This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

Workload Scheduling for Massive Storage Systems with Arbitrary Renewable Supply

Daping Li, Xiaoyang Qu, Jiguang Wan*, Jun Wang, *Senior Member, IEEE,*
Yang Xia, Xiaozhao Zhuang, and Changsheng Xie

Abstract—As datacenters grow in scale, increasing energy costs and carbon emissions have led data centers to seek renewable energy, such as wind and solar energy. However, tackling the challenges associated with the intermittency and variability of renewable energy is difficult. This paper proposes a scheme called GreenMatch, which deploys an SSD cache to match green energy supplies with a time-shifting workload schedule while maintaining low latency for online data-intensive services. With the SSD cache, the process for a latency-sensitive request to access a disk is divided into two stages: a low-energy/low-latency online stage and a high-energy/high-latency off-line stage. As the process in the latter stage is off-line, it offers opportunities for time-shifting workload scheduling in response to variations of green energy supplies. We also allocate an HDD cache to guarantee data availability when renewable energy is inadequate. Furthermore, we design a novel replacement policy called Inactive P-disk First for the HDD cache to avoid inactive disk accesses. The experimental results show that GreenMatch can make full use of renewable energy while minimizing the negative impacts of intermittency and variability on performance and availability.

Index Terms—Renewable energy; Storage system; SSD cache.

1 Introduction

With the growth of online cloud-based services, the scale of datacenters is rapidly expanding, which makes datacenters consume much more energy than ever before. The ever-growing grid energy consumption increases datacenters energy costs and indirectly grows carbon emissions, forcing datacenters to employ the cheaper and environmental-friendly renewable energy, such as solar and wind energy[3]. Many IT companies, therefore, have built (or will build) their own onsite solar/wind power farms to reach renewable energy datacenters[3]. For example, Apple has built the largest corporation-owned solar farm for its data center in North Carolina. Datacenters powered by 100% renewable energy will greatly help datacenters energy saving. However, not all the energy-hungry components, especially storage systems, are ready for it in a short time.

In datacenters, storage systems require uninterrupted energy supply to work for computing components, which results in as high as 25%[1] of the total energy consumed by storage systems. When the workload gets low, the energy proportion of disk-based storage systems becomes even higher[2]. This is because disks are the least energy-proportional among all datacenter major components. Even though renewable energy provides a considerable amount of energy for energy-hungry datacenters, the contradiction between stable energy supply for storage systems and the intermittent nature of renewable energy make it difficult for datacenters to adopt renewable energy as the only energy source in the short term. Hence, the combination of renewable and traditional energy seems to have become an effective way to solve the energy consumption issues of datacenters.

Many efforts try to tame storage systems to work better with green energy. Based on their trade-offs between storage performance and the energy efficiency, there are two kinds of power management schemes: workload-driven and supply-following. (1) Workload-driven power management schemes involve changing the number of active nodes in response to workload variations [5][4][6]. This kind of schemes try to maintain the best storage performance, while benefits less from green energy. (2) Supply-aware power management schemes involve changing the number of active storage nodes to be proportional to the green power supply [12][11][15][16][1]. These schemes are more capable of taking full advantage of renewable energy, but intermittence and variability of the green energy supply will result in unintended storage nodes unavailability, which in turn will result in high latency in the case of low green power supplies and heavy workloads. This paper devotes to optimizing supply-aware power management schemes, and the ultimate goal is to power datacenter storage systems with 100% green energy.

Existing supply-aware workload management policies work well on latency-insensitive workloads. They schedule workloads to match renewable energy supplies by deferring delay-tolerant workloads [15][16], or migrating loads between nodes in a cluster [11]. Figure 1 shows the variation of a real-world workload trace [8] and a green power supply trace [7] across 168 hours. For latency-sensitive

- *Daping Li, Xiaoyang Qu, Jiguang Wan*, Xiaozhao Zhuang, Changsheng Xie are with National Laboratory for Optoelectronics, Huazhong University of Science and Technology, 430074, Wuhan, Hubei, P.R. China. E-mail: daping_li@outlook.com quxiaoy@gmail.com 1074311818@qq.com cs_xie@mail.hust.edu.cn*
  ** Corresponding author: jgwan@mail.hust.edu.cn*
- *J. Wang is with School of Electrical Engineering and Computer Science, University of Central Florida, Orlando, FL32816, USA. E-mail: jwang@eecs.ucf.edu*
- *Y.Xia is with School of Computer Science and Engineering, The Ohio State University, Columbus, OH43210. E-mail:xia.425@osu.edu*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

workloads, it is difficult for disk-based storage to schedule workloads in response to variations of green energy supplies without a latency penalty. However, many applications, such as web services, online analysis, and transaction processing are latency-sensitive. Compared with faster computation and network technologies, disk-based storage becomes further bottleneck-prone. Straightforwardly using high performance devices, such as SSDs, as the cache [33][34][36][35][37] will help boost performance, but how to effectively integrate SSDs into green energy powered storage systems remains to be improved.

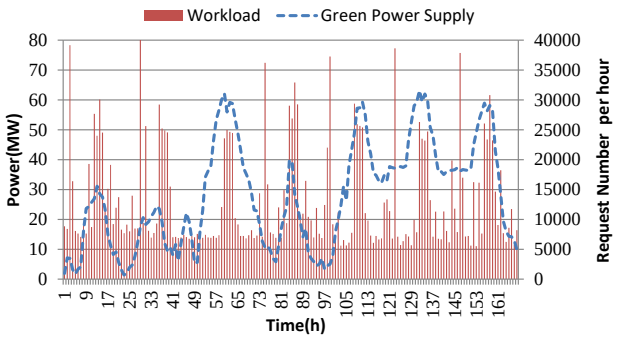


Fig. 1. The variation of a workload and a green power supply

Based on forementioned concerns, this paper aims to obtain a supply-aware workload matching without a latency penalty for critical workloads. We propose a new scheme for storage systems called GreenMatch, which leverages green energy efficiently without sacrificing performance and availability. We allocate several SSDs as an SSD cache for a storage system with a HDD storage pool. And HDDs in the storage pool are named **P-HDDs** or **P-disks**. In this way, the process for a non-deferrable request is split into two stages: a low-latency/low-energy on-line stage, and a high latency/high-energy off-line stage. The online stage can offer lower latency response, while the off-line stage, which consumes a large percentage of the total energy, can provide the opportunity to match the variations of a green supply. The response time is decreased due to the high performance of SSDs. And a supply-following workload matching scheme is achieved by deferring writes to P-disks and prefetching data from P-disks with a cache solution. To guarantee data availability, we deploy an HDD cache as a supplement of the SSD cache with a novel replacement algorithm called Inactive P-disk First, which aims to reduce the latency penalty caused by inactive P-disk access. The experimental results show that our GreenMatch can make full use of green energy while lowering the latency for sensitive-workloads.

In summary, we make the following contributions:

- In order to match the renewable energy supply with the energy consumption in a storage system, the process for time-sensitive requests to access P-HDDs is split into two stages: an online stage processed by SSDs, and an off-line stage processed by P-HDDs. While the online stage provides low latency with low energy consumption, the off-line stage achieves renewable energy aware workload matching.
- We deployed an HDD cache to guarantee data availability when green power is low. As the HDD cache mainly stores warm data, we designed an algorithm called Inactive P-disk First to reduce inactive disk accesses. We also designed a prefetch algorithm for SSDs that considers the lifetime and capacity of SSDs.
- We implemented GreenMatch based on a block-level distributed storage called Sheepdog [9]. Compared with a grid-only SSD cache-based storage system without power management, GreenMatch reduces the response time and achieves up to 97.5% renewable energy utilization, and up to 97.54% grid energy reduction. Meanwhile, GreenMatch enables graceful enhancement of latency-sensitive services to match system energy demands to green energy supplies.

The rest of this paper is organized as follows: Section 2 and Section 3 describe the design of GreenMatch in general and in detail respectively, the implementation is introduced in Section 4, and the experiments are illustrated in Section 5. And a simulation test is described in Section 6. Section 7 discusses related work, and Section 8 gives a summary of our work.

2 Overall Design

2.1 Main Idea of GreenMatch

The motivation of GreenMatch is to maximize the use of on-site green energy while minimizing the negative effects on the performance of critical workloads. Considering that SSDs outperforms HDDs (Table 1), adopting SSDs as a caching layer for disk-based storage systems is common [33][34][36][35][37]. But, using SSDs as the cache layer is not only because of the performance benefits. SSDs also have lower energy consumption than HDDs (Table 1), and GreenMatch takes advantages of this feature to help disk-based storage systems effectively cooperate with variable green energy supply.

TABLE 1
The normal features of SSDs and HDDs

	Read speed peak	Write speed peak	Active energy	Idle energy
SSD	475MB/S	200MB/S	0.6W	0W
HDD	128MB/S	128MB/S	7.9W	5.9W

To maximize the use of on-site green energy, GreenMatch offers a time-shifting workload schedule. The SSD cache can process the write requests to defer writes to P-HDDs and prefetch data from P-HDDs to process the coming read requests. Specifically, GreenMatch divides the process for a non-deferrable request into two stages: a high-performance/low-energy on-line stage, and a low-performance/high-energy off-line stage. The on-line stage can reduce latency due to the high performance of SSDs, while the off-line stage aims to match the variations of green supplies.

Note that if a request has been processed at the on-line stage, an acknowledgement could be returned to the client.

In Figure 2, GreenMatch splits a process for P-HDD write into two stages: an SSD cache write operation and a P-HDD write operation. The P-HDD write operation is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

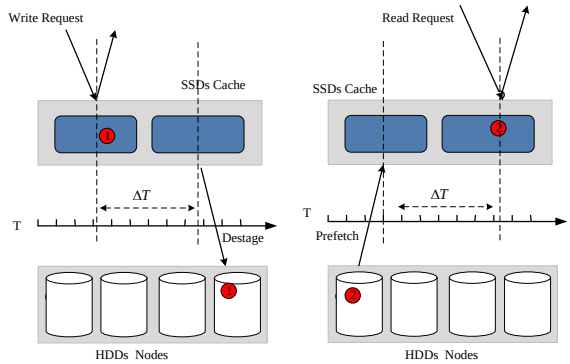


Fig. 2. The processes for P-disk write and P-disk read are divided into two stages: online stage and off-line stage

delayed to be performed until the green energy supply is sufficient or the SSD cache is almost full. Because SSDs consume much less energy than HDDs, the process in the off-line stage contributes a large percentage of the total power consumption. Thus, the majority of the total energy is consumed when renewable energy is sufficient. The process for a P-disk read is also split into two stages: fetching data from a P-HDD to the SSD cache in advance and reading an SSD. GreenMatch can achieve supply-aware workload matching because SSDs have lower energy consumption and lower response time than HDDs.

Figure 3 illustrates the disk-based storage system access requests distribution without and with matching the variations of green power. The white bubbles represent the requests processed by the SSD cache with lower latency and lower energy. The blue bubbles represent requests processed by P-HDDs. The bubbles with black lines means the deferred write data to P-HDDs, and the red heart bubbles represent the prefetched read data from P-HDDs. We find that an SSD cache solution provides the opportunity to match the number of active nodes the green power could supply.

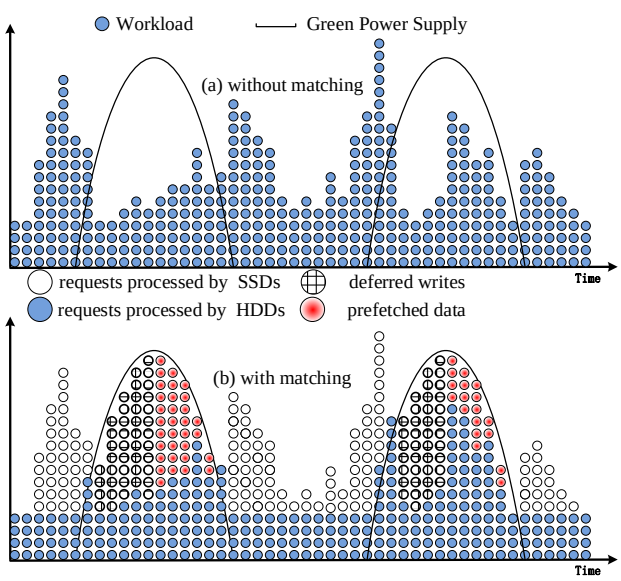


Fig. 3. The disk access requests distribution without and with matching the variations of the green power

2.2 Architecture of GreenMatch

Figure 4 presents the architecture of our GreenMatch, which consists of two parts: a green power supply and a storage system. The former is a combination of solar power, wind power, and grid power. We use a hybrid green power to decrease the time interval of power outages. The grid power is used as backup power supply when the green power fails. The storage system includes an SSD cache, an HDD cache, and a main storage pool.

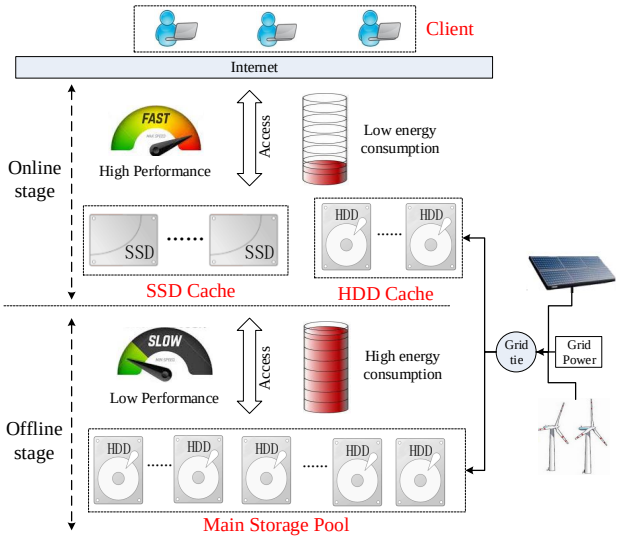


Fig. 4. The architecture of GreenMatch

For the power supply, we use the grid-tie to sync the energy from renewable sources and the grid power. The primary supply is the green power, including solar power and wind power. The secondary supply is the grid power, which is used as supplement when the green energy supply cannot meet the minimum demands.

The main storage pool is a distributed HDD storage system, which uses n-way replication to retain reliability. There are many data layout policies in distributed storage systems [5]. In this paper, GreenMatch employs three-way replications, and our main storage pool is grouped into three pools: a primary pool, a secondary pool, and a tertiary pool.

The SSD cache is used to store the frequently and recently accessed data to retain high performance.

The HDD cache is logically divided into two areas: a logging area and a reserve data area. The reserve data area is a supplement of the SSD cache, as SSDs have a lower capacity/price rate compared with HDDs according to Table 1. The reserve data area is also a transfer station of the primary replica when the primary replica is not totally powered on or there is a need to prefetch some data from the disks that will be powered down in the primary pool. The logging area is used to guarantee the reliability of the dirty data in the SSD cache and the reserve data area of the HDD cache. (see section 3.3 for details.)

The coming write requests will be written into the SSD cache and the logging area of the HDD cache simultaneously. When there is a need to evict data from the cache, the dirty data in the SSD cache will be moved to the reserve data area of the HDD cache or the primary pool. And the dirty

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

data in the logging area and the reserve area will be moved to the non-primary pools and primary pool respectively. Of course, the clean data will be discarded directly.

When a read request comes, it will be firstly served by the SSD cache and the reserve data area of the HDD cache. If a cache miss happens, the read request will be sent to the primary pool. And the non-primary pools only exist to guarantee the reliability of the old data in the primary pool.

3 Detailed Design

3.1 Supply-following Power Control

As our GreenMatch has a supply-aware workload matching policy, the first principle of our power control is to follow the variation of a green energy supply. In other words, the number of active disks is power-proportional to the green energy supply.

3.1.1 Overall Power Control Policy

The main principles of our power control policy are shown as follows. The SSD cache is always on to guarantee performance whether the renewable supply is adequate or not. The HDD cache will not power off unless the green energy supply can power on the whole system and there is no dirty data in the HDD cache. When the green power is adequate, the green energy is used to power on disks in the storage pool as much as possible to retain performance, guarantee availability, and maintain consistency. For every long-term period (one week, one month, and so on), all of the replicas of the storage pool are switched in Round-Robin order to leverage the wear of these replicas.

The power modes are defined based on the range of the renewable energy supply. The characteristics are listed in Table 2. In this paper, the peak green supply is equal to the peak power demand. In the Table 2, P_{SCache}, P_{HCache} and P_{HGroup} represent the maximum power value to power on the SSD cache, the HDD cache and the primary storage pool respectively.

- **Low-power mode.** The green power supply, whose range is $[0, P_{SCache} + P_{HCache})$, is not able to power on both the SSD cache and the HDD cache. In our GreenMatch, as the SSD cache and the HDD cache must be powered on, we use brown power as a supplement during this period.
- **Medium-power mode.** The green power supply, whose range is $[P_{SCache} + P_{HCache}, P_{SCache} + P_{HCache} + P_{HGroup})$, can power on the SSD cache and the HDD cache. But it does not have the capability to power on the primary group in the meantime.
- **High-power mode.** The green power supply, whose range is $[P_{SCache} + P_{HCache} + P_{HGroup}, \infty)$, can power on the SSD cache, the HDD cache and the primary group. The majority of off-line operations are handled in high-power mode to make full use of green energy.

3.1.2 Power Control During Low- or Medium-Power Phase

During low-power phase, it is necessary to utilize the grid power to keep the SSD cache and the HDD cache powered on. In other words, brown power is used as a supplement to meet the minimum demand during this period. During medium-power phase, as the amount of green supply can meet minimum demand, the green power may be used to reduce the performance degradation. Data miss of both the SSD cache and the HDD cache will result in access to inactive disks, which need the help of auxiliary grid energy.

Based on the observations of many real-life workload traces, we found that the fraction of the time in the heavy-intensive workload mode is small. We exploit two kinds of power control policies: performance-first and energy-first. Heavy-load ratio R_{heavy} means the fraction of time the workload is in heavy state, namely: $Ratio_{heavy} = T_{heavyload}/(T_{heavyload}+T_{lightload})$. $T_{heavyload}$ and $T_{lightload}$ respectively represent the time the workload is in heavy mode and the time workload is in light mode.

For performance-first power control policy, the storage nodes in the primary pool are always powered on to avoid inactive disk access during low or medium-power phases. This policy can avoid the long tail of response distribution. However, whether the workload is light or heavy, the primary storage pool is always powered on. This will result in a large amount of energy waste during idle periods. This mode is used for the high-intensive heavy workload. Thus, when R_{heavy} is low, we can use the HDD cache to avoid the energy wasted during idle time.

For energy-first power control, the grid power is used only when the workload becomes heavy or when inactive disks are accessed. This power mode can save more energy. When the workload is heavy, the GreenMatch spins up all storage nodes in the primary storage pool. When the predicted workload in the next interval exceeds the predefined threshold, all storage nodes in the primary storage pool will be spun up. In this way, we can reduce the degree of performance degradation and the amount of energy wasted. During the light workload period in the medium mode, the grid power is used only when there is a request to access the inactive disk. There is a cache policy for avoiding the inactive disk access. During the light workload in low power mode, grid power is used to power on Top-M disks which have more warm data to reduce inactive disk accesses.

Another critical issue of power control is to decide which disk to spin up and spin down. During the low- or medium-power period, in order to avoid inactive disk access, the disks which have more warm data have a higher priority to be powered on. The details are shown in Section 3.2.

3.1.3 Power Control During High-Power Phase

During the high-power period, the majority of off-line operations are processed. The disks in the secondary pool and the tertiary pool are switched on and off. Scheduling the disks in a Round-Robin manner is easy to implement, but not the best choice. An alternative is to choose the disk which is associated with most amount of dirty data. In this way, the dirty data in the caches are destaged as much as possible during this period.

GreenMatch prefers to power on disks whose dirty data rank in the front. If the renewable power can support M extra disks(except for the cache and the primary disk group), disks whose dirty data amount rank top M will be powered on. In the next interval, these M disks are powered

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

TABLE 2
The comparison among three power modes

	Low power mode	Medium power mode	High power mode
Range of renewable supply	$[0, P_{SCah} + P_{HCah})$	$[P_{SCah} + P_{HCah}, P_{SCah} + P_{HPri})$	$[P_{SCache} + P_{HCache} + P_{HPri}, \infty)$
Brown energy used for	powering on all caches	handling inactive disk access	none
Processes for off-line	none	a small fraction	a large fraction
Power state of SSD-cache	all powered-on	all powered-on	all powered-on
Power state of HDD-cache	all powered-on	all powered-on	all powered-on
Power state of primary group	all powered-down	partial powered-on	all powered-on
Power state of other groups	all powered-down	all powered-down	partial or all powered-on

down, and several disks from the remaining inactive disk set are selected to be powered on. In this way, we can ensure that the dirty data are destaged from the cache as much as possible. If all disks in the secondary pool and the tertiary pool have been switched on and off once during this high power period, the current active disks will remain on during the remaining time to reduce performance degradation.

3.1.4 Fine Tuning

In this paper, solar power is used as the dominant green supply, and the other green supplies are used as supplementary supplies. After mixing wind power and solar power, the hybrid green supply curve is at its peak at noon every sunny day. Thus, after the wind power and solar power is mixed, on sunny days, the mixed power remains at a level which contains pre-peak, peak, and post-peak. At night or on cloudy days, the supply curve may present frequent fluctuation periods.

When there are conflicts between global trends and local trends, this will cause frequent switchings of power states, which do harm to disks and result in performance degradation. We need fine tuning to address this problem. In order to avoid unnecessary frequent operations for storage nodes switching, there are three cases to alleviate this conflict. First, during pre-peak periods, while the global energy supply is increasing, the local energy supply is decreasing. We can utilize some brown energy to make up this gap. Second, during post-peak periods, while the global energy supply is decreasing, the local energy supply is increasing. We do not power on disks to avoid frequent switching. Third, during stable power generation periods, while there is a variable local power generation, we use brown energy or waste green energy to keep the current power state of all devices.

3.2 Multi-level Cache Management

In this paper, data are categorized into three types: hot data, warm data, and cold data. Hot data refers to frequently and recently accessed data. Warm data represents infrequently but recently accessed data and frequently but not recently accessed data. Cold data represents the infrequently and not recently accessed data. The HDD cache is logically partitioned into two areas: a **logging area** and a **reserve data area**. The reserve data area is used to guarantee data availability in low green-power mode and buffer the cold dirty data evicted from the SSD cache whose corresponding disks in the primary pool are inactive in low-power mode. The logging area is used to retain the reliability of dirty data in the SSD cache and the reserve data area of the HDD cache.

3.2.1 Inactive P-disk First Replacement Policy for HDD-Cache

If a read request cannot find the required data in the SSD cache or the HDD cache, the request will be issued to the primary storage pool. However, if the corresponding P-disk is inactive, the P-disk need to be powered on to ensure the availability. And we call this access inactive P-disk Access. The inactive P-disk access will bring extra cost due to several factors. First, frequently switching the power mode of P-disks will accelerate the aging of disks. Second, powering on a P-disk results in a large access latency penalty. Third, the operation to power on or power down will consume extra energy.

State_Table	Disk_1	Disk_2	…	Disk_N
SCache_Num	s1	s2	…	sn
HCache_Num	h1	h2	…	hn
Primary_Num	p1	p2	…	pn
Power State	on	on	…	off

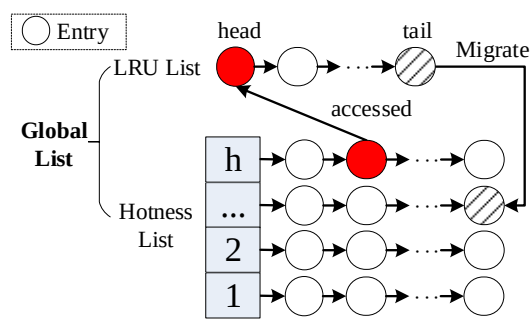


Fig. 5. The State_table and the Global_List

In order to avoid inactive P-disk accesse, we exploit a replacement policy called Inactive P-disk First, which means that the data of inactive P-disk would be cached first. And we use the Global_List to manage all the data in the storage system, which consists of an LRU list and a hotness list. The LRU list ranks the entries based on recent access, while the hotness list is organized as a sorted array to classify data by the hotness, where every element points to a list having the same hotness. We also use a State_Table to record the state of the primary pool disks. In detail, for each P-disk, the State-Table contains the power state (on or off) and the number of data objects in the SSD cache (SCache_Num), the reserve data area of the HDD cache (HCache_Num), and the primary storage pool (Primary_Num). And each entry is designed as (objest_id, disk_id, disk_state, hotness,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

location, dirty_flag). The objest_id means the logical address, the disk_id means the pool disk in the primary pool that the data belong to, the disk_state means the power state of the disk the data belong to, the hotness means the popularity, the location means the device the data is stored, and the dirty_flag means the dirty(dirty_flag=1) or clean(dirty_flag=0) state.

The details of this scheme are shown algorithm 1.

Algorithm 1 Inacitve P-disk First

Require: $State\_Table$, $HCache\_Size$, $SCache\_Size$, A(the active disk number), E(the number of disks the green energy can supply, excluding the caches)

while an inactive pool-disk access to disk x happens **do**
  power on disk x with grown energy
  $disk\_state$ of the entries belong to disk x = on
  move the accessed data to the reserve data area of the HDD cache
  $HCache\_Num$ of disk x = $HCache\_Num$ of disk x + 1
  while $Sum\_of\_HCache\_Num > HCache\_Size$ **do**
    evict the coldest entry whose $disk\_state$ is on
    $Sum\_of\_HCache\_Num = Sum\_of\_HCache\_Num - 1$
  end while
  move the accessed entry to the $head$ of the LRU List
  if the LRU List is full **then**
    remove the tail entry and migrate it to the Hotness List
  end if
end while
while $E > A$ **do**
  comparing the $HCache\_Num$ of each disk in the $State\_Table$
  if disk y has the largest $HCache\_Num$ **then**
    power on disk y
    change the $disk\_state$ of the entries belong to disk y from off to on
  end if
end while
while $E < A$ **do**
  comparing the $Primary\_Num$ of each disk in the $State\_Table$
  if disk z has the smallest $Primary\_Num$ **then**
    power down disk z
    change the $disk\_state$ of the entries belong to disk z from on to off
  end if
end while

3.2.2 Global_List based Prefetching for Read Cache of SSD Cache

Many research papers [33][**?**] have shown that popular data only accounts for a small fraction of the whole data set. With an appropriate cache management scheme, the storage system can achieve a significant reduction of response time. We use the Global-List shown in the previous section to identify hot data to improve the SSD cache hit ratio.

We designed a prefetch algorithm that considers the capacity and lifetime of the SSDs. This algorithm is designed to decrease the number of erase cycles in the SSDs and prefetch the hottest possible data. The time interval of replacement operations is set to be very small to be adapted to the popular data set shifting. For the prefetching algorithm, we need to manipulate two dynamic thresholds: T_{Hot} represents the hotness threshold to identify whether the data in the HDDs is hot or not, and T_{Cold} represents the coldness threshold to identify whether the data in the SSDs is cold or not.

We use a scheme called Top-M to dynamically manipulate the hotness threshold T_{Hot}. As shown in previous sections, the hotness list will record the popularity and location of every object, so we can lookup the Global-List to calculate two variables $SCnt(x)$ and $HCnt(x)$. While $SCnt(x)$ is used to record the number of objects located in SSDs and that have the hotness x, $HCnt(x)$ is used to record the number of objects located in HDDs and that have the hotness x. With the value x varying from maximum value to minimum value step by step, we sum the value of $(SCnt(x)+HCnt(x))$ until the value M meets the demand of the following equation.

$$\sum_{x=i}^{N+1}(SCnt(x)+HCnt(x)) < M \leq \sum_{x=i-1}^{N+1}(SCnt(x)+HCnt(x)). \quad (1)$$

M represents the number of objects the SSD can store. In order to limit the number of erase cycles to prolong the lifetime of SSDs, we add a constraint shown in the following equation.

$$\sum_{x=j}^{N+1} HCnt(x) < M/R \leq \sum_{x=j-1}^{N+1} HCnt(x) \quad (2)$$

where M/R represents the maximum number of objects to be prefetched at a time. Thus, $T_{Hot} = max(i-1, j-1)$ is the hotness threshold we need.

Correspondingly, we use a scheme called Bottom-N to manipulate the coldness threshold T_{Cold}. To find the N coldest data in SSDs, Bottom-N sums the value of $SCnt(x)$ from minimum value to maximum value until N is in range $[\sum_{x=0}^{k} SCnt(x), \sum_{x=0}^{k+1} SCnt(x))$. Thus, $T_{Cold} = k+1$ is the coldness threshold we need. The oid of objects whose access number is smaller the T_S will be added to a list for eviction.

3.3 Logging and Destage

3.3.1 Logging Scheme

In order to ensure the reliability of dirty data in the SSD cache and the reserve data area of the HDD cache, we deploy a logging area in the HDD cache. There are two options for the logging area: replica-aware logging mode and erasure-coding-aware logging mode. For replica-aware logging mode, the logging area of the HDD cache will host $n-1$ replicas for an n-way replicated distributed storage system. In this mode, the cached dirty data have the same level of failure tolerance to the normal replicas in the main storage pool. However, it will introduce a large extra amount of space in the HDD cache to host the replicas. On the contrary, the erasure-coding-aware logging mode can obtain a high capacity utilization without significant reliability attenuation.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

For replica-aware logging mode, we take a three-way replicated storage system as an example. The RAID10 can be used for the logging area in the HDD cache to store the replicated data. It will keep the same level of fault protection to the normal data. However, it gives rise to high cost and low capacity utilization. If the workload is very heavy, our GreenMatch will host a large amount of logged data. And this will reduce the capacity of the reserve data area.

For erasure-coding-aware logging mode, we take the RAID5/RAID6 arrays as examples. The reason that we prefer to use erasure-coding to log the dirty data is that it can decrease the overhead of capacity and maintain reliability compared with replica-aware logging. In order to reduce the negative impacts of logging on performance, we set a small buffer to hold the working parity. As shown in Figure 6, for an incoming write request, the request will be issued to the SSD-cache, the logging area of the HDD cache and the buffer. The parity can be read quickly from, and written back to the buffer. When the entire stripe is logged into the logging area, the corresponding parity is written back to the logging area. In this way, the logging performance can be enhanced.

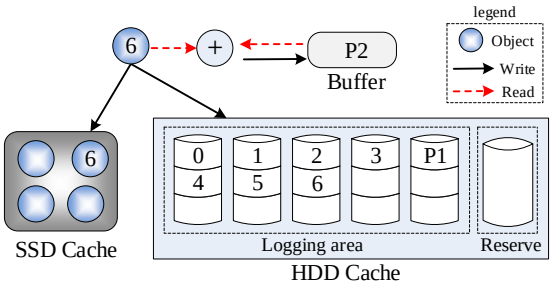


Fig. 6. Erasure-coding-aware logging in logging area

3.3.2 Destage Policy

In our design, the destage process means that synchronizing the dirty data in the SSD cache and the HDD cache into the storage pools. In low-power mode, as the reserve data area of the HDD cache is a supplement of the SSD cache, destage operation is prohibited unless the reserve data area of the HDD cache is full. In medium-power and high-power mode, destage operations under a heavy-workload period will affect the performance. In contrast, no destage during a light-workload period will make idle disks a waste of energy. So the optimal time for destage operations is during the high-power/light-workload period. And the policy is described in Algorithm 2.

To ensure the data consistency of the storage system and make it simple, the dirty data in the SSD cache and the reserve data area are only in charge of the primary storage pool. Because the reserve data area of the HDD cache is a supplement of the SSD cache in low-power and medium-power mode.

Apart from guaranteeing the reliability of the dirty data in the SSD cache and the reserve data area of the HDD cache, the logging area of the HDD cache is only responsible for the remaining non-primary replicas. For erasure-coding mode, we have to use the 1-to-N mapping scheme. For example, a flag of (0, 1) means that the logged data is clean for the secondary replica, but dirty for the tertiary replica. The cold clean data whose flag is (0, 0) can be reclaimed directly, but the cold dirty data will remain in the logging area waiting for the destage process. For replica-aware mode, we use 1-to-1 mapping. There is only one data copy in the SSD cache, which is responsible for the primary storage pool, and the $n-1$ data copies in the HDD cache logging area are in charge of the $n-1$ remaining storage pools, which could be processed in parallel.

Algorithm 2 Destage policy

```
if the SSD cache is almost full then
    Part of the dirty data will be moved to the reserve
    data area of the HDD cache
end if
if the HDD cache is full then
    do destage now
    power on the pool disks with the largest number of
    dirty data
    synchronize data from the SSD cache to the primary
    replica
    synchronize data from the reserve data area of the
    HDD cache to the non-primary replica
end if
while medium-power and high-power mode do
    if workload is light then
        destage for the active pool disks
    end if
    if workload is heavy then
        pause some of the destage
    end if
end while
```

3.4 Fault-Tolerance

Fault-tolerance capability is a key feature of the distributed storage system. Within GreenMatch, there are three kinds of nodes: SSD-cache nodes, HDD-cache nodes, and the HDD storage nodes in the main storage pool.

The nodes in the main storage pool are organized in a distribution storage system. When some P-disks fail, the corresponding dirty data will be acquired from the SSD cache and HDD cache, as mentioned in section 3.3.2. The remaining clean data could be recovered by the recovery algorithm of a practical distribution storage system. In this section, we focus on analyzing and designing the fault-tolerance of cache nodes.

If an SSD-cache node fails, we can use the Global-List to recover the clean data by prefetching the hot data from the main storage pool. As the replicas of the dirty data in the SSD cache are logged in the logging area of the HDD cache, dirty data can be fetched from the logging area in the HDD cache or written back to the corresponding storage nodes in the primary storage pool.

For HDD-cache node failures, there are three types of data to be processed. First, as the HDD cache is used as a cache for the main storage pool, the clean data can be recovered from the main storage pool using the Global-List. Second, as the dirty data in the reserve data area and the corresponding logging data are not in the same

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

storage device, the dirty data can be recovered from the logging area. Third, as mentioned in section 3.3.1, data in the logging area can be calculated by the erasure-coding mode, or be replicated by the replica-aware logging mode. And the recovered dirty data can be directly written back to the corresponding nodes in the secondary and tertiary storage pools.

4 IMPLEMENTATION

Figure 7 presents the implementation of GreenMatch. There are five main components: the SSD cache, the HDD cache, the main storage pool, the power mode scheduling service, and the client. Data in SSD cache is distributed based on hash partitions. For details, we install a sheepdog system on each SSD to make up a distributed system. The logging area of the HDD cache is offered with two implementation options, and here we choose the erasure-aware logging mode (please see section 3.3.1 for details). In addition, the reserve data area of the HDD cache is separated from the logging area physically, including several disks. The main storage pool is organized as a block-level distributed storage system to increase the scalability, with three replicas. The power mode scheduling service is used to decide the number of disks could be active according to the green energy supply.

Figure 7 also shows the data paths of our GreenMatch. The write request is mainly processed by the SSD cache and the logging area of the HDD cache synchronously unless it hits the reserve data area of the HDD cache. If the write request hits the reserve data area, it will write the reserve data area and the logging area synchronously instead. The read request is usually processed by the SSD cache and the reserve data area of the HDD cache. And a cache miss read will be done by the primary storage pool. There is also data exchange between the SSD cache and the reserve data area of the HDD cache according to the data hotness. Besides, during the destage period, the SSD cache and the reserve data area of the HDD cache are responsible for the primary storage pool, and the logging area of the HDD cache is responsible for the non-primary storage pools. Meanwhile, both of the cache can prefetch some hot data from the primary storage pool when the corresponding disks are going to be inactive.

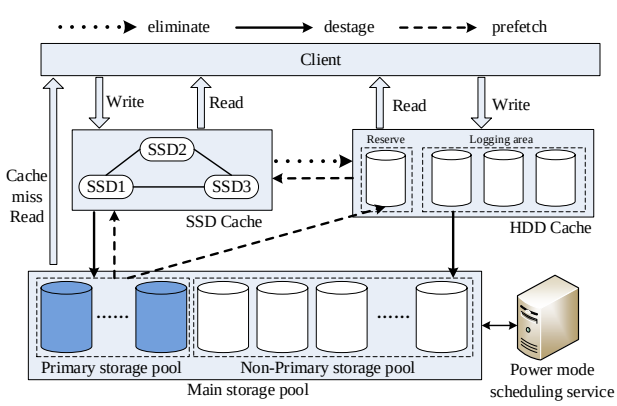


Fig. 7. Data path of GreenMatch

TABLE 3
Hardware details

OS	Linux version 2.6.35.6-45.fc14.x86_64	
CPU	Intel (R) Xeon (R) CPU E5506@2.13GHz	
Hard Disk	Western Digital RE4 1TB WD1003FBYX 3.5 SATA/64MB Cache 3Gb/S 7200rpm	
SSD	SanDisk SATA Revision3.0 6Gb/s SSD 64G	
Disk Parameters	Individual Disk Capacity	1TB
	Average Seek Time	4.2 ms
	Average Rotation Delay	2.08 ms
	Internal Transfer Rate	128 MB/Sec
	Idle Power	5.9 W
	Standby Power	0.7 W
	Active Power	7.9 W
SSD Parameters	Individual SSD Capacity	64GB
	Sequential Read(up to)	475 MB/s
	Sequential Write(up to)	200MB/S
	Random Read (up to)	8200 IOPS
	Random Write (up to)	1100 IOPS
	Active Power	0.6W

5 EVALUATION

5.1 Experiment Setup

We evaluated GreenMatch using a seven-server cluster, where each server has a four-core CPU and a 32GB RAM memory. The servers are inter-connected by a gigabit ethernet switch. While we use disks as storage nodes, our experiment deploys three SSDs as the SSD cache, three SATA disks as the HDD-cache and 42 SATA disks with 1TB of free space as the main storage pool. GreenMatch also includes a server to receive renewable trace data and store power consumption data for power management. The parameters of the SSDs and HDDs are shown in Table 3.

The workload traces are collected from the MSR with a total of 36 different traces[8]. We carefully chose four week-long traces from these traces, whose characteristics are listed in Table 4. Because running a seven-day trace non-stop is infeasible, our experiment accelerates the test by a factor of 64.

TABLE 4
The features of traces

Traces	Write Ritio	IOPS	Avg. Req	Function
usr	59%	83.87	22.66KB	User home directories
web	70%	50.32	14.99KB	Web SQL server
mds	88%	18.41	9.19KB	Media server
rsrch	91%	21.17	8.93KB	Research projects

Renewable energy supply data in our paper is collected from the National Renewable Energy Laboratory[7]. The wind and solar generation datasets are time-series of power output. For the solar trace, because the average sunshine intensity is different in winter and summer, we carefully chose two week-long traces respectively in summer and winter from a location in Washington. These two week-long solar traces are named $ST1$ and $ST2$. For the wind traces, since we want to study the impacts of supply variation and intermittency, we choose two wind traces based on the wind power intensity. These two week-long wind traces are referred to as $WT1$ and $WT2$. However, this paper aims to study GreenMatch based on a hybrid power combination of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

TABLE 5
The characteristics of green power traces

	ID	Avg power(MW)	Peak power(MW)	Power outage ratio	low power ratio	high power ratio	note
Solar Trace	ST1	8.1	27.7	45.8%	54.2%	31.0%	summer
	ST2	5.65	25.1	62.5%	66.7%	24.4%	Winter
Wind Trace	WT1	2.8	13.0	17.2%	75%	0%	Low
	WT2	9.99	18.95	2.9%	34.5%	41.1%	High
Combined Trace	SWT1	10.9	40.7	8.3%	45.2%	34.5%	ST1+WT1
	SWT2	18.07	45.07	0%	20.2%	60.1%	ST1+WT2
	SWT3	8.47	35.20	11.9%	54.1%	27.4%	ST2+WT1
	SWT4	15.65	42.95	0%	23.2%	55.4%	ST2+WT2

wind power and solar power. Thus, our experiments combine four hybrid power traces (*SWT1, SWT2, SWT3, SWT4*) using solar traces and wind traces, whose characteristics are listed in Table 5. Besides, the power traces can be linearly scaled to match the storage scale.

To evaluate GreenMatch, there are five configurations used for comparison shown as follows.

- **Standard Sheepdog**: a block-level distributed system called Sheepdog[9], which can provide volume and container services and manage storage nodes intelligently. To ensure balanced data placement and workload services, Sheepdog uses Consist Hash Table(CHT) to distribute primary objects and utilizes Chained-Declustering[31] to spread corresponding replicas. Without any power management policy, Standard Sheepdog is provisioned with 45 SATA disks with 1TB of free space as the storage pool.
- **WDS-CP** (**W**orkload-**D**riven **S**heepdog with **C**oarse-granularity **P**ower control policy): To conserve energy under light workload, this scheme employs a coarse-granularity energy-proportional power control scheme. To retain high availability, this scheme uses Group-aware De-clustering as the data placement policy. The other configurations of this scheme are the same to Standard Sheepdog.
- **WDS-FP** (**W**orkload-**D**riven **S**heepdog with **F**ine-granularity **P**ower control policy): To achieve a larger amount of energy savings, this scheme exploited a fine-granularity energy-proportional power control scheme. Namely, the number of storage nodes is power-proportional to workload in fine-granularity.
- **SSD-based Sheepdog**: This scheme is an SSD-cache-based Sheepdog without any power management policy. It is different from Standard Sheepdog that SSD-based Sheepdog employs three SSDs as an ensemble cache.
- **GreenMatch**: an SSD-cache-based Sheepdog with a novel cache management scheme and an appropriate power control policy to match the supplies of renewable power. Three SSDs as the SSD cache, three SATA disks as the HDD-cache and 42 SATA disks with 1TB of free space as the main storage pool.

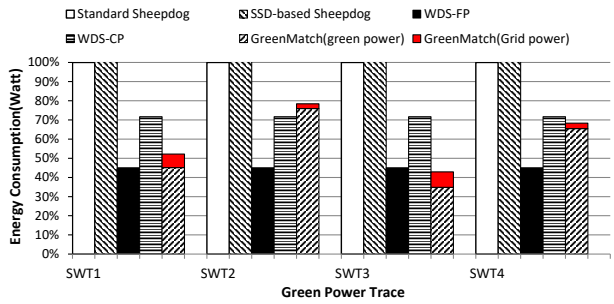
5.2 Energy Consumption Impact

Figure 8(a) shows the energy consumed by the five configurations under various power traces. First, GreenMatch can produce significant reductions in grid power under various power traces. Under *SWT2*, GreenMatch can reduce grid power up to 97.54%, because the low-power mode ratio (20.2%) combining with the Power outage ratio(0%) is the smallest as shown in Table 5. The grid power is mainly consumed in low-power mode when the renewable energy cannot meet the minimum energy demand to power on the SSD cache and HDD cache. Thus, only a small amount of grid power is used by GreenMatch under various power traces. Second, Our GreenMatch can reduce total energy consumption with our proposed cache solution and supply-following power management scheme. $SWT3$ delivers the highest total energy savings(57.08%), but has the largest response time as shown in the next section. As our power control scheme aims to use renewable energy as much as possible, the amount of the renewable energy supply has impacts on total energy savings. Third, under the *SWT2* power trace, GreenMatch consumes more energy than WDS-FP due to the large amount of green supply in *SWT2*. The impact of the amount of green supply will be analyzed in a latter section. Forth, WDS-FP with fine-granularity power control scheme will consume less energy than WDS-CP. However, it will introduce worse performance and a larger amount of data migration due to frequent P-disk power-state switching. Lastly, as shown in Figure 8(a), Standard Sheepdog and SSD-based Sheepdog without power management are non energy-proportional systems.

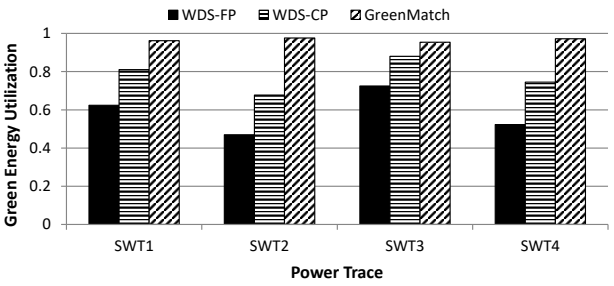
Renewable energy utilization represents the ratio of the used green energy to the total green energy supplies. The main principle of our power control is to power on P-disks as many as possible to make full use of renewable energy. The extra green energy supplies that are not able to power on at least one storage device are wasted. Thus, the amount of wasted energy, i.e. generated but not used energy, is very small. Figure 8(b) presents green utilization under various power traces. First, GreenMatch can achieve high renewable energy utilization under various renewable power traces. Among these renewable power traces, $SWT2$ and $SWT4$ get the maximum utilization(97.5%) and minimum utilization(95.4%), respectively. As the SSD cache is used to split the request into two phases and the second phase is used to match the green supply. In other words, our GreenMatch is a supply-following power control scheme. The energy is only wasted when the extra energy cannot support one storage device. Second, WDS-FP has lower green energy utilization than WDS-CP. This is because the green energy utilization is also decided by the amount of overlay area for the workload and supply.

Figure 9 shows the power consumed by GreenMatch

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems



(a) Total energy consumption under various power traces



(b) Green utilization under various power traces

Fig. 8. Green utilization and total energy consumption under various power traces

and SSD-based Sheepdog over one week under the $SWT2$ power trace and the usr workload trace. First, the grid power consumed by GreenMatch is very small. As mentioned in previous sections, brown energy is mainly consumed in low-power mode, when the green power supply cannot meet the minimum energy demand. The low-power mode ratio under the $SWT2$ trace is very small due to an adequate supply of green energy. Second, no grid power is consumed between the 105th and 168th hours. During this period, the wind turbines generate sufficient power, so GreenMatch works in high-power mode. Third, the power consumption of the standard Sheepdog is almost flat(the power range in storage servers is small), because this distributed storage system without power management is a non energy-proportional system. Thus, our GreenMatch, with its careful power management, can make full use of green energy and achieve significant power reductions.

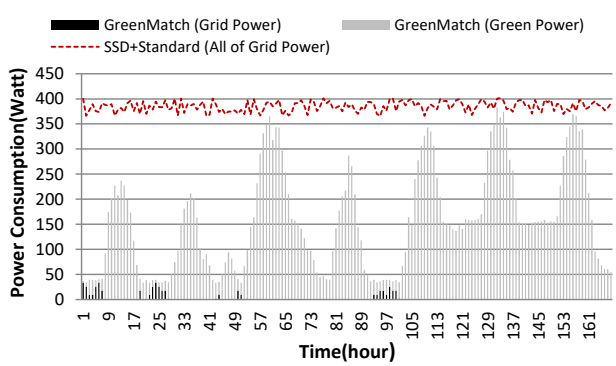


Fig. 9. Power consumption under the usr workload trace and the $SWT2$ power trace

5.3 Performance Impact

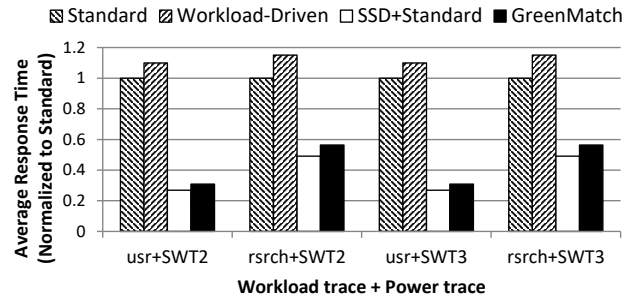


Fig. 10. The performance under various power traces and workload traces

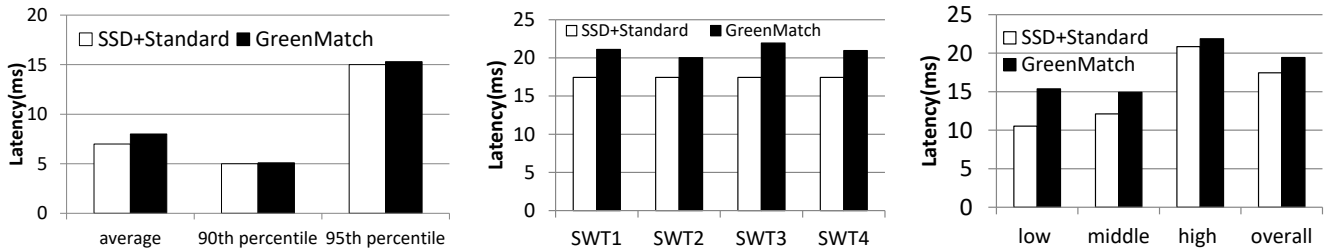
Figure 10 shows the performance comparisons among four configurations under various workload traces. We can find that after deploying an SSD cache to a standard distributed storage system, there is a considerable improvement in performance. This is because SSDs have higher performance than HDDs. In addition, GreenMatch has a lower response latency than standard Sheepdog, because the high SSD cache hit ratio reduces the latency for the majority of online requests. However, GreenMatch creates a small amount of performance degradation compared with the SSD-based Sheepdog because an SSD cache miss may result in accessing a powered-down P-disk with a large latency penalty. Furthermore, the parallelism is decreased in low-power and medium-power, thus an SSD cache miss in GreenMatch has a larger response latency than the SSD-based sheepdog.

Figure 11(a) show the performance comparison between the SSD-based Sheepdog and GreenMatch under $rsrch$. We can find that GreenMatch creates a small amount of performance degradation compared with the SSD-based Sheepdog, because the data misses in the SSD cache may result in accesses to a powered-down P-disk with a large latency penalty. Furthermore, the parallelism is decreased in low-power and medium-power mode. Thus an SSD-cache miss within GreenMatch has a larger response time than the SSD-based Sheepdog. However, the latency of GreenMatch is very close to the latency of SSD-based Sheepdog at the 90th percentile of the distribution. With high green energy utilization and significant power reductions as shown in the previous section, our GreenMatch achieves a graceful upgrade of performance.

Figure 11(b) shows the performance comparisons of two configurations under various renewable power traces. GreenMatch delivers graceful performance degradation of latency-sensitive services under various green power traces. For our power control scheme, the SSD cache is always on whether renewable supplies are adequate or not. The latency penalty is caused by SSD-cache misses. Thus, with a very high SSD-cache hit ratio, the intermittency and variability of renewable supplies have a negligible impact on response latency. Our GreenMatch can match green energy supplies with graceful degradation of performance.

Figure 11(c) illustrates the performance comparisons of two configurations under various power modes. First, compared with the SSD-based Sheepdog, GreenMatch has a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems



(a) Under $rsrch$ trace (b) Under various power traces (c) Under various power modes

Fig. 11. Response time comparison under different scenes

significant degradation of performance in low-power mode. In low-power mode, the HDD cache has worse parallelism than the main storage pool. Thus, an SSD-cache miss will cause a greater latency than the SSD-based Sheepdog. Furthermore, if misses happen both in the SSD and HDD cache, there may give rise to an inactive P-disk access with a large latency penalty. Second, GreenMatch performs well under medium power and high power due to the high hit ratio of the SSD cache. And the data access missing the SSD-cache will be redirected to main storage pool, which has a higher parallelism than the HDD cache. Third, the overall performance of GreenMatch is similar to the SSD-based Sheepdog. Most of the time, GreenMatch under the $SWT2$ is in high power mode, because the $SWT2$ offers sufficient supplies of green energy.

Figure12 shows the response time variations under the usr and the $SWT2$. Although the response time under GreenMatch is greater than the SSD-based Sheepdog, GreenMatch performs better than standard Sheepdog. Between the 15th and 29th hours, the response time of GreenMatch is greater than the SSD-based Sheepdog. This is because only the SSD and the HDD caches are powered on during this period. The parallelism of the HDD cache limits the performance in low-power mode. Between the 106th and 162nd hours, GreenMatch performs very well due to adequate renewable supplies. During this period, the renewable supplies are sufficient to power on the primary storage group to guarantee data availability, so that inactive P-disk accesses will not happen. Furthermore, the SSD cache hit ratio is very high, so only a small fraction of data accesses will be redirected to the main storage pool.

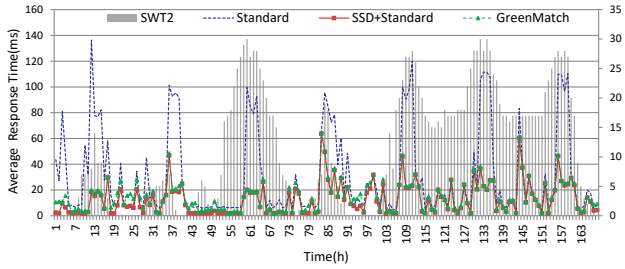


Fig. 12. Observed distribution of average response time under the usr workload trace and the $SWT2$ power trace

5.4 The Effect of Caches and Power Control

Figure 13(a) shows the impact of an HDD cache on the amount of energy saved under various power traces. First, compared with GreenMatch with a performance-first power control policy, GreenMatch with an energy-first power control policy can save more energy. Because the I/O intensity is light during most of the time and the numbers of active storage nodes could be reduced. Second, the energy consumed by GreenMatch with a performance-first policy under $SWT3$ is the largest, because the amount of $SWT3$ is the smallest among all power traces. Third, the energy consumed by GreenMatch with an energy-first policy under $SWT2$ is the smallest. As shown in the Table 5, the high-power mode of $SWT2$ is the largest among all traces.

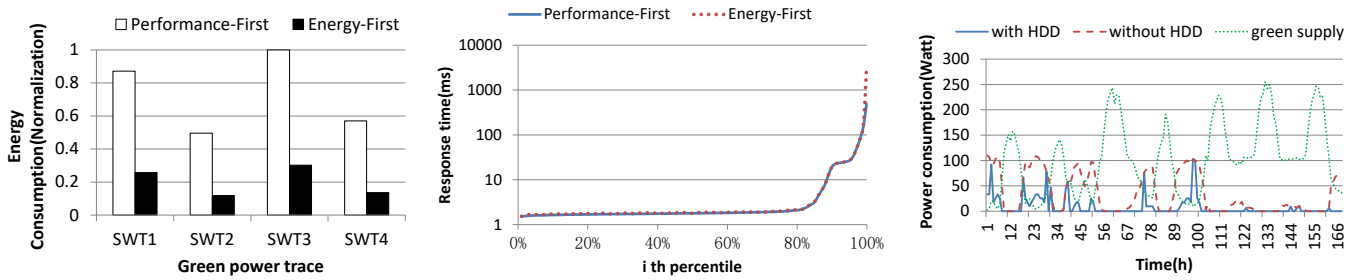
Figure 13(b) shows the tail latency of GreenMatch. Tail latency is an important metric to measure the responsiveness of datacenter services. First, GreenMatch with an energy-first policy has a long distribution tail due to the inactive P-disk access caused by missing the SSD and HDD caches. Accessing the inactive P-disks will consume seconds. However, GreenMatch with an energy-first policy is close to GreenMatch with a performance-first policy in the 90th/95th/99th response time percentiles, because our inactive P-disk first replacement policy is designed to reduce the number of inactive P-disk accesses. If the size of the HDD cache is large enough, the inactive P-disk access may be avoided entirely.

Figure 13(c) compares the observed power consumption between GreenMatch with and without an HDD cache under the usr workload trace and the $SWT2$ power trace. First, The GreenMatch with an HDD cache consumes less energy than GreenMatch without an HDD cache at any time. For GreenMatch without an HDD cache, the storage nodes in the primary storage pool are always powered on. The grid power is mainly used at night to supplement the green supply. Between the 105th and 161st hours, the green power supplies are so sufficient that grid power is seldom utilized during this period.

5.5 The Impact of the Amount of Green Supply

The green energy utilization and the total energy savings have been illustrated in previous sections. The green supply ratio is defined as the fraction of green supply to total energy. Figure 14 shows the impact of the supply peak value on these three metrics under $SWT1$ power trace and usr workload trace. First, the green utilization ratio is steady. Because our power control policy is proportional to the green supply, the wasted energy is not related to the amount of green supply peak value. Second, with the growth of peak value, the green supply ratio is increasing. This is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems



(a) Energy saving (b) Latency tail (c) Observed power consumption under usr and $SWT2$

Fig. 13. The impact of power control schemes, caches and the amount of green supply

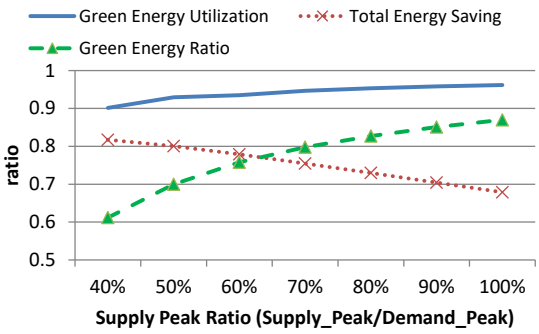


Fig. 14. Green utilization and total energy consumption under $SWT1$ power trace and usr trace

because the amount of grid power is decreasing along with the increasing green supply. Third, with the green supply peak value increasing, the amount of total energy savings is decreasing. Thus, we can conclude that the relationship between the amount of green supply and the overall effect is not a case of "more is better". With more green supply, the green supply will surpass the actual demand.

6 SIMULATION

6.1 Experiment Setup

To evaluat the effect of the GreenMatch in large-scale data centers, we choose the simulation method due to the limited resources. The overview of our prototype is shown in Figure 15.

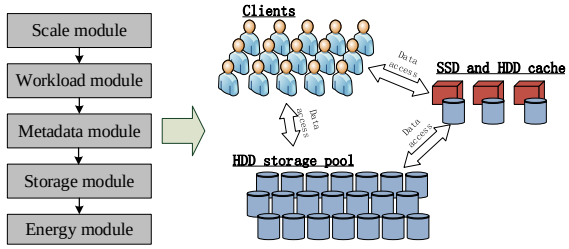


Fig. 15. Simulation architecture

The Scale module is responsible for the scalability of the system. As the storage pool is based on an n-way replicated distributed storage system, and the SSD cache is organized by consistent hashing, we can scale up and down the SSD cache and the HDD storage pool easily. As for HDD cache, if RAID cannot match the massive scale system, we can also use n-way replicated distributed system to manage the HDD cache. Thus, our GreenMatch is also suitable for the massive scale system. The Energy module is used to adapt the renewable energy supply collected from the National Renewable Energy Laboratory [7] to supply the system of different scales. In the Workload module, there are many clients, and each client will create some threads to dispatch the same trace, such as usr trace, to the Storage module at a time. And we can control the workload to match the peak workload of different scale systems by increasing or decreasing the number of clients or threads. The Storage module is responsible for the simulation of actual disks and SSDs, we just return the request access time calculated by the request type, size and target storage medium, based on statistical numbers. Besides, the Metadata module deals with the Stable_table and the Global_List, which manage all the data in the storage system.

Moreover, we install the modified sheepdog [9] in three real servers to simulate all the primary-replica server nodes, all the second-replica server nodes and all the third-replica server nodes respectively. In a real server, to simulate a large number of virtual server nodes, we create many threads and every thread represents a virtual server node. Thus, the network module of sheepdog has been modified to address the communication between the virtual servers by using the message queue mode. Therefore, there are two types of network communication: one is the original network module of sheepdog working when there is a communication between the real servers, and the other is the message queue mode used for virtual server nodes within a real server. Besides, we also set a network latency for the message queue mode according to the statistical data.

To evaluate the GreenMatch in the simulation, there are four configurations used for comparison shown as follows.

- **Standard**:This scheme is an SSD-cache based data center without any power management policy.
- **WDS** (**W**orkLoad-**D**riven **S**ystem):We keep the cache and the primary replica active, and will power on any P-disk when the upper layer desires.
- **GreenMatch-C** (GreenMatch **C**ache):A standard mode data center with a novel cache management scheme and an appropriate power control policy to match the supplies of renewable power. Besides, both the SSD and HDD cache are always kept active.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

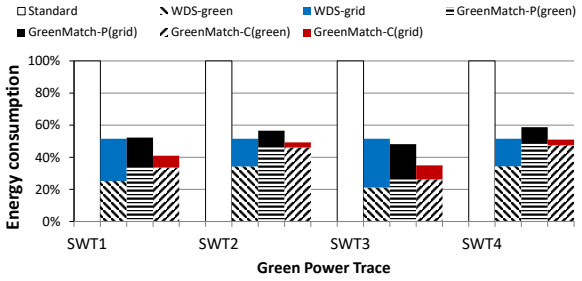


Fig. 16. Total energy consumption under various power traces

- **GreenMatch-P** (GreenMatch **P**rimary):The only difference with GreenMatch-C is that GreenMatch-P keep both the cache and primary replica active.

6.2 Large Scale Analysis

6.2.1 Massive Scale Analysis

To make a large-scale test, we set the number of disks in the storage pool as 30000, thus providing a 0.3MW energy consumption system. Figure 16 shows the energy consumed by the four configurations under various power traces with 1000 cluster units, and Figure 17 shows the performance comparisons. In Figure 16, the upper part of the splicing cylinder represents the grid energy actually used, and the lower part represents the green energy actually used. Similar to the previous test, first, GreenMatch-C delivers the highest grid energy saving, reducing grid power up to 91%-97%. But GreenMatch-C has the worst performance as it keeps the main storage pool inactive when the green energy is not enough. Second, GreenMatch-P and WDS have almost the same performance, better than GreenMatch-C, as they should keep both the cache and primary replica active, which results in more grid energy consumption than GreenMatch-C. Third, WDS has the lowest green energy utilization, i.e. 70%-80%, while it is about 98% for two GreenMatch schemes. Fourth, WDS consumes more grid energy than GreenMatch-P, when keeping all the replications in sync, WDS does it according to the workload of the upper layers, but GreenMatch-P does it mainly decided by the supply of the green energy.

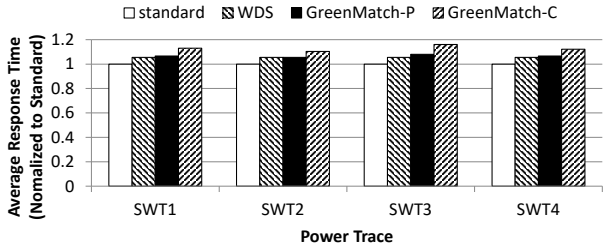


Fig. 17. Performance comparison under various power traces

6.2.2 Comparison of Different Scales

To evaluate GreenMatch in different scale sizes, we have done many tests. For simplicity, we take the test of 3000, 15000, 21000 and 30000 storage disks under *SWT2* power trace as examples for comparison. As shown in Figure 18 and Figure 19, the results demonstrate the same regularity as our previous tests. Therefore, though there are some limitations in our simulation tests, our GreenMatch could match different storage scale sizes theoretically as well.

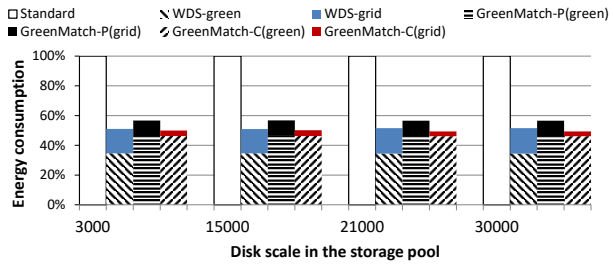


Fig. 18. Energy consumption under different storage scales

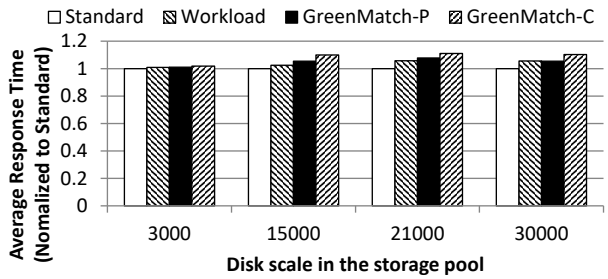


Fig. 19. Performance comparison under different storage scales

7 RELATED WORK

Renewable Energy Driven Data Centers. Blink [12] deploys two wind turbines and two solar panels to power on a small cluster, and uses a staggered blinking schedule to cope with energy variability. Researchers from Rutgers University have built a solar-powered micro datacenter called Parasol, whose workloads and energy sources are managed by a system called GreenSwitch[13]. A Net-Zero datacenter partially powered by renewable energy has been built by HP Labs[10]. Several researchers have studied the capacity planning to minimize cost and carbon emissions, such as [19][20]. Chao Li et al. have developed studies on green-driven data center regarding sustainability [27] scalibity [28] and power security [26]. SolarCore[17] leverages per-core DVFS on multi-core systems to temporarily lower server power demands when solar power drops. The energy storage devices [30][29] are typically used to smooth the unpredictable green supply. Article [25] exploits the on-site solar power and batteries to keep the power peak. CADRE [24] uses an online, distributed algorithm to decide the replication policy based on the footprint-replication curve, and provides huge carbon footprint savings. But CADRE primarily addresses the data placement among many datacenters, and there are no servers turned off to save the grid energy when green energy is low. Our work differs in that we are aiming to design supply-aware workload matching for latency-sensitive workloads without adding a latency penalty.

Workload Scheduling for Sustainable Data Centers. GreenSlot [16], GreenHadoop [15], and GreenSwitch [13]

suggest workload scheduling to match renewable energy supplies by deferring delay-tolerant workloads, but these papers focuse on latency-insensitive batch processing jobs rather than latency-sensitive online workloads that we're working on in this article. B. Aksanli, et al. have exploited two separate job arrival queues to respectively process interactive and batch workloads in datacenters [14]. Li, et al. [11] proposed an architecture where two sets of servers are respectively powered by grid power and wind farms. The servers migrate loads between them to match renewable energy supplies. GreenGear [18] utilizes the heterogeneity and workload schedule to match dispatchable solar power. GreenCassandra [42] guarantees that at least one replica of every data is always available by powering on servers of the covering subset. However, our GreenMatch adopts a more radical policy that powers off more servers, as in most cases just the hot data stored in the covering subset servers will be accessed. In addition, there are some papers aimed at workload scheduling across datacenters [21][22][23][24], while the GreenMatch aims at only a datacenter.

SSDs: cache and storage nodes. Flash memory solid-state disk (SSD) has emerged as an important media for mass storage [32]. Generally, most SSDs are used as cache for traditional HDDs, such as Sievestore [33], iTransformer [34], S4D-Cache [36] and iBridge [35]. Liu et al. [37] simulated a system using SSD storage on I/O nodes as buffer to handle burst I/O requests. SSD cache is also used in industry, such as flash cache [38], and Fusion-io [39]. It is proven that SSD cache can improve storage performance for most applications. In addition, treating SSDs as persistent storage device to make up a hybrid storage is another popular way, such as I-CASH [40] and Hystor [41]. But how to effectively integrate SSDs into green energy powered storage systems is still unsettled.

8 Conclusion

For disk-based storage systems mainly powered by green energy, the workload-driven schemes miss the opportunity to match green energy supplies while supply-following schemes degrade performance for latency-sensitive workloads. This paper proposes a scheme called GreenMatch, which deploys an SSD cache to match green energy supplies with a time-shifting workload schedule while maintaining low latency for online data-intensive services. We also exploited a novel cache management policy to reduce latency and guarantee data availability. Compared with a grid-only storage system without power management, GreenMatch achieves up to 97.5% green energy utilization, and reduces grid energy consumption by up to 97.54%. Compared with an SSD cache-based storage system, GreenMatch enables a significant upgrade of latency-sensitive services to match energy demands with green energy supplies.

Acknowledgments

This work was sponsored in part by the National Natural Science Foundation of China under Grant No.61472152, NO.61300047, No.61432007, and No.61572209; the 111 Project (No.B07038); the Director Fund of WNLO; the Key Laboratory of Data Storage System, Ministry of Education.

References

[1] A. Verma, R. Koller, L. Useche, and R. Rangaswami, "SRCMap: Energy proportional storage using dynamic consolidation," in *Proc. of the File and Storage Sys.*, 2010, pp. 267-280.

[2] L. A .Barroso, J. Clidaras, and U. H, "The datacenter as a computer: An introduction to the design of warehouse-scale machines," *Synthesis lectures on computer architecture*, 8(3), 2013, pp. 1-154.

[3] G. Cook, T. Dowdall, D. Pomerantz, and Y. Wang, "Clicking clean: how companies are creating the green internet," *Greenpeace Inc*, Washington, DC, 2014.

[4] H. Amur, J. Cipar, V. Gupta, G. R. Ganger, M. A. Kozuch, and K. Schwan, "Robust and flexible power-proportional storage," in *Proc. of the 1st ACM symp. on Cloud computing*, 2010, pp. 217-228.

[5] E. Thereska, A.Donnelly, and D. Narayanan, "Sierra: practical power-proportionality for data center storage," in *Proc. of the sixth conf. on Compu. syst.*, 2010, pp. 169-182.

[6] J. Wan, X. Qu, J. Wang, and C. Xie, "ThinRAID: Thinning down RAID array for energy conservation," *IEEE Trans. Parallel Distrib. Syst.*, 2015, 26(10): pp. 2903-2915.

[7] Western Wind and Solar Integration Study. [Online]. Available: http://www.nrel.gov/electricity/transmission/western_wind.html

[8] MSR Cambridge Traces. [Online]. Available: http://iotta.snia.org/traces/388.

[9] Sheepdog. [Online]. Available: http://sheepdog.github.io/sheepdog/.

[10] Z. Liu, Y. Chen, C. Bash, A. Wierman, D. Gmach, Z. Wang, and C. Hyser,"Renewable and cooling aware workload management for sustainable data centers," *ACM SIGMETRICS Performance Evaluation Review*, Vol. 40, No. 1, pp. 175-186, 2012.

[11] C. Li, A. Qouneh, and T. Li, "iSwitch: coordinating and optimizing renewable energy powered server clusters," in *Proc. of 39th Int. Symp. on Compu. Arch.*, pp. 512-523, 2012.

[12] N. Sharma, S. Barker, D. Irwin, and P. Shenoy, "Blink: managing server clusters on intermittent power," *ACM SIGPLAN Notices*, Vol. 46, No. 3, pp. 185-198, 2011.

[13] I. Goiri, W. A. Katsak, K. Le, T. D. Nguyen, and R. Bianchini, "Parasol and greenswitch: Managing datacenters powered by renewable energy," *ACM SIGARCH Compu. Arch. News*, Vol. 41, No. 1, pp. 51-64, 2013.

[14] B. Aksanli, J. Venkatesh, L. Zhang, and T. Rosing, "Utilizing green energy prediction to schedule mixed batch and service jobs in data centers," *ACM SIGOPS Operating Systems Review*, 45(3), 2011, pp. 53-57.

[15] I. Goiri, K. Le, T. D. Nguyen, J. Guitart, J. Torres, and R. Bianchini, "GreenHadoop: leveraging green energy in data-processing frameworks," in *Proc. of 2012 ACM European Conf. Compu. Syst.*, 2012, pp. 57-70.

[16] I. Goiri, K. Le, M.E. Haque, R. Beauchea, T.D. Nguyen, J. Guitart, J. Torres, and R. Bianchini, "GreenSlot: Scheduling Energy Consumption in Green Datacenters," in *Proc. of 2011 Int. Conf. for High Performance Computing, Networking, Storage and Analysis*, Seattle, Washington, 2011, pp. 20:1-0:11.

[17] C. Li, W. Zhang, C. B. Cho, and T. Li, "SolarCore: Solar energy driven multi-core architecture power management," in *Proc. of IEEE 17th Int. Symp. High Performance Computer Architecture (HPCA)*, pp. 205-216, 2011.

[18] X. Zhou, H. Cai, Q. Cao, H. Jiang, L. Tian, and C. Xie, "GreenGear: Leveraging and Managing Server Heterogeneity for Improving Energy Efficiency in Green Data Centers," in *Proc. of the 2016 Int. Conf. on Supercomputing*, 2016, pp. 12:1-12:14.

[19] D. Gmach, J. Rolia, C. Bash, Y. Chen, T. Christian, A. Shah, R. K. Sharma, and Z. Wang, "Capacity planning and power management to exploit sustainable energy," in *Proc. of Int. Conf. Network and Service Management (CNSM)*, pp. 96-103, 2010.

[20] M. Brown, and J. Renau, "Rerack: Power simulation for data centers with renewable energy generation," *ACM SIGMETRICS Performance Evaluation Review*, 39(3), 2011, pp.77-81.

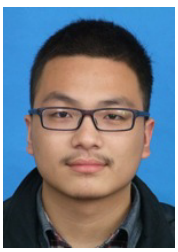
[21] K. Le, O.Bilgir, R. Bianchini, M. Martonosi, and T. D. Nguyen, "Managing the cost, energy consumption, and carbon footprint of internet services," *ACM SIGMETRICS Performance Evaluation Review*, Vol. 38, No. 1, pp. 357-358, 2010.

[22] Y. Zhang, Y. Wang, and X. Wang, "Greenware: Greening Cloud-scale Data Centers to Maximize the Use of Renewable Energy," in *Proc. Middleware*, Lisboa, Portugal, 2011, pp. 143-64.

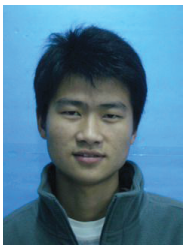
[23] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. H. Andrew, "Greening geographical load balancing," *Proc. ACM SIGMETRICS*, San Jose, CA, 2011, pp. 233-44.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPDS.2018.2820070, IEEE Transactions on Parallel and Distributed Systems

[24] Z. Xu, N. Deng, C. Stewart, and X. Wang, "CADRE: Carbon-Aware Data Replication for Geo-Diverse Services," in *Proc. of IEEE Int. Conf. on Autonomic Compu.*, pp. 177-186, 2015.
[25] X. Zhou, Q. Cao, H. Jiang, and C. Xie, "Underprovisioning the Grid Power Infrastructure for Green Datacenters," in *Proc. of the 29th ACM on Int. Conf. on Supercomputing*, pp. 229-240, 2015.
[26] C. Li, Z. Wang, X. Hou, H. Chen, X. Liang, and M. Guo, "Power Attack Defense: Securing Battery-Backed Data Centers," in *Proc. the 43rd ACM/IEEE Int. Symp. on Compu. Arch. (ISCA)*, pp. 493-505, 2016.
[27] C. Li, Y. Hu, L. Liu, J. Gu, M. Song, J. Yuan, X. Liang, and T. Li, "Towards Sustainable in-Situ Server Systems in the Big Data Era," in *Proc. the 42nd ACM/IEEE Int. Symp. on Compu. Arch. (ISCA)*, pp. 14-26, 2015.
[28] C. Li, Y. Hu, R. Zhou, M.Liu, L. Liu, J. Yuan, and T. Li, "Enabling Datacenter Servers to Scale Out Economically and Sustainably," in *Proc. the 46th ACM/IEEE Int. Symp. on Microarchitecture (MICRO)*, pp. 322-333, 2013.
[29] L. Liu, C. Li, H. Sun, Y. Hu, J. Gu, T. Li, J. Xin, and N. Zheng, "HEB: Deploying and Managing Hybrid Energy Buffers for Improving Datacenter Efficiency and Economy," in *Proc. the 42nd ACM/IEEE Int. Symp. on Comp. Arch. (ISCA)*, pp. 463-475, 2015.
[30] L. Liu, C. Li, H. Sun, Y. Hu, J. Gu, and T Li, "BAAT: Towards Dynamically Managing Battery Aging in Green Datacenters," *Ieee/ifip Int. Conf. on Dependable Syst. and Netw. IEEE Computer Society*, 2015, pp. 307-318.
[31] H.-I. Hsiao, and D. J. DeWitt, "Chained declustering: A new availability strategy for multiprocessor database machines," in *Proc. of Int. Conf. on Data Engineering* , pp. 456-465, 1990.
[32] F. Chen, R. Lee, and X. Zhang, "Essential roles of exploiting internal parallelism of flash memory based solid state drives in high-speed data processing," in *IEEE Int. Symp. on High PERFORMANCE Compu. Arch.*, pp. 266-277, 2011.
[33] T. Pritchett, and M. Thottethodi, "SieveStore: a highly-selective, ensemble-level disk cache for cost-performance", *ACM SIGARCH Compu. Arch. News*, Vol.38, No.3, pp.163-174, 2010.
[34] X. Zhang, K. Davis, and S. Jiang, "iTransformer: using SSD to improve disk scheduling for high-performance I/O," in *Proc. of 26th IEEE Int. Parallel and Distrib. Proc. Symp.*, Washington, DC, 2012, pp. 715-726.
[35] X. Zhang, K. Davis, and S. Jiang, "iBridge: improving unaligned parallel file access with solid-state drives", in *Proc. of 27th IEEE Int. Parallel and Distrib. Proc. Symp.*, Boston, MA, 2013, pp. 381-392.
[36] S. He, X.-H. Sun, and B. Feng, "S4D-Cache: Smart selective SSD cache for parallel I/O Systems," in *Proc. Int. Conf. Distrib. Comput. Syst.*, pp. 514-523, 2014
[37] N. Liu, J. Cope, P. Carns, C. Carothers, R. Ross, G. Grider, A. Crume, and C. Maltzahn, "On the role of burst buffers in leadership-class storage systems," in *Proc. of the IEEE 28th Symp. on Mass Storage Systems and Technologies*, Washington, DC, USA, 2012, pp. 1-11.
[38] Flash cache. [Online]. Available: https://github.com/facebookarchive/flashcache.
[39] Fusion-io. iocache. [Online]. Available: http://www.fusionio.com/products/iocache.
[40] Q. Yang, and J. Ren, "I-CASH: intelligently coupled array of SSD and HDD," in *Proc. of the IEEE 17th Int. Symp. on High Performance Compu. Arch.*, Washington, DC, USA, 2011, pp. 278-289.
[41] F. Chen, D. Koufaty, and X. Zhang, "Hystor: making the best use of solid state drives in high performance storage systems," in *Proc. of the Int. Conf. on Supercomputing*, New York, NY, USA, 2011, pp. 22-32.
[42] W. A. Katsak, I. Goiri, R. Bianchini, and Thu D. Nguyen, "Green-Cassandra: Using renewable energy in distributed structured storage systems," in *Proc. of the Sixth Int. Green and Sustainable Compu. Conf.(IGSC)*, 2015, pp. 1-8.



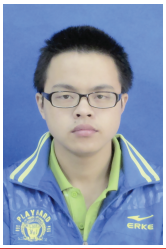
Daping Li received the bachelor's degree in information security from Huazhong University of Science and Technology, China, in 2014. He is currently working towards the PhD degree in computer science department at Huazhong University of Science and Techology. His research interests include computer architecture, parallel and distributed system.



Xiaoyang Qu received the bachelor's degree in software engineering from Fujian Normal University, China, in 2010, the MS degree in software engineering from Huazhong University of Science and Technology, China, in 2013. He is currently working towards the PhD degree in computer science department at Huazhong University of Science and Technology. His research interests include computer architecture, parallel and distributed system.

Jiguang Wan received the bachelor's degree in computer science from Zhengzhou University, China, in 1996, the MS and PhD degrees in computer science from Huazhong University of Science and Technology, China, in 2003 and 2007, respectively. His research interests include computer architecture, networked storage system, embedded system, parallel and distributed system.

Jun Wang received the PhD degree in computer science and engineering from the University of Cincinnati in 2002. He is a Charles N. Millican faculty fellow and an associate professor of computer engineering at the University of Central Florida, Orlando. He has authored more than 100 publications in premier journals such as IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems and leading HPC and systems conferences such as IPDPS, HPDC, EuroSys, ICS, Middleware, FAST. He has conducted extensive research in the areas of computer systems and high-performance computing. His specific research interests include massive storage and file System in local, distributed and parallel systems environment. He has graduated six PhD students who upon their graduations were employed by major US IT corporations (e.g., Google, Microsoft, etc). He serves as an associate editor for IEEE Transactions on Parallel and Distributed Systems and IEEE Transactions on Cloud Computing. He is the recipient of National Science Foundation Early Career Award 2009 and the Department of Energy Early Career Principal Investigator Award 2005. He is a senior member of the IEEE and the IEEE Computer Society.



Yang Xia received the bachelor's degree in computer science from Huazhong University of Science and Technology(HUST), China, in 2016. He is currently working towards the PhD degree at School of Computer Science and Engineering form Ohio State University, USA. His research interests include computer architecture, parallel and distributed system.

Xiaozhao Zhuang received the bachelor's degree in computer science from Huazhong University of Science and Technology(HUST), China, in 2015. He is currently working towards the MS degree in computer science department at Huazhong University of Science and Techology. His research interests include computer architecture networked storage system, parallel and distributed system.

Changsheng Xie received the BS and MS degrees in computer science from Huazhong University of Science and Technology(HUST), China, in 1982 and 1988, respectively. Presently, he is a professor in the Department of Computer Engineering at Huazhong University of Science and Technology. He is also the director of the Data Storage Systems Laboratory of HUST and the deputy director of the Wuhan National Laboratory for Optoelectronics. His research interests include computer architecture, disk I/O system, networked data storage system, and digital media technology. He is the vice chair of the expert committee of Storage Networking Industry Association (SNIA), China.