Reinforcement Learning for Rack-Level Cooling

Yanduo Duan[1], Jianxiong Wan[1,2(✉)], Jie Zhou[1], Gaoxiang Cong[1], Zeeshan Rasheed[1], and Tianyang Hua[1]

[1] Inner Mongolia University of Technology, Inner Mongolia, People's Republic of China
jxwan@imut.edu.cn

[2] Inner Mongolia Autonomous Region Engineering and Technology Research Center of Big Data Based Software Service, Hohhot, China

Abstract. In recent years, we have seen a rapid growth in big data and cloud computing industry, because of the rapid development in Internet technology. That's why the number and scale of data center have increased rapidly. A data center is a warehouse-level IT facility that hosts many servers. Because of the uneven heat production and heat dissipation of the servers in a rack, the hot spots emerges. In order to maintain the CPU temperature hence the computing performance, the server cooling is very critical. A common solution is to increase the speed of the Computer Room Air Handler (CRAH) blower and increase the flow of cold air. Nevertheless, this solution can only partially address the issue and raise the cooling energy consumption.

In this paper, we study how to mitigate rack hot spots without significantly increasing the power of air conditioning system. We propose the Active Ventilation Tiles (AVTs), i.e., ordinary ventilation tiles with attached fans, to enhance the local cold air delivery and improve the cooling performance. In particular, we propose an AVT control algorithm adapted from the Reinforcement Learning techniques to tackle the complex data center environment and thermo dynamic process. The reinforcement learning algorithm adjusts the temperature distribution of the rack by controlling the fan speed installed on the ventilation tile, and guides the fan speed according to the feedback temperature to mitigate hot spots. Due to the slow learning speed of the traditional Tabular-Q-Learning algorithm, we integrate the Tabular-Q-Learning algorithm with the Dyna architecture to accelerate the learning speed and improve the algorithm performance in the early stage. Experimental results reveal that Tabular-Q-learning based on Dyna has better performance.

Keywords: Reinforcement learning · Data center · Hotspot

1 Introduction

A data center is a server hosting facility that provides users with more storage and computing resources. In recent years we has seen the rapid development in

© ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2020
Published by Springer Nature Switzerland AG 2020. All Rights Reserved
W. Li and D. Tang (Eds.): MOBILWARE 2020, LNICST 331, pp. 167–173, 2020.
https://doi.org/10.1007/978-3-030-62205-3_16

Internet technology and 5G communication technology. That's why number and scale of data center has increased rapidly.

However, a reality often faced by the data center operators is that the local server overheating, or hot spot, is commonplace in practice. Usually the main causes of hot spots are server load fluctuations and cold air supply imbalance[2]. To solve this problem, someone used AVT. Ventilation tiles in the raised-floor data centers can be classified as Passive Ventilation Tiles (PVTs) and Active Ventilation Tiles (AVTs), AVTs have attached fans and PVTs where the tile flow completely determine by the pressure differential between above- and underfloor spaces [1]. AVTs are more flexible since fans can draw the cold air actively to cold aisles even if the under-floor pressure is ill-distributed [7]. In previous paper, researcher focused on measurement or simulation based performance modeling and evaluation. The impact of various factor, such as airflow angle, tile position, containment structure CRAC blower speed, on the tile flow was investigated in [3–5]. The general conclusion of these articles is that AVTs can improve local cooling delivery. No one has proposed a relevant study on the control problem of AVTs under actual working conditions.

In this paper, we propose an AVTs control algorithm based on reinforcement learning. The difficulty with AVTs control is the diversity of different data center environments. Due to the diversity of the environment and the lack of a complete grasp of the environment, reinforcement learning is more appropriate solution to solve such problems. We use Tabular-Q-Learning and Tabular-Q-Learning Based On Dyna algorithms to solve the control problem of AVTs. Because the algorithm is completely online learning. It can learn the external environment through limited environmental exploration, without considering the establishment of complex airflow and heat exchange model, thus improving the universality of AVTs and control algorithm.

2 Markov Decision Process Formulation

Reinforcement learning is a computational approach to understanding and automating goal-directed learning and decision making. It is distinguished from other computational approaches by its emphasis on learning by an agent from direct interaction with its environment, without requiring exemplary supervision or complete models of the environment. Reinforcement learning uses the formal framework of Markov decision processes to define the interaction between a learning agent and its environment in terms of states, actions, and rewards [6]. Our main target is control the fan speed of AVT and provide the average temperature distribution to the rack. Tabular based approach requires that the state and behavior of the MDP model should have discrete value and space of the state and behavior should be small. Because all values in this method are stored in the tabular (Q tabular), Q tabular is the result of the algorithm update and learning. The advantage of this method is that the optimal solution of the problem learned by agent. Following is the tabular approach of MDP modeling:

- System states. We discretize the duty cycle of PWM and divide it by 25% equidistance, that is, the single state dimension of the system is one, the system state space $\mathcal{S} = \{0\%, 25\%, 50\%, 75\%, 100\%\}$.
- Actions. Action behavior defined the different condition of fan speed, which is increase, decrease and static state of fan speed. The behavior space of the system $\mathcal{A} = \{inc, dec, nop\}$, for ease of calculation $\mathcal{A} = \{1, -1, 0\}$.
- Reward. The main purpose of this paper is provide the average temperature to the rack and reduced the fan energy consumption. Therefore, the immediate reward is divided into two parts:1) The temperature part of the reward: $R_{t,T} = \frac{\sum_{i \in t}(T_{t,i} - \overline{T_t})^2}{|\mathcal{I}|}$ Where, $T_{t,i}$ is the temperature of the ith sensor on the front panel of the rack at time t, $\mathcal{I}$ is the temperature and humidity sensor set, and $|\mathcal{I}|$ is the total number of sensors. $\overline{T_t}$ is the frame reference temperature at time t. It is obvious that the smaller $R_{t,T}$ is when the sensor temperature is closer to the reference temperature. That is, the more uniform the temperature distribution of the rack. On the premise of 1), reduce fan energy consumption. Energy consumption and fan speed is 3 power relationship, so energy consumption part of the reward $R_{t,E} = \left(\frac{s_t}{S_{ref}}\right)^3$,among them $s_t \in \mathcal{S}$,S_{ref} is the reference value that keeps $R_{t,T}$ and $R_{t,E}$ of the same order. Obviously, the smaller the fan speed, the lower the energy consumption. Therefore, the immediate reward of the system is set as:

$$R_t = -(1 - \omega)R_{t,T} - \omega R_{t,E}. \tag{1}$$

Where ω is to determine which part of $R_{t,E}$ or $R_{t,T}$ contributes more weight to R_t. Therefore, all of them have negative values, and maximizing R_t (tending to 0) is the optimization direction of this topic. Finally, the AVT control problem can be defined as the following decision problem:

$$\max_{a_t} \sum_{t=0}^{\infty} \gamma^k R_t, a_t \in \mathcal{A}, \forall t \in \{0, 1, ..., \infty\}. \tag{2}$$

3 Algorithm Design

3.1 Tabular-Q-learning

In order to solve the problem of Eq. (2), in this paper we used the Q function to quantify the advantages of behavior at under state s_t:

$$Q(s, a) = E\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \middle| s = s_t, a = a_t\right]. \tag{3}$$

There is a unique optimal Q function satisfying Bellman equation:

$$Q^*(s_t, a_t) = \max_{s_{t+k}, \mathcal{A}} E\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \middle| s_t, a_t\right] = E\left[R_{t+1} + \gamma \times \max_{a_{t+1} \in \mathcal{A}} Q^*(s_t, a)\right]. \tag{4}$$

According to formula (4), the optimal behavior can be selected as follows:

$$a_t = \arg\max Q^*(s_t, a) \tag{5}$$

In Q-learning, the update of Q function is as follows:

$$\begin{cases} Q_{t+1,target} = R_{t+1} + \gamma \max\limits_{a \in \mathcal{A}} Q(s_{t+1}, a). \\ Q(s_{t+1}, a) = Q(s_{t+1}, a) + \alpha \times (Q_{t+1,target} - Q(s_t, a_t)). \end{cases} \tag{6}$$

where, $Q_{t+1,target}$ is the Q sample received at time t+1, s_{t+1} is the system state at time t+1, and $\alpha \in [0, 1]$ is the learning rate. Tabular q-learning algorithm, after the initialization of the Q tabular, carries out Tabular update learning according to the above formula.

Algorithm 1. A basic Tabular-Q-learning algorithm for solving the AVT control problem.

1: Define
   t = Time index;
   s_t = State at time t;
   a_t = Action at time t;
   q_table = Matrix with size:$|S| * |A|$;
   ε = Exploration factor;
   ω = Reward the weight;
2: t=0,$\varepsilon = [0, 0.99]$,$\omega = 0.3$;
3: Randomly initialize q_table with 0 except that q_table [0, -1] = min and q_table [4, 1] = min(min is a very small number);
4: Initialize S;
5: **loop**

$$a_t = \begin{cases} \arg\max Q(s_t, a), \textit{with probability } \varepsilon. \\ \textit{a random action from } \mathcal{A}, \textit{ with probability } 1 - \varepsilon. \end{cases} \tag{7}$$

6: Observe the next state s_{t+1} and the temperature distribution of the rack inlet,compute reward R_{t+1}(see Eq.(1));
7: Update the $q\_table[s_t, a_t]$(see Eq.(6));
8: **end loop**

Q-Learning requires a Q tabular. If Q tabular has to many states, then required large amount of space for storage and need large amount of time to search data. Q-Learning has the problem of overestimation, because Q-Learning uses the action corresponding to the optimal value at the next moment when updating Q function, which cause to the "overestimation". In order to solve the slow convergence rate of Q-Learning, we proposed Q-Learning integrated with Dyna framework.

3.2 Tabular-Q-learning Based on Dyna

Dyna is a generic term for a class of algorithmic frameworks that integrate model-based reinforcement learning with non-model-based reinforcement learning. Dyna can learn both from models and from the experience of environmental interactions.

Compared with Q-Learning alone, Tabular-Q-learning Based On Dyna adds an L-size experience replay buffer B to store the learning samples as the model of interaction with the environment. In addition, a thread was opened in the program. After one step of algorithm training, M samples were randomly selected from the experience replay buffer for model training.

Tabular-Q-learning Based On Dyna Algorithm adds the experience replay buffer storage transition (s,a,r_{t+1},s_{t+1}) between 5–6 lines of the above algorithm, and selects M -size samples for model training.

4 Result Analysis

This experiment we did in a real data center. We collect the temperature by installing sensors on the rack and air conditioning, and then transferred the data to the computer for intensive learning algorithm training. In algorithm Tabular-Q-learning, $\varepsilon = [0, 0.99]$, $\omega = 0.3$, $\alpha = 0.01$; In algorithm Tabular-Q-learning Based On Dyna, $L = 5000$, $M = 100$.

The total reward based on Q-Leaning with Dyna is closer to 0 than the total reward based on Q-Leaning. As can be seen from the figure. The average of algorithm Tabular-Q-learning Based On Dyna total rewards is higher than algorithm Tabular-Q-learning, (see Fig. 1).

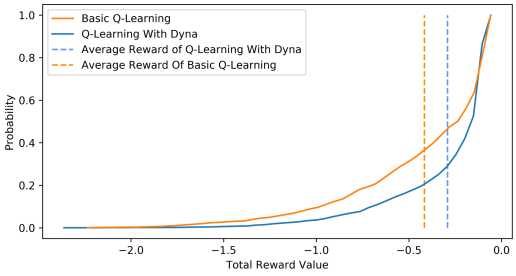

Fig. 1. Total reward (CDF)

The average Q value of algorithm Tabular-Q-learning Based On Dyna tends to converge in 40 steps, and fluctuates slowly in 400–500 steps. After 500 steps, the average value of Q Tabular becomes stable, and the average value of Q Tabular converges in the range of [−0.2, −0.3]. However, in Tabular-Q-learning the time steps of the average Q Tabular value from 0 to 1750 was continuously declined. The average value of Q Tabular is down from –1.1, and there is no

convergence. The convergence speed of Tabular-Q-learning Based On Dyna is much faster than tabular Q learning, and the average value of Tabular-Q-learning Based On Dyna is larger, which proves that the algorithm learns a great reward and its performance is better, (see Fig. 2).

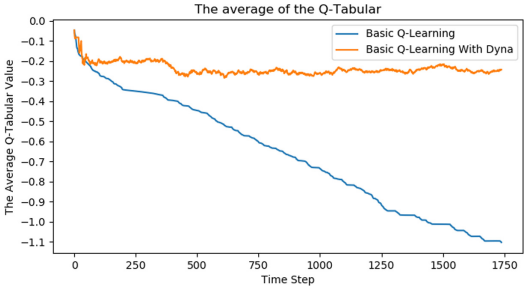

Fig. 2. The average of the Q-tabular

5 Conclusion

In this paper, we describe the rack hot spot issues facing today's data centers, as well as the shortcomings of solving these problems in the past. Due to the complexity of the data center environment, we propose a solution based on the reinforcement learning algorithm AVT control. We introduce the establishment of Markov decision process, and introduce reinforcement learning algorithm Tabular-Q-Learning and Tabular-Q-Learning Based on Dyna. By comparison, For algorithm Tabular-Q-learning Based On Dyna, it can accelerate the convergence of Q tabular. It is found that algorithm Tabular-Q-learning Based On Dyna has better performance in dealing with Tabular problems. Algorithm Tabular-Q-Learning converges at least four times faster than algorithm Tabular-Q-Learnig Based on Dyna.

Acknowledgements. This work was funded in part by the National Natural Science Foundation of China (NSFC) under Grants. 61862048, 61762070, and 61962045, Inner Mongolia Key Technological Development Program (2019ZD015), Key Scientific and Technological Research Program of Inner Mongolia Autonomous Region (2019GG273), and Inner Mongolia Autonomous Region Special Program for Engineering Application of Scientific and Technical Payoffs (2020CG0073).

References

1. Khalili, S., Tradat, M.I., Nemati, K., Seymour, M., Sammakia, B.: Impact of tile design on the thermal performance of open and enclosed aisles. J. Electron. Packag. **140**(1), 010907 (2018)
2. Patankar, S.V.: Airflow and cooling in a data center. J. Heat Transf. **132**(7), 073001 (2010)
3. Song, Z.: Numerical cooling performance evaluation of fan-assisted perforations in a raised-floor data center. Int. J. Heat Mass Transf. **95**, 833–842 (2016)
4. Song, Z.: Thermal performance of a contained data center with fan-assisted perforations. Appl. Thermal Eng. **102**, 1175–1184 (2016)
5. Song, Z.: Studying the fan-assisted cooling using the Taguchi approach in open and closed data centers. Int. J. Heat Mass Transf. **111**, 593–601 (2017)
6. Sutton, R.S., Barto, A.G., et al.: Introduction to Reinforcement Learning, vol. 2. MIT press, Cambridge (1998)
7. Wan, J., Gui, X., Kasahara, S., Zhang, Y., Zhang, R.: Air flow measurement and management for improving cooling and energy efficiency in raised-floor data centers: a survey. IEEE Access **6**, 48867–48901 (2018)