Workload Migration across Distributed Data Centers under Electrical Load Shedding

Linfeng Shen*, Fangxin Wang†, Feng Wang‡, Jiangchuan Liu*
*School of Computing Science, Simon Fraser University, Canada
†Future Network of Intelligence Institute (FNii) and School of Science and Engineering
The Chinese University of Hong Kong, Shenzhen, China
‡Department of Computer and Information Science, The University of Mississippi, USA

2021 IEEE/ACM 29th International Symposium on Quality of Service (IWQOS) | 978-1-6654-1494-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/IWQOS52092.2021.9521297

***Abstract*—Data centers are essential components in the current digital world. The number and scales of data centers have both increased a lot in recent years. The distributed data centers are standing out as a promising solution due to the development of modern applications which need a massive amount of computation resource and strict response requirement. However, compared to centralized data centers, distributed data centers are more fragile when the power supply is unstable. Power constraints or outages because of electrical load shedding or other reasons will significantly affect the service performance of data centers and damage the quality of service (QoS) for customers. Moreover, unlike conventional data centers, distributed data centers are often unattended, so we need a system that can automatically calculate the best workload schedule to maximize profit in such situations. In this paper, we closely investigate the influence of electrical load shedding in distributed data centers and construct a physical model to estimate the relationship among power, heat and workload. We then use queueing theory to approximate the tasks' response time and aim to minimize the overall response time of tasks by migration. Our extensive evaluations show that our method can improve the response time with more than 9% reduction.**

I. Introduction

Recent years have witnessed the rapid development of cloud computing. It brings tremendous and pay-as-you-go computation resources to fill the gap between the computing demands of ever-increasing mobile devices and the limited onboard computation power [1] where data centers (DCs), as the core component of cloud computing, have become the brain of nowadays digital world to provide storage, computation, and management for those inter-connected devices over the high-speed Internet. On the other hand, many emerging modern applications, such as autonomous driving, VR/AR, video analytics, etc., require not only a massive amount of computation resources but also ultra-low delay to guarantee the fast response, rendering the conventional centralized DC-based task offloading ineffective. To this end, distributed data centers stand out as a promising solution. Distributed DCs are usually small-scale server clusters deployed at the edge of the Internet, which handle the various tasks from numerous nearby sources and provide instant feedbacks with lower operation cost and delay compared to centralized data centers.

978-0-7381-3207-5/21/$31.00 © 2021 IEEE

Different from centralized DCs, which are usually installed with powerful backup batteries or UPSs and well attended with sophisticated water cooling systems [2], to allow dense deployment in large regions, distributed DCs often use much less backup power supply and cheaper air cooling systems to reduce the operational costs. These factors render a key challenge lies in the confluence of the ever-changing workload demand at each distributed DC and the unreliable power supply therein, particularly due to load shedding (which means the utilities will cut back the supply voltage when electrical generation and transmission systems cannot meet the demand requirements), as well as planned or accidental power outages [3], which tends to cause severe service delays or even service interruptions. For example, the government of China limits the power in several provinces due to the impact of weather or green energy policy [4]. Power supplies are being cut to some industrial and commercial customers in Hunan and Jiangxi provinces, where demand has jumped by at least 18% over the previous year. Severe winter storm causes power outages to about 220,000 utility customers in Texas recently [5]. Load shedding in South Africa has crippled many data centers and incurs lots of extra costs [6]. It is estimated that a four-and-a-half-hour load shedding could cost the operator of a data center 100,000 rand. As such, when power supply becomes constrained, the server utilization of a DC will be affected accordingly, usually causing the scale-down of the task processing capacity, so as to reduce the heat generation and thus the load of its power-hungry cooling system.

On the other hand, workload migration has been proposed as an effective solution to maximize the resource efficiency by scheduling tasks of congested DCs to those light-loaded ones. And pioneer works have explored many solutions to achieve this goal with such considerations as different data center capacities, migration cost, resource efficiency and electricity price [7] [8]. Different from previous work, in this paper, we for the first time to carefully consider the impacts of load shedding on the distributed DCs, especially on their server utilization as well as the resulting ineffective resource usage and degraded QoS, and we further propose to use workload migrations to effectively tackle such power constraint caused issues therein. To achieve this, we first examine the relationship between the power constraints and the corresponding server utilization by fully considering the cooling requirement

978-1-6654-1494-4/21/$31.00 ©2021 IEEE

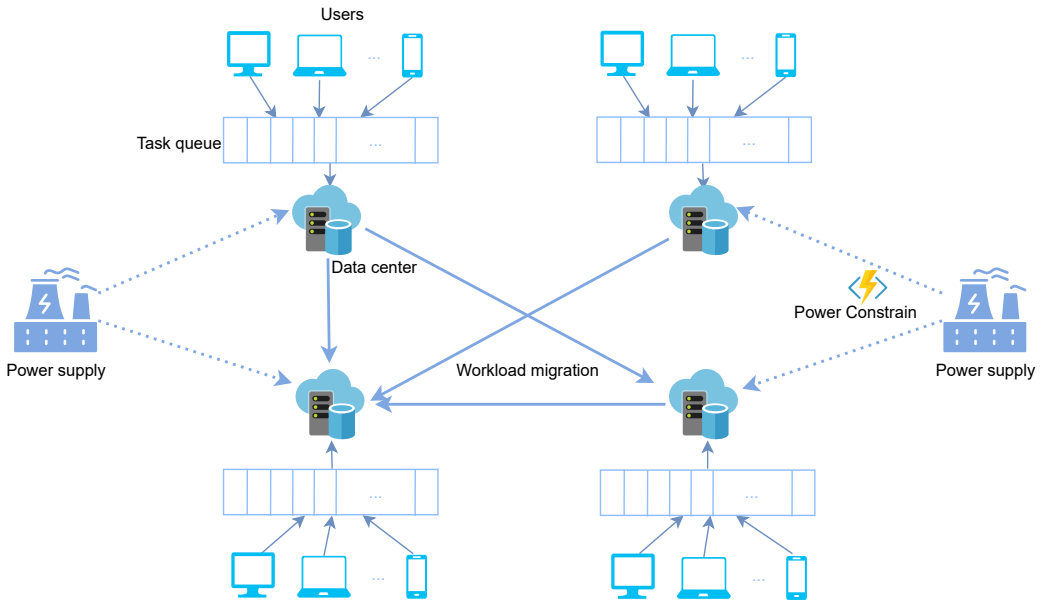


Fig. 1: Overall system model of distributed data centers

in real-world distributed data centers. We then develop a queuing model for the task service and formulate an optimization problem for workload migration, aiming to minimize both the service response time and the migration cost. We have conducted extensive simulations to evaluate our method. The results demonstrate that when the power supply becomes constrained, our method can significantly improve the overall response time with over 9% reduction with minimal cost.

The rest of the paper is organized as follows. Section II introduces the system model and problem formulation. Section III proposes our method to reschedule the workload under electrical load shedding. Section IV shows our simulation results and analysis. We conclude our work and provide some discussions in Section V.

II. System Model and Problem Formulation

In this section, we first introduce the overall system model and then formulate the workload migration problem with the consideration of constrained power supply.

A. System Model

Fig. 1 shows the overall system model of the distributed data centers considered in this paper. In particular, a number of distributed data centers are deployed in a region with different scales and different power supply conditions. Normally, users send tasks to their closest data center, and these tasks are usually scheduled and executed with the first-come-first-serve (FCFS) policy [9]. Each data center has a single-channel queue and all the servers in a data center are homogeneous. However, due to the instability of the power supply, tasks may suffer from considerable delays, where a long response time will severely affect the quality of service (QoS).

As each data center has different scales, and the server utilization will be heterogeneous under different power supply situations, we first model the relationship between the power supply and server utilization with the consideration of the power hungry cooling system. Then we present our model for task arrival, server service rate and workload migration.

1) Workload, Power and Heat:

To estimate the server utilization under power constraints, we first construct a physical model of data center. Data center is an energy-intensive system. There are many devices in this system which consume lots of energy. Meanwhile, these devices will generate a large amount of heat. So to protect these devices, a cooling system is essential in data center. Similar to [10], we construct a model with several intermediary data flows and relationships between sub-components in a data center, which is further illustrated in Fig. 2. The work in [11] shows that a server's power consumption has a linear relationship with its utilization as shown below:

$$P = P_{idle} + (P_{peak} - P_{idle})u \tag{1}$$

where P is the power consumption of the server and u is the percentage of its utilization. P_{idle} and P_{peak} are the idle and peak load power consumption of the server. We use Q to represent the heat generated by the server, which is also roughly the heat that should be removed by the cooling system. Then we have the following convective heat transfer equation:

$$Q = hA(T_{outside} - T_{inside}) \tag{2}$$

where A is the area of the object, h is the heat transfer coefficient, $T_{outside}$ and T_{inside} are the temperature of the environment inside and outside the data center. From [12], we know h is exponentially related to velocity (V) of the flow, and the exponent depends on the type of flow. The forced convection in the server is typically turbulent, which makes the exponent $\frac{4}{5}$. We thus have:

$$h \propto V^{\frac{4}{5}} \tag{3}$$

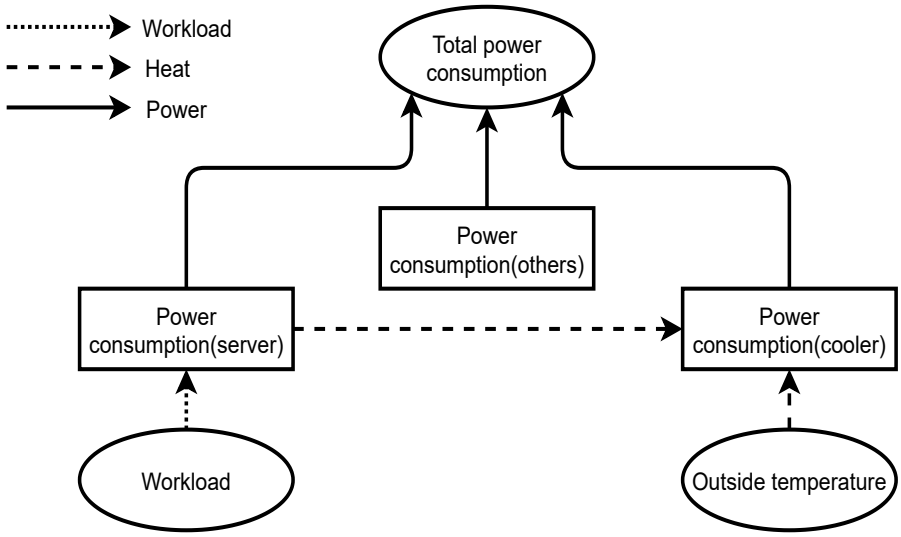


Fig. 2: Data flow in the model

To remove more heat generated by the server, the velocity of airflow should increase at an even greater rate. The fan laws [13] tell us that this increase is first order, but power consumption increases to the 3^{rd} power:

$$V \propto C^3 \tag{4}$$

where C is the power consumption of cooling system. Combine the Eq. (2), (3) and (4), we can get the relationship between power consumption C and generated heat Q:

$$C \propto Q^{\frac{5}{12}} \tag{5}$$

Next we consider the data center's server utilization under power constraints, and the power supply relationship among different components can be represented like:

$$P' + C' + O' \leq \alpha S \tag{6}$$

where α is the percentage of power constraints and S is the total power consumption. We can easily get the the total server power consumption P' from Eq. (1), and the total power consumption of other components O' will not change under power constraints. Then the only challenge is to get the total power consumption of the cooling system C'. From the Eq. (5) we know that the power consumption of the cooling system has a relationship with the heat. Based on this, we can estimate the C' if we can know the predicted heat under a specific situation. If the total amount of generated heat is Q_{total} and Q'_{total} with and without power constraints, respectively, where we use β to denote the ratio between them, then we have:

$$Q'_{total} = \beta Q_{total} \tag{7}$$

Combine with Eq. (5), we can get the relationship between C'_{total} and C_{total}:

$$C'_{total} = \beta^{\frac{5}{12}} C_{total} \tag{8}$$

It is easy to get C_{total} and β in the general situation. So we can easily get C'_{total} under power constraints and calculate the server utilization based on Eq. (1) and (6).

2) Task-Arriving and Server-Service Rates:

Let λ_i denote the task-arriving rate in data center i, and μ_i denote the mean rate at which tasks are finished by a single server in the data center i, which is directly proportional to server utilization. The number of servers in the data center i is represented by s_i.

Let W_i denote the average waiting time of tasks in data center i which is determined by the task-arriving rate λ_i, server-service rate μ_i, and the number of servers s_i. Moreover, if μ_i and s_i do not change, the average waiting time should decrease when λ_i decreases. That is the reason why power constraints can affect the response time of tasks. In next section, we will propose an approach to approximate this average waiting time.

3) Workload Migration:

Our model assumes that all data centers can connect to each other and migrate some workloads to others. When the workload migration happens between two data centers, both of their task-arriving rates will change accordingly. In particular, if data center i migrates some workload to another data center j, then its task-arriving rate will decrease from λ_i to $(\lambda_i - \Delta\lambda_{ij})$. The task-arriving rate of data center j will then increase from λ_j to $(\lambda_j + \Delta\lambda_{ij})$.

We use U_i and L_i to denote the maximum upload and download capacity allowed for workload migration in data center i. The migration will also incur some cost, and we use c_{ij} to denote the cost of migrating one unit of workload from data center i to j.

B. Problem Formulation

If there are some power outages or constraints in some distributed data centers, the original task schedule will suffer from low QoS. Tasks in some data centers will have too long response time while others may still have extra computing resources. To obtain better QoS and the lower response time, the original workloads may need to be rescheduled with some being migrated to other distributed data centers. Let X_{ij}

denote the volume of workload migrated from data center i to j. If X_{ij} is negative, it means data center i receives workload from j. Let D denote the number of distributed data centers considered in a region, and then we can get the new task-arriving rate ${\lambda_i}'$ in data center i as below:

$$ {\lambda_i}' = \lambda_i - \sum_{j=1}^{D} X_{ij} \tag{9} $$

Our objective is to minimize the response time of all the tasks and minimize the migration cost simultaneously. Let ${W_i}'$ denote the new average waiting time after workload migration. The problem can be formulated as

$$ \textbf{Min:} Response = \sum_{i=1}^{D} {\lambda_i}' {W_i}' \tag{10} $$

$$ \textbf{Min:} Cost = \sum_{i=1}^{D}\sum_{j=1}^{D} max\{X_{ij}, 0\} c_{ij} \tag{11} $$

s.t.

$$ X_{ij} = -X_{ji}, \forall i, j \tag{12} $$

$$ \frac{{\lambda_i}'}{s_i \mu_i} < 1, \forall i \tag{13} $$

$$ -L_i \leq \sum_{j=1}^{D} X_{ij} \leq U_i, \forall i \tag{14} $$

Here, constraint (13) ensures the model is stable, which means the queue of tasks will not increase infinitely. Constraint (14) ensures the migration limitation is not surpassed.

III. Solutions

In this section, we first use queuing theory to approximate the tasks' average response time. Then we transform the optimization problem into two subproblems and design an algorithm to solve them efficiently.

A. Approximation of Average Waiting Time

Assume the task-arriving time and server-service time are both exponentially distributed [8] [14]. All the tasks in a data center can be modeled as an M/M/c queue. Then we can get the utilization of the servers in data center i as below:

$$ \rho_i = \frac{\lambda_i}{s_i \mu_i} \tag{15} $$

where ρ_i also represents the probability that the server is busy or the proportion of time that the server is busy. The probability that no task in queue is

$$ P_{0_i} = [\sum_{n=0}^{s_i - 1} \frac{(s_i \rho_i)^n}{n!} + \frac{(s_i \rho_i)^s_i}{s_i!(1-\rho_i)}]^{-1} \tag{16} $$

Therefore, the average waiting time of tasks in data center i can be calculated as

$$ W_i = \frac{P_{0_i} \rho_i \lambda_i^{s_i - 1} + s_i!(1-\rho_i)^2 \mu_i^{s_i - 1}}{s_i!(1-\rho_i)^2 \mu_i^{s_i}} \tag{17} $$

B. Problem Transformation

Recall that our objectives are minimizing both the response time Eq. (10) and overall migration cost Eq. (11). This optimization problem is actually a multi-objective programming problem, where the migration variable X_{ij} must be in a specific range with some constraints, making the problem NP-hard. Here we transform it into two subproblems and design an algorithm to solve it efficiently.

The two objectives in our formulated problem are contradictory and thus cannot be optimized simultaneously. Here we first optimize the response time, because for the providers, guaranteeing the QoS of users is usually the most important. As overall response time is only determined by the new task-arriving rate λ' and does not depend on the process to migrate the workload, all the constraints can be transformed into new forms without variable X_{ij}. Let ${\lambda_i}'$ become the new variable instead of X_{ij}, with the same objective (10), the new constraints can be formulated as:

$$ \sum_{i=1}^{D} {\lambda_i}' = \sum_{i=1}^{D} \lambda_i \tag{18} $$

$$ \lambda_i - U_i \leq {\lambda_i}' \leq \lambda_i + L_i, \forall i \tag{19} $$

$$ {\lambda_i}' < s_i \mu_i, \forall i \tag{20} $$

To solve this, we start from the barrier method [15]. We first introduce a punish function $I(u) = -(1/t)log(-u)$ to remove the inequality constraints, where $t > 0$ is a parameter that sets the accuracy of the approximation. We will replace u in the punish function with Eq. (19) and Eq. (20) and add them to the objective. The basic idea here is that the punish function will be close to zero when constraints are satisfied and very large when constraints are not satisfied. So the solution of the new objective will be closer and closer to the origin problem when t increases. Now the problems becomes

$$ \begin{aligned} \textbf{Min:} f(\vec{\lambda}') = \sum_{i=1}^{D} \{ & t{\lambda_i}' W_i - log({\lambda_i}' - \lambda_i + U_i) - \\ & log(\lambda_i + L_i - {\lambda_i}') - log(s_i \mu_i - {\lambda_i}') \} \end{aligned} \tag{21} $$

s.t. Eq. (18).

We have transformed the original problem to an equality constrained minimization problem, and we now develop a Newton's method based algorithm [15] to solve it. Unlike classical Newton's method, the start point must be feasible for this problem, and the calculation of the Newton step must consider the equality constraint. Here we can use the original task-arriving rate $\vec{\lambda} = \{\lambda_1, \lambda_2, ..., \lambda_D\}$ as the start point since it must be feasible. At this feasible point $\vec{\lambda}$, we replace the objective with its second-order Taylor approximation near $\vec{\lambda}$, the objective becomes

$$ \textbf{Min:} \hat{f}(\vec{\lambda} + \Delta\vec{\lambda}) = f(\vec{\lambda}) + f'(\vec{\lambda})\Delta\vec{\lambda} + \frac{1}{2} f''(\vec{\lambda}) \Delta\vec{\lambda}^2 \tag{22} $$

TABLE I: Server configuration

Item	Quantity	Configuration
Processor	4	Intel 8158 3.0 GHz 150W 12C/24.75MB Cache/DDR4 2666MHz
Memory	48	128GB DDR4-2666-MHz TSV-RDIMM/PC4-23100/octal rank/x4/1.2v
Power Supply	4	1600W PSU1
Dedicated Storage Controller	1	RAID-Cisco 12G SAS Modular Controller 4GB FBWC
Storage	16	480GB 6Gb SATA 2.5-inch SSD Enterprise Value
Adapter	12	GPU-NVIDIA V100 32GB

where $\Delta\vec{\lambda}$ denotes the Newton step, and it can be calculated by the KKT system for the equality constrained quadratic optimization problem [15]. The optimality conditions are

$$\sum_{i=1}^{D} \Delta\lambda_i = 0, f''(\vec{\lambda})\Delta\vec{\lambda} + f'(\vec{\lambda}) + \sum_{i=1}^{D} \hat{\lambda}_i = 0 \tag{23}$$

where $\hat{\lambda}$ is the associated optimal dual variable for the quadratic problem. After we get the $\Delta\vec{\lambda}$, we can use $\vec{\lambda} + \Delta\vec{\lambda}$ to approximate the optimal solution for the current t. More details can be found in [16].

IV. Performance Evaluation

In this section, we will introduce our simulation setup and present the performance evaluation of our solution.

A. Simulation Setup

We evaluate our solution with extensive simulations, which include up to 100 distributed data centers in total. The number of servers in a single distributed data center varies from 2 to 20. The average task-arriving and server-service rates are generated by uniform distribution, where the former vary from 10 to 15 MB/s and the latter from 15 to 20 MB/s. Here the server-service rate is higher so as to make sure that the origin task queue in each data center is stable. The upload and download limits U and L vary from 5 to 10 MB/s, and the migration cost c varies from 1 to 5. Finally, each data center's power constraints are generated randomly, varying from 70% to 100%. We use Cisco's UCS Power Calculator [17] to generate the server configurations. Taking a Cisco UCS C480 M5 server as an example, the exact configuration parameters are illustrated in Table I.

To evaluate the performance of our method, we compare it with the following baselines:

- **Origin:** Tasks are processed based on the FCFS policy without any migration.
- **None-power constraint:** This one uses the same algorithm as our method to schedule all the tasks, but assuming the power supply is full all the time.
- **Random:** This one migrates the workload in a random way and does not consider the migration cost.

B. Evaluation Results

We first evaluate the response time in different situations. Fig. 3 (a) shows how the response time changes with different number of data centers. It is easy to see that our method can always get lower response time with more than 9% comparing to the origin approach. Then we check the influence of power constraints. As illustrated in Fig. 3 (b), we change the average power constraints from 80% to 98%, and the results show that our method is more effective when the power supply percentage is low. Indeed when all the data centers have a nearly full power supply, our method can still benefit the response time with the workload migration, performing the same as the non-power constrain one and much lower than the origin approach.

Next, we evaluate the migration costs among different methods, which are presented in Fig. 4. Fig. 4 (a) shows that the migration cost will increase significantly when the number of data centers increases. Fig. 4 (b) shows that the percentage power constraints do not have a significant correlation with the migration cost. However, all the results show that our method saves lots of costs than the random migration method.

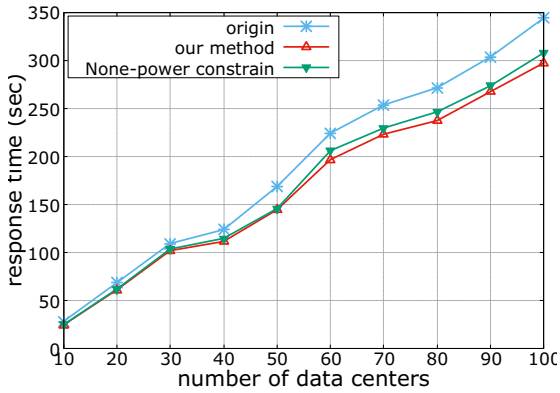
In general, the evaluation results show that our proposed method can minimize the response time with a relatively low migration cost. Different factors have different effects on our objective, but all the results show that power constraints are essential for workload scheduling, and our method can significantly improve the effectiveness of distributed data centers under different situations.

V. Conclusion

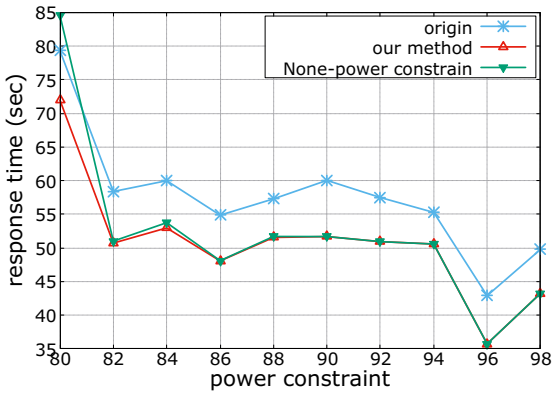
In this paper, we concentrate on enhancing the QoS of distributed DCs with workload migration under load shedding. We first construct a physical model of a data center to investigate the effect of power constraints. Then we use the queueing theory to approximate the response time and regard this as an objective in the problem formulation. To get the minimum response time, we need to migrate some of the workloads to other distributed DCs, so another objective is to minimize migration costs. Since this multi-objective optimization problem is NP-hard, we propose a method that combines the barrier method and modified Newton's method to solve it. Our simulation results show that our algorithms can significantly improve the performance of distributed data centers under load shedding.

Acknowledgments

This research is supported by a Canada NSERC Discovery Grant. Fangxin Wang's work was supported in part by the Key Area R&D Program of Guangdong Province with grant No. 2018B030338001.

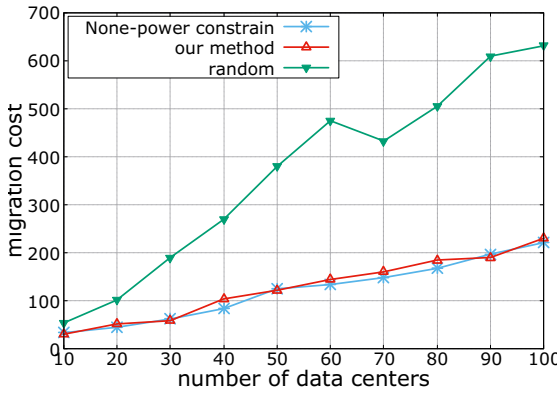


(a) Response time with different number of data centers

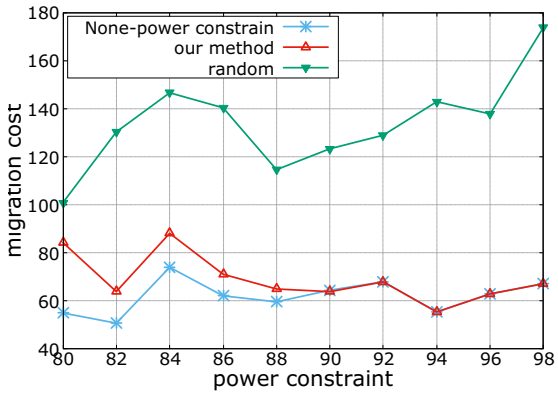


(b) Response time with different power constraints

Fig. 3: Comparison of overall response time



(a) Migration cost with different number of data centers



(b) Migration cost with different power constraints

Fig. 4: Comparison of overall migration cost

REFERENCES

[1] C. V. N. Index, "Forecast and methodology, 2015–2020," *White paper*, pp. 1–41, 2016.

[2] W. Jiang, Z. Jia, S. Feng, F. Liu, and H. Jin, "Fine-grained warm water cooling for improving datacenter economy," in *Proc. of ACM/IEEE 46th Annual International Symposium on Computer Architecture (ISCA)*, 2019.

[3] F. Wang, X. Fan, F. Wang, and J. Liu, "Backup battery analysis and allocation against power outage for cellular base stations," *IEEE Transactions on Mobile Computing*, vol. 18, no. 3, pp. 520–533, 2018.

[4] Bloomberg. China limits power supplies as demand surges on cold weather. [Online]. Available: https://www.bnnbloomberg.ca/china-limits-power-supplies-as-demand-surges-on-cold-weather-1.1539909

[5] R. S. Madeline Holcombe. A final round of freezing temperatures strikes texas as it struggles to recover from winter storms. [Online]. Available: https://www.cnn.com/2021/02/20/weather/texas-winter-storm-saturday/index.html

[6] J. McKane. How load-shedding cripples south african data centres. [Online]. Available: https://mybroadband.co.za/news/energy/288586-how-load-shedding-cripples-south-african-data-centres.html

[7] J. Guo, Z. Chang, S. Wang, H. Ding, Y. Feng, L. Mao, and Y. Bao, "Who limits the resource efficiency of my datacenter: An analysis of alibaba datacenter traces," in *Proc. of IEEE/ACM 27th International Symposium on Quality of Service (IWQoS)*, 2019.

[8] H. Yuan, H. Liu, J. Bi, and M. Zhou, "Revenue and energy cost-optimized biobjective task scheduling for green cloud data centers," *IEEE Transactions on Automation Science and Engineering*, pp. 1–14, 2020.

[9] A. Ghassami and N. Kiyavash, "A covert queueing channel in fcfs schedulers," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 6, pp. 1551–1563, 2018.

[10] S. Pelley, D. Meisner, T. F. Wenisch, and J. W. VanGilder, "Understanding and abstracting total data center power," in *Workshop on Energy-Efficient Design*, vol. 11, 2009, pp. 1–6.

[11] X. Fan, W.-D. Weber, and L. A. Barroso, "Power provisioning for a warehouse-sized computer," *ACM SIGARCH computer architecture news*, vol. 35, no. 2, pp. 13–23, 2007.

[12] D. P. DeWitt, *Fundamentals of heat and mass transfer*. John Wiley & Sons, 2002.

[13] R. Jorgensen, "Fan engineering: An engineer's handbook on fans and their applications; howden buffalo," 1999.

[14] H. Yuan, J. Bi, M. Zhou, and A. C. Ammari, "Time-aware multi-application task scheduling with guaranteed delay constraints in green data center," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1138–1151, 2017.

[15] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[16] L. Shen, F. Wang, F. Wang, and J. Liu, "Workload migration across distributed data centers under electrical load shedding," Technical Report 2021-05-(001), School of Computing Science, Simon Fraser University, BC, Canada, May 2021.

[17] Cisco. Ucs power calculator. [Online]. Available: https://ucspowercalc.cloudapps.cisco.com/public/