
\end{aligned} \tag{4.19}$$

By using the same arguments as before, $W^*(t) := W(\mathtt{x}(t; t_0, z))$ is nonincreasing on $[t_0, \infty)$. Moreover, we have $\gamma(z) \in K \cap \Psi$ and thus W^* is bounded from below on $[t_0, \infty)$. It results that

$$\lim_{\tau \to \infty} W(\mathtt{x}(\tau; t_0, z)) = \alpha, \tag{4.20}$$

for some $\alpha \in \mathbb{R}$.

Limit points correspond to sublevel sets of W: Since $z \in \Lambda(\mathtt{x}_0)$, there exists a sequence $\{\tau_i\} \subset [t_0, \infty)$ such that $\tau_i \to \infty$ and $\lim_{i\to\infty} \mathtt{x}(\tau_i, t_0, \mathtt{x}_0) = z$. By the uniqueness of solutions we have that $\mathtt{x}(\tau; t_0, \mathtt{x}(\tau_i; t_0, \mathtt{x}_0)) = \mathtt{x}(\tau - t_0 + \tau_i; t_0, \mathtt{x}_0)$. By taking the limit $i \to \infty$ and the continuity with respect

to the initial condition we therefore have that

$$\lim_{i\to\infty} \mathtt{x}(\tau; t_0, \mathtt{x}(\tau_i; t_0, \mathtt{x}_0)) = \mathtt{x}(\tau, t_0, z) \\ = \lim_{i\to\infty} \mathtt{x}(\tau - t_0 + \tau_i; t_0, \mathtt{x}_0).$$

As a result, by (4.20)

$$\lim_{\tau\to\infty} W(\lim_{i\to\infty} \mathtt{x}(\tau - t_0 + \tau_i; t_0, \mathtt{x}_0)) = \lim_{t\to\infty} W(\mathtt{x}(t; t_0, \mathtt{x}_0)) = \alpha.$$

By repeating the same arguments as for (4.10), we have

$$W(y) = \alpha \qquad \forall y \in \Lambda(\mathtt{x}_0).$$

Limit points are equilibria: In particular, $W(z) = \alpha$ and

$$W(\mathtt{x}(t; t_0, z)) = \alpha \qquad \forall t \geq t_0,$$

as $\mathtt{x}(t; t_0, z) \in \Lambda(\mathtt{x}_0), \forall t \geq t_0$. It results that

$$\frac{d}{dt} W(\mathtt{x}(t; t_0, z) = \mathbb{0} \qquad \text{a.e. } t \geq t_0,$$

and thus $c = \nabla g_{I(x)}(x)\mu_{I(x)} = \nabla g(x)\mu$ by (4.19), stating that $\dot{\mathtt{x}}(t; t_0, z) = \mathbb{0}, \forall t \geq t_0$. Hence, $z \in \Omega$. Since z was chosen arbitrary, it follows that $\Lambda(\mathtt{x}_0) \subset \Omega$.

Asymptotic stability of Ω: Since $\mathtt{x}_0 \in K, \bar{\mathtt{x}} \in \Omega$ were chosen arbitrary, each point in Ω is Lyapunov stable. In addition, by (4.11) and $\Lambda(\mathtt{x}_0) \subset \Omega$ we obtain for each $\mathtt{x}_0$,

$$\lim_{\tau\to\infty} d(\mathtt{x}(\tau, t_0, \mathtt{x}_0), \Omega) = 0. \tag{4.21}$$

Hence, the set Ω is asymptotically stable. Finally, we show that the convergence is to a point.

Convergence to a point: Let $\mathtt{x}_0 \in K$ and consider the function $\tilde{V}(\mathtt{x}) = \frac{1}{2}\|\mathtt{x} - z\|^2$ where $z \in \Lambda(\mathtt{x}_0) \subset \Omega$. Then there exists a sequence $\{\tau_i\} \subset [t_0, \infty)$ such that $\mathtt{x}(\tau_i; t_0, \mathtt{x}_0) \to z$. Given $\epsilon > 0$, let $k \in \mathbb{Z}$ be such that

$\frac{1}{2}\|\mathtt{x}(\tau_k;t_0,\mathtt{x}_0)-z\|^2 \leq \epsilon$. Then we know that $\frac{1}{2}\|\mathtt{x}(t;t_0,\mathtt{x}_0)-z\|^2 \leq \epsilon$ for all $t \geq \tau_k$ as the sublevel set $\{\mathtt{x} : \tilde{V}(\mathtt{x}) \leq \epsilon\} = \{\mathtt{x} : \frac{1}{2}\|\mathtt{x}-z\|^2 \leq \epsilon\}$ of $\tilde{V}$ is forward invariant by (4.9), taking $V^*(t) = \tilde{V}(\mathtt{x}(t;t_0,\mathtt{x}_0))$. As $\epsilon > 0$ can be taken arbitrary small, we conclude that the convergence is to a point. □

Remark 4.2 (Comparison with (Goebel, 2017))**.** The dynamics (4.3) can be interpreted as special case of the projected saddle point dynamics of (Goebel, 2017). To see this, define

$$H(x,\lambda) = \begin{cases} h(x,\lambda) & \text{if } g(x) \preccurlyeq \mathbb{0} \\ \infty & \text{if } g(x) \not\preccurlyeq \mathbb{0} \end{cases},$$
$$h(x,\lambda) = f(x) + \lambda^T(Ax-b) + \|Ax-b\|_\Xi^2,$$

and note that $\dot{x} = P_{T_{K_x}(x)}(-\nabla_x h(x,\lambda)), \dot{\lambda} = \nabla_\lambda h(x,\lambda)$ with $K_x = \{x \in \mathbb{R}^n : g(x) \preccurlyeq \mathbb{0}\}$ and $T_{K_x}(x)$ denoting the tangent cone at x with respect to K_x. However, it remains an open question whether (Goebel, 2017, Assumption 3.2) holds for this case, which in that work is required for establishing pointwise asymptotic stability.

4.2.1 Primal-dual dynamics with gains

Now we discuss briefly how we can extend the previous results to the modified projected primal-dual dynamics

$$\begin{aligned} L_x L_x^T \dot{x} &\overset{\text{a.e.}}{=} -\nabla f(x) - A^T\lambda - A^T\Xi(Ax-b) - \nabla g(x)\mu, \\ L_\lambda L_\lambda^T \dot{\lambda} &\overset{\text{a.e.}}{=} Ax - b, \\ \mathbb{0} &\overset{\text{a.e.}}{\preccurlyeq} g(x) \perp \mu \overset{\text{a.e.}}{\succcurlyeq} \mathbb{0}. \end{aligned} \tag{4.22}$$

with symmetric gain matrices of the form $L_x L_x^T > 0, L_\lambda L_\lambda^T > 0, L_x \in \mathbb{R}^{n\times n}, L_\lambda \in \mathbb{R}^{m\times m}$. Define $\tilde{x} = L_x^T x, \tilde{\lambda} = L_\lambda^T \lambda$ and define $\tilde{f} : \mathbb{R}^n \to \mathbb{R}$ as $\tilde{x} \mapsto f(L_x^{-T}\tilde{x})$, and $\tilde{g} : \mathbb{R}^n \to \mathbb{R}^{1\times q}$ as $\tilde{x} \mapsto g(L_x^{-T}\tilde{x})$. In addition, let $\tilde{A} = L_\lambda^{-1} A L_x^{-T}, \tilde{b} = L_\lambda^{-1} b, \tilde{\Xi} = L_\lambda \Xi L_\lambda^T, \tilde{\mu} = \mu$. We observe that the

system (4.22) can then be rewritten as

$$\begin{aligned}
&\dot{\tilde{x}} \overset{\text{a.e.}}{=} -\nabla \tilde{f}(\tilde{x}) - \tilde{A}^T \tilde{\lambda} - \tilde{A}^T \tilde{\Xi}(\tilde{A}\tilde{x} - \tilde{b}) - \nabla \tilde{g}(\tilde{x})\tilde{\mu}, \\
&\dot{\tilde{\lambda}} \overset{\text{a.e.}}{=} \tilde{A}\tilde{x} - \tilde{b}, \\
&\mathbb{0} \overset{\text{a.e.}}{\succcurlyeq} \tilde{g}(\tilde{x}) \perp \tilde{\mu} \overset{\text{a.e.}}{\succcurlyeq} \mathbb{0}.
\end{aligned} \tag{4.23}$$

It is easily seen that $\tilde{f}, \tilde{g}_i, i = 1, \ldots, q$ are convex functions. Hence, by applying Theorem 4.1 to (4.23), we establish convergence to an (optimal) equilibrium for both the transformed system (4.23) as well as the original system (4.22) with positive definite gain matrices.

4.2.2 Strict convexity case

The convergence result of Theorem 4.1 relies on the fact that Ξ which appears in (4.3) is a positive definite matrix. Indeed, if this assumption is not satisfied, then oscillations may occur or the trajectories are divergent.

Example 4.1 (No convergence if $\Xi \not\succ 0$)**.** Consider the simple optimization problem

$$\begin{aligned}
&\underset{x \in \mathbb{R}}{\text{minimize}} \;\; x \\
&\text{subject to} \;\; x = 1
\end{aligned}$$

which by (4.3) results in the following primal-dual dynamics

$$\begin{aligned}
\dot{x} &= -1 - \lambda - \Xi(x - 1) \\
\dot{\lambda} &= x - 1
\end{aligned} \tag{4.24}$$

with $\Xi \in \mathbb{R}$. The convergence of (4.24) is guaranteed for $\Xi > 0$ by Theorem 4.1. However, the trajectories are oscillatory for $\Xi = 0$ and divergent for $\Xi < 0$.

However, under the assumption that the objective function f is strictly convex, the convergence result is unaffected for general positive semi-definite Ξ.

Proposition 4.3. *[Convergence of* (4.3) *for* $\Xi \geq 0$*] Consider system* (4.3) *and let Assumption 4.1 hold. Assume furthermore that* f *is strictly convex and* Ξ *is a positive semi-definite matrix. The set of optimizers* Ω *is asymptotically stable. Moreover, the convergence of each trajectory* $\mathtt{x}(t, t_0, \mathtt{x}_0)$ *of* (4.3) *with* $\mathtt{x}_0 \in K$ *is to a point in* Ω*.*

Proof. The proof of Proposition 4.3 is analogous to the proof of Theorem 4.1 with the following changes. Let $(z_x, z_\lambda) \in \Lambda(\mathtt{x}_0)$. Since f is strictly convex it follows by (4.9) and (4.12) that $x(t; t_0, z_x) = \bar{x}$ and $\lambda(t; t_0, z_\lambda) = z_\lambda$ for all $t \geq t_0$. As a result, $\Lambda(\mathtt{x}_0) \subset \Omega$. □

4.3 Data center case study

Now that we have established a primal-dual algorithm for dynamically solving convex optimization problems, we can apply the algorithm in the case of data centers. In Theorem 3.1 we have shown that the optimal data center operating point can be found by solving a linear optimization problem. That optimization problem is given by

$$\min_{T_{\text{out}}} \quad -C_1^T T_{\text{out}} \tag{4.25a}$$

$$s.t. \quad \mathbb{0} \preccurlyeq C_3 T_{\text{out}} + C_4(D^*) \preccurlyeq D_{\max} \tag{4.25b}$$

$$T_{\text{out}} \preccurlyeq T_{\text{safe}}, \tag{4.25c}$$

where C_1, C_3, C_4 are defined in Lemma 3.2 and Lemma 3.3, D^* is the total workload which has to processed by the data center at a given time, T_{safe} is the maximally allowed temperature of the units, $D_{\max}$ is the computational capacity of the units, and $C_3 T_{\text{out}} + C_4(D^*) = D$ is the relation between the chosen temperature profile and the necessary workload allocation to achieve that temperature profile.

The solution to (4.25), $\bar{T}_{\text{out}}$, is the desired temperature distribution which guarantees the minimal energy consumption of the cooling equipment in the data center. With this solution, (4.25b), and a similar relation for T_{sup}, it is then possible to design controllers for the cooling equipment and the

workload distribution, D, which steer the system to the optimal temperature profile.

The controllers for the supply temperature, (3.27), and the workload distribution, (3.28), are derived in chapter 3. These controllers are designed to dynamically adjust T_{sup} and D based on temperature measurements at the units,

$$\begin{aligned}\dot{T}_{\text{sup}} &= \mathbb{1}^T A^T Z(T_{\text{out}} - \bar{T}_{\text{out}}),\\ \dot{D} &= (\frac{\mathbb{1}\mathbb{1}^T}{n} - I_n)B^T Z(T_{\text{out}} - \bar{T}_{\text{out}}),\end{aligned}$$

where the above parameters are defined in section 3.6.

Ideally $\bar{T}_{\text{out}}$ is equal to T_{safe}, however at very high or very low workload levels the computational bounds will cause the optimal solution to deviate from T_{safe}. To allow for these edge cases we apply the proposed primal-dual algorithm from this chapter to find the optimal temperature profile. Adapted to the example of data centers, and augmented with an arbitrary gain $L_{\bar{T}_{\text{out}}} \in \mathbb{R}^{n\times n}$, this algorithm is given by

$$\begin{aligned}L_{\bar{T}_{\text{out}}}\dot{\bar{T}}_{\text{out}} &= C_1 - \begin{bmatrix}-C_3^T & C_3^T & I\end{bmatrix}\mu\\ \mathbb{0} &\preccurlyeq \begin{bmatrix}-C_3\\ C_3\\ I\end{bmatrix}\bar{T}_{\text{out}} + \begin{bmatrix}-C_4(D^*)\\ C_4(D^*) - D_{\max}\\ -T_{\text{safe}}\end{bmatrix} \perp \mu \succcurlyeq \mathbb{0}.\end{aligned} \tag{4.26}$$

4.3.1 Simulation results

To test the performance of the algorithm we simulate a realistic data center setting where a high level of workload is applied to the data center, i.e. 91.7% of the total computing capacity of the data center. The simulations results are given in Figure 4.1 to 4.3. The same simulation setup is used as in chapter 3, where the data center consists of 30 units, each with a maximum allowed temperature of $T_{\text{safe}} = 30$ °C, and a computational capacity of $D_{\max} = 20$ tasks. To have the convergence time within acceptable limits we set the gain $L_{\bar{T}_{\text{out}}} = \frac{1}{20}I_n$. The simulation is initialized relatively far away

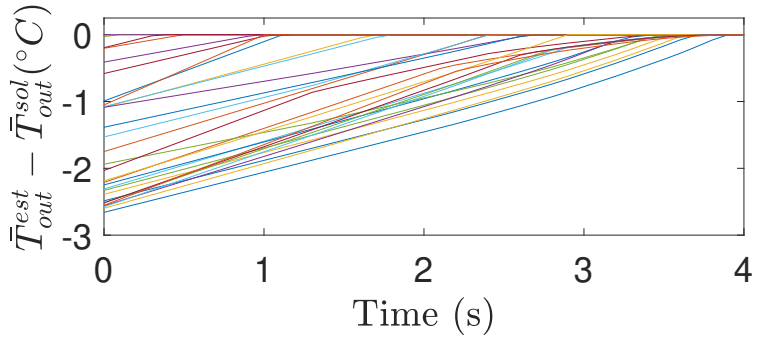


FIGURE 4.1: Convergence of $\bar{T}_{\text{out}}^{est}$ to the solution of optimization problem (4.25), $\bar{T}_{\text{out}}^{sol}$. Within 4 seconds our primal-dual algorithm converges to the real optimal solution.

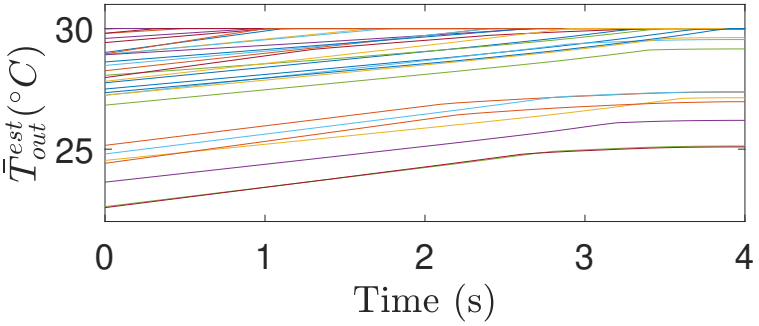


FIGURE 4.2: Evolution of the estimated optimal temperature of each unit. The safe temperature threshold of 30 °C is not violated during the transient.

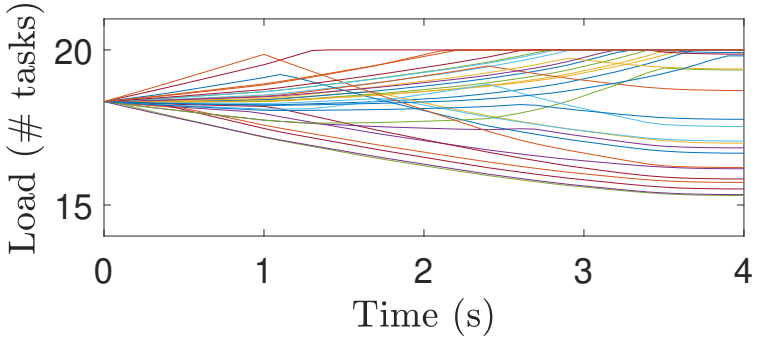


FIGURE 4.3: Evolution of the estimated optimal workload assignment to each unit. The computational capacity of each unit, between 0 and 20 tasks is never violated during transient.

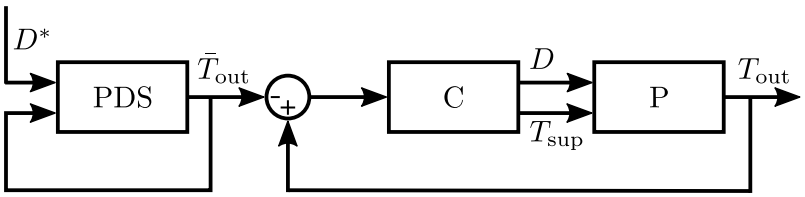


FIGURE 4.4: Interconnection between the primal-dual algorithm (PDS) and the integral controllers (C) and the thermodynamics (P).

from the optimal solution and we see in Figure 4.1 that using our primal-dual algorithm, the estimated optimal solution $\bar{T}_{\text{out}}^{est}$ converges to the real optimal solution of (4.25), $\bar{T}_{\text{out}}^{sol}$, in 4 seconds. To check that the constraints are indeed not violated during the transient, the temperature evolution is plotted in Figure 4.2, and the workload assignment is plotted in Figure 4.3. Here we see that the temperature never exceeds the safe threshold of 30 °C and that the assigned workload never exceeds the computational bound of 20 tasks.

4.4 Interconnection with physical system

Now we have established a primal-dual controller that can dynamically converge to the optimal solution in any situation, we interconnect the primal-dual controller with integral controllers (C), given by (3.27) and (3.28). In Figure 4.4 a schematic of this interconnection is depicted. The integral controllers (C) are connected in feedback with the thermodynamic system (P), and are driven by the reference point $\bar{T}_{\text{out}}$ generated by the primal-dual algorithm (PDS) (4.26). The primal-dual algorithm takes its own output combined with the total workload (D^*) in the data center and converges to the optimal solution, $\bar{T}_{\text{out}}^{sol}$. The mathematical formulation of the total controller

resulting from this interconnection is given by

$$\begin{aligned}
\dot{T}_{\text{sup}} &= \mathbb{1}^T A^T Z(T_{\text{out}} - \bar{T}_{\text{out}}),\\
\dot{D} &= (\frac{\mathbb{1}\mathbb{1}^T}{n} - I_n) B^T Z(T_{\text{out}} - \bar{T}_{\text{out}}),\\
L_{\bar{T}_{\text{out}}} \dot{\bar{T}}_{\text{out}} &= C_1 - \begin{bmatrix} -C_3^T & C_3^T & I \end{bmatrix} \mu,\\
\mathbb{0} &\preccurlyeq \begin{bmatrix} -C_3 \\ C_3 \\ I \end{bmatrix} \bar{T}_{\text{out}} + \begin{bmatrix} -C_4(D^*) \\ C_4(D^*) - D_{\max} \\ -T_{\text{safe}} \end{bmatrix} \perp \mu \succcurlyeq \mathbb{0}.
\end{aligned} \tag{4.27}$$

Currently it remains an open problem whether this interconnection is stable and the temperature distribution and the control inputs will converge to the optimal values, as standard Lyapunov analysis fails to prove stability. Alternatively stability might be found by proving passivity of the integral controllers and the primal-dual controller, however this has not been shown yet. Practically however, the system will be stable if we can guarantee that the primal-dual controller converges quickly enough to the optimal solution. That is, the optimal solution, $\bar{T}_{\text{out}}^{sol}$, is reached fast enough such that the integral controllers are given enough time to converge to the desired setpoint. Given that we can tune the convergence time of the primal-dual controller by adjusting the controller gain, $L_{\bar{T}_{\text{out}}}$, practically this should be possible to achieve.

4.4.1 Simulating interconnection

To study the practical stability of the interconnection, we update the simulation results in section 3.7 to include the primal-dual algorithm. This time we study the response of the interconnection under two different workload traces, similarly as in Figure 6a and 7a in (Vasic, Scherer, and Schott, 2010). The first workload trace is the synthetic workload trace used in section 3.7 and depicted again here in Figure 4.5. This workload trace is characterized by piecewise constant load levels with (larger) jumps between different loads. The second workload trace, Figure 4.6, is based on real workload traces from the NASA Kennedy Space Center web server from Monday, July 3 of 1995. This workload trace is characterized by much smoother and smaller

transitions between different time steps albeit with a much higher jump frequency. For the synthetic workload trace the load level is changed every 7.5 minutes, for the realistic workload trace the load level is changed every 1.5 minute.

As stated before, the practical stability of the interconnection is dependent on whether the workload remains constant long enough for the primal-dual algorithm to converge to the optimal solution. With both workload traces we can study the behavior of the controller under frequent, small jumps, and less frequent, larger jumps. We will study the convergence rate and stability with both workload traces.

In Figure 4.7 to 4.22, the figures on the left side depict the simulation results for the synthetic workload trace and the figures on the right side depict the results for the realistic workload trace. The full simulation is run for all the units in the system, however for clarity the results are shown for 4 selected units. The insets show the full simulation whereas the main window shows a zoom of single transitions in order to see the system behavior in more detail. The simulation starts when the optimal temperature distribution is the ideal distribution, every unit has a desired temperature equal to the maximally allowed temperature of T_{safe}. In the middle of the simulation, the workload level rises such that the optimal temperature distribution deviates from the ideal distribution. Finally load levels lower again and the optimal distribution is again the ideal distribution. In Figure 4.7 and 4.8 the setpoint as calculated by the primal-dual controller is depicted, and in Figure 4.9 and 4.10 the deviation of this setpoint from the optimal setpoint is depicted. We see that the larger jumps in the synthetic workload trace cause larger deviations from the optimal setpoints during load level transitions compared to the more realistic workload trace. The convergence time of the primal-dual algorithm however is still very short, 6.5 seconds for the synthetic trace versus 1.5 seconds for the realistic trace.

In Figure 4.11 to 4.16 the deviation of the temperature of 4 selected units, the supply temperature, and the workload distribution for the 4 selected units, with respect to the optimal values is plotted. Again we see larger deviations for the synthetic workload trace during load changes than for the realistic workload trace. The convergence time for both types of workload however is similar here, both around 0.01 hour = 36 seconds.

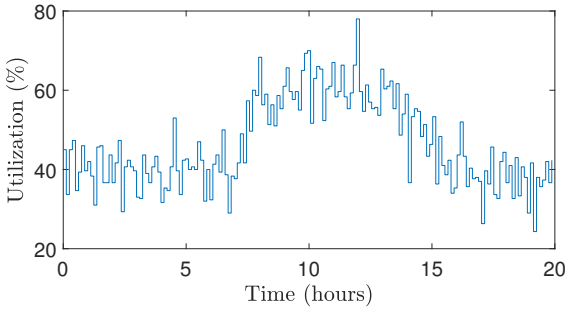


FIGURE 4.5: Synthetic workload trace used in section 3.7 as well. Characterized by large jumps between consecutive load levels.

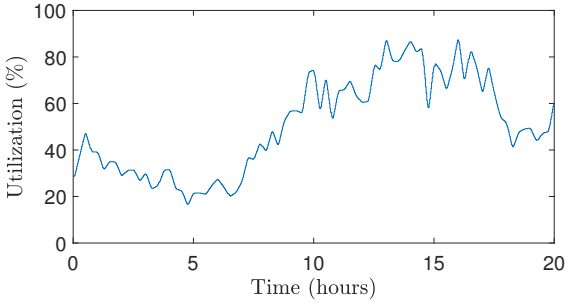


FIGURE 4.6: Workload trace based on real workload traces from the NASA Kennedy Space Center web server from Monday, July 3 of 1995. Characterized by much smoother transitions between consecutive load levels.

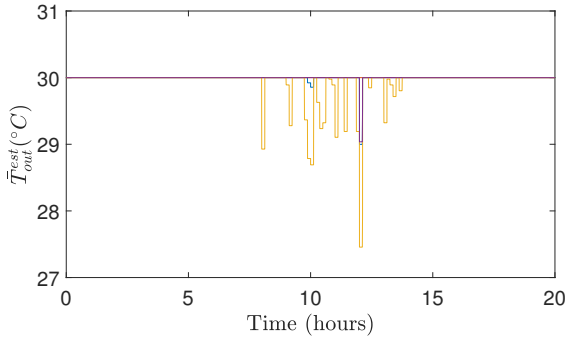


FIGURE 4.7: Temperature setpoint for thermal-aware controller for the synthetic workload trace.

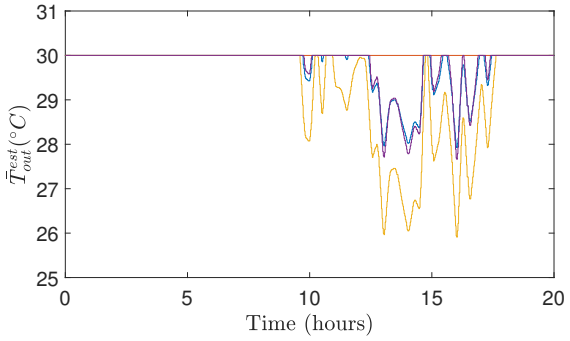


FIGURE 4.8: Temperature setpoint for thermal-aware controller for the realistic workload trace.

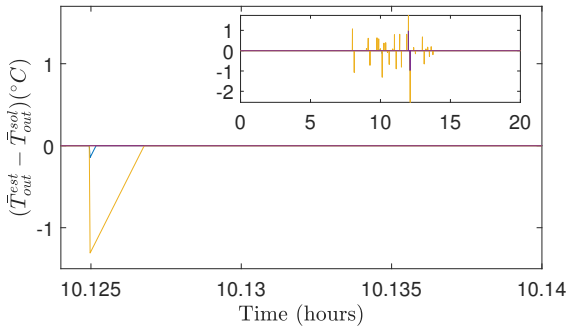


FIGURE 4.9: Difference between temperature setpoint and optimal setpoint for the synthetic workload trace.

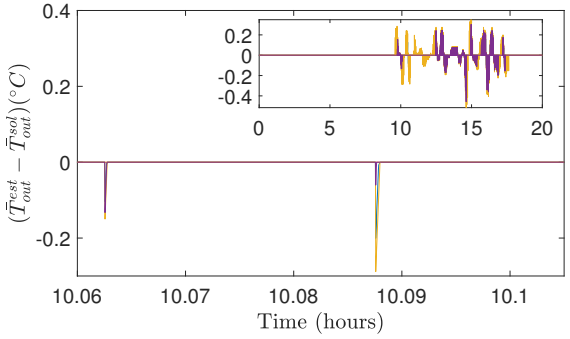


FIGURE 4.10: Difference between temperature setpoint and optimal setpoint for the realistic workload trace.

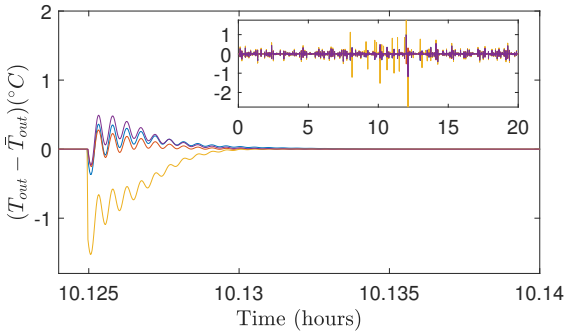


FIGURE 4.11: Difference between actual temperature distribution and optimal temperature distribution for the synthetic workload for 4 selected units.

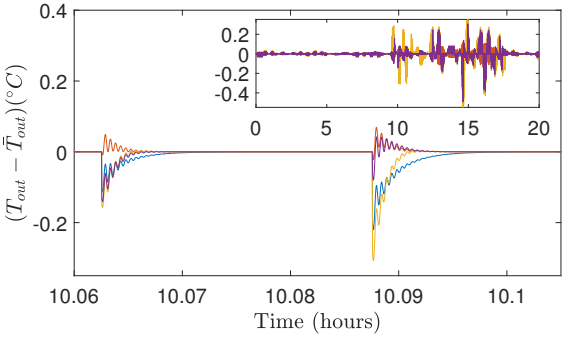


FIGURE 4.12: Difference between actual temperature distribution and optimal temperature distribution for the realistic workload for 4 selected units.

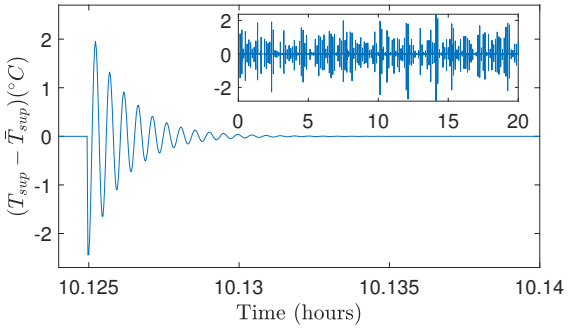


FIGURE 4.13: Difference between actual supply temperature and optimal supply temperature for the synthetic workload.

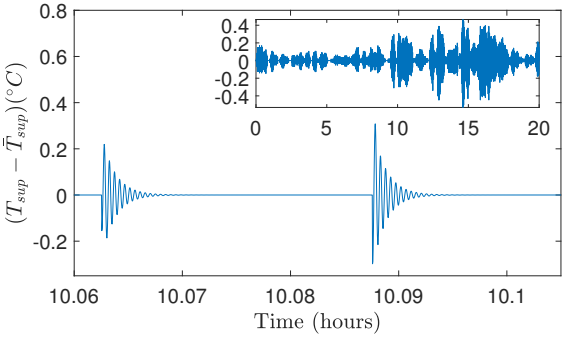


FIGURE 4.14: Difference between actual supply temperature and optimal supply temperature for the realistic workload.

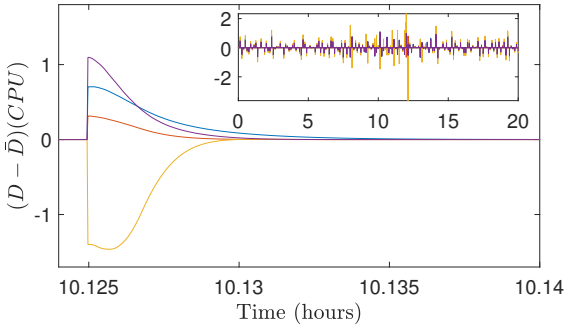


FIGURE 4.15: Difference between actual workload distribution and optimal workload distribution for the synthetic workload for 4 selected units.

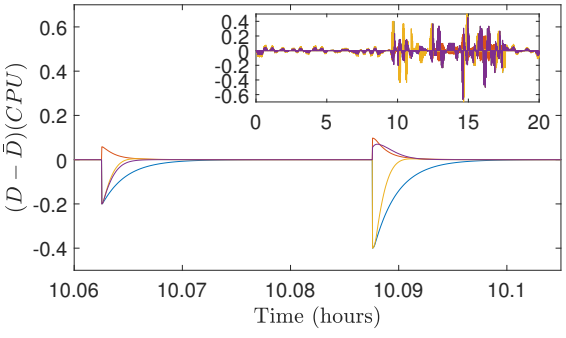


FIGURE 4.16: Difference between actual workload distribution and optimal workload distribution for the realistic workload for 4 selected units.

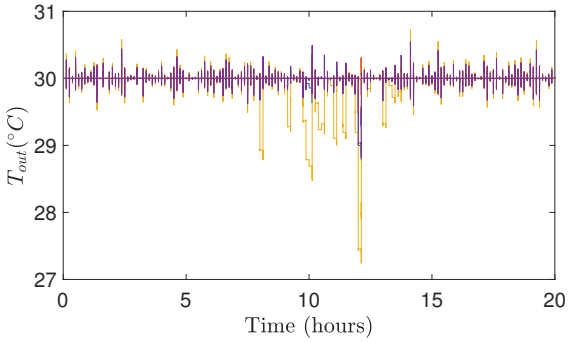


FIGURE 4.17: Actual temperature distribution for the synthetic workload for 4 selected units.

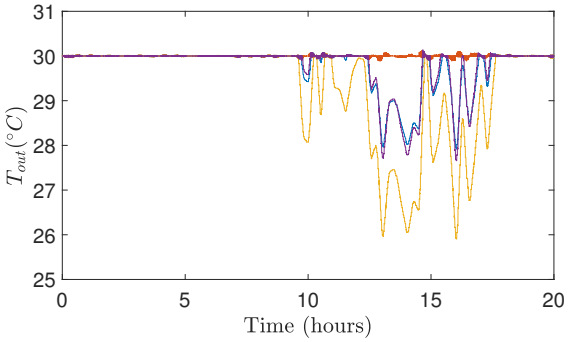


FIGURE 4.18: Actual temperature distribution for the realistic workload for 4 selected units.

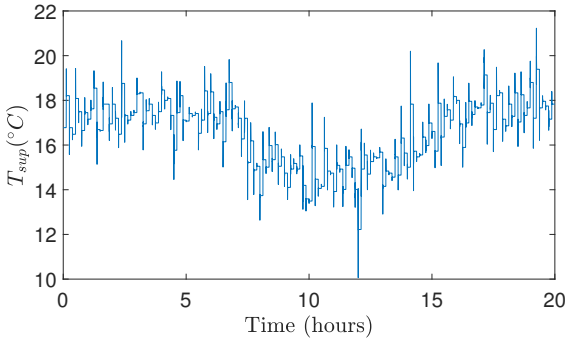


FIGURE 4.19: Actual supply temperature for the synthetic workload.

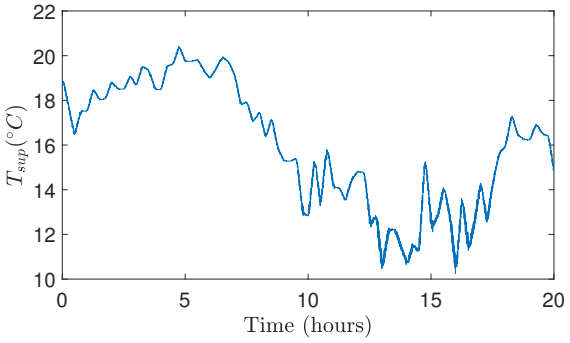


FIGURE 4.20: Actual supply temperature for the realistic workload.

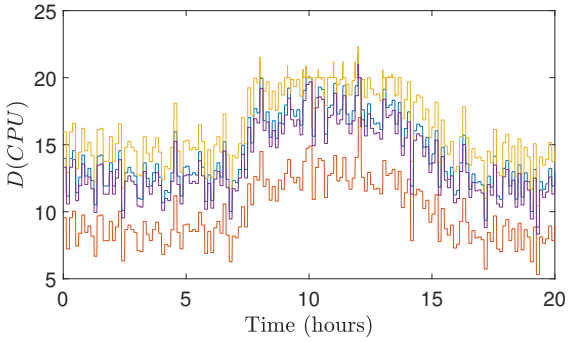


FIGURE 4.21: Actual workload distribution for the synthetic workload for 4 selected units.

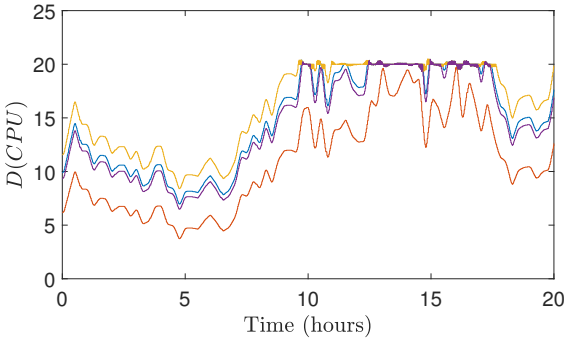


FIGURE 4.22: Actual workload distribution for the realistic workload for 4 selected units.

Finally in Figure 4.17 to 4.22 the actual values for the temperature distribution, supply temperature, and the workload distribution are given. The integral controllers are not designed to take the temperature and workload constraints into account. Therefore during the transient we see that the temperature and workload distribution sometimes violate these constraints, 30°C for the temperature of the units and 20 CPU's for the workload distribution. Again for the synthetic workload trace we see larger violation because the changes in load level are larger as well. Still as intended, the temperature and workload distribution stay very close to the constraint boundaries.

4.5 Conclusions

We considered the stability of constrained primal-dual dynamics represented by a complementarity system. For each unique (slow) solution initialized in the feasible set, we established the convergence to a primal-dual optimizer of the underlying constrained optimization problem. The stability proof involves the use of a generalized invariance principle using two different storage functions and the result relies only on mild assumptions including the convexity of the objective function and the existence of at least one primal-dual optimizer.

The primal-dual algorithm is interconnected with the integral controllers designed in chapter 3. Practical stability has been shown for the data center simulation considered in this thesis. Two workload traces were studied, one with slower but larger jumps, the other with more frequent but smaller jumps. In both situations the interconnected controller is able to drive the system to the optimal state after every jump in load. Showing that in practice the interconnection can behave in a stable manner.

Extensions to the primal-dual algorithm includes studying the robustness of the proposed primal-dual algorithm, possibly in the lines of (Cherukuri et al., 2017) and (Richert and Cortés, 2015). Another research direction is to study initialization-free algorithms as for example in (Yi, Hong, and Liu, 2016) which allows for a plug-and-play implementation of the constrained

primal-dual dynamics. Lastly we would like to further study the interconnection of the proposed primal-dual algorithm with other physical systems to analyze more real-time applications of the algorithm in dynamic environments.

CHAPTER 5

Combining thermodynamics with power-aware control techniques in data centers: A simulation study

ABSTRACT

Advanced power management and cooling techniques for data centers often co-exist as separate entities in current-day operation of data centers. In this chapter we propose to combine these techniques to achieve greater power savings. To this end, a theoretical thermal-aware model is integrated in an extensive simulation framework for data centers using power and performance models, which allows for a detailed study in power, performance and thermal metrics. In this chapter we compare four distinct cases for studying the effect on these metrics: a data center with (i) basic functionality; (ii) advanced cooling; (iii) advanced power management; and (iv) a combination thereof. The combined case shows a significant reduction in the energy consumption compared to the other cases while performance and thermal demands are kept intact. The combination of these techniques shows improvements in energy savings and shows it is meaningful to investigate further into smart combined energy saving techniques.

5.1 Introduction

From 2000 to 2006 the annual energy consumption of United States data centers increased from 28.5 billion kWh to 61.8 billion kWh, whereas in the years from 2006 to 2014, the annual energy consumption only increased

to 69.8 billion kWh (Shebabi et al., 2016). This small growth in energy consumption comes from efforts among data center owners to push back the energy consumption of their data centers.

Energy savings in data centers can be achieved by means other than looking at thermodynamics as well. According to (Shebabi et al., 2016), the three main energy-efficiency improvements that contribute to this flattening are (i) advanced cooling strategies, (ii) power proportionality, and (iii) server consolidation. Advanced cooling strategies focus on techniques that increase the thermal efficiency of the data center like hot aisle isolation, economizers, and liquid cooling, techniques studied up to now in this thesis. Power proportionality is achieved with power management software and hardware, whereas server consolidation focuses on running current workload on as few servers as possible, in order to decrease the amount of hardware necessary in the data center.

While these three areas separately show many improvements, we believe that more improvements can be gained by combining these areas, specifically combining the area of advanced cooling strategies with the area of power proportionality. In this chapter, we investigate the cooperation between strategic power management control and strategic thermal control. Besides possible energy consumption benefits, this study allows us to show the general applicability of both these modeling approaches.

Recently, a simulation framework has been introduced to analyze models for both power and performance in data centers that use power management techniques to reduce its energy consumption (Postema and Haverkort, 2015; Postema and Haverkort, 2017). In this framework it is easy to study power and performance metrics of high-level models for any given data center configuration and workload characteristic. Already these kind of analyses provide helpful insights in the design phase of data centers. In chapter 2-3, a thermodynamical framework has been introduced with accompanying controllers for CRAC cooling control and workload distribution in order to minimize the energy consumption of the cooling system.

In this chapter we propose to integrate the thermal-aware controllers in the existing simulation framework to study the interaction between the power management strategies from (Postema and Haverkort, 2017) and the thermal-aware controllers from (Van Damme, De Persis, and Tesi, 2018).

Individually these areas have received much attention by researchers, e.g. (Hameed et al., 2014) and references therein, however the combination of these two fields is much less studied (Zhang et al., 2016). These results contribute to the existing state-of-the-art by providing an extensive simulation study that shows the viability of combining these two distinct control strategies and study the improvements that can be made by combining the two approaches.

The remainder of this chapter is organized as follows. In section 5.2 the models are introduced and their integration into the simulation framework is discussed. Next, the simulation configuration is given in section 5.3 and different control scenarios are described in section 5.4. Finally, the simulation results are studied in section 5.5.

5.2 Model integration

In this section we explain how the different models are integrated in the simulation framework as well as explaining the background of each of the models.

Overview. Figure 5.1 shows an overview of how the different models are connected and how they interact with each other. In order to get the most realistic models, we interact with industrial partners in order to obtain realistic data center configurations and parameters. These characteristics are given to each of the models, **1–3**. The temperature of each unit and the optimal job distribution is communicated to the server models and to the job dispatcher, **4**. Moreover, the performance of the data center is monitored via power metrics available in the models, **7** and **8**. Depending on the chosen control strategy the dispatcher schedules jobs among the servers using the optimal thermal-aware distribution or using strategic power management related criteria according to the current state of the metrics, or both. The energy consumption of the computer room air conditioning (CRAC) is calculated using the thermodynamics and is communicated to the cascade model, **6** and **9**; the energy consumption of the other infrastructural components remain linearly dependent on the energy consumed by IT equipment,

5. The total data center energy consumption is sent to the power management module, **10**.

Data flow. The models are initialized and calibrated with the data center parameters only once at the start of the simulation, **1–3**. During the simulation, the thermal models are updated every 0.1 seconds. The power consumption and related metrics are updated in an event-based fashion: every time a job enters or leaves the data center. Every time the total power consumption of the units changes, the optimal workload distribution according to the thermal-aware controller and/or the number of required active units is calculated and send to the server/dispatcher models, **4 and 7**. After every change in unit power consumption, the new power consumption is transmitted to the power management controllers, **8**, and the cascade model, **5**. The supply temperature setpoint is updated every time the temperature changes, every 0.1 seconds, **6**, and the new power consumption is transmitted to the cascade model, **9**.

5.2.1 Data center infrastructure

The data center structure used in this simulation framework is the same as in chapter 2. The hierarchical structure remains the same and as a reminder is depicted in Figure 5.2.

The power and performance models will adhere closely to this hierarchy as explained in the following subsections.

5.2.2 Thermodynamical model

The thermodynamical model is taken from chapter 2 and remains unchanged in this chapter.

5.2.3 Power and Performance Models

The models for power and performance are based on earlier work from (Postema and Haverkort, 2015). Here we explain how each model is adapted to fit in the current framework.

Performance. The performance models are extended with a two-level scheduling algorithm, see Figure 5.3. A central dispatcher distributes jobs

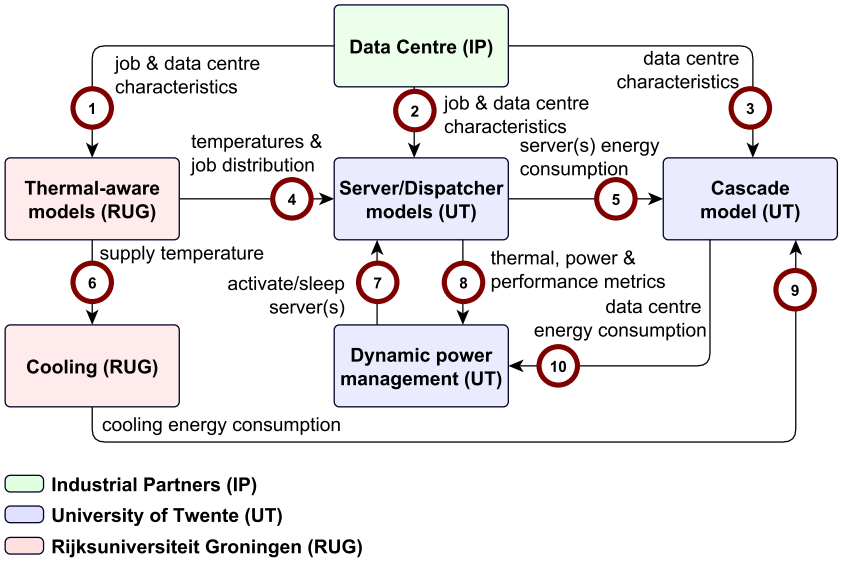


FIGURE 5.1: Detailed integration of thermal-, power- and performance-aware models for data centers.

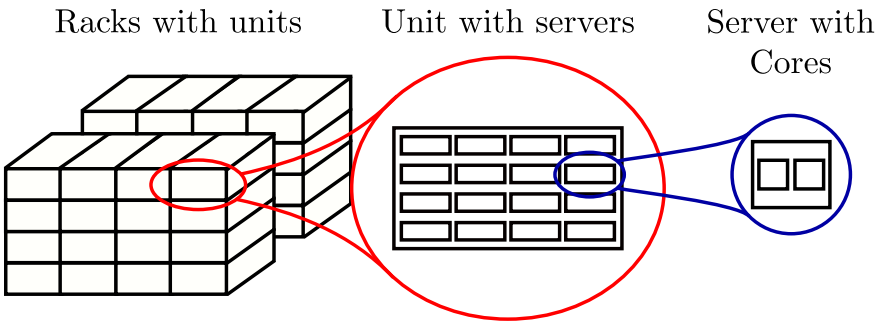


FIGURE 5.2: Hierarchical overview of the data center. The same hierarchy is used as in chapter 2

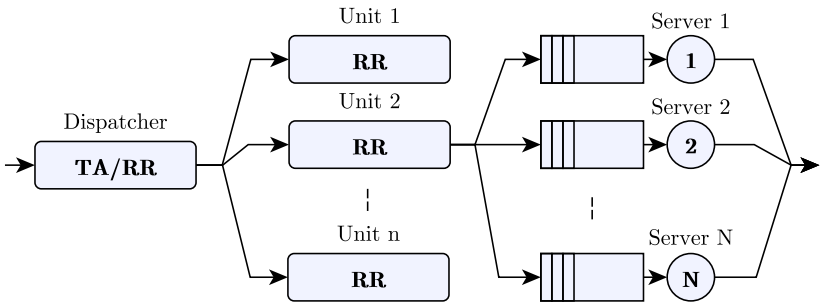


FIGURE 5.3: Our extended dispatcher schedules jobs to the queues of the servers 1 to N via the hierarchy of the units 1 to n using a two-level scheduling algorithms in *Thermal-Aware* (TA) or *Round-Robin* (RR) fashion.

to one of the n units using a scheduling algorithm of choice. Then, jobs are scheduled in round-robin fashion to servers 1 to N inside the unit. As in the original work, each server comprises a $G|G|1|\infty|\infty$ queue with a FIFO buffer.

Power consumption of IT equipment. The power consumption at time t for each of these servers is equal to the predefined amount $R(k)$ for each power state k as can be seen in Figure 5.4. Each state has a fixed power consumption with the exception of the processing state. As each server can have multiple computing cores, the power consumption of the processing state is also dependent on the number of active cores. The power consumption for the processing state is therefore given by $R[\text{pc}] = R[\text{id}] + w_i D_s(t)$, where D_s is the number of active cores in server s, and w_i, the power consumption per active core for the unit the server resides in. The power consumption of unit i is then given by the sum of the power consumption of the servers inside the unit.

The main power management feature is the ability to switch between global power states. This allows data center operations to adapt power consumption levels at the cost of time spent switching between global power states and therefore decreased performance.

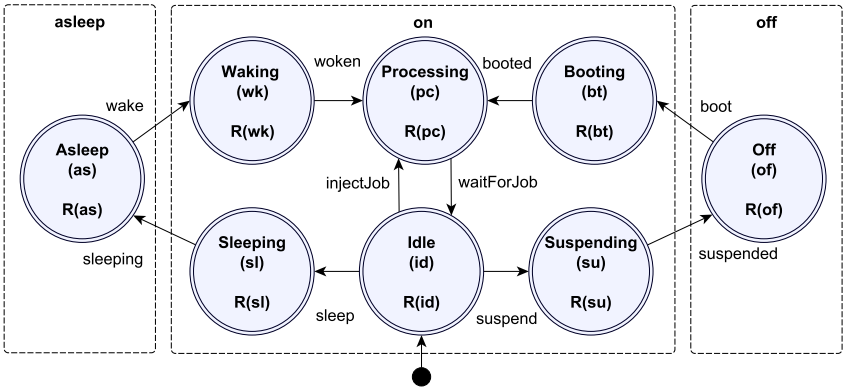


FIGURE 5.4: The power model for switching between three global power states: *Asleep*, *On*, and *Off*. The power consumption of each state is denoted by $R(k)$ where k denotes the current power state.

Power consumption of data center. Based on the IT equipment an estimation of the power consumed by other necessary infrastructural components can be computed using simple linear functions, which is called the *cascade model*. The power consumption of the CRAC is calculated with

$$P_{AC}(T_{\text{out}}(t), T_{\text{sup}}(t)) = \frac{Q_{\text{rem}}(t)}{\text{COP}(T_{\text{sup}}(t))}, \tag{5.1}$$

where each part of this calculation is explained in section 2.5. The total data center power consumption is then calculated by the sum of P_{AC}, and the power consumption of the IT equipment and the other infrastructural components, as calculated by the cascade model.

5.2.4 Advanced Cooling Control

The thermodynamical control in the simulation is done via the integral controllers designed in chapter 3. The simple form of the controllers will be used, i.e. we will assume that the optimal temperature distribution is the

safe temperature threshold, as calculating the primal-dual controller from chapter 4 is a computationally heavy task. For clarity we repeat the integral controllers here

$$\frac{d}{dt}T_{\text{sup}}(t) = \mathbb{1}^T A^T Z(T_{\text{out}}(t) - T_{\text{safe}}), \tag{5.2}$$

$$\frac{d}{dt}D(t) = (\frac{\mathbb{1}\mathbb{1}^T}{n} - I_n)(M^{-1}W)^T Z(T_{\text{out}}(t) - T_{\text{safe}}). \tag{5.3}$$

The control input calculated by the controllers cannot be implemented directly in the data center. Therefore the calculated inputs will be used as a setpoint by the other parts. A local (internal) CRAC controller will steer the CRAC supply temperature slowly to the calculated T_{sup}. The job dispatcher will assign jobs to the unit which shows the largest deviation from the calculated workload distribution, D.

Since we use the simple form of the integral controllers, the controllers only work for the operation range where the optimal temperature distribution is equal to the safe temperature threshold. The exact range depends on the values of the parameters of the data center, e.g. power consumption of servers, and recirculation flow. For our set of parameters the safe temperature threshold is optimal for workload levels between $10 - 55\%$ of the total data center computing capacity. Whenever higher load levels are applied in the data center, the integral controllers will be disabled. At first sight this limited operating range might seem restrictive, however in real life data center operations the workload levels are very often around 30% in order to have enough spare capacity when equipment fails, or there is a temporary load spike. Therefore the simulation results will still be very relevant for real-life operations.

5.2.5 Advanced Power Management

An advanced power management strategy outlines a global directive to achieve certain energy saving goals during operation. Every r seconds, certain performance metrics are evaluated, then according to the specifics of the chosen power management strategy a number of servers that could switch global power states is computed. To formalize this description we define a power

management strategy Θ by the 3-tuple (Postema and Haverkort, 2017)

$$\Theta = (G, \Phi_{\mathcal{S}}, \Phi_{\mathcal{C}}(s)). \tag{5.4}$$

Here G is a vector containing all possible global power states, the vector $\Phi_{\mathcal{S}}$ contains the constraints on the global power levels, and the vector $\Phi_{\mathcal{C}}(s)$ contains the constraints on each server s that is appointed to switch global power state.

In our strategy there are three possible global power states, Asleep (as), On (on), and Off (of), denoted by $G = (as, on, of)$. Furthermore there are eight server power states: asleep (as), waking (wk), processing (pc), booting (bt), sleeping (sl), idle (id), suspending (su), and off (of), denoted by $k =$(as, wk, pc, bt, sl, id, su, of). The server power states are visualized in Figure 5.4. The metrics used in this power management strategy are calculated in two possible ways, either using exponentially moving averages (eavg) or by instantaneous (ins) calculation, denoted by $\gamma =$ (eavg, ins). The metrics used in our strategy are various observable quantities related to the performance and current state of the servers and data center, and are denoted as

RT(γ) is the response time, a measure of performance that indicates the total time a job takes from start to end of execution,

PU(k, γ) is the percentage of time that servers spend in global power state k,

PS(s) is the current server power state of server s,

TO(s, k) is the time server s spends in server power state k,

R_{SLA} is the average response time threshold as specified by the Service-Level Agreements (SLAs), a service demand required by data center customers.

Now that the metrics have been defined, we can set up the constraints, $\Phi_{\mathcal{S}}$ and $\Phi_{\mathcal{C}}(s)$, that define power management strategy Θ:

$$\Phi_{\mathcal{S}} = \begin{pmatrix} \phi_{\mathcal{S}}^{on} & := & \mathrm{RT}(\mathrm{eavg}) > 0.75 \cdot R_{\mathrm{SLA}} \\ \phi_{\mathcal{S}}^{as} & := & (\mathrm{RT}(\mathrm{eavg}) \leq 0.75 \cdot R_{\mathrm{SLA}}) \\ & & \wedge\, (\mathrm{PU}(\mathrm{id}, \mathrm{ins}) \geq 0.3) \\ \phi_{\mathcal{S}}^{of} & := & (\mathrm{PU}(\mathrm{id}, \mathrm{ins}) \geq 0.3) \\ & & \wedge\, (\mathrm{PU}(\mathrm{of}, \mathrm{ins}) \leq 0.3) \\ & & \wedge\, (\mathrm{PU}(\mathrm{bt}, \mathrm{ins}) \leq 0.05) \end{pmatrix}, \tag{5.5a}$$

$$\Phi_{\mathcal{C}}(s) = \begin{pmatrix} \phi_{\mathcal{C}}^{as}(s) & := & \mathrm{PS}(s) = \mathrm{id} \\ \phi_{\mathcal{C}}^{on}(s) & := & (\mathrm{PS}(s) = \mathrm{sl} \vee \mathrm{PS}(s) = \mathrm{of}) \\ & & \wedge\, (\neg(\mathrm{PU}(\mathrm{as}, \mathrm{ins}) \geq 0.0) \\ & & \qquad \wedge\, (\mathrm{PS}(s) = \mathrm{as})) \\ \phi_{\mathcal{C}}^{of}(s) & := & \mathrm{TO}(s, \mathrm{as}) \geq 100.0 \end{pmatrix}. \tag{5.5b}$$

Our strategy requires the data center to be able to observe the (exponentially moving average) response times and the (current) utilization. If the exponentially moving average response times are more than 75% of the threshold described in the Service-Level Agreement, then additional servers need to switch to global power state *On*. Conversely, when the exponentially moving average response times are below the 75% threshold, then servers can move to the global power state *Asleep*. Here we require that at least 30% of the server remain idle; this ensures that enough servers are active to process the current workload and ensure sufficient capacity to be able to (de)activate servers. Also for servers to be switched off, less than 30 % of the total amount of servers need to be switched off and less than 5% of the servers must be in the booting state, in order to prevent overly active power state switching.

Secondly a server can only move to the *Asleep* global power state if the server is currently idle. Servers which have then been in the *Asleep* global power state for a duration of at least 100 s can be shut down. Lastly, a server can only be switched on if it is currently in the sleeping (sl) or the off (of) power state. However, to remain efficient a server can only be woken from the global *Off* power state if no server can be woken from the global *Asleep*

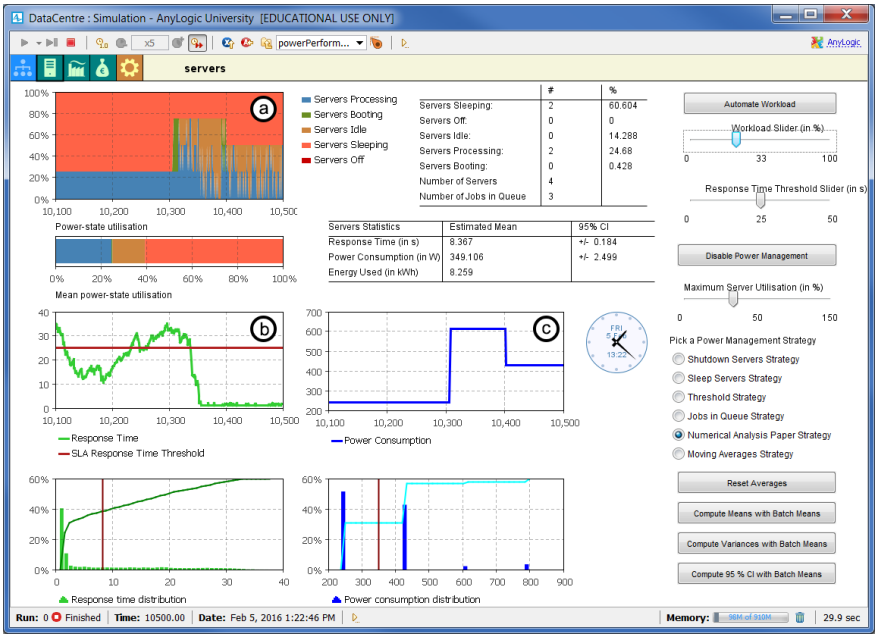


FIGURE 5.5: The ANYLOGIC dashboard

power state.

5.2.6 General overview of the DACSIM simulator

In (Postema and Haverkort, 2015), a simulation framework has been proposed that allows for analysing the trade-offs between power consumption and performance in data centers. The aim of this framework is understanding ways to save energy via power management using the power and performance models from subsection 5.2.3. A copy of the source code of the *Data center Simulation Framework* (also known as DACSIM) is publicly accessible via a GITHUB repository (Postema, 2013).

An image of the simulation dashboard is given in Figure 5.5. The framework is developed in ANYLOGIC and allows for easy implementation of

combinations of discrete-event and agent-based models. The framework features an intuitive dashboard that actively controls and obtains insights during each simulation run. Transient and steady-state behavior can be analyzed for (i) *power-state utilization*, (ii) *response times* and (iii) *power consumption*. At the end of each simulation run, relevant data is exported for optional post-processing and more extensive analysis.

For the purpose of integrating the thermal-aware models in DACSIM, the matrix library EJML (Abeles, 2017) is included to handle the differential equations. A module is set up that allows for (i) all the computations related to the thermal-aware models, (ii) transient analysis of the computed values during a simulation run and (iii) full logs of all the computed values.

5.3 Model Parameters and Output

5.3.1 Job and Data center Characteristics

Similar as in earlier simulations, the data center in this simulation consists of 30 Dell PowerEdge 1855 server racks, i.e. units in Figure 5.2. Each unit has 10 dual-processor blade servers, i.e. a total of 20 CPU cores per unit. The base power consumption of a server in an idle state is $R[\text{id}] = 172.8$ W. The power consumption of each active CPU core is $w_i = 145.5$ W (Tang et al., 2006b). The power consumption of server s in the sleep or off power states is respectively $R[\text{as}] = 14$ W and $R[\text{of}] = 0$ W (Gandhi et al., 2013). The power consumption of all other power states $R[\text{wk}]$, $R[\text{sl}]$, $R[\text{bt}]$ and $R[\text{su}]$ for global power state switching are regarded as if all CPUs in the server are in use.

The global power state switching time is distributed deterministically with mean $1/\alpha_{\text{wk}} = 1/\alpha_{\text{sl}} = 0.1$ (10 s) and $1/\alpha_{\text{bt}} = 1/\alpha_{\text{su}} = 0.01$ (100 s). The coefficients of the cascade model are taken from (Postema and Haverkort, 2015).

The data center parameters were obtained from measurements by Vasic et al. (Vasic, Scherer, and Schott, 2010) at the IBM Zurich Research Laboratory. The safe temperature threshold for the units is set at 30 °C. The initial temperature distribution of the units is set to 27.5 °C for all units.

Jobs arriving at the data center are characterized by HTTP requests. The inter-arrival times and service times distributions in the model are calibrated with two data sets of HTTP requests from a real data center, with each set having a duration of about 21 days (about 27.2 million entries), using a fitting algorithm in cooperation with Better.be. These distributions are exponential with a rate λ that is proportional to the desired workload in the case of the inter-arrival time, and a mixture of normal distributions with an average service time of about 107 ms in the case of the service times. The Service-Level Agreement (SLA) requires response times of HTTP requests to be below 1 s and an average response time of 300 ms.

5.3.2 Simulation Settings

Time units are set to seconds. The duration of the simulation was 3600 seconds. The warmup period for the system to adapt to the initial transient phase (e.g. sleeping the right number of servers) has been set to 1000 seconds. All simulations have been performed on a machine equipped with a 2.70 GHz Intel® Core™ i7-4800MQ CPU, 8 GB of RAM and Windows 7 64-bit with AnyLogic v8.1.0. The execution time of a single simulation run was between approximately 1 minute for the lowest workload and approximately 30 minutes for the highest workload. Results required a total of 40 simulation runs.

5.4 Case studies

To study the impact of each control strategy on energy, performance and thermal measures, we have devised four different control scenarios in a realistic data center setting. In Table 5.1 an overview of the different scenarios is given.

In the **base case** scenario (*Scenario I*), *no* advanced control mechanics are applied, i.e., there is *no* feedback in control decisions. This scenario represents current day heuristics in many data centers. In this scenario, basic control of the data center is applied at two levels, namely (i) *cooling*, (ii) *job scheduling*. The supply temperature of the cooled air of the CRAC is controlled in a way that keeps the temperature of the units below a certain safe

Scenario	Scheduler	Cooling	PM Strategy
I: Base Case	RR-RR	Static	Always On
II: Advanced Cooling	TA-RR	Dynamic	Always On
III: Advanced PM	RR-RR	Static	Strategy Θ
IV: Combined	TA-RR	Dynamic	Strategy Θ

TABLE 5.1: Overview of the four scenarios

threshold. If the maximum unit temperature is above the safe threshold, then the supply temperature will decrease, otherwise it will increase. Jobs arriving at the data center are scheduled in round-robin fashion to the units. The power management strategy is inactive, i.e., all servers are always turned on.

In the **advanced cooling strategy** scenario (*Scenario II*), only advanced cooling control is applied, there is *no* active power management, i.e. servers are always turned on. Controllers (5.2) and (5.3) are applied according to the steps described in subsection 5.2.4. This control is tested up to and including a workload of 50% of the total data center workload capacity.

In the **advanced power management strategy** scenario (*Scenario III*), cooling control and job distribution are the same as in the base case, whereas advanced power management strategy (5.5) is applied as specified in subsection 5.2.5.

The **combined cooling and power management strategy** scenario (*Scenario IV*) allows for the investigation of a combination of both advanced power management strategies and advanced thermal-aware control. In this scenario, global power states are switched according to strategy (5.5) for energy-efficiency, *and* the job dispatcher follows the set point of the job distribution using controllers (3.27) and (3.28) for thermal-efficiency. Same as in Scenario II, advanced cooling control is applied up to workloads of 50% of the total data center capacity. For workloads higher than 50%, this scenario is identical to Scenario III.

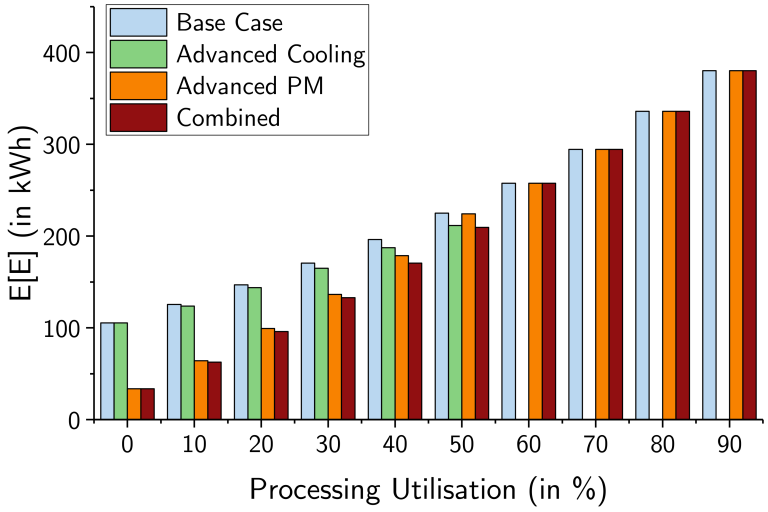


FIGURE 5.6: Total expected energy consumed by the data center for the four scenarios with varying workloads from 0% to 90% of the total data center capacity in increments of 10%.

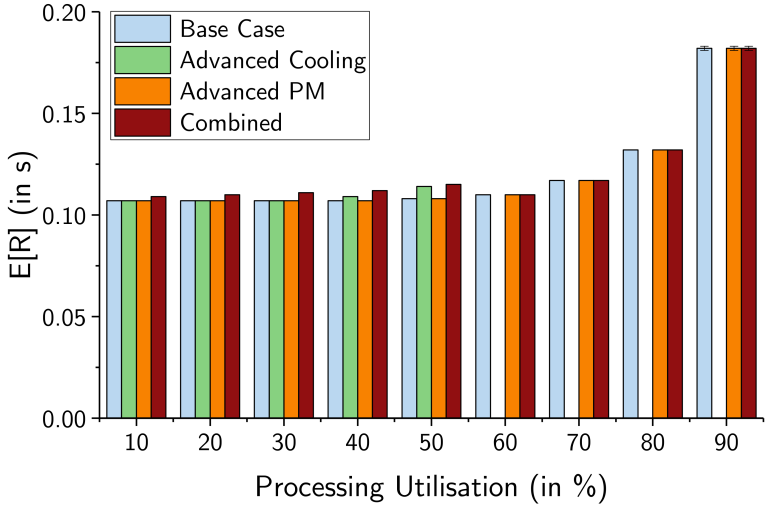


FIGURE 5.7: Mean response time for the four scenarios with varying workloads from 10% to 90% of the total data center capacity in increments of 10%.

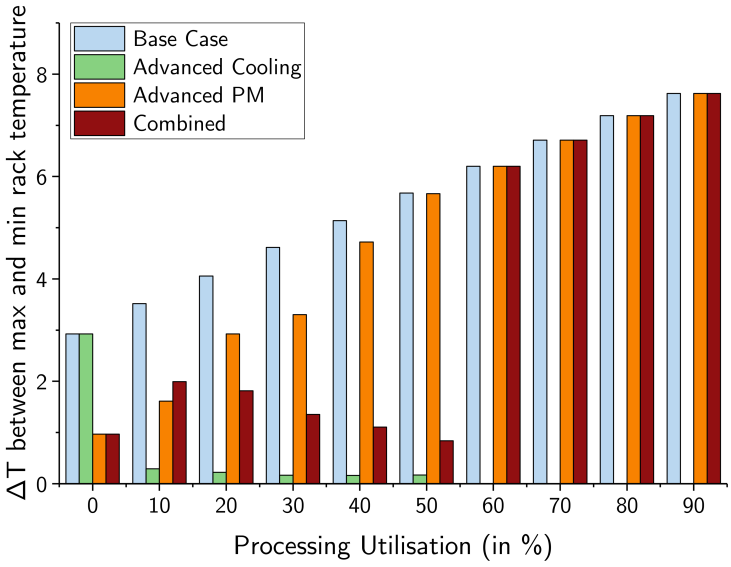


FIGURE 5.8: The temperature difference between the average maximum and minimum temperature for the four scenarios with varying workloads from 0% to 90% of the total data center capacity in increments of 10%.

5.5 Results

5.5.1 Energy

The total expected energy consumption $E[E]$ of the data center for the full duration of the simulation is plotted in Figure 5.6 for all the scenarios, with different utilization levels varying from 0% to 90% with increments of 10%. Note that with a processing utilization of 100% the system would become unstable.

First, it is observed from Figure 5.6 that the higher the utilization level becomes, the greater the energy reduction of the *advanced cooling* strategy becomes with respect to the *base case*. Secondly, a large energy reduction

is observed at lower utilization levels when only *advanced power management* is applied. However, the best energy savings for all utilization levels are obtained when the two control approaches are combined as in Scenario IV. At higher utilization levels, our strategies have almost no room to control anything, and therefore no significant energy savings are observed.

5.5.2 Performance

For each of the simulation runs, the Service-Level-Agreement violations are recorded as a percentage of the overall number of jobs. The percentage of SLA violations for all processing utilizations has been 0% with an outlier of 0.011% at 90% processing utilization due to the stochastic nature of the simulation.

Figure 5.7 shows the mean response times for the four scenarios with varying workload levels from 10% to 90%. The 0% case is skipped as there are no jobs arriving in the system in this case. The figure shows an increase of at most 100 ms in the average response times for all scenarios. It is seen that distributing jobs in a thermal-aware fashion gives rise to a slight increase in response times, with the biggest impact seen at 50% utilization. The SLA requirements are still met however, because response times should be at most 1 s and the average response time should not exceed 300 ms. So, the overall performance is maintained while energy is being saved.

5.5.3 Thermodynamics

In order to plot the temperature data in an understandable way, the spread in temperatures among the units is studied. To do this, the difference between the average maximum and minimum unit temperature over the full simulation run is calculated for all simulation runs. This temperature difference is plotted in Figure 5.8.

Comparing the temperature differences of Scenario I with Scenario II, we see that the *advanced cooling* strategy results in a very balanced temperature profile among the units. This is the reason for the energy savings between the two scenarios, observed in Figure 5.6. When comparing the temperature differences between Scenario III and Scenario IV, we again see

large improvements in favour of the *combined case*, where *advanced cooling* is applied. Same as before, this smaller spread results in less energy consumed.

Note that in the case of 0% workload, not much interesting can be done as there are no jobs available for redistribution. Also in the case of 10% workload it is seen that Scenario IV has an increased spread compared to Scenario III. However when considering all units, less heat is generated overall, as can be deduced from Figure 5.6 from the lower energy consumption of Scenario IV in this case.

5.6 Conclusions

In order to analyze a potential power-, performance- and thermal-aware data center, thermodynamical models have successfully been integrated in an existing extensive simulation framework with power and performance models. Moreover, advanced energy-aware control strategies are studied in a realistic simulation setting. Energy consumption, performance and thermodynamics are analyzed in four scenarios where different control strategies are applied. From the simulation runs we see that combining thermal-aware control strategies with power- and performance-aware strategies yields the best energy savings without suffering any SLA violations. Furthermore it is seen that the thermal-aware controller successfully balances output temperatures of the units.

Future work includes studying the combined controllers for all workload levels and studying different ways of combining power- and performance-aware controllers with thermal-aware controllers. Also, current analysis can be extended by studying the transient phases as a consequence of fluctuating workload conditions.

CHAPTER 6

Characterizing heat recirculation parameters in data centers

ABSTRACT

Knowledge of the thermodynamics is vital for the correct operation of the controllers presented in this work. The thermodynamics are characterized by a static mapping of the airflow within the data center. To allow portability of the thermal-aware controllers between different data centers it is important to have methods to reconstruct these system parameters from easy-to-perform measurements. Subspace identification methods allow for such easy characterization, independent of the specific data center context, allowing any data center operator to implement these algorithms. We show that for our data center context we are able to identify the recirculation parameters with a 2-norm error of the order of 10^{-7}.

6.1 Introduction

The thermodynamical models rely heavily on the knowledge of the way the heat recirculates among the units in the data center. So far in this thesis we have shown that if the characterization of this recirculation is known, then it is possible to apply thermodynamical measures in order to save energy in the cooling system. In this chapter we will focus on developing a way for data center operators to obtain the parameters associated to the recirculation via easy-to-perform experiments that are independent of specific data center layouts.

From chapter 2 we have that the thermodynamics in a data center can be modeled as

$$\frac{d}{dt}T_{\text{out}}(t) = \underbrace{\rho c_p M^{-1}(\Gamma^T - I_n)F}_{A}(T_{\text{out}}(t) - \mathbb{1}T_{\text{sup}}(t)) + M^{-1}P(t). \quad (6.1)$$

The system parameters, or air recirculation parameters, are represented in these dynamics by the matrix $\Gamma \in \mathbb{R}^{n\times n} = [\gamma_{ij}]$. Each element γ_{ij} represents the proportion of heat flowing from unit i to unit j. As such there are n^2 elements that are unknown for n equations in total. In order to identify these parameters we turn to the field of system identification, and subspace identification in particular.

Subspace identification methods are very attractive for identifying MIMO systems due to their simple and general parametrization. They are especially attractive compared to other input-output systems as there is no linear input-output parametrization that is general enough for all linear MIMO systems, see (Katayama, 2006). The main benefits of subspace identification methods can be summarized by three key points: (Van Overschee and De Moor, 2012)

Parametrization Subspace identification methods require very little user-specified parametrization from the user because it makes use of state space model. This makes that there is no difference in complexity between SISO and MIMO systems. The models only require the order of the model to be user-specified, which can be determined from inspection of certain singular values.

Convergence When implemented *correctly*, subspace identification algorithms are very fast, despite the fact that they use QR and singular value decompositions. Since these methods are not iterative, there are no convergence problems, and numerical robustness is guaranteed.

Model reduction When applying subspace identification methods, it is possible to directly obtain a reduced model from input-output data, without having to compute the higher order model first.

The downside of using subspace identification methods is that they require a lot of input-output data samples due to the statistical properties of the geometrical methods involved. However in the data center setting this isn't a major hurdle since data centers come equipped with a lot of sensors for tracking a plethora of operational metrics. Therefore it is straightforward to design experiment runs, that can readily be run by data center operators in order to identify the missing recirculation parameters.

While there are many different types of models, such as canonical variate analysis (CVA) (Larimore, 1990), N4SID (Van Overschee and De Moor, 1994), subspace splitting (Jansson and Wahlberg, 1996), MOESP (Verhaegen and Dewilde, 1992), and others (Rao and Unbehauen, 2006), in this chapter we will focus on the numerical N4SID methods described in detail in (Van Overschee and De Moor, 2012).

The remainder of this chapter is organized as follows: in section 6.2 the current system is rewritten to a discretized system. Next, the subspace identification method is explained in detail, section 6.3, after which the algorithm to identify the parameters of interest is summarized, section 6.4. Lastly a possible experiment design is discussed in section 6.5, and the simulated identification run on our specific data center setting is covered in section 6.6.

6.2 Discretized state space model

The subspace identification method described in (Van Overschee and De Moor, 2012) is restricted to a discrete time, linear, time-invariant, state-space model. In our work we have a continuous time, linear, time-invariant, state-space model, therefore we have to discretize the dynamics used in this thesis, which we will do in this section.

Before we can rewrite the dynamics in state space form, we will simplify the dynamics in order to simplify the identification process. In order to identify the recirculation parameters we need to measure the temperature change due to heat added to the data center, i.e. work processes by the units. Since the supply temperature does not affect single units, but the data

center as a whole, any changes made to the supply temperature will not provide insights in the recirculation of the air flows. Therefore we will assume that the supply temperature of the CRAC unit remains constant during the identification run, effectively removing one of the inputs.

Assumption 6.1. The supply temperature, T_{sup}, remains constant during the identification run.

By taking the supply temperature constant, it can be combined in the state giving the new state variable $x(t) \triangleq: T_{\text{out}}(t) - \mathbb{1}T_{\text{sup}}$. For the new state variable it holds that $\dot{x}(t) = \dot{T}_{\text{out}}(t)$, so that the new dynamics are still the desired dynamics. In state space form the data center thermodynamics are then given by

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{6.2a}$$
$$y(t) = Cx(t), \tag{6.2b}$$

where

$$\begin{aligned} x(t) &= T_{\text{out}}(t) - \mathbb{1}T_{\text{sup}}, \\ A &= \rho c_p M^{-1}(\Gamma^T - I_n)F, \\ B &= M^{-1}, \\ u(t) &= P(t) = V + WD(t), \\ C &= I_n. \end{aligned}$$

Recall from Property 2.1 that system matrix A is Hurwitz and therefore invertible. The next step is to discretize the system. We start by solving system (6.2) and then characterizing the solution at time $t + dt$. For discretization purposes the input is considered to be a piece-wise constant signal, being constant during each time interval.

Assumption 6.2. The input $u(t)$ is a piecewise constant signal, i.e. during each time interval $[t, t + dt)$, where $t \in \{0, dt, 2dt, \dots\}$, the input is given by $u(t)$.

The general solution to system (6.2) is given by

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau. \tag{6.3}$$

Starting at time-step t, combined with Assumption 6.2, the solution at time-step $t + dt$ is given by

$$\begin{aligned} x(t+dt) &= e^{Adt}x(t) + \int_t^{t+dt} e^{A(t+dt-\tau)}d\tau Bu(t) \\ &= e^{Adt}x(t) + \left[-A^{-1}e^{A(t+dt-\tau)}\right]_t^{t+dt} Bu(t) \\ &= \underbrace{e^{Adt}}_{\tilde{A}}x(t) + \underbrace{A^{-1}\left(e^{Adt} - I_n\right)B}_{\tilde{B}}u(t), \end{aligned}$$

where the second step is possible as A is Hurwitz and therefore invertible in this case. Having found the dependency of the value $x(t + dt)$ on $x(t)$ we can finish the discretization: Each time-step is taken of equal length dt, furthermore if we have $k \in \{0, 1, 2, \dots\}$ such that $x(k = 0)$ coincides with x_0, $x(k = 1)$ coincides with $x(dt)$, $x(k = 2) = x(2dt)$, etc., then the discretized state space is given by

$$x(k+1) = \tilde{A}x(k) + \tilde{B}u(k), \tag{6.4a}$$

$$y(k) = \tilde{C}x(k), \tag{6.4b}$$

where

$$\begin{aligned} \tilde{A} &= e^{Adt}, \\ \tilde{B} &= A^{-1}\left(e^{Adt} - I_n\right)B, \\ \tilde{C} &= C. \end{aligned}$$

System (6.4) is the system we will identify. The recirculation parameters can then be found by finding A from $\tilde{A}$ via the relation above.

6.3 Subspace identification method

In this section we will explain the subspace identification method which is used for identification. We will treat our system as being completely deterministic, i.e. there is no measurement noise nor process noise in the system. It is possible to extend the algorithm to include these noises however this falls outside the scope of this thesis. In original system, (6.2), only the system matrix A is unknown, while the other system matrices are all known. Therefore the identification method will focus on finding $\tilde{A}$. To start we will formalize the problem statement

Problem 6.1. Given s measurements of the output, the temperature of the units $T_{\text{out}} \in \mathbb{R}^n$, and the input, the applied workload distribution $D \in \mathbb{R}^n$, generated by system (6.4) of order n, determine the system matrix $\tilde{A} \in \mathbb{R}^{n \times n}$ (up to within a similarity transformation). Afterwards determine the air recirculation parameters from $\tilde{A}$.

6.3.1 Theoretical background

The notation in the method can become a bit involved, therefore in this subsection some notation will be introduced.

Block Hankel matrices and state sequences

Block Hankel matrices play an important role in subspace identification. Both input and output block Hankel matrices can be determined from the input-output data. The block matrices are divided in two equal parts of i block rows that are somewhat loosely called past and future. Input block Hankel matrices are defined as:

$$
U_{0|2i-1} \triangleq \begin{array}{c} \\ i \\ \\ \\ \\ i \\ \\ \\ \end{array}
\overbrace{\begin{bmatrix}
u_0 & u_1 & u_2 & \cdots & u_{j-1} \\
u_1 & u_2 & u_3 & \cdots & u_j \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
u_{i-1} & u_i & u_{i+1} & \cdots & u_{i+j-2} \\
\hline
u_i & u_{i+1} & u_{i+2} & \cdots & u_{i+j-1} \\
u_{i+1} & u_{i+2} & u_{i+3} & \cdots & u_{i+j} \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
u_{2i-1} & u_2 i & u_{2i+1} & \cdots & u_{2i+j-2}
\end{bmatrix}}^{j}
\begin{array}{l} \\ \\ \\ \\ \\ \\ \\ \\ \end{array}
\begin{array}{l} \\ \\ \\ \text{"past"} \\ \\ \\ \\ \text{"future"} \end{array}
\tag{6.5a}
$$

$$
\triangleq \left[\frac{U_{0|i-1}}{U_{i|2i-1}}\right] \triangleq \left[\frac{U_p}{U_f}\right], \tag{6.5b}
$$

$$
U^+_{0|2i-1} \triangleq \begin{array}{c} \\ i+1 \\ \\ \\ \\ i-1 \\ \\ \\ \end{array}
\overbrace{\begin{bmatrix}
u_0 & u_1 & u_2 & \cdots & u_{j-1} \\
u_1 & u_2 & u_3 & \cdots & u_j \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
u_{i-1} & u_i & u_{i+1} & \cdots & u_{i+j-2} \\
u_i & u_{i+1} & u_{i+2} & \cdots & u_{i+j-1} \\
\hline
u_{i+1} & u_{i+2} & u_{i+3} & \cdots & u_{i+j} \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
u_{2i-1} & u_2 i & u_{2i+1} & \cdots & u_{2i+j-2}
\end{bmatrix}}^{j}
\begin{array}{l} \\ \\ \\ \\ \\ \\ \\ \\ \end{array}
\begin{array}{l} \\ \\ \\ \text{"past"} \\ \\ \\ \\ \text{"future"} \end{array}
\tag{6.5c}
$$

$$
\triangleq \left[\frac{U_{0|i}}{U_{i+1|2i-1}}\right] \triangleq \left[\frac{U^+_p}{U^-_f}\right]. \tag{6.5d}
$$

The difference between the matrices is the point where the past input ends and the future input starts. Furthermore:

- The horizontal line in the matrices is for visual guidance and only indicates the partitioning of the matrix.
- The subscript of $U_{0|2i-1}, U_{0|i-1}$, etc., indicates the first and last element respectively of the first column of the chosen partitioning. The

subscript p stands for past, and the subscript f stand for future. Lastly the superscripts $+$ and $-$, indicate that a partitioning contains one row extra or less than i rows.

- The number of block rows (i) is a user-defined index which is large enough, i.e. it should be at least larger than the maximum order of the system that is identified. Technically the number of block rows should only be larger than the largest observability index, however generally this index is unknown so it is safest to assume $i > n$. Note that each block row contains n (number of inputs) rows, therefore the Hankel block matrix, $U_{0|2i-1}$, contains $2ni$ rows.
- The number of columns (j) is typically equal to $s - 2i + 1$, which implies that all given data samples are used.
- Note again that the definition of past and future inputs is quite loose as both U_p and U_p^+ are denoted by "past inputs". The loose definition however helps in explaining concepts in a more intuitive way.

The output Hankel block matrices, $Y_{0|2i-1}, Y^+_{0|2i-1}, Y_p, Y_p^+, Y_f, Y_f^-$, are defined in the same way as the input Hankel block matrices. Following the notation of (Willems, 1986a; Willems, 1986b; Willems, 1987), we define the block Hankel matrices that combine past inputs and past outputs as $W_{0|i-1} \triangleq: W_p$ and $W_{0|i} \triangleq: W_p^+$

$$W_p = \begin{bmatrix} U_p \\ Y_p \end{bmatrix}, \quad W_p^+ = \begin{bmatrix} U_p^+ \\ Y_p^+ \end{bmatrix}. \tag{6.6}$$

State sequences play an important role in the derivation and interpretation of subspace identification algorithms. A state sequence is a collection of the state values starting at time-step i up to time-step $i + j - 1$. A state sequence is defined as

$$X_i \triangleq: \begin{bmatrix} x_i & x_{i+1} & \cdots & x_{i+j-2} & x_{i+j-1} \end{bmatrix} \in \mathbb{R}^{n \times j}, \tag{6.7}$$

where i denotes the first element of the state sequence. Analogous as before we define past and future state sequence by X_p and X_f respectively

$$X_p = X_0, \quad X_f = X_i, \tag{6.8}$$

where X_0 starts at element x_0 and ends at element x_{j-1}.

Observability matrix

The subspace identification method used here makes use of observability and controllability matrices and of their structure. For the implementation needed in this work, we only require the observability matrix. The extended observability matrix W_o^i with $i > n$ denoting the number of block rows is defined as:

$$W_o^i := \begin{bmatrix} \tilde{C} \\ \tilde{C}\tilde{A} \\ \vdots \\ \tilde{C}\tilde{A}^{i-1} \end{bmatrix} \in \mathbb{R}^{ni \times n}. \tag{6.9}$$

Here it is assumed that the pair $\{\tilde{A}, \tilde{C}\}$ is observable, which implies that the rank of the observability matrix is equal to n.

Covariance matrix

The covariance matrix $\Phi_{[A,B]}$ between two matrices $A \in \mathbb{R}^{m \times j}$ and $B \in \mathbb{R}^{m \times j}$ is defined as

$$\Phi_{[A,B]} \stackrel{\Delta}{=}: \lim_{j \to \infty} \frac{1}{j}(AB^T) \tag{6.10}$$

6.3.2 Main theorem

Now that we have introduced some notation we can explain the main theorem behind subspace identification. The main theorem uses the notion of persistency of excitation. In the theorem the definition of (Ljung, 1987) is adopted:

Definition 6.1 (Persistency of excitation). The input sequence $u_k \in \mathbb{R}^n$ is persistently exciting of order $2i$ if the input covariance matrix $\Phi_{[U_{0|2i-1}, U_{0|2i-1}]}$ has full rank, that is a rank of $2ni$.

Theorem 6.1. *Under the assumptions that*

1. *The input u_k is persistently exciting of order $2i$, see Definition 6.1.*
2. *The intersection of the row space of U_f (the future inputs) and the row space of X_p (the past states) is empty.*
3. *The user-defined weighting matrices $W_1 \in \mathbb{R}^{ni \times ni}$ and $W_2 \in \mathbb{R}^{j \times j}$ are chosen such that W_1 is of full rank and W_2 satisfies: rank(W_p) = rank($W_p W_2$), where W_p is the block Hankel matrix in* (6.6) *containing the past inputs and past outputs.*

And with the oblique projection, $\mathcal{O}_i$, defined as

$$\mathcal{O}_i \stackrel{\Delta}{=}: Y_f /_{U_f} W_p, \tag{6.11}$$

and the singular value decomposition

$$W_1 \mathcal{O}_i W_2 = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} S_1 & \mathbb{0} \\ \mathbb{0} & \mathbb{0} \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \tag{6.12}$$

$$= U_1 S_1 V_1^T, \tag{6.13}$$

we have that

1. *The matrix $\mathcal{O}_i$ is equal to the product of the extended observability matrix and the states*
$$\mathcal{O}_i = W_o^i X_f. \tag{6.14}$$
2. *The order of system* (6.4) *is equal to the number of singular values in* (6.12) *different from zero.*
3. *The extended observability matrix W_o^i is equal to*
$$W_o^i = W_1^{-1} U_1 S_1^{1/2} T, \tag{6.15}$$

where $T \in \mathbb{R}^{n \times n}$ an arbitrary non-singular similarity transformation.

4. *The part of the state sequence X_f that lies in the column space of W_2 can be recovered from*

$$X_f W_2 = T^{-1} S_1^{1/2} V_1^T. \tag{6.16}$$

5. *The state sequence X_f is equal to*

$$X_f = W_o^{i\dagger} \mathcal{O}_i, \tag{6.17}$$

where † denotes the Moore-Penrose pseudo-inverse.

The theorem introduces two user-specified weighting matrices W_1 and W_2. The specific choice for weighting matrices W_1 and W_2 will determine the state space basis in which the identified model will be identified. The deeper meaning of these matrices and why they are introduced becomes clear in extensions of the identification method for combined stochastic-deterministic systems, however this is beyond the scope of this thesis. The interested reader is referred to (Van Overschee and De Moor, 2012, Chapter 5). Furthermore the identified state matrices will be correct up to a similarity transformation to the original system matrices. The similarity transformation is given by

$$\tilde{A}_{id} = T^{-1} \tilde{A} T, \quad \tilde{B}_{id} = T^{-1} \tilde{B}, \quad \tilde{C}_{id} = \tilde{C} T \tag{6.18}$$

In our case we have that $\tilde{C} = I_n$, such that we immediately know the similarity transformation after having found $\tilde{C}_{id}$. Therefore in our case it is possible to obtain the original state matrices.

The theorem also introduces the oblique projection. The oblique projection is a decomposition of one matrix in linear combinations of two non-orthogonal matrices and the orthogonal complement of those matrices. Written out in full, the oblique projection is given by

$$\mathcal{O}_i = Y_f (I - U_f^T (U_f U_f^T)^\dagger U_f)(W_p (I - U_f^T (U_f U_f^T)^\dagger U_f))^\dagger W_p \tag{6.19}$$

The proof of this theorem is omitted here and can be found in (Van Overschee and De Moor, 2012, Section 2.3).

6.4 Subspace identification algorithm

From Theorem 6.1 it is possible to design two methods for identifying the desired system parameters. Below we will summarize the algorithm which has been implemented. For the thermodynamical system considered here, it is only necessary to find the system matrix, $\tilde{A}$ and $\tilde{C}$, therefore this algorithm was most suitable.

The algorithm consists of 6 steps:

1. Calculate the oblique projection:
$$\mathcal{O}_i = Y_f /_{U_f} W_p.$$
2. Calculate the singular value decomposition of the weighted oblique projection:
$$W_1 \mathcal{O}_i W_2 = USV^T.$$
3. Determine the order of the system by inspecting the singular values in S and partition the singular value decomposition accordingly to obtain U_1, U_2, and S_1.
4. Determine W_o^i as
$$W_o^i = W_1^{-1} U_1 S_1^{1/2}$$
5. Determine $\tilde{A}_{id}$ from W_o^i as $\tilde{A}_{id} = \underline{W_o^i}^{\dagger} \overline{W_o^i}$. Find $\tilde{C}_{id} = T$ from the first n rows of W_o^i, and use T to transform the system to the exact state matrices.
6. Find A from $\tilde{A}$ as $A = \frac{\log \tilde{A}}{dt}$, where $\log \tilde{A}$ denotes the matrix logarithm.

The weighting matrices W_1 and W_2 are chosen here as identity matrices of appropriate dimensions.

6.5 Identification experiment

The thermodynamical system consists of two sets of inputs, the supply temperature and the workload distribution among the units, and one set of outputs, the unit temperature. In our setting we are able to observe the temperature of all the units, i.e. output matrix $C = I_n$. Secondly, in section 6.2 we have assumed that the supply temperature remains constant. Changing the supply temperature would affect all the units equally, therefore any changes made to this input will not result in any helpful insights. As such the only requirement is to design a proper workload input.

Following Theorem 6.1, the input should be persistently exciting, therefore the input sequence should be chosen carefully. A constant input during the whole experiment run will result in a non full rank input covariance matrix, invalidating Theorem 6.1. By varying the workload input randomly at each time-step it is possible to ensure a full rank input covariance matrix however. This randomness is achievable by applying a workload trace of many small, short tasks. This will result in a randomly varying workload trace, fulfilling the theorem requirement.

6.6 Simulations

The workload input for the identification run in this chapter is initialized by a random, uniform distribution between 50-75% of each unit's computational capacity. At each time-step the workload of each unit is increased or decreased by a random amount following a uniform distribution with mean $\mathbb{0}$. As a result the total load applied during the identification run stays roughly around 62% of the total computational capacity, see Figure 6.1.

The supply temperature is kept at 9°C in order to keep the units below the safe temperature threshold of 30°C. The temperature of each unit is initialized at 27°C. The temperature of the units is depicted in Figure 6.2. Finally the workload distribution among the units is depicted in Figure 6.3. The load on each unit is varying slightly around $50\% - 75\%$ of the units total computational capacity (20 CPU's available).

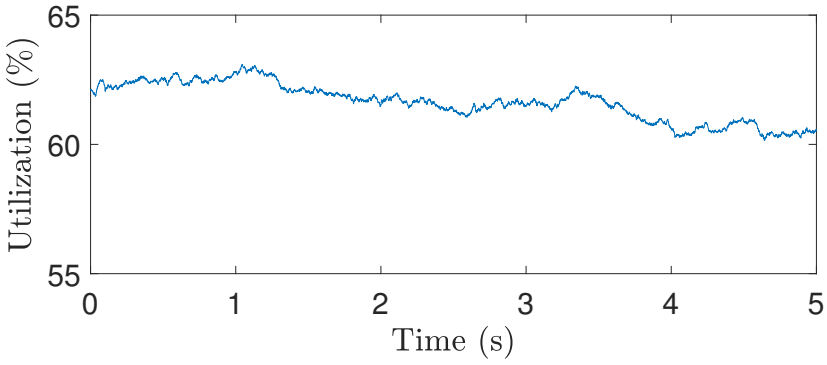


FIGURE 6.1: The total workload applied in the data center. It stays roughly around 62% of the total data center computational capacity.

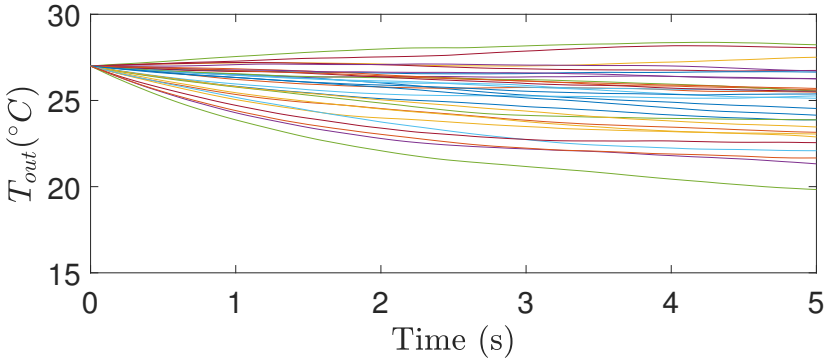


FIGURE 6.2: Temperature change of the units during identification run.

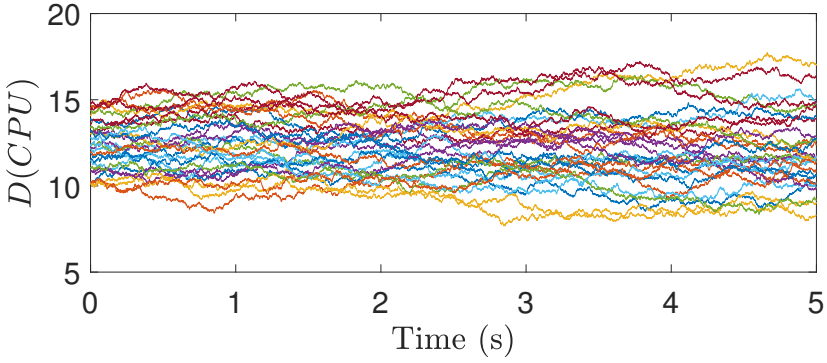


FIGURE 6.3: The workload distribution among the units. The vary slowly but randomly around 10-15 CPU's, 50-75% of the total unit computational capacity.

Following the algorithm described in section 6.4 the recirculation parameters of the system are identified. The success of the algorithm is calculated with the 2-norm of the error between the identified recirculation parameters and the original recirculation parameters. The 2-norm of the error is in the order of 10^{-7} so we can conclude that the subspace identification is indeed successfully identifying the recirculation parameters.

6.7 Conclusion

A subspace identification algorithm following the design of (Van Overschee and De Moor, 2012) is implemented in a data center setting in order to identify the heat recirculation parameters, that are crucial for implementing the controllers presented in this thesis. The subspace identification algorithm provided a very accurate estimate of the parameters with a 2-norm of the error in the order of 10^{-7}. The identification run itself is easy to implement and is independent from the context of the specific data center, making the procedure very easy to adopt by any data center operator interested in implementing the results of this thesis.

The algorithm presented here works for deterministic systems, however it is possible to extend the identification to include measurement noise, see (Van Overschee and De Moor, 2012). This is an interesting extension as in most real-life application there is always some measurement noise.

CHAPTER 7

Conclusions and future work

The online world is nowadays more than ever a vital part of our current society. Online data usage, cloud storage, and internet interactions are mostly done via (big) data centers, clusters of computer, server, and networking systems and components. The data centers are growing ever bigger and accompanied with this growth is an ever increasing power hunger. Already from the early days, about 15-20 years ago, scientists have worked on improving data centers, both by reducing energy consumption as by improving data center layouts and computing equipment. The work in this thesis tries to add to the field by bringing a theoretic understanding in an otherwise very practical oriented field. The main question addressed here: **How can we apply control engineering techniques in a data center context in order to reduce the energy consumption of the data centers**.

Energy reduction can be achieved in multiple ways, (1) by server consolidation, trying to run the same workload on less equipment, (2) by power management techniques, switching servers to low power consuming modes, or adjusting CPU frequency, or (3) by applying advanced cooling strategies, dividing workload among servers in order to equalize the thermal load, or more physical techniques such as optimizing the data center layout in order to improve thermal flows. In this work we contribute to the existing work by developing a theoretical framework from which we can design techniques to better equalize thermal load.

In section 7.1 we will present the conclusions of each chapter and in section 7.2 we will suggest some possible extensions to the current work.

7.1 Conclusions

The thesis starts in chapter 2 by constructing a thermodynamical model which relates two inputs, (1) the temperature setpoint of the cold air exiting the cooling equipment, and (2) the assigned workload of each unit, to the output, i.e. the temperature change of that unit. Data center equipment is usually air-cooled via big air conditioning units (CRAC's). The key part of the model is the thermodynamical unbalance which is created due to the air flows. Because air flows are typically very complex, the cold air doesn't reach each unit in equal amounts, furthermore the hot exhaust air cannot be fully extracted which leads to heat leakages, "air recirculation", among the units. Furthermore the model relates the temperature of the units to the power consumption of the cooling equipment such that it is possible to relate the temperature change to a change in power consumption.

With the model ready we set up an optimization problem in order to tackle the main research question. In chapter 3 a static optimization problem is formulated which minimizes the energy consumption subject to physical constraints, such as a maximum allowable unit temperature, computational capacity of the equipment, and thermal equilibrium at the optimal solution. The optimization problem itself is non-convex which is why, under some mild assumptions, the problem is rewritten as a linear optimization problem. Via the Karush-Kuhn-Tucker conditions it is possible to characterize the optimal solution exactly.

Two operating scenarios are described in the chapter: (1) the most common scenario, all the units have been assigned some jobs but are not maximally loaded, and (2) a scenario which might happen in certain cases, where some racks have been maximally loaded while the others are not but have some work assigned to them. Which one of these scenarios happens depends on the total load present in the data center, but in most current data center the default operational condition will be scenario (1). Coincidentally the optimal solution is easiest in this scenario and, as long as you stay within the correct operating range, independent on the exact total workload in the data center. With this knowledge it is possible to design integral controllers that will dynamically adjust the supply temperature and the workload

division among to units in order to steer the temperature distribution to the optimal distribution.

The integral controllers designed here only work when the total load is such that the optimal solution adheres to scenario (1). In order to extend the controllers to also work for scenario (2) it is necessary to dynamically calculate the optimal solution such that the optimal temperature setpoint will adjust to possible changing operating conditions. In chapter 4 we use a projected dynamical system in order to detect whenever we have hit the constraint boundary, i.e. computational capacity bound, for some of the units. We take a primal-dual algorithm which works for strictly convex systems, adapt it, and proof that it works for linear systems as well. Finally we interconnect the projected dynamical system with the integral controllers designed in chapter 3, and simulate the interconnected system. Although it is not possible (yet) to show theoretically that the interconnected system is stable, we have shown that in our simulation setting the interconnection does yield a stable system, i.e. under varying load conditions the primal-dual algorithm is able to follow the changing optimal setpoint and the integral controllers are able to drive the temperature distribution to the changing setpoint.

As stated before there are multiple approaches to reduce the energy consumption of the data center. In the literature, results tend to focus on only one area at the same time, similar as to how we only focused on cooling strategies so far. In chapter 5 a first attempt is made in order to combine advanced cooling strategies with power management strategies in order to further reduce data center energy consumption. To that end, the integral controllers developed in chapter 3 have been integrated in a simulation framework developed elsewhere. In order to study the influence of each modeling strategy on the energy consumption and performance of the data center, four operational scenarios are studied: (1) the base case, no control strategy is applied, (2) advanced cooling, only the thermodynamical controllers are applied, (3) advanced power management, only the power management strategies developed elsewhere are implemented, (4) combined, here both cooling and power management strategies are applied simultaneously. We show via simulations that while each strategy has merits on its own, the best results were obtained by combining both energy reduction techniques. This shows

that there is promise in further investigation in combining multiple energy saving techniques in order to achieve the greatest savings.

Finally in chapter 6 we introduce a subspace identification algorithm that can help identifying the air recirculation parameters, a vital part of the thermodynamical model. The thermodynamical model is defined by a static mapping of the heat recirculation/heat leakages among the units. In order for data center operators to implement our results it is important that these parameters are known. System identification methods, like subspace identification, have produced easy-to-implement methods in order to find these parameters by running simple identification runs. We show that the 2-norm of the error between the recirculation parameters and the identified parameters is in the order of 10^{-7}.

7.2 Future work

While the results presented in this thesis are promising, the framework is not finished. However the framework can serve as a starting point upon which new results can be built. In the following we will suggest several extensions to the current work.

7.2.1 Power state switching

In the current work it is assumed that the power state of the equipment is fixed, i.e. fully powered on. However in power management techniques, energy consumption is reduced by switching the equipment to different power states given the current operating conditions. In chapter 5 we have made a first attempt to combine power management techniques with advanced cooling strategies. The two strategies worked as two separate entities trying to cooperate to achieve greater energy consumption reduction. It would be valuable to combine the power state switching into the theoretical model in order to better understand how this dynamic influences the thermodynamical workload scheduling. This would significantly increase the complexity of the model as this would introduce a switching dynamic, but could possibly lead to a better characterization of the optimal operating point, thus further reducing energy consumption while still ensuring good data center

performance. The power consumption of the computing equipment is then given by

$$P = \sum_{i \in G} P_i = \sum_{i \in G} \sum_{j \in G(i)} (v_{j,i} + w_{j,i} D_j), \tag{7.1}$$

where G is the set that contains all possible power states, $v_{j,i}$, and $w_{j,i}$ is the idle power consumption and CPU power consumption respectively of unit j in power state i, and D_j is the number of jobs assigned to unit j.

In order to make this work it is necessary to define switching criteria that dictate when server will switch power state. Furthermore to make this work, a notion of data center performance has to be added to the model, such as job execution time, SLA violations, or timing requirements. The performance can then be taken into account in the optimization problem, either as part of the cost function, or less restrictively as a constraint.

Alternatively for the switched system, the power switching can also be modeled as a continuous function dependent on CPU frequency and voltage. The idea behind this is that CPU frequency and voltage can be adjusted dynamically to lower the energy consumption of the CPU at the cost of decreased performance. In times of low workload demand, the data center equipment can then be scaled down in order to reduce the energy bill. These techniques are already widely used in battery-powered applications and embedded systems under the denominator of dynamic voltage and frequency scaling (DVFS). The switching power dissipated by a CPU can be approximated by

$$P_{CPU} = CV^2Af, \tag{7.2}$$

where C is the capacitance being switched per clock cycle, V is the supply voltage, A is the switching activity, i.e. number of switches per clock cycle, and f is the switching frequency. It is important to note that frequency and voltage are intimately coupled, i.e. the voltage required for stable operation is determined by the frequency at which the circuit is clocked. Therefore the voltage can only be reduced if the frequency is also reduced. Sometimes researchers simplify the equation by eliminating the voltage dependency and writing the power consumption of the CPU and other components as a function of CPU frequency. The power consumption of a unit for processing

a job is then given by

$$P_{proc} = af^3 + bf + c, \tag{7.3}$$

where a, b, and c are constants (Elnozahy, Kistler, and Rajamony, 2002; Chen et al., 2005; Sarood et al., 2014). With the frequency model for the power consumption, distinct power states do not exist, rather it is a continuum of possible states, however dealing with a third-order model has its own downsides. Still it could be valuable to attempt to incorporate this model in the existing framework.

All-in-all these extension aim at merging power management techniques with advanced cooling strategies in order to further reduce the energy consumption of the computational equipment. Which, as we have shown in chapter 5, seems to be a promising direction.

7.2.2 Power characteristics equipment

We make the assumption in this work that the data center equipment is all identical. While this is true for the larger data centers, other data centers might work in situations where equipment variety exists, e.g. two or three generations or sets of equipment. This variety will add a second factor in the energy optimization, namely that the specific choice of server to which a job is assigned, matters. This invalidates the homogeneity assumption and makes the linearization of the non-convex optimization problem not direct. A generalization in this direction will increase the utility of the results presented here greatly.

7.2.3 Integrated PDS-integral control

In chapter 4 we designed a projected dynamical system (PDS) to dynamically find the optimal solution under varying operating conditions. We were not able to show that the interconnection between the PDS-controller and the integral controller is stable. However in a practical situation it does show stable behavior. While intuitively this can be argued, a characterization of the stability is desirable, i.e. if the interconnection is stable in any given connection or operating condition, or if not, when it is not stable. The proof

for this could possibly be found by investigating passivity properties of both systems, or by characterizing the convergence time for both systems individually and stating stability conditions based on those convergence times.

7.2.4 System identification

The subspace identification algorithm suggested here, is designed for deterministic systems. Practical systems however will very often be corrupted by measurement noise, especially a volatile parameter like temperature. Since algorithmic extensions for including these measurement noises exist, these extended algorithms could be studied in the data center context.

Secondly, it is assumed that it is possible to measure the temperature of every unit in the data center, i.e. the output matrix $C = I_n$. However, given the versatility of subspace identification methods, it might be possible to reconstruct the air recirculation parameters from less temperature information. For example, if a temperature sensor would cover more than one unit, it is possible to reduce the number of sensors significantly. As a result the amount of temperature data which needs to be transmitted, stored, and processed by the data center would reduce significantly.

7.2.5 Time delays

For the results in this thesis it is assumed that the cold air instantly reaches the server equipment and that there is zero delay. Of course implementing delays would be more accurate, however a formal investigation of time delays poses significant challenges for the theoretical analysis. From a technical viewpoint, one main difficulty is that the storage functions seem not suitable any longer and it is not immediate to see how to suitably modify them. Another difficulty is that, to the best of our knowledge, there is no clear characterization of how cooling delays actually enter the system dynamics. In the literature, one can find models describing "short" (Bash and Forman, 2007) as well as "long" (Liu et al., 2009) propagation delays. This makes it hard to qualitatively value this aspect and, as such, we have decided to neglect this aspect in our work. Future research can focus on studying the impacts of these transportation delays on the stability of the suggested controllers.

Bibliography

Abeles, P. (2017). *Efficient Java Matrix Library*. `http://ejml.org`.

Albea, C., A. Seuret, and L. Zaccarian (Dec. 2014). "Hybrid control of a three-agent network cluster". In: *53th IEEE Conference on Decision and Control*, pp. 5302–5307.

Arrow, J., L. Hurwicz, H. Uzawa, and H. B. Chenery (1958). *Studies in linear and non-linear programming*. Stanford, CA: Stanford University Press.

Arsie, A. and C. Ebenbauer (2010). "Locating omega-limit sets using height functions". In: *Journal of Differential Equations* 248.10, pp. 2458–2469.

ASHRAE (2011). *2011 Thermal Guidelines for Data Processing Environments - Expanded Data Center Classes and Usage Guidance*. Tech. rep. Ashrae.

Banerjee, A., T. Mukherjee, G. Varsamopoulos, and S. K. Gupta (2011). "Integrating cooling awareness with thermal aware workload placement for HPC data centers". In: *Sustainable Computing: Informatics and Systems* 1.2, pp. 134–150.

Bash, C. and G. Forman (2007). "Cool Job Allocation: Measuring the Power Savings of Placing Jobs at Cooling-Efficient Locations in the Data Center." In: *USENIX Annual Technical Conference*. Vol. 138, p. 140.

Blatch, D. (Feb. 2014). "Is the industry getting better at using power?" In: *Datacenter Dynamics Focus* 3.33, pp. 16–17.

Boucher, T. D., D. M. Auslander, C. E. Bash, C. C. Federspiel, and C. D. Patel (2006). "Viability of dynamic cooling control in a data center environment". In: *Journal of electronic packaging* 128.2, pp. 137–144.

Boyd, S. and L. Vandenberghe (2004). *Convex optimization*. Cambridge university press.

Brogliato, B. and D. Goeleven (2005). "The Krakovskii-LaSalle invariance principle for a class of unilateral dynamical systems". In: *Mathematics of Control, Signals and Systems* 17.1, pp. 57–76.

Brogliato, B., A. Daniilidis, C. Lemaréchal, and V. Acary (2006). "On the equivalence between complementarity systems, projected systems and differential inclusions". In: *Systems & Control Letters* 55.1, pp. 45–51.

Chen, Y., A. Das, W. Qin, A. Sivasubramaniam, Q. Wang, and N. Gautam (2005). "Managing server energy and operational costs in hosting centers". In: *ACM SIGMETRICS performance evaluation review*. Vol. 33. 1. ACM, pp. 303–314.

Cherukuri, A., E. Mallada, and J. Cortés (2016). "Asymptotic convergence of constrained primal–dual dynamics". In: *Systems & Control Letters* 87, pp. 10–15.

Cherukuri, A., E. Mallada, S. Low, and J. Cortés (2017). "The role of convexity on saddle-point dynamics: Lyapunov function and robustness". In: *IEEE Transactions on Automatic Control*.

Dai, X., J. M. Wang, and B. Bensaou (2016). "Energy-efficient virtual machines scheduling in multi-tenant data centers". In: *IEEE Transactions on Cloud Computing* 4.2, pp. 210–221.

Dayarathna, M., Y. Wen, and R. Fan (2016). "Data center energy consumption modeling: A survey". In: *IEEE Communications Surveys & Tutorials* 18.1, pp. 732–794.

Doyle, J., F. Knorn, D. O'Mahony, and R. Shorten (Nov. 2013). "Distributed thermal aware load balancing for cooling of modular data centres". In: *IET Control Theory and Applications* 7 (4), pp. 612–622.

Elnozahy, E. M., M. Kistler, and R. Rajamony (2002). "Energy-efficient server clusters". In: *International Workshop on Power-Aware Computer Systems*. Springer, pp. 179–197.

Emerson Network Power (2009). "Energy logic: Reducing data center energy consumption by creating savings that cascade across systems". In: *White Paper of Emerson Electric Co*.

Enerdata (Aug. 2016). *Global domestic electricity consumption*.

Fan, X., W.-D. Weber, and L. A. Barroso (2007). "Power provisioning for a warehouse-sized computer". In: *ACM SIGARCH Computer Architecture News*. Vol. 35. ACM, pp. 13–23.

Feijer, D. and F. Paganini (Sept. 2010). "Stability of primal-dual gradient dynamics and applications to network optimization". In: *Automatica* 46, pp. 1974–1981.

Gaggero, M. and L. Caviglione (Dec. 2014). "A predictive control approach for energy-aware consolidation of virtual machines in cloud computing". In: *53th IEEE Conference on Decision and Control*, pp. 5308–5313.

Gandhi, A., S. Doroudi, M. Harchol-Balter, and A. Scheller-Wolf (2013). "Exact analysis of the M/M/k/setup class of Markov chains via recursive renewal reward". In: *ACM SIGMETRICS Performance Evaluation Review*. Vol. 41. 1. ACM, pp. 153–166.

Gao, Y., H. Guan, Z. Qi, B. Wang, and L. Liu (2013). "Quality of service aware power management for virtualized data centers". In: *Journal of Systems Architecture* 59.4-5, pp. 245–259.

Goebel, R. (2017). "Stability and robustness for saddle-point dynamics through monotone mappings". In: *Systems & Control Letters* 108, pp. 16–22.

Gupta, V., R. Nathuji, and K. Schwan (2011). "An analysis of power reduction in datacenters using heterogeneous chip multiprocessors". In: *ACM SIGMETRICS Performance Evaluation Review* 39.3, pp. 87–91.

Haddad, W. M. and V. S. Chellaboina (2008). *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*. Princeton University Press.

Hameed, A., A. Khoshkbarforoushha, R. Ranjan, P. P. Jayaraman, J. Kolodziej, P. Balaji, S. Zeadally, Q. M. Malluhi, N. Tziritas, A. Vishnu, S. U. Khan, and A. Zomaya (June 2014). "A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems". In: *Computing*, pp. 1–24.

Hatanaka, T., X. Zhang, W. Shi, M. Zhu, and N. Li (2017). "An integrated design of optimization and physical dynamics for energy efficient buildings: A passivity approach". In: *Conference on Control Technology and Applications (CCTA)*. IEEE, pp. 1050–1057.

Heath, T., A. P. Centeno, P. George, L. Ramos, Y. Jaluria, and R. Bianchini (2006). "Mercury and freon: temperature emulation and management for server systems". In: *12th international conference on Architectural support for programming languages and operating systems*. ASPLOS XII, pp. 106–116.

Jansson, M. and B. Wahlberg (1996). "A linear regression approach to state-space subspace system identification". In: *Signal Processing* 52.2, pp. 103–129.

Jokić, A., M. Lazar, and P. P. Van den Bosch (2009). "On constrained steady-state regulation: dynamic KKT controllers". In: *IEEE Transactions on Automatic Control* 54.9, pp. 2250–2254.

Katayama, T. (2006). *Subspace methods for system identification*. Springer Science & Business Media.

Khalil, H. K. (2002). *Nonlinear systems*. Vol. 3. Prentice hall Upper Saddle River, NJ.

Larimore, W. E. (1990). "Canonical variate analysis in identification, filtering, and adaptive control". In: *Decision and Control, 1990., Proceedings of the 29th IEEE Conference on*. IEEE, pp. 596–604.

Lauri Minas, B. E. (2009). *The problem of power consumption in servers*. Tech. rep. Santa Clara, CA, USA: Intel.

Li, S., H. Le, N. Pham, J. Heo, and T. Abdelzaher (June 2012). "Joint Optimization of Computing and Cooling Energy: Analytic Model and A Machine Room Case Study". In: *32nd Int. Conf. on Distributed Computing Systems*. IEEE, pp. 396–405.

Liu, J., F. Zhao, X. Liu, and W. He (2009). "Challenges towards elastic power management in internet data centers". In: *Distributed Computing Systems Workshops, 2009. ICDCS Workshops' 09. 29th IEEE International Conference on*. IEEE, pp. 65–72.

Ljung, L. (1987). *System identification: theory for the user*. Prentice-hall.

Moore, J., J. Chase, R. Parthasarathy, and S. Ratnesh (2005). "Making scheduling 'cool' temperature-aware workload placement in data centers". In: *USENIX Annual Technical Conference*, pp. 61–74.

Mukherjee, T., Q. Tang, C. Ziesman, S. K. Gupta, and P. Cayton (2007). "Software architecture for dynamic thermal management in datacenters". In: *Communication Systems Software and Middleware, 2007. COMSWARE 2007. 2nd International Conference on*. IEEE, pp. 1–11.

Mukherjee, T., A. Banerjee, G. Varsamopoulos, S. K. Gupta, and S. Rungta (2009). "Spatio-temporal thermal-aware job scheduling to minimize energy consumption in virtualized heterogeneous data centers". In: *Computer Networks* 53.17, pp. 2888–2904.

Parolini, L., B. Sinopoli, B. H. Krogh, and Z. Wang (Jan. 2012). "A cyber-physical systems approach to data center modeling and control for energy efficiency". In: *Proceedings of the IEEE* 100 (1), pp. 254–268.

Postema, B. F. (2013). *DaCSim: A Data Centre Simulation Framework*. `https://github.com/bjornpostema/DaCSim`.

Postema, B. F. and B. R. Haverkort (Aug. 2015). "An AnyLogic Simulation Model for Power and Performance Analysis of Data Centres". In: ed. by C. P. Engineering. Vol. 9272. Lecture Notes in Computer Science. Springer Link, pp. 258–272.

— (2017). "Specification of Data Centre Power Management Strategies". In: *Proc. of the 6th Int. Workshop on Energy-Efficient Data Centres (E2DC)*. Hong Kong.

— (2018). "Evaluation of Advanced Data Centre Power Management Strategies". In: *Electronic notes in theoretical computer science* 337, pp. 173–191.

Postema, B. F., T. Van Damme, C. De Persis, P. Tesi, and B. R. Haverkort (2018). "Combining Energy Saving Techniques in Data Centres using Model-Based Analysis". In: *Companion of the 2018 ACM/SPEC International Conference on Performance Engineering*. ACM, pp. 67–72.

Ranganathan, P., P. Leech, I. David, and C. J. S. (2006). "Ensemble-level Power Management for Dense Blade Servers". In: *33rd annual international symposium on Computer Architecture*. ISCA, pp. 66–77.

Rao, G. and H Unbehauen (2006). "Identification of continuous-time systems". In: *IEE Proceedings-Control theory and applications* 153.2, pp. 185–220.

Richert, D. and J. Cortés (2015). "Robust distributed linear programming". In: *IEEE Transactions on Automatic Control* 60.10, pp. 2567–2582.

Rosen, J. B. (1965). "Existence and uniqueness of equilibrium points for concave n-person games". In: *Econometrica: Journal of the Econometric Society*, pp. 520–534.

Sarood, O., A. Langer, A. Gupta, and L. Kale (2014). "Maximizing throughput of overprovisioned hpc data centers under a strict power budget". In: *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE Press, pp. 807–818.

Schmidt, R. R. (2004). "Thermal Profile of a High-Density Data Center-Methodology to Thermally Characterize a Data Center". In: *ASHRAE Transactions* 110, p. 635.

Sepulchre, R, M Jankovic, and P Kokotovic (1997). *Constructive Nonlinear Control*.

Shebabi, A., S. Smith, D. Sartor, R. Brown, M. Herrlin, J. Koomey, E. Masanet, N. Horner, I. Azevedo, and W. Lintner (2016). *United States Data Center Energy Usage Report*. Tech. rep. Lawrence Berkeley National Laboratory.

Simpson-Porco, J. W. (2016). "Input/output analysis of primal-dual gradient algorithms". In: *Allerton Conference on Communication, Control, and Computing*. IEEE, pp. 219–224.

Stegink, T. W., C. De Persis, and A. J. Van Der Schaft (2015). "Port-Hamiltonian formulation of the gradient method applied to smart grids". In: *IFAC-PapersOnLine* 48.13, pp. 13–18.

— (2017). "A unifying energy-based approach to stability of power grids with market dynamics". In: *IEEE Transactions on Automatic Control* 62.6, pp. 2612–2622.

Stegink, T. W., T. Van Damme, and C. De Persis (2018). "Convergence of projected primal–dual dynamics with applications in data centers". In: *7th IFAC Workshop on Distributed Estimation and Control in Networked Systems*.

Tang, Q., T. Mukherjee, S. K. S. Gupta, and P. Cayton (Dec. 2006a). "Sensor-Based Fast Thermal Evaluation Model For Energy Efficient High-Performance Datacenters". In: *Fourth International Conference on Intelligent Sensing and Information Processing, 2006. ICISIP 2006*. IEEE, pp. 203–208.

Tang, Q., S. K. S. Gupta, and G. Varsamopoulos (Nov. 2008). "Energy-efficient thermal-aware task scheduling for homogeneous high-performance computing data centers: a cyber-physical approach". In: *IEEE trans. on Parallel and distributed systems* 19 (11), pp. 1458–1472.

Tang, Q., S. K. S. Gupta, D. Stanzione, and P. Cayton (Sept. 2006b). "Thermal-Aware Task Scheduling to Minimize Energy Usage of Blade Server Based Datacenters". In: *2nd IEEE Int. Symp. on Dependable, Autonomic and Secure Computing*. IEEE, pp. 195–202.

Van Damme, T., C. De Persis, and P. Tesi (2017). "Optimized Thermal-Aware Job Scheduling and Control of Data Centers". In: *Proceedings of the IFAC World Congress*.

— (2018). "Optimized Thermal-Aware Job Scheduling and Control of Data Centers". In: *IEEE Transactions on Control Systems Technology*, pp. 1–12.

Van Overschee, P. and B. De Moor (1994). "N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems". In: *Automatica* 30.1. Special issue on statistical signal processing and control, pp. 75–93.

— (2012). *Subspace identification for linear systems: Theory - Implementation - Applications*. Springer Science & Business Media.

Vasic, N., T. Scherer, and W. Schott (2010). "Thermal-Aware Workload Scheduling for Energy Efficient Data Centers". In: *Proceedings of the 7th international conference on Autonomic computing*, pp. 169–174.

Verhaegen, M. and P. Dewilde (1992). "Subspace model identification Part 1. The output-error state-space model identification class of algorithms". In: *International Journal of Control* 56.5, pp. 1187–1210.

Wen, J. T. and M. Arcak (2004). "A unifying passivity framework for network flow control". In: *IEEE Transactions on Automatic Control* 49.2, pp. 162–174.

Willems, J. C. (1986a). "From time series to linear system - Part I". In: *Automatica* 22.5, pp. 561–580.

— (1986b). "From time series to linear system - Part II". In: *Automatica* 22.6, pp. 675–694.

— (1987). "From time series to linear system - Part III". In: *Automatica* 23.1, pp. 87–115.

Yi, P., Y. Hong, and F. Liu (2016). "Initialization-free distributed algorithms for optimal resource allocation with feasibility constraints and application to economic dispatch of power systems". In: *Automatica* 74, pp. 259–269.

Yin, X. and B. Sinopoli (2014). "Adaptive robust optimization for coordinated capacity and load control in data centers". In: *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE, pp. 5674–5679.

Zhang, W., Y. Wen, Y. W. Wong, K. C. Toh, and C.-H. Chen (2016). "Towards Joint Optimization Over ICT and Cooling Systems in Data Centre: A Survey". In: *IEEE Communications Surveys & Tutorials* 18.3, pp. 1596–1616.

Summary

Shortly after the widespread introduction of the world wide web, and the rise of computing equipment in households and businesses, there has been a growing desire for more centralized computing centers. As the computational demands of organizations increased, there was a drive to cluster computational power, in order to reduce maintenance requirements, lower operational costs, and improve computational capacity. These computational clusters, called data centers, have grown to enormous proportions where nowadays companies like Google, Facebook, and Dropbox base their complete business model on these constructions.

One of the major focus points in operating and maintaining data centers, is reducing operational costs and in particular reducing the energy consumption. Ever since the early 2000s researchers have picked up on this topic and suggested many methods to lower energy consumption in different ways. As a result, the growth in energy consumption has slowed down significantly compared to the steady growth of data centers. However in the upcoming years the growth in data centers is not going to slow down, as companies and services are still moving to the cloud en masse. Therefore it is ever more important that research is done in ways to improve the efficiency of data centers and reduce the energy consumption as much as possible.

While there are different ways in which it is possible to reduce the energy consumption, the work in this thesis focuses on thermal-aware strategies. Computing equipment in data centers produce large amounts of heat, that needs to be extracted in order to maintain the computing equipment at acceptable temperature levels. Prolonged operation at higher temperatures than recommended leads to increased equipment failure, which in turn leads to unacceptable monetary losses. Hence, the goal of thermal-aware strategies is to improve the efficiency of the cooling equipment, while also preventing these heat spikes.

In order to understand and study the problem in a fundamental way, the problem of minimizing energy consumption is cast as an optimization

problem in this work. With this theoretical framework, it is possible to understand what constitutes an optimal operating point that minimizes the cooling energy consumption. Furthermore it is shown that it is possible to design controllers that will automatically steer data center operations to the optimal operating point, while having limited operational information available. One limiting factor to the first control design, is the workable operating range. In order to extend the usability of the proposed controllers, a second controller is introduced that dynamically solves the optimization problem. The two controllers are interconnected, and simulations show the usability of the new, interconnected controller.

Alternatively it is possible to reduce energy consumption with power-aware strategies. The focus of these strategies lies on improved power management techniques, for example more efficient operational techniques such that operators can reduce the necessary quantity of servers, or implementing power state switching that utilizes low-power states of the computing equipment whenever less services are demanded from the data center. Research results typically focus on only one type of strategy, while further improvements can be obtained by combining multiple techniques. In a simulation study we show the potential benefits to be had by combining theoretical thermal models with power management techniques in a large simulation framework, as combined thermal- and power-aware strategies yield the largest reductions in power consumption compared to each strategy individually.

Finally the thesis concludes with a system identification study where it is shown how the thermodynamical model can be reconstructed for any given data center. Easy-to-perform experiments are designed from which it is possible to deduce the thermodynamical structure inside the data center.

Samenvatting

Kort na de wijdverspreide introductie van het 'world wide web', en de opmars van computers in alle aspecten van de samenleving, ontstond er een groeiend verlangen om die rekenkracht te centraliseren. Een data center is de overkoepelende term voor zo'n rekencluster. De reden dat data centers steeds meer in trek waren, en nog steeds zijn, is dat het onderhoud van servers en de bijbehorende infrastructuur efficiënter en goedkoper georganiseerd kan worden. Bovendien is het mogelijk om rekenkracht van de servers beter te benutten in een data center. Tegenwoordig hebben data centers enorme proporties en baseren bedrijven zoals Google, Facebook, en Dropbox hun complete zakenmodel op deze constructies.

Een van de kernzaken in het onderhoud en het draaiende houden van data centers, is het reduceren van de operationele kosten. Met name het verminderen van het energieverbruik is hierbij belangrijk aangezien dit een grote kostenpost is. Vandaar dat sinds begin 21ste eeuw veel onderzoek is gedaan naar verschillende manieren om dit energieverbruik te verminderen. Als gevolg hiervan zijn data centers steeds kunnen blijven doorgroeien in grootte terwijl het totale energieverbruik bijna gelijk is gebleven. Deze groei zal de komende tijd nog niet stoppen, vandaar dat het noodzakelijk blijft om op zoek te gaan naar nieuwe manieren om de efficiëntie van data centers te verbeteren, en om het energieverbruik verder te reduceren.

Hoewel energieverbruik via meerdere manieren gereduceerd kan worden, zijn de resultaten in dit proefschrift gebaseerd op thermisch-bewuste strategieën. Servers en de bijbehorende infrastructuur genereren veel hitte. Deze hitte moet vervolgens uit het data center getransporteerd worden om er voor te zorgen dat de apparaten niet oververhit raken. Langdurige blootstelling aan overmatige hitte leidt namelijk tot een verkorte levensduur van de servers en daarmee tot onacceptabele kosten. Thermisch-bewuste strategieën hebben als doel om de efficiëntie van het koelsysteem te verbeteren, terwijl ze hittepieken moeten voorkomen.

Om het probleem vanuit een fundamenteel oogpunt te bestuderen en te begrijpen, wordt het minimaliseren van het energieverbruik opgesteld als een optimalisatieprobleem. Met dit theoretische kader is het mogelijk om te begrijpen welke operationele keuzes leiden tot een minimalisatie in energieverbruik. Daarbij worden regelaars ontworpen die, met beperkte informatie, het data center automatisch richting het optimale operationele punt sturen. Een beperkende factor in het eerste ontwerp voor de regelaars is een gelimiteerd werkbereik. Om deze beperking te verhelpen, wordt een tweede regelaar geïntroduceerd die het optimalisatieprobleem op een dynamische wijze oplost. Vervolgens worden de twee regelaars gekoppeld en wordt de werking ervan middels een simulatiestudie geverifieerd.

Een tweede manier om het energieverbruik te reduceren, is met stroomgerichte strategieën. Deze strategieën richten zich op het verbeteren van het beheer van de servers. Hierdoor wordt het mogelijk om het totaal aantal benodigde servers om dezelfde hoeveelheid werk te verzetten, te verminderen. Als tweede mogelijkheid proberen ze de verbruikstoestand actief te reguleren zodanig dat servers in een lage verbruikstoestand worden gezet ten tijde van lage vraag. Onderzoek richt zich normaal gesproken enkel op een type strategie. Echter zijn verdere verbeteringen te behalen door het combineren meerdere type strategieën. In een uitvoerige simulatiestudie laten we de potentie zien van het combineren van thermisch-bewuste en stroom-gerichte strategieën.

Dit proefschrift wordt afgerond met een identificatiestudie naar het thermodynamisch model. Hiermee is het mogelijk om de thermodynamische structuur van de luchtstromen te karakteriseren voor elk data center. Eenvoudig uitvoerbare experimenten worden beschreven waarmee deze identificatie mogelijk wordt.