Scheduling Challenges for Variable Capacity Resources

Chaojie Zhang[1] and Andrew A. Chien[1,2]

[1] University of Chicago, Chicago IL, USA
[2] Argonne National Lab, Chicago IL, USA
chaojie@uchicago.edu,achien@cs.uchicago.edu

Abstract. Datacenter scheduling research often assumes resources as a constant quantity, but increasingly external factors shape capacity dynamically, and beyond the control of an operator. Based on emerging examples, we define a new, open research challenge: **the variable capacity resource scheduling problem**. The objective here is effective resource utilization despite sudden, perhaps large, changes in the available resources.

We define the problem, key dimensions of resource capacity variation, and give specific examples that arise from the natural world (carbon-content, power price, datacenter cooling, and more). Key dimensions of the resource capacity variation include dynamic range, frequency, and structure. With these dimensions, an empirical trace can be characterized, abstracting it from the many possible important real-world generators of variation.

Resource capacity variation can arise from many causes including weather, market prices, renewable energy, carbon emission targets, and internal dynamic power management constraints. We give examples of three different sources of variable capacity.

Finally, we show variable resource capacity presents new scheduling challenges. We show how variation can cause significant performance degradation in existing schedulers, with up to 60% goodput reduction. Further, initial results also show intelligent scheduling techniques can be helpful. These insights show the promise and opportunity for future scheduling studies on resource volatility.

Keywords: Resource variability, Data center, Batch scheduling, Power limits

1 Introduction

The extensive research studies on job scheduling and resource management generally focus on problems where the quantity of resources is fixed or constant. In this paper, we define a new, open research challenge: **the variable capacity resource scheduling problem**. That is, in data centers or clusters of the future it will be common to have variable capacity, and that capacity determined by external factors. Changing resource capacity is a challenge for job schedulers

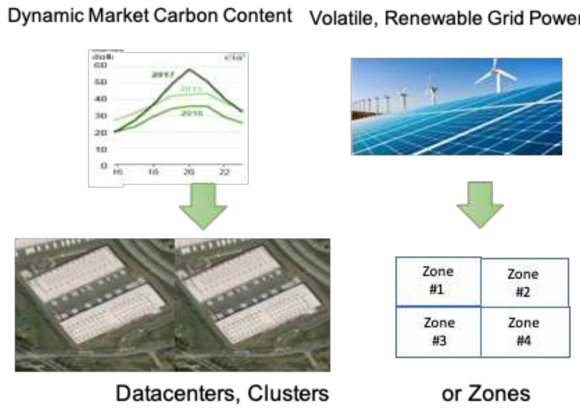


Fig. 1. Management to minimize carbon emissions or power cost combined with power grid, power markets, and renewable generation producing variable capacity. This is because changing power level directly affects the available computing resources [17, 34].

and resource managers because of the *uncertainty* about future resource capacity. On one hand, this means that even if job runtime is known at start time, the resources may not be available long enough to complete it. On the other hand, resources can increase rapidly, challenging the availability of workload to utilize them.

A wide variety of sources can produce variable resource capacities. For example, power limits are constraining the scale of world's largest supercomputers [6] and already define datacenter size. With the largest supercomputers approaching 50 megawatts, and predicted to grow well beyond 150 megawatts by 2025 [43]. These limits make dynamic power management for cost, cooling, sharing, or simply to be a good citizen in a fluctuating or stressed power grid a source of variable capacity for datacenters. At another level, carbon emission management can give rise to dynamic capacity. Concerned about climate, governments around the world have adopted policies to reduce carbon emissions whenever possible at the same time hyperscale cloud operators (e.g. Amazon, Microsoft, Google, etc.) are growing rapidly, accelerated further by exploding popularity of machine learning [19, 36]. This means that they must reduce datacenter power, perhaps on a dynamic basis in concert with use of renewable generation[27, 32, 29].

The importance of power and carbon as both a limit and a key cost has spawned a large and vibrant body of research on synergizing use and load with the grid (ZCCloud with renewables and low price [42, 8]) or with the availability of local renewables [16, 20, 12]. These approaches all suggest that future data centers will have variable capacity, determined by external factors such as the general (grid-wide) or local (on site) availability of renewable generated power.

Beyond power, there are a number of other scenarios where variable capacity is of interest for resource management. For example, a dependent cloud (a meta-cloud that forms its resource pool from spare resources of others) typically experiences frequent capacity change. One example of this would be the meta-cloud formed from a collection of AWS spot instances and Google's preemptible virtual machines. Another example source of variation might include partition-shutdowns for software upgrades, response to a security emergency, and so on. The latter examples may seem less compelling as they may perhaps be more controllable in theory. However, in practice they may not be controllable.

These varied scenarios suggest clusters, availability zones, scheduling domains, even entire data centers will have variable capacity, driven by external factors such as power allocation, market prices, or even general (grid-wide) or local (on-site) availability of renewable energy. This is the core motivation for the variable capacity resource scheduling problem. As shown in Figure 1, an external factor such as varying power creates variation in capability/capacity and the resource manager must effectively manage this varying capacity as it changes over time, as in Figure 2(b).

Today's resource management systems and schedulers generally assume full knowledge of resource capacity, and presume that it is stable going forward. While resource managers have dealt with the addition and removal of resources, these have typically been rare events with either unpredictable (failures) or simply structured (upgrade)[11]. Further, these are typically small-scale compared to cluster size. In contrast, many of the sources of variation we consider are continually varying, have complex correlation with external factors (e.g. weather), and have large-scale effect on cluster resources. It is not known how to achieve high goodput (useful throughput) in the face of continual resource capacity variability.

To define the problem, in subsequent sections we first define the key dimensions of resource capacity variation. With this framework of dynamic range, frequency, structure, and foresight in place, an empirical trace can be characterized, abstracting it as a generic problem. Second, we give several specific examples in the natural world (carbon-content, power price, datacenter cooling, and more) that give rise to variation. We illustrate how varied and challenging these examples are. Third, we present simulation results that show that variable resource capacity presents new scheduling challenges. Without change, current schedulers suffer significant performance loss, up to 60% goodput degradation. Finally, we present initial studies which show that intelligent scheduling techniques can be helpful.

Specific contributions of the paper include:

- Formal definition of a new scheduling problem, variable capacity resource management in datacenters
- Examples of and empirical traces of sources that lead to resource capacity variability
- Study of variable capacity that show today's schedulers suffer significant performance degradation

– Study of scheduler improvements shows that intelligent scheduling techniques are promising in regaining performance loss.

The rest of the paper is organized as follows. In Section 2 we formally define the scheduling problem of variable resource capacity. In Section 3, we discuss some empirical examples and cover metrics in Section 4. In Section 5, simulation results show how resource variability impacts scheduler performance and scheduling techniques that can mitigate performance degradation. We discuss some future directions and opportunities in Section 6 and related work in Section 7. Finally, we summarize in Section 8.

2 Scheduling Problem with Resource Capacity Variations

2.1 Scheduling Problem Definition

We formally state the job scheduling problem as follows. In a data center or cluster, let M denote the number of total machines, where each machine m has $r(m)$ resources. We want to schedule a set of jobs J on M machines. Each job $j \in J$ has submission time $s(j)$, resource requirement $r(j)$ and execution time $t(j)$. The data centers need to decide j_{mt}, which is the decision variable of running job j on machine m at time t. In traditional systems, such placements are subject to each machine's resource constraint:

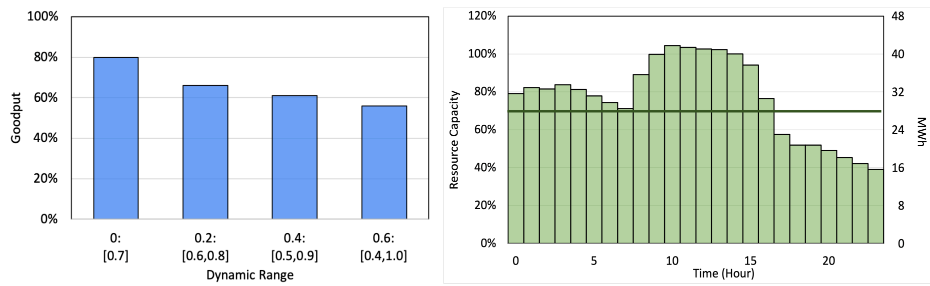
$$\forall t \in T, \forall m \in M, \sum_{j \in J} j_{mt} \times r(j) \leq r(m) \tag{1}$$

where the left hand side calculates the number of active resources that are processing jobs on each machine.

However in the new scheduling problem with resource capacity variations, the available resource capacity is a function of time t, denoted as $R(t)$ where $R(t) \leq M$. Hence, all job placements are now subject to a time-varying resource capacity constraint at each time slot t:

$$\begin{gathered}
\forall t \in T, \forall m \in M, \sum_{j \in J} j_{mt} \times r(j) \leq r(m) \\
\text{subject to} \\
u_{mt} = 1 \iff \exists j \in J \; s.t. \; j_{mt} = 1 \\
\sum_{m \in M} u_{mt} \times r(m) \leq R(t)
\end{gathered} \tag{2}$$

This constraint ensures that the total number of machines which have active running jobs do not exceed current resource capacity $R(t)$, where u_{mt} indicates whether a machine is active or not.



(a) Goodput (b) Hourly Capacity

Fig. 2. Scheduler goodput for a batch HPC workload under variable capacity (a); as dynamic range increases, performance degrades. Example of hourly capacity variation (b), assuming with enough capacity headroom.

2.2 Challenges of Job Scheduling

When resource capacity varies, even if the average capacity does not change, significant losses in system goodput(useful resource utilization based on total available resources) can result. In Figure 2(a), we present the resulting system goodput under dynamic capacity, even when a state-of-the-art scheduler [9] is used! As the dynamic range of variation increases from 0 to 0.6 (around an average capacity of 0.7), goodput decreases by 30%. Results are shown for capacity variability with random walk structure with stepsize of one-fourth the dynamic range. Figure 2(b) shows an example of capacity variation based on constant hourly carbon emissions from the Germany electricity market on 12.03.2020[15]. The quantity of compute resources available $R(t)$ can vary significantly and on short time scales compared to job runtimes.

What accounts for this degradation in goodput? Traditional schedulers assume constant resource capacity. Based on the assumption that current capacity will continue, these schedulers make decisions that commit resources into the future. Because they have been designed to maximize goodput, they strive to fill as much of this capacity as possible. So if resource capacity decreases, expressed as $R(t) < R(t-1)$, the schedule reflects an overestimate, and the resource capacity constraint in Equation 2 can be violated. This results in that some scheduled jobs may have to be terminated (fail) to release the machine. If resource capacity increases, the situation is a little better. No jobs need to be disturbed, but the schedule reflects an underestimation, and the scheduler has missed an opportunity to increase goodput.

In this new world, key open research questions include:

1. How do current schedulers respond to capacity variation?
2. How can scheduler performance be improved in these challenging situations?
3. How should we best limit or shape capacity variability for performance and other benefits?

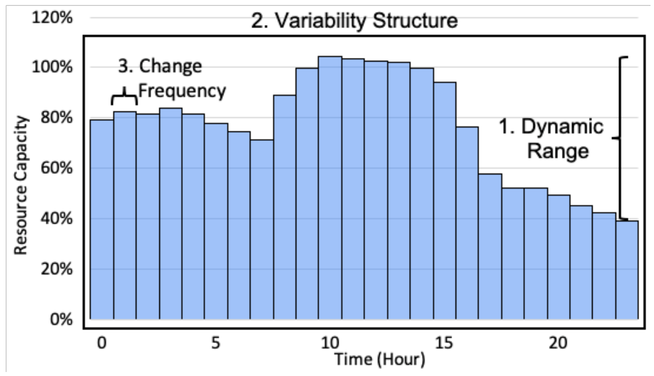


Fig. 3. Modeled dimensions of capacity variation include (1) dynamic range, (2) variability structure and (3) change frequency (temporal granularity) on a time-sequence of datacenter capacity from Figure 2(b).

2.3 Approach

To characterize the challenge to conventional schedulers under dynamic resource capacity, we study workloads and schedulers drawn from both HPC and commercial environments. These workloads are well-known exemplars of their respective environments. For each workload, we use a system model that varies the resource capacity available to the scheduler and evaluate performance. Constant resources is a simple model; variable resources can have many different dimensions of variation. We consider three:

– Dynamic range: minimum to maximum capacity
– Variability Structure: random uniform, random walk
– Change Frequency: frequency of capacity variation

We consider these key dimensions as abstract framework, where specific examples can be characterized and generalized. Dynamic range captures the distance over which resource capacity varies – from a low to high watermark and back. It is the most foundational element of resource capacity change. Variability structure reflects how capacity is constrained to change from one time period to the next. Such constraints often reflect the realities of physical systems - inductance, momentum, inertia and more – that prevent large instantaneous change. Change frequency reflects our choice to model time discretely – capacity varies only at time period boundaries – so change frequency reflects the size of those periods. In a real system, periods could be defined by external structures (power markets), datacenter physicals (cooling and power sharing control systems), or other factors.

Using these workloads and schedulers, we execute a set of scheduler experiments that explore this multi-dimensional capacity change space, characterizing scheduler performance. In effect, each experiment explores scheduler performance when actual resource capacity diverges from the scheduler's simple fixed

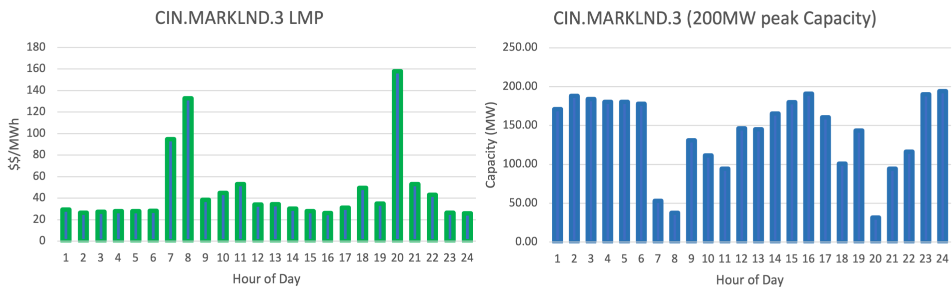


Fig. 4. Power price ($/MWh) (left) and resulting resource capacity for a 200 megawatt datacenter (right), using constant cost purchase approach. Exemplar 24-hour day from MISO January 9, 2018, CIN.Marklnd grid node.

estimate of stable resources. Our goal is to understand the capabilities of existing state-of-the-art schedulers. With a broad characterization of the negative impacts of capacity variation, we explore several scheduling ideas for how to mitigate performance degradation due to capacity variability.

3 Resource Capacity Variations from Empirical Traces

We focus on a few such factors that give rise to variable resource capacity and derive variable resource traces from them that can be used to evaluate scheduling systems. For each of these sources, we produce a set of sample traces of one-year duration with a variety of temporal resolutions (spanning 5 minutes to hourly). These exemplar capacity traces are generated based on several simple policies, e.g. constant (hourly) carbon budget.

3.1 Variation from Price

In order to manage a supply cost (e.g. power), a common strategy is to constrain expenditures to a constant rate for an operating period. In datacenters or many types of machinery, this couples dynamic market price to resource capacity as illustrated in Figure 4, showing capacity variation of 5-fold [0.2,1.0] or more. Variation can be large over time periods as short as 5-minutes, and with very low (even negative) prices variable capacity may be limited by physical capacity.

3.2 Variation from Carbon Emissions

Concern is increasing about climate change, and thereby associated carbon emissions with power consumption. Carbon budgets must be managed against power grids with large fluctuations in carbon content. A basic strategy is a constant carbon budget for each time period as shown in Figure 5. Carbon emissions often vary not only daily, but also with patterns that differ by month of the year. Note that workload SLOs such as “catchup by end of day” can have difficult interactions with the shape of variation curves.

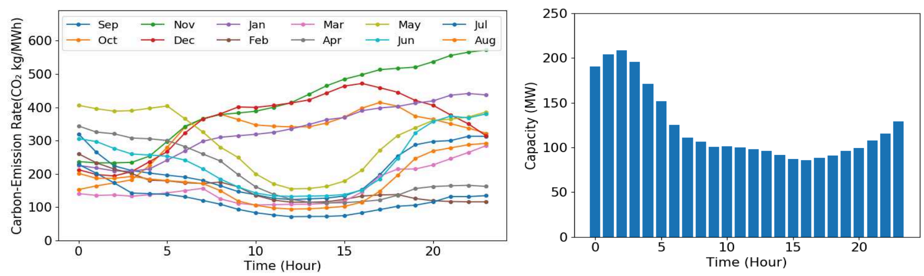


Fig. 5. Carbon-emissions rate (mT/MWh) (left) and resulting resource capacity at Constant Carbon purchase approach (December 2019, right).

3.3 Variation from Stranded Power

A different approach to lower carbon emissions is stranded renewable power [8, 42], where excess renewable energy (power with zero-marginal carbon) can be used to power datacenters intermittently. This excess case may be important for combatting climate [42, 43], and produces a nearly binary on-off resource capacity (Figure 6, ERCOT), while operating at zero carbon emissions. The graphs illustrate 15-minute intervals, and reflect variation over a weeklong period. The power availability variation is day-to-day, week-to-week, and also by season of the year.

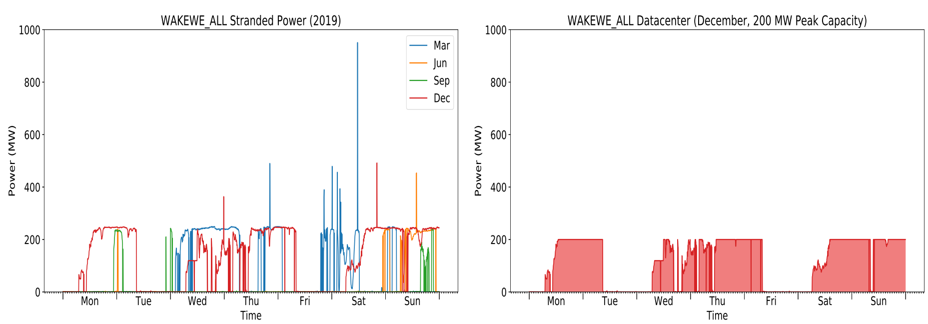


Fig. 6. Stranded Power (curtailed and negative priced power) in 15-minute intervals for a node in the ERCOT power grid (left, each line is a different week), and the resulting resource capacity for a 200 megawatt datacenter for the week in December (right).

4 Metrics

In this section, we discuss the metrics for resource capacity variation and measuring system performance.

4.1 Capacity Variation

Since resource capacity variation is produced by external sources, such as power prices, carbon footprint rates, and renewable generations, it can be viewed as a stochastic process. To better characterize and explore capacity variation, we look at three dimensions:

– **Dynamic range**: the range over which the resource capacity can vary. We define the lower and upper bound of resource capacity, expressed as $lbound, ubound$, as a fraction of the maximum datacenter capacity. Therefore, $R(t)$, the resource capacity at any time t, will be within the dynamic range, $lbound \leq R(t) \leq ubound$.
  We consider variation ranges of 0 (constant), 0.2, 0.4 and 0.6 as a fraction of maximum datacenter capacity. To normalize average capacity at 0.7, this produces dynamic ranges and intervals: 0: [0.7], 0.2: [0.6, 0.8], 0.4: [0.5, 0.9], and 0.6: [0.4, 1.0].
– **Variability Structure**: defines how much the capacity can change between adjacent time periods. *Random Uniform:* Resource capacity can be any level within the dynamic range at each interval and is drawn from a uniform distribution $\mathcal{U}([lbound, ubound])$, and *Walk:* Resource capacity can be any level within the dynamic range, but can only change by a maximum of *stepsize* in adjacent time intervals. Stepsize is one-fourth of the dynamic range.

$$R(t) = \begin{cases} \mathcal{U}([lbound, ubound]), & \text{if Random Uniform} \\ R(t-1) \pm stepsize, & \text{if Random Walk} \end{cases}$$

– **Temporal Granularity** represents the length of each time slot t. Between any time t and $t-1$, the capacity is constant. We vary the change frequency from 0.25 per hour (every 240 minutes) to 4 per hour (every 15 minutes).

4.2 Performance

Scheduling performance is measured by a group of widely-adopted metrics. Here we formally define these metrics which address system expectation and user experiences.

– **Goodput** is a measure of useful cluster utilization. It is calculated as total completed work divided by total available resource capacity: $\frac{\sum_{j \in J_{completed}} r(j)}{\sum_{t \in T} R(t)}$.
– **Failure Rate** represents the percentage of jobs that fail to complete due to resource capacity changes. It is calculated as $\frac{|J_{failed}|}{|J_{completed}|+|J_{failed}|}$.
– **Average Job Wait Time** measures the average of interval between job arrival time in the queue and job start time, which can be expressed as $\frac{\sum_{j \in J} START_j - ARRIVAL_j}{|J|}$.
– **SLO Miss Rate** represents the percentage of jobs that fail to complete before Service-Level-Objective (SLO) required deadline. For each job j, SLO miss $SM(j)$ is true if $FINISH_j - ARRIVAL_j - t(j) \geq X\% \times t(j)$, where X%

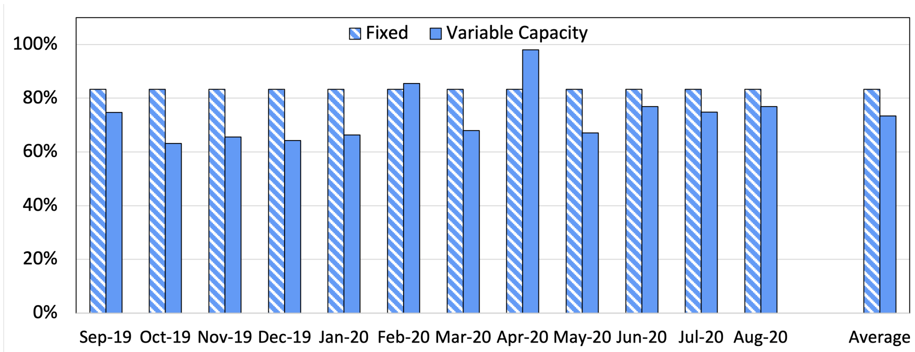


Fig. 7. Goodput for 12 exemplar days, comparing fixed and variable capacity.

is a threshold and usually set to 10%. The total SLO miss rate is therefore calculated as $\frac{\sum_{j \in J} SM(j)}{|J|}$.

There are many other widely-used metrics targeting different goals, such as response time and slowdown for cloud workloads and scheduling fairness. In addition to metrics, there are also various constraints that a system must consider. For example, "catch-up" constraint that bound the maximum start time of jobs, and hardware constraints that limit system's ramping capabilities or headroom limits that constrain system's maximum capacity.

5 Example studies of Variable Resource Capacity Data Centers

5.1 Experiment methodology

We considered a variety of publicly available workloads. While all of them are relevant and useful to study, we pick a few examplars that are widely-studied with distinct characteristics to understand new scheduling challenges. We use a month-long production trace from ALCF/Mira with a full range of job runtimes and large parallelism as the exemplar of large-scale HPC workload[5]. We pick Azure[10], Borg V2 traces [33] as node-sharing commercial cloud workload. Compared to Azure, the Borg trace has more small and short jobs, as well as significant load from long-running jobs.

For the Mira workload, we study the corresponding Cobalt[9] scheduler with the Mira supercomputer, a 10-petaflops IBM Blue Gene/Q system, deployed at the Argonne Leadership Computing Facility. Mira contains 49,152 nodes (786,432 cores) and 760 TB memory [28]. We model an Azure commercial cluster with 1,250 nodes (20,000 cores) and 160 TB of memory. This system is a close match in scale in resource utilization to the Mira system. We also model a Borg cluster with 630 nodes (336 GCU - Google-Compute-Unit) and 300 normalized bytes of memory. This system is sized to match the sampled Borg V2 trace used. Both cloud clusters use a FCFS first-fit scheduling policy.

5.2 Impact of Capacity Variation Dimensions

To illustrate the impact of variable resource capacity on scheduling performance in a real-world scenario, we consider a hypothetical 40-megawatt datacenter, which dynamically acquires power and resource capacity based on carbon emission rate, operating in the German Power Market[15]. Because the power market varies every day, and has a strong seasonal structure, we pick a set of exemplar days from the 12 most recent months (Sept 2019 - August 2020). When using constant carbon emissions per hour, they have power variation such as shown in Figure 5. These twelve days have 24-hour capacity increases from 6% to 16% with an average of 11%.

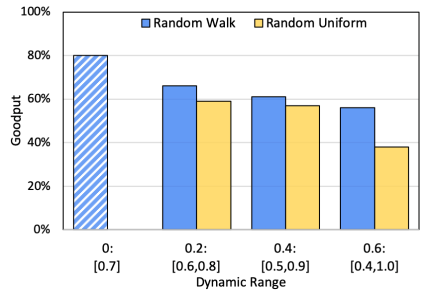
We use an HPC Mira workload, Mira system, and Cobalt HPC schedulers. For reference, we include a baseline mode (fixed power), comparing the variable capacity resulting from constant hourly carbon emissions, and showing the resulting goodput in Figure 7. Each blue bar depicts the results for a single exemplar day. Shifting from fixed to variable capacity produces a large drop in goodput as large at 24% on some days and 12% on average.

To further understand the impact on scheduling performance, we systematically vary the variability dimensions of dynamic range, structure, and change frequency while keeping average available capacity constant to understand how features of capacity variation affect scheduler performance, so we can highlight what is most important to address with scheduling techniques.

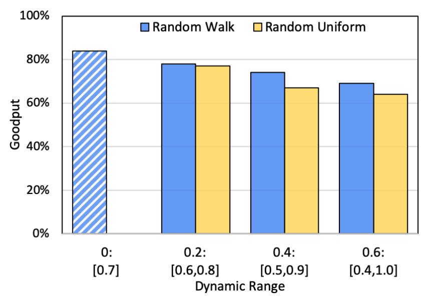
Dynamic Range First, let's consider how resource capacity variation impact varies as we increase dynamic range. In Figure 8, we first consider random walk structure (blue, left), comparing to no variation (patterned). The x-axis shows different dynamic ranges, and stepsizes are always one-fourth of the dynamic range. As the dynamic range increases, the scheduler performance degrades, and with the largest range, 0.6: [0.4,1.0], the goodput has declined by 25-45%.

Variability Structure We consider two variability structures, random walk and random uniform. Now we compare random uniform (yellow, right in Figure 8). The resource schedulers experience goodput degradation as much as 35% (for a total degradation of 55%). This is because random uniform allows large jumps in capacity, disrupting the job schedule with terminations or wasted resources. It appears variation structure can be as important as dynamic range in degrading scheduler performance.

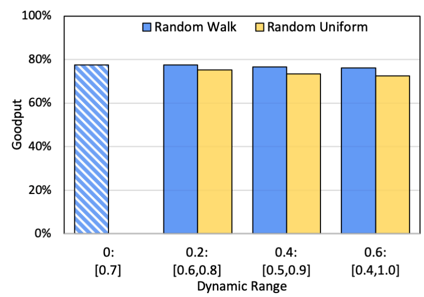
Change Frequency Change frequency is another dimension of capacity variation, so we start with a low rate (0.25 changes/hour), and increase to a high rate (4 changes/hour). Note that all prior experiments used a change frequency



(a) HPC



(b) Azure



(c) Borg

Fig. 8. Scheduling performance with random walk and random uniform resource variability structure, varying dynamic range.

of 1 change/hour. We focus on dynamic range of 0.6: [0.4, 1.0] with stepsize of 0.15 first. In Figure 9, significant goodput drop is observed across all structures and workloads as frequency increases. For HPC workload, goodput has fallen by as much as 50%. For Azure workload, higher change frequencies cause clear degradation in goodput (up to 30% overall, but 15% attributable to frequency); Borg V2 exhibits clear, but lesser degradation. These commercial workloads are less sensitive to resource variation because of their lower parallelism and shorter duration.

We combine change frequency with the other parameters (dynamic range and structure), putting it all together in Figure 10. With very low change frequency of 0.25 changes/hour, performance approaches the fixed capacity case. The negative impact of increasing change frequency on goodput remains but less extreme across all dynamic ranges.

We find that resource capacity variation can have a large impact on goodput, reducing it by up to 60%. Goodput in HPC and both commercial resource models are particularly sensitive to dynamic range, structure (and stepsize), and change frequency.

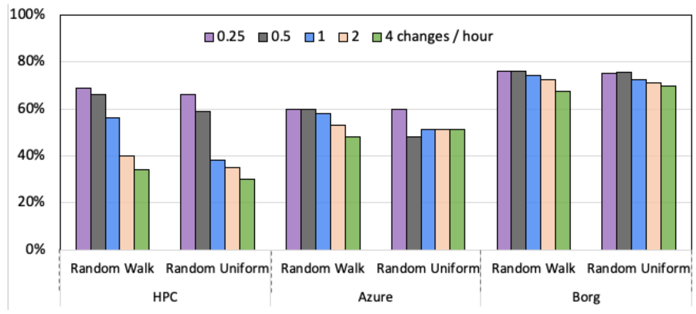


Fig. 9. Goodput versus change frequency (dynamic range 0.6: [0.4, 1.0]).

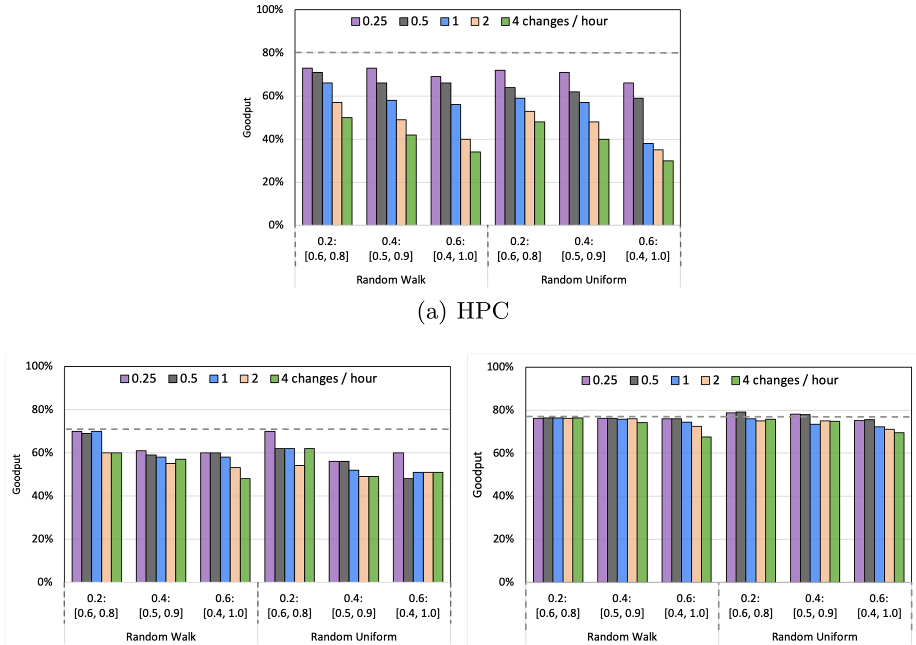
5.3 Scheduling Potential for Improvement

To show there is opportunity, we examine scheduling policies to mitigate performance degradation from capacity variation. When capacity decreases below the scheduled workload, to meet the capacity constraint, jobs must be terminated (fail). We explore how to choose the jobs for termination with the goal of maximizing goodput. Selective terimination or preemption is frequently adapted while facing mis-estimates based on priority or resource consumption[38, 30]. Here we consider three policies:

- **Random**: Select a node randomly, terminate the associated job, and free its resources.
- **Least Wasted Work (LWW)**: Select the job whose termination wastes least work (smallest $nodes \times (t - start\ time)$, where t is the current time) and free its resources.
- **Least Fraction Done (LFD)**: Terminate the job which is least fraction completed (minimum $\frac{(t - start\ time)}{runtime_j}$, where t is the current time) and free its resources.

For each policy, we repeat until the desired (lower) resource level is reached. For the HPC workloads, we use the requested runtime to compute LFD; for the commercial workloads we use the trace information for actual job length. However in production, this information is not generally available. We compare the termination policies, using scheduler performance metrics of goodput and failure rate.

Broadly, Figure 11 presents goodput results for a variety of dynamic ranges and variability structures. The results show that intelligent termination policies make a big difference. For HPC both intelligent termination algorithms improve performance, but best performance is achieved with LWW (rightmost, gray). The goodput achieved by LWW approaches the stable resource capacity, and is an average of 44% improvement over Random. For Azure and Borg V2 workloads, the algorithm preference is similar, with LWW producing highest goodput, but with smaller benefits.



(a) HPC

(b) Azure

(c) Borg

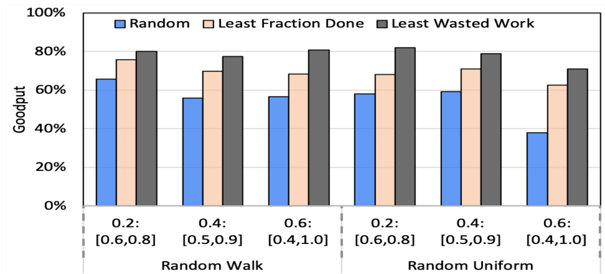
Fig. 10. Goodput versus change frequency, varying dynamic range and structure of capacity variation.

These policies show that scheduling strategies can provide improvement, and in this case increase performance to match the fixed-resource scenario (no variation), increasing goodput by 30% on average. These results show that intelligent scheduling techniques are of interest in variable capacity data centers.

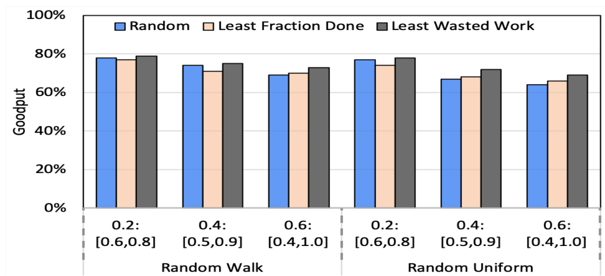
6 Further Directions and Opportunities

While we have outlined the core aspect of the open scheduling problem variable capacity, where resource capacity changes under external control on time scales shorter than many scheduled jobs. There are several dimensions that significantly broaden the space of interesting research.

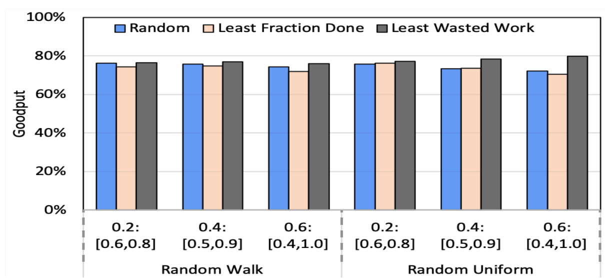
Complex SLO requirements Many workloads have complex dependencies amongst jobs and tasks that constrain scheduling, and correlate task failures [37]. Complex dependence structures make variable capacity scheduling challenging. Further, service-level objectives for jobs and tasks create further constraints on scheduling and opportunities for improvement. For example, a 24-hour time shifting model might have an asymmetic “catch-up” constraint.



(a) HPC



(b) Azure



(c) Borg

Fig. 11. Goodput versus termination policy, varying dynamic ranges and structures

More Sources of Variability and Correlation Another dimension of challenges comes from different sources of variability. Beyond power management, weather can produce variation in time and space, power availability and cooling efficiency (external temperature or humidity). Variation can be correlated across space and time – cloud cover can be correlated with weather, affecting solar and wind and temperature. Power grid element failures have correlated, cascading effects, and load changes can spill over from one cloud network to another. Unlike local failures, resource capacity variations coming from various external factors can be informed, estimated, or predicted through other correlated information. Power grid carbon-content can be correlated with price, and power availability can depend on competition.

Resource Heterogeneity The addition of heterogeneity to the variable capacity problem creates new challenges. Variants include fixed ratios, dictated ratios, partially controlled, or even fully controlled ratios of each type as capacity changes. All of these problems represent interesting challenges, both creating more complex and changing scheduling problems or in some cases added new critical decisions such as to invest the power relative to the potential heterogeneity.

Complex external metrics; e.g. overall cost optimization, overall carbon optimization One more additional dimension is the notion that metrics might depend on the input metrics that cause variation. Such a dependence not only affects the assessment of success, but therefore may also affect the scheduling strategies used. For example, power costs might be passed through to cloud users, and likewise responsibilty for carbon-emissions. Combinations of these metrics combined with traditional time-based techniques – 5s at high carbon, but overnight latency at low carbon – might make sense for some applications.

7 Related Work

We study resource management for both supercomputer and datacenter scenarios responding to capacity changes that could arise from carbon-emission-aware dynamic power acquisition. Other potential sources of resource capacity variation include cluster, datacenter, and site power management [34] or power grid dynamics [42, 22, 8]. While many other scheduling studies have also dealt with variations and uncertainty, they mainly focus on fluctuation of the load and job information[35, 14, 18]. It is an open question how well these techniques apply to the variation that is our focus, and perhaps more interesting if they can be adapted to cope better by exploiting the properties of the variation.

Burstable Instances and Turbo Modes In several cloud environments, virtual machines can have variable performance [1], but the resource consumption is typically controlled by the application. Bursting credit is accumulated over time and expended as the application demands. Turbo modes are similar, where heat capacity is akin to credit. This differs from variable capacity where resource constraint is enforced on the workload/resource manager.

Resource Revocation Many systems have volatile resource management (e.g. PC's in desktop grids [26, 7], and more recently AWS Spot Instances[3] and Google Preemptible VM's[4]), employing checkpointing and a range of statistical techniques to achieve high throughput through revocations [39, 40]. Commercial versions include [2, 21]. Most of these systems are application-oriented, and deal with collections of single-node jobs. The capacity variation problem is large-scale resource-oriented, and formulated for a job scheduler managing a workload with complex mixes of co-run, run-before, and other kinds of task dependencies in the face of a rich set of service-level objectives (SLO's).

Meta-schedulers There have been some systems that do this, but they typically manage batch queue delay (Condor Glideins with known durations). These systems schedule revocable resources, but the focus has traditionally been on managing across several resource pools and assumes new resources can be immediately obtained while revocations happen, not the scheduling efficiency within one (our focus here).

Power Capping and Large-scale Power Management Power capping generally limit power, a fixed capacity. Then, the challenge is managing the performance of the applications within a fixed cap [13, 31, 23]. Large-scale power management, power oversubscription, and power capping is common in commercial datacenters to improve power efficiency (e.g. Facebook's Dynamo[41], IBM's CapMaestro[25], and Google [34]). These studies do not model schedulers, and interestingly suggests that smaller and therefore more variable power pools may be preferrable, suggesting variable capacity.

Green Scheduling Researchers have also explored the use of local renewables or integration of grid demand-response with job scheduling [16, 24]. Local renewables are a simpler instance of the variable capacity scheduling problem – many variants exist. The grid demand-response examples are also related – but deal with rare circumstances (e.g. 4 hours a year). Our formulation of the variable capacity problem admits a rich, general externally imposed variation. It can vary at many time scales, with correlation or dependence across sites, and focuses on typical performance, but could perhaps include rare events.

8 Summary

We have proposed a new scheduling challenge: the variable resource capacity scheduling problem. We have defined the key dimensions (dynamic range, frequency, and structure) of resource capacity variations and provided empirical traces of such variation from several real-world scenarios. Using real HPC and commercial workloads, our results show that the negative impact of resource variability on goodput can be severe – as much as 60%, and 30% on average.

Further, we find that intelligent scheduling techniques such as job termination policies can reduce goodput losses for both workloads. These results not only show that variable resource capacity imposes new challenges, but also suggest that intelligent scheduling solution is of benefit. And we look forward to both exploring this space, and exploring the coupling of these studies with the complex systems which are also producing capacity changes [34, 22].

References

1. Amazon burstable instances (2019), https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/burstable-performance-instances.html

2. Amazon spot fleet. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/spot-fleet.html (2019)
3. Amazon spot instance (2019), https://aws.amazon.com/ec2/spot/
4. Google preemptible VMs. https://cloud.google.com/preemptible-vms/ (2019)
5. Allcock, W., Rich, P., Fan, Y., Lan, Z.: Experience and practice of batch scheduling on leadership supercomputers at Argonne. In: Workshop on Job Scheduling Strategies for Parallel Processing. pp. 1–24. Springer (2017)
6. Bourzac, K.: Supercomputing poised for a massive speed boost. Nature (November 2017), https://www.nature.com/articles/d41586-017-07523-y
7. Chien, A., Calder, B., Elbert, S., Bhatia, K.: Entropia: architecture and performance of an enterprise desktop grid system. Journal of Parallel and Distributed Computing **63**(5), 597–610 (2003)
8. Chien, A.A., Yang, F., Zhang, C.: Characterizing curtailed and uneconomic renewable power in the mid-continent independent system operator. AIMS Energy **6**(2), 376–401 (December 2018)
9. Cobalt: Component based lightweight toolkit. https://github.com/ido/cobalt (2019), iBM Commercial Job scheduler for Mira, JuQueen, and other Blue Gene systems
10. Cortez, E., Bonde, A., Muzio, A., Russinovich, M., Fontoura, M., Bianchini, R.: Resource central: Understanding and predicting workloads for improved resource management in large cloud platforms. In: Proceedings of the 26th Symposium on Operating Systems Principles. pp. 153–167. ACM (2017)
11. Curino, C., Krishnan, S., Karanasos, K., Rao, S., Fumarola, G.M., Huang, B., Chaliparambil, K., Suresh, A., Chen, Y., Heddaya, S., et al.: Hydra: a federated resource manager for data-center scale analytics. In: NSDI. pp. 177–192 (2019)
12. Derin, O., Ferrante, A.: Scheduling energy consumption with local renewable micro-generation and dynamic electricity prices. In: First Workshop on Green and Smart Embedded System Technology: Infrastructures, Methods, and Tools. vol. 12, pp. 1–6 (2010)
13. Ellsworth, D.A., Malony, A.D., Rountree, B., Schulz, M.: Dynamic power sharing for higher job throughput. In: SC '15: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. pp. 1–11 (2015)
14. Feitelson, D.G., Tsafrir, D., Krakov, D.: Experience with using the parallel workloads archive. Journal of Parallel and Distributed Computing **74**(10), 2967–2982 (2014)
15. Electricity market data: Generations, prices, power (2020), https://www.smard.de
16. Goiri, Í., Katsak, W., Le, K., Nguyen, T.D., Bianchini, R.: Parasol and greenswitch: Managing datacenters powered by renewable energy. In: ACM SIGPLAN Notices. vol. 48, pp. 51–64. ACM (2013)
17. Google: White paper: Moving toward 24x7 carbon-free energy at Google data centers: Progress and insights. Tech. rep., Google (October 2018)
18. Gupta, V., Harchol-Balter, M., Wolf, A.S., Yechiali, U.: Fundamental characteristics of queues with fluctuating load. In: Proceedings of the joint international conference on Measurement and modeling of computer systems. pp. 203–215 (2006)
19. Hao, K.: Training a single AI model can emit as much carbon as five cars in their lifetimes. Technology Review (June 2019)
20. Haque, M.E., Goiri, I., Bianchini, R., Nguyen, T.D.: Greenpar: Scheduling parallel high performance applications in green datacenters. In: Proceedings of the 29th ACM on International Conference on Supercomputing. pp. 217–227. ACM (2015)

21. Harlap, A., Chung, A., Tumanov, A., Ganger, G.R., Gibbons, P.B.: Tributary: spot-dancing for elastic services with latency slos. In: 2018 USENIX Annual Technical Conference (ATC'18). pp. 1–14 (2018)
22. Kim, K., Yang, F., Zavala, V., Chien, A.A.: Data centers as dispatchable loads to harness stranded power. IEEE Transactions on Sustainable Energy (2016)
23. Le, K., Bianchini, R., Zhang, J., Jaluria, Y., Meng, J., Nguyen, T.D.: Reducing electricity cost through virtual machine placement in high performance computing clouds. In: Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis. pp. 22:1–22:12. SC '11, ACM, New York, NY, USA (2011). https://doi.org/10.1145/2063384.2063413, http://doi.acm.org/10.1145/2063384.2063413
24. Le, T.N., Liu, Z., Chen, Y., Bash, C.: Joint capacity planning and operational management for sustainable data centers and demand response. In: Proceedings of the Seventh International Conference on Future Energy Systems. pp. 1–12 (2016)
25. Li, Y., Lefurgy, C.R., Rajamani, K., Allen-Ware, M.S., Silva, G.J., Heimsoth, D.D., Ghose, S., Mutlu, O.: A scalable priority-aware approach to managing data center server power. In: 2019 IEEE International Symposium on High Performance Computer Architecture (HPCA). pp. 701–714. IEEE (2019)
26. Litzkow, M.J., Livny, M., Mutka, M.W.: Condor-a hunter of idle workstations. Tech. rep., University of Wisconsin-Madison Department of Computer Sciences (1987)
27. Megerian, C., Panzar, J.: Gov. Brown signs climate change bill to spur renewable energy, efficiency standards. Los Angeles Times (9 2015)
28. Mira-ALCF: MIRA: A 10-Petaflop, 4 MW IBM Supercomputing at Argonne. https://www.alcf.anl.gov/mira (2019)
29. New York State Energy Planning Board: The energy to lead: 2015 New York state energy plan (2015), http://energyplan.ny.gov/Plans/2015.aspx
30. Park, J.W., Tumanov, A., Jiang, A., Kozuch, M.A., Ganger, G.R.: 3sigma: distribution-based cluster scheduling for runtime uncertainty. In: Proceedings of the Thirteenth EuroSys Conference. pp. 1–17 (2018)
31. Patel, T., Tiwari, D.: Perq: Fair and efficient power management of power-constrained large-scale computing systems. In: Proceedings of the 28th International Symposium on High-Performance Parallel and Distributed Computing. p. 171182. HPDC 19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3307681.3326607, https://doi.org/10.1145/3307681.3326607
32. Penn, I., Kang, I.: California today: A move to mandate 100% carbon-free electricity. New York Times (August 2018)
33. Reiss, C., Wilkes, J., Hellerstein, J.L.: Google cluster-usage traces: format+ schema. Google Inc., White Paper pp. 1–14 (2011)
34. Sakalkar, V., Kontorinis, V., Landhuis, D., Li, S., De Ronde, D., Blooming, T., Ramesh, A., Kennedy, J., Malone, C., Clidaras, J., et al.: Data center power oversubscription with a medium voltage power plane and priority-aware capping. In: Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems. pp. 497–511 (2020)
35. Schroeder, B., Harchol-Balter, M.: Web servers under overload: How scheduling can help. ACM Transactions on Internet Technology (TOIT) **6**(1), 20–52 (2006)
36. Strubell, E., Ganesh, A., McCallum, A.: Energy and policy considerations for deep learning in NLP. arXiv preprint arXiv:1906.02243 (2019)

37. Tirmazi, M., Barker, A., Deng, N., Haque, M.E., Qin, Z.G., Hand, S., Harchol-Balter, M., Wilkes, J.: Borg: the next generation. In: Proceedings of the Fifteenth European Conference on Computer Systems (2020)
38. Verma, A., Pedrosa, L., Korupolu, M., Oppenheimer, D., Tune, E., Wilkes, J.: Large-scale cluster management at google with borg. In: Proceedings of the Tenth European Conference on Computer Systems. p. 18. ACM (2015)
39. Wolski, R., Brevik, J.: Providing statistical reliability guarantees in the aws spot tier. In: Proceedings of the 24th High Performance Computing Symposium. pp. 1–9 (2016)
40. Wolski, R., Brevik, J., Chard, R., Chard, K.: Probabilistic guarantees of execution duration for amazon spot instances. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. ACM (2017)
41. Wu, Q., Deng, Q., Ganesh, L., Hsu, C.H., Jin, Y., Kumar, S., Li, B., Meza, J., Song, Y.J.: Dynamo: facebook's data center-wide power management system. ACM SIGARCH Computer Architecture News **44**(3), 469–480 (2016)
42. Yang, F., Chien, A.A.: Zccloud: Exploring wasted green power for high-performance computing. In: 2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS). pp. 1051–1060. IEEE (2016)
43. Yang, F., Chien, A.A.: Large-scale and extreme-scale computing with stranded green power: Opportunities and costs. IEEE Transactions on Parallel and Distributed Systems **29**(5) (December 2017)