Stochastic Programming Approach for Resource Selection under Demand Uncertainty

Tanveer Hossain Bhuiyan[1], Mahantesh Halappanavar[2], Ryan Friese[2], Hugh Medal[1], Luis de la Torre[3], Arun Sathanur[2], and Nathan Tallent[2]

[1]Mississippi State University [2]Pacific Northwest National Laboratory
[3]Washington State University
tb2038@msstate.edu, {FirstName.LastName}@pnnl.gov,
hmedal@ise.msstate.edu, luis.delatorre@wsu.edu

Abstract. Cost-efficient selection and scheduling of a subset of geographically distributed resources to meet the demands of a scientific workflow is a challenging problem. The problem is exacerbated by uncertainties in demand and availability of resources. In this paper, we present a stochastic optimization based framework for robust decision making in the selection of distributed resources over a planning horizon under demand uncertainty. We present a novel two-stage stochastic programming model for resource selection, and implement an L-shaped decomposition algorithm to solve this model. A Sample Average Approximation algorithm is integrated to enable stochastic optimization to solve problems with a large number of scenarios. Using the metric of stochastic solution, we demonstrate up to **30%** cost reduction relative to solutions without explicit consideration of demand uncertainty for a 24-month problem. We also demonstrate up to **54%** cost reduction relative to a previously developed solution for a 36-month problem. We further argue that the composition of resources selected is superior to solutions computed without explicit consideration of uncertainties. Given the importance of resource selection and scheduling of complex scientific workflows, especially in the context of commercial cloud computing, we believe that our novel stochastic programming framework will benefit many researchers as well as users of distributed computing resources.

1 Introduction

Scheduling of large-scale scientific workflows on geographically distributed resources is a challenging problem. Optimal selection of a subset of available resources to meet the projected demand is usually the first step in scheduling. Given a wide range of resources, from dedicated high-performance clusters to commercial cloud computing platforms, that are available to a scientific workflow, cost-efficient selection of resources is a challenging problem. Further, uncertainties not only in demand but also in the availability of resources exacerbates the problem. To address this problem, we present a stochastic programming based approach in this paper. Our goal is to compute cost-efficient selection of resources under demand uncertainties. We build on our prior work [6], where

we introduced the problem of resource selection under demand uncertainties. Here, we develop a stochastic programming based framework that significantly improves the quality of solutions.

Our study is motivated by complex workflows from the Belle II experiments, a high energy physics experiment to probe the interactions of fundamental constituents of our universe [9]. Computing and storage resources to support the Belle II experiments span several continents with users across the globe. Data is generated both from the Belle II detector and Monte Carlo simulations, and is expected to reach 350 peta bytes (a peta byte is 10^{15} bytes) by the end of the experiment in 2022. Complex workflows run across multiple computing and storage resources distributed worldwide. Multiple research and commercial cloud computing resources are also used. With a wide variety of user jobs and resource types, the Belle II experiment is an ideal case study to develop efficient solutions for scheduling of complex workflows.

Inspired by a resource selection problem in electric power grids, we model our problem as a *unit commitment* problem, where the goal is to meet a forecasted demand with a subset of resources at a minimum cost [8]. We describe the problem in §2. We then introduce the notion of uncertainties in demand, where the forecasted demand varies due to several factors. This formulation enables us to propose a *two-stage mixed-integer stochastic program* as an efficient solution technique. We detail the mathematical formulation in §3. Intuitively, stochastic programming is a mathematical programming technique for modeling optimization problems that involve uncertainties [3]. Stochastic programming can exploit the fact that probability distributions governing the data are known or can be estimated. For workflows with demand uncertainties, our goal is to develop a large set of scenarios of the forecasted demand, drawn from known probability distributions. Stochastic programming will compute solutions that are feasible for all scenarios and maximizes the expectation of an objective function. In the two-stage model, we make a resource selection decision in the first stage, after which each realization of the demand (a scenario) is considered that affects the outcome of the first-stage decision. A penalty is added for any unsatisfied demand from the second-stage, and the first-stage problem is adjusted accordingly. We employ the *Sample Average Approximation* (SAA) method as a sampling strategy to improve computational complexity, and use an *L-shaped decomposition algorithm* within the SAA procedure to solve the mixed-integer stochastic programming problem. We detail this approach in §4.

Using a carefully designed synthetic workflow inspired from the Belle II experiment, we present experimental evaluation of the proposed solution in §6. We demonstrate the superior quality of the proposed solution not only with respect to the previously developed method, but also with optimal solutions that are computed without explicit consideration of demand uncertainties. We demonstrate up to **30%** cost reduction relative to solutions without explicit consideration of uncertainty for a 24-month use case, and up to **54%** cost reduction relative to a Genetic Algorithm based solution for a 36-month use case. We further argue that the proposed solution method leads to resource compositions that are superior and robust to price fluctuations. To the best of our knowledge,

this is the first detailed work on employing stochastic programming approach for scheduling of complex workflows with demand uncertainties.

We make the following contributions in this paper:

- Develop a novel two-stage stochastic programming model for the allocation of distributed resources over a long range planning horizon with demand uncertainties to minimize the total expected cost.
- Implement and evaluate a stochastic optimization algorithm (L-shaped decomposition algorithm) to efficiently solve the proposed optimization problem.
- Integrate Sample Average Approximation method with the L-shaped decomposition algorithm to solve problems with continuous distribution of demand uncertainties with a large number of scenarios.
- Present numerical results to demonstrate the benefit of considering uncertainty in resource selection relative to a deterministic approach considering only the base demands (without uncertainty) in decision making.
- Present numerical results to demonstrate the robustness of the solution computed using a stochastic optimization approach relative to existing approaches.

2 Problem Description

Given a set of diverse computing resources with varying costs of usage, the objective is to compute the most cost-efficient subset of resources to meet the forecasted demand. This problem is analogous to the *unit commitment* problem in the context of electric power grid [19]. Unit commitment is a resource utilization problem where the objective is to select a subset of power generators at a minimum cost to satisfy a given demand that varies over time. Different power generators have different start-up and operation costs. Mathematically, the unit commitment problem can be formulated as shown in Equation 1, where, f is the total cost of the system for N power generators chosen to satisfy the demand over the planning horizon T. Variables S_j and C_j are the start-up and operating cost for each generator j respectively, to generate P_j units of power. The binary variable x_{jt} represents whether the generator j is on or off at time period t. The system also specifies a reserved demand, R_t, for every time period that is satisfied by the spinning reserve (spare capacity), r_t.

$$min\ f = \sum_{t=1}^{T}\sum_{j=1}^{N}(S_j x_{jt} + C_j P_{jt}) \tag{1a}$$

$$s.t.\ \sum_{j=1}^{N} P_{jt} \geq D_t \qquad \forall t = 1, .., T \tag{1b}$$

$$\sum_{j=1}^{N} r_{jt} \geq R_t \qquad \forall t = 1, .., T. \tag{1c}$$

In this paper, we introduce a similar resource utilization problem in the context of large-scale workflows, where we need to allocate geographically distributed computing resources to satisfy the demand over a planning horizon. The selection problem is pronounced in the context of cloud computing where different types of resources are available with varying cost structures. For example, Amazon EC2 offers several types of resources with different fixed (subscription) and usage (operating) costs (detailed in §5). Using a computing resource incurs two types of costs: A subscription cost and an operating cost. A resource can only be used within a given period that it has been subscribed for. There are three broad types of machine usage policies: Total-upfront, partial-upfront (hybrid), and on-demand. In total-upfront, a machine is subscribed or paid upfront for a contiguous block of time without an operating cost for its use. In partial-upfront, a machine is also subscribed for several contiguous time periods, but incur an operating cost for using the machine during those periods. On-demand machines do not require any subscription cost, but incur higher operating costs when used. We also assume a penalty cost for any unmet demand for a given period that can be considered as *spot pricing* in the cloud computing literature.

Our objective is to compute the minimum-cost allocation of resources to satisfy forecasted demands under uncertainty. The demand fluctuates significantly over a planing horizon relative to the forecasted baseline. Demand fluctuations are addressed in unit commitment through reserves, r_t. However, estimating and maintaining spare capacity at every time period is an expensive solution. We therefore develop a stochastic programming approach to address this problem. We will describe our approach in the following section.

3 Two-Stage Stochastic Programming Model

We formulate the resource selection problem under demand uncertainty as a two-stage mixed-integer stochastic programming model. In order to address uncertainties in demand, which can arise from several factors such as errors in planning and unforeseen circumstances, we construct specific demand scenarios by sampling from a continuous distribution of base demands over the horizon with estimated probability distribution functions (§4.1).

Each scenario represents a particular demand curve spanning the entire horizon. In the first stage of the two-stage programming model, subscription decisions are made before realizing the uncertainty. In other words, the machines are subscribed at the beginning of the planning horizon when the actual demand for each period is unknown to the decision maker. In contrast to the first-stage problem, the second-stage problem considers the uncertainties, where the decisions are made as to whether to use or not to use a machine that has already been subscribed for a given period of time. Feedback from the second-stage problem is used to improve the decision making in the first-stage problem. Our objective is to minimize the total subscription cost as well as the expected operating costs while satisfying the demand under uncertainties over the planning horizon. We detail the mathematical formulation in this section. Notations used in this paper

are summarized in Table 1.

$$min\ f(x) = \sum_{t=1}^{T}\sum_{j=1}^{N} S_j x_{jt} + E[Q(x,\omega)] \tag{2a}$$

$$s.t. \quad \sum_{t'=t}^{min\{T,t+(u1-1)\}} x_{jt'} \leq 1 \qquad \forall j \in J_{NS1}, \forall t = 1,..,T \tag{2b}$$

$$\sum_{t'=t}^{min\{T,t+(u2-1)\}} x_{jt'} \leq 1 \qquad \forall j \in J_{NS2}, \forall t = 1,..,T \tag{2c}$$

$$x_{jt} \in \{0,1\} \qquad \forall j \in J, \forall t = 1,..,T \tag{2d}$$

$$Q(\hat{x},\omega) = min \sum_{t=1}^{T}\sum_{j=1}^{N} C_j p_{jt}^{\omega} + \sum_{t=1}^{T} \lambda l_t^{\omega} \tag{3a}$$

$$s.t. \quad \sum_{j=1}^{N} P_j(p_{jt}^{\omega}) + l_t^{\omega} \geq d_t^{\omega} \qquad \forall t = 1,..,T \tag{3b}$$

$$P_{jt}^{\omega} \leq \hat{x}_{jt} \qquad \forall j \in J_{OND}, \forall t = 1,...,T \tag{3c}$$

$$P_{jt}^{\omega} \leq \sum_{t'=max\{1,t-(u1-1)\}}^{t} \hat{x}_{jt'} \qquad \forall j \in J_{NS1}, \forall t = 1,..,T \tag{3d}$$

$$P_{jt}^{\omega} \leq \sum_{t'=max\{1,t-(u2-1)\}}^{t} \hat{x}_{jt'} \qquad \forall j \in J_{NS2}, \forall t = 1,..,T \tag{3e}$$

$$0 \leq P_{jt}^{\omega} \leq 1 \qquad \forall j \in J, \forall t = 1,..,T \tag{3f}$$

The two-stage stochastic programming problem is formulated as follows. The objective function, Equation 2a, of the first-stage model is to subscribe (select) a set of machines for the entire planning horizon, such that the total subscription cost and the expected total operating cost are simultaneously minimized across all the scenarios. Constraints 2b and 2c ensure that if a machine is subscribed at a period t for p number of periods, the subscription costs are incurred only once. Constraint 2d models the binary nature of the subscription decision variables. The second-stage model stands for each realization of a randomly sampled demand scenario.

The objective function, Equation 3a, of the second-stage model minimizes the total operating cost over all periods for a given scenario. A penalty is added to the objective function for any unsatisfied demand. The objective function has two components. The first component computes the total operating cost of machines over the planning horizon T, and the second component computes the total penalty cost for unmet demand across T. Constraint 3b represents the demand to be satisfied for each time period. A variable is introduced to satisfy

Table 1: A summary of the notations used in this paper.

Notation	Description
Sets	
J	Set of machines, $j \in J$
J_{OND}	On-demand machines
J_{NS1}	12 month-subscription machines
J_{NS2}	36 month-subscription machines
Parameters	
u	No. of periods for which machine j is subscribed
N	No. of machines
d_t^ω	Demand at period t in scenario $\omega \in \Omega$
S_j	Subscription cost for machine j
C_j	Operating cost of machine j for a single period
T	Total planning horizon
λ	Penalty cost for unsatisfied demand
P_j	Computing power of machine j
Variables	
x_{jt}	1 if machine j is subscribed at period t, 0 otherwise
p_{jt}^ω	Fraction of period t machine j is used in scenario ω
l_t^ω	Computing power shortage in period t in scenario ω

the shortage in demand. Constraint 3c imposes the subscription requirement to use on-demand machines. In order to use a total-upfront or partial-upfront machine j for period t, that period should be within the range of periods p for which the machine has been subscribed in the first-stage model. This requirement is satisfied for machines with 12 months and 36 months subscription periods in Constraints 3d and 3e, respectively. Constraint 3f represents the bounds for usage of machines. We present an approach to efficiently solve the two-stage stochastic programming model in §4.

4 Solution Approaches

We now present our solution to the two-stage stochastic programming model described in Section 3. A particular challenge in the solution of this problem arises due to the difficulty in computing the expected operating cost in the first-stage objective function, 2a. For a given first-stage solution, we need to compute the expected operating cost over all the realizations of the uncertain demand. If we consider the distribution of the uncertain demand to be continuous for each period, the computation of the expected operating cost requires taking multiple integrals, which will be computationally challenging [18]. On the other hand, if we consider the demand distribution to be discrete, we will have a large number of scenarios (realizations) to consider. For example, consider a 36-month problem. If the demand for each period has 10 different discrete values, the total number of possible scenarios to be considered will be 10^{36}. Thus, the challenge will be to compute a large number of scenarios, and consequently, solve a large

number of linear programming problems corresponding to these scenarios which is computationally infeasible.

To reduce the computational complexity in solving the two-stage stochastic program with infinitely many scenarios, we implement a sampling strategy called the Sample Average Approximation (SAA) [15, 20]. SAA enables the solution of stochastic allocation problem with continuous distributions for demand uncertainties. We integrate SAA with the L-shaped decomposition algorithm [1, 21] to solve the two-stage problem efficiently. This approach is motivated by the success of our own work [2] and of other researchers [18] to solve similar problems in different domains. We briefly describe the integrated approach in this section.

4.1 Sample Average Approximation

We use Sample Average Approximation (SAA) to deal with the difficulty in computing the expectation in the objective function 2a. SAA approximates the expected cost component, $E[Q(x,\omega)]$, of the objective function by a sample average function, $\frac{1}{|\Omega|}\sum_{s=1}^{|\Omega|} Q(x,\omega^s)$. SAA generates a set of random samples $(\omega^1,\omega^2,...,\omega^{|\Omega|})$ of size $\mid \Omega \mid$, where Ω is the set of scenarios (realizations) indexed by ω. Thus, the original problem in Equation 2 is approximated as:

$$min_{x\in X}\, \hat{f}(x) := \sum_{t=1}^{T}\sum_{j=1}^{N} S_j x_{jt} + \frac{1}{\mid \Omega \mid}\sum_{s=1}^{|\Omega|} Q(x,\omega^s). \tag{4}$$

We denote the optimal solution and optimal objective value of the approximation problem (Equation 4) by $\hat{x}$ and V, respectively. Here, $\hat{x}$ and V are stochastic as they are computed based on random samples. As described by Kleywegt *et al.,* the values for $\hat{x}$ and V get closer to the optimal values and the objective value of the original problem, with a probability of approximately one, as the sample size increases [13]. Thus, with a moderately large sample size, SAA scheme provides relatively good solutions to the original problem.

Key steps of the SAA algorithm are as follows:

1. Set iteration count to zero; $SAALB$ to zero; $SAAUB$ to ∞; and, the optimality gap to α. Generate M independent samples of size $\mid \Omega \mid$ and solve the SAA problem (Equation 4) for each sample. For sample n, let $\hat{x}^n$ and V^n represent the optimal solution and the optimal objective value respectively.
2. Compute the average of the optimal objective values over all samples, $\bar{V}$, which provides a statistical lower bound of the optimal objective value of the original problem. The average $\bar{V}$ and its associated variance $\sigma^2_{\bar{V}}$ are computed as follows:

$$\bar{V} := \tfrac{1}{M}\textstyle\sum_{n=1}^{M}(V^n)$$
$$\sigma^2_{\bar{V}} := \tfrac{1}{(M-1)M}\textstyle\sum_{n=1}^{M}(V^n-\bar{V})^2$$

Update variable $SAALB = \bar{V}$.

3. Select a feasible solution $\bar{x}$ of the true problem from the solutions computed in Step 1. Generate an independent reference sample of size $\mid \Omega_R \mid$, much larger than the sample size used in computing the solutions $\hat{x}^n$. Using this reference sample and one of the feasible solutions, estimate the objective function value of the true problem as follows:

$$\hat{f}(\bar{x}) := \sum_{t=1}^{T} \sum_{j=1}^{N} S_j \bar{x_{jt}} + \frac{1}{|\Omega_R|} \sum_{s=1}^{|\Omega_R|} Q(\bar{x}, \omega^s)$$

Update the variable $SAAUB = \hat{f}(\bar{x})$. Usually, $\bar{x}$ chosen from $\hat{x}^n$ results in the smallest value for $\hat{f}(\bar{x})$. Variance of the estimate, $\hat{f}(\bar{x})$, can be computed using Equation 5.

4. If $(SAAUB - SAALB) \leq \alpha$ then go to next step. Else, go to Step 1.
5. Compute an estimate of the optimality gap and the associated variance as follows:

$$Gap := SAAUB - SAALB$$
$$\sigma^2_{gap} = \sigma^2(\bar{x}) + \sigma^2_{\bar{V}}.$$

$$\sigma^2(\bar{x}) := \frac{1}{(|\Omega_R|-1)|\Omega_R|} + \sum_{s=1}^{|\Omega_R|} \left(\sum_{t=1}^{T} \sum_{j=1}^{N} S_j \bar{x_{jt}} + Q(\bar{x}, \omega^s) - \hat{f}(\bar{x}) \right) \quad (5a)$$

$$\begin{aligned}
LB \;=\; & min \sum_{t=1}^{T} \sum_{j=1}^{N} S_j x_{jt} + \theta && (6a)\\
s.t. \quad & \sum_{t'=t}^{min\{T, t+(u1-1)\}} x_{jt'} \leq 1 \qquad \forall j \in J_{NS1}, \forall t \in T && (6b)\\
& \sum_{t'=t}^{min\{T, t+(u2-1)\}} x_{jt'} \leq 1 \qquad \forall j \in J_{NS2}, \forall t = 1, .., T && (6c)\\
& \theta \geq \sum_{t=1}^{T} a_t^k + \sum_{t=1}^{T} \sum_{j \in J_{OND}} b_{jt}^k x_{jt} && (6d)\\
& \quad + \sum_{t=1}^{T} \sum_{j \in J_{NS1}} c_{jt}^k \left(\sum_{t=max\{1, t-(u1-1)\}}^{t} x_{jt} \right) && (6e)\\
& \quad + \sum_{t=1}^{T} \sum_{j \in J_{NS2}} d_{jt}^k \left(\sum_{t=max\{1, t-(u2-1)\}}^{t} x_{jt} \right) && (6f)
\end{aligned}$$

$$Q(\hat{x}^k, \omega) = min \sum_{t=1}^{T} \sum_{j=1}^{N} C_j P_{jt}^{\omega} + \sum_{t=1}^{T} \lambda l_t^{\omega} \tag{7a}$$

$$s.t. \quad \sum_{j=1}^{N} P_j(P_{jt}^{\omega}) + l_t^{\omega} \geq d_t^{\omega} \qquad \forall t = 1, .., T \quad (\pi) \tag{7b}$$

$$P_{jt}^{\omega} \leq \hat{x}_{jt}^k \qquad \forall j \in J_{OND}, \forall t = 1, ..., T \quad (\mu) \tag{7c}$$

$$P_{jt}^{\omega} \leq \sum_{t'=max\{1,t-(u1-1)\}}^{t} \hat{x}_{jt'}^k \quad \forall j \in J_{NS1}, \forall t = 1, .., T \quad (\gamma) \tag{7d}$$

$$P_{jt}^{\omega} \leq \sum_{t'=max\{1,t-(u2-1)\}}^{t} \hat{x}_{jt'}^k \quad \forall j \in J_{NS2}, \forall t = 1, .., T \quad (\rho) \tag{7e}$$

$$0 \leq P_{jt}^{\omega} \leq 1 \qquad \forall j \in J, \forall t = 1, .., T \tag{7f}$$

4.2 L-shaped Decomposition Algorithm

The L-shaped decomposition algorithm is used in the previously described SAA algorithm. In the the SAA algorithm, we solve the sample average problem (Equation 4) for each sample, which is a two-stage stochastic programming problem with a finite number of scenarios. We use the L-shaped decomposition algorithm to solve the sample average problem for each sample. The algorithm can be described as follows:

1. Let $LB = 0$, $UB = \infty$, iteration counter $k = 0$, and optimality gap be ϵ. Solve the following lower bound formulation (Master problem) to get the lower bound of the algorithm as given by Equation 6, where $\hat{x}^k$ is the optimal solution of the Master problem at iteration k.
2. Given $\hat{x}^k$, solve the second-stage problem for each scenario ω, described by Equation 7. The dual variables corresponding to the constraints are represented by symbols: π, μ, γ, and ρ.
3. Use the objective values of all the second-stage problems to compute the total objective function value at iteration k, as follows:

   $$f(\hat{x}^k) = \sum_{t=1}^{T} \sum_{j=1}^{N} S_j \hat{x}_{jt}^k + \frac{1}{|\Omega|} \sum_{\omega \in \Omega} Q(\hat{x}^k, \omega).$$

   If $f(\hat{x}^k) < UB$, update the upper bound $UB = f(\hat{x}^k)$, and store the solution $\hat{x} = \hat{x}^k$.
4. If $(UB - LB) < \epsilon$, then stop, and return $\hat{x}$ as the optimal solution and UB as the optimal objective value. Otherwise, go to Step 5.
5. Use optimal dual solutions of each second-stage problems corresponding to scenarios, $\omega = 1, 2, 3, ..., \mid \Omega \mid$, from Step 2 to compute the coefficients of optimality constraints. Aggregate the coefficients of the optimality constraints

from all the scenarios to compute the coefficients of the aggregated optimality constraint (cut) as follows:

$$a_t^{k+1} = \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \hat{\pi}_t^{\omega} d_t^{\omega}$$
$$b_{jt}^{k+1} = \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \hat{\mu}_{jt}^{\omega}$$
$$c_{jt}^{k+1} = \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \hat{\gamma}_{jt}^{\omega}$$
$$d_{jt}^{k+1} = \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \hat{\rho}_{jt}^{\omega}.$$

Now, construct the new optimality cut with these coefficients and add the cut to the Master problem. Update $k = k + 1$ and go to Step 1.

We empirically evaluated the efficacy of the integrated SAA and L-shaped decomposition approach using two synthetic datasets that were inspired from the Belle II experiment and real-world data from cloud computing platforms. We provide the details in §6. We describe the genetic algorithms based approach next.

4.3 A Genetic Algorithms based approach

Genetic Algorithms (GA) are common evolutionary optimization techniques that are used to solve problems containing large and complex search spaces. GAs emulate the process of natural selection to produce better solutions as time progresses. GAs consist of a set of candidate solutions called a population. Each solution within the population is called a chromosome. Chromosomes can be compared with one another by evaluating their fitness, i.e., how well they optimize a given objective. Individual decision variables within a chromosome are called genes. During the execution of a GA, various genetic operations (e.g., mutation of individual genes, swapping genes between chromosomes) are performed to enable progress through the search space.

A GA based approach for cost-efficient selection and scheduling of resources with demand uncertainty was introduced in [6]. This method implements a multi-objective Genetic algorithm based on NSGA-II [5]. In this approach, genes represent individual months within the planning horizon, and will specify the amount of each resource type allocated for that month. Chromosome are represented as $P \times Q$ matrices, where P and Q represent the number of months and the number of resource types available, respectively. If the resources determined by a chromosome are unable to meet the specified demand, additional on-demand resources are subscribed to fill the gap. To speed up evaluation of the search space, parallel execution of the GA is achieved using a modified island model. We refer you to Friese *et al.,* for further details [6]. The primary reason to consider this algorithm in the paper is to provide a baseline evaluation of the two-stage stochastic programming approach.

5 Experimental Setup

Computation and data storage of Belle II experiments span a geographically distributed set of resources across several continents. The experiments can be

classified into three main activities: (i) processing of raw data from the Belle II detector, (ii) Monte Carlo simulations of physical phenomena, and (iii) physics analysis of experimental and simulation data. While the computational demand for Monte Carlo campaigns is fairly stable, the demand for user analysis tends to be chaotic leading to uncertainties in computational and storage demands. Inspired from this setting, we use a representative setup for demand and supply in our experiments that are detailed in this section.

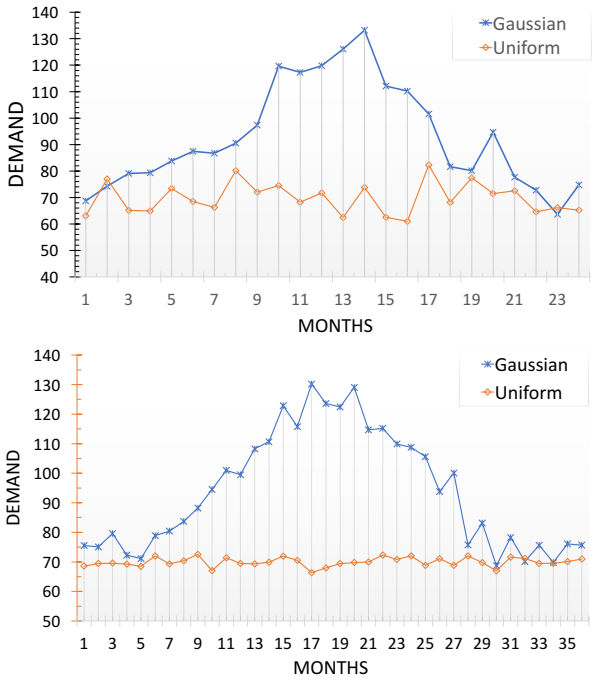

Fig. 1: Base demand curves for 24 months (top) and 36 months (bottom) with uniform and Gaussian distributions.

Numerical experiments are conducted for 24-month and 36-month planning horizons. Additionally, for each planning horizon, we study two probability distributions – uniform and Gaussian. For each distribution, we construct five unique base demand curves. Figure 1 illustrates base demand curves for uniform and Gaussian distributions over a 24-month (left) and 36-month (right) planning horizons. Each base demand curve is used to construct random demand scenarios. Specifically, for each month in the base curve, a uniform distribution, $U(d_b - a,\, d_b + b)$, is sampled to realize the actual demand for the corresponding month in a given scenario. We conduct experiments to evaluate scenarios that were constructed using two different levels of variation: smaller variation, $U(d_b - 7.5,\, d_b + 15)$, and larger variation, $U(d_b - 15,\, d_b + 20)$. All experiments are carried out for 10 SAA samples, where each sample consists of 80 scenarios. The

size of the reference sample is set to 1000 scenarios. Our experimental results will show that these parameters of SAA can obtain good quality solutions and can provide better approximation of the true problem.

We use representative computation and cost models of cloud computing resources based on Amazon EC2, as shown in Table 2. ECU, Period, S, and C respectively denote computing power, subscription period, subscription cost (in dollars), and usage cost per month for each machine. We only list a subset of machine types for illustrative purposes. A full list is provided in [6]. Please note that the prices in the table may not reflect current Amazon EC2 prices. In our experiments, we assume that we can purchase/utilize no more than 10 units of a resource for any given month. We implement the integrated Sample Average Approximation and the L-shaped decomposition algorithm in Python 2.7 with Gurobi optimizer [7] that is used to solve the mixed-integer programming master problem and the linear programming second-stage problems. The experiments are run on a laptop with Intel core i7 2.80GHz processor and 8GB RAM. We compare the results of our stochastic optimization methodology with a genetic algorithms based approach (§4.3). A fundamental difference between the two approaches is that the GA based approach is deterministic, in that it does not consider demand uncertainty during the fitness evaluation of the chromosomes [6]. However, the GA independently evaluates all the demand scenarios as part of the SAA framework. The solution that minimizes cost across all the scenarios is chosen as the best resource allocation strategy.

Table 2: A subset of Amazon EC2 resources used in our experiments.

Index	Machine type	ECU	Period	S	C
1	on-demand	0.2	1	0	19.04
3	hybrid	0.2	12	102	4.38
4	subscription	0.2	12	151	0
18	on-demand	13	12	0	126.29
19	hybrid	13	12	648	54.02
20	subscription	13	12	1271	0
33	on-demand	124.5	12	0	1264.36
34	hybrid	124.5	12	6482	540.2
35	subscription	124.5	12	12706	0

6 Experimental Results

We now present the results and observations from our experiments. We will first analyze the solutions computed by the two-stage stochastic programming approach by studying the convergence, variation and composition of the solutions. We will then assess the quality of solutions relative to deterministic solutions that do not include demand uncertainty. We will finally look at the quality with respect to the solutions computed by the Genetic Algorithms based approach.

6.1 Stochastic programming based solutions

Convergence: Each sample in the Sample Average Approximation (SAA) problem is solved using the L-shaped decomposition algorithm. Performance of L-shaped algorithm is therefore critical for the overall performance. We analyze the convergence behavior of the L-shaped algorithm. Figure 2 illustrates convergence of the upper and lower bounds of the algorithm over iterations for a sample with 24-month horizon. As the samples are randomly generated, the optimal cost at which the algorithm converges for different samples vary within a small range which is evident from Table 3 where the variances associated with the optimal costs are In general, we observe that the upper bound of the algorithm decreases over iterations as the Master problem produces better solutions at each iteration until convergence. Similarly, the lower bound increases as the optimality constraints force the Master problem to purchase the best possible machines to satisfy demands. The algorithm converges when no better machine configurations are available to reduce the total cost. At which point, the upper and lower bounds of the algorithm converge to the same value.

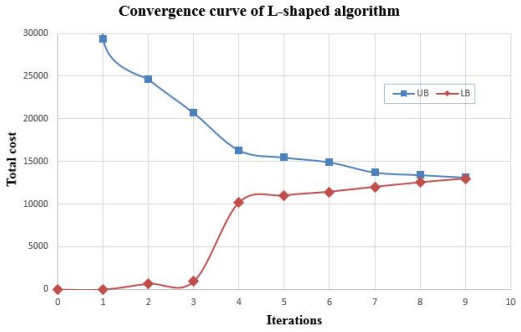

Fig. 2: Convergence of the L-shaped algorithm fora sample with 24-month horizon.

Composition: With explicit consideration of demand uncertainties, the proposed solutions compute optimal machine subscriptions that are robust against random demand scenarios. The solutions also indicate the optimal subscription time for each selected resource since we assume that the resources can be used partially for a given period. Thus, the composition of the solution – different machines selected in the optimal solution – is an important factor. Intuitively, a cost-effective decision is to subscribe total-upfront or partial-upfront (hybrid) machines at the beginning of the planning horizon and then use them to the maximum extent at each time period to meet the demand. Smaller portions of unmet demand can be satisfied by on-demand machines. In Figure 3, we illustrate the machine composition for a 24-month planning horizon where the base demands follow a Gaussian distribution. We observe that a large portion of the

demand is satisfied by partial-upfront (hybrid) resources, and the rest of the demands are satisfied by on-demand resources.

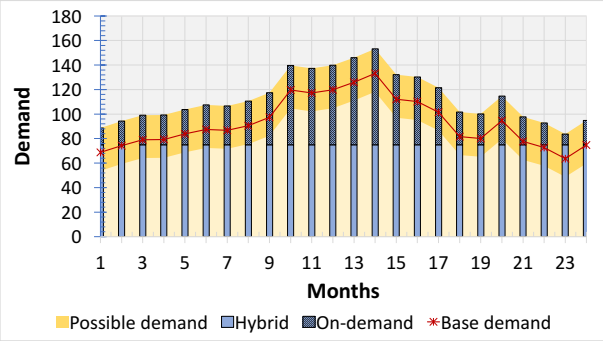

Fig. 3: Composition of the solution for a problem with 24-month horizon. The orange shaded region shows possible variation of demand from the base curve shown in red. Each bar represents machine type composition for a given month. Hybrid (partial upfront) machines shown in blue are purchased during Period 1, and used in subsequent periods. On-demand purchases are shown with hatched blue bars.

Variance: Since SAA approximates the true problem by a set of scenarios, the total cost computed by our approach is only an approximation of the true cost (§4.1). Therefore, we now present data on the variance of the estimates, σ^2_{gap}, from their true values. Intuitively, the lower the variance, the better is the approximation of the SAA-based solution. In Table 3, we present the optimal cost and the associated variance for a case with 24-month planning horizon. The base demands follow both Uniform and Gaussian distributions, and with smaller and larger variation (§5). The optimal cost is the total cost corresponding to the optimal solution, where the total cost consists of the total subscription cost and the expected operating cost over all the scenarios. We observe that the variance is small, which indicates that our approach provides high quality estimates of the true problem. We also observe that the variance increases as the uncertainty increases. For example, variance is larger for larger variation runs, and for Gaussian distributions. Consequently, the solutions include subscription of machines with larger computing power to satisfy penalties from large variations in demand, which in turn, increases the total cost.

6.2 Value of Stochastic Solution

An important research question of our work is: *How much benefit do we really get from considering demand uncertainty?* To quantify the answer, we use a metric known as *the value of stochastic solution* (VSS). VSS measures the difference between the optimal cost resulting from a solution considering uncertainty and the optimal cost resulting from applying the expected value problem (EVP) to

Table 3: Variance of the approximate solutions for a 24-month case

Smaller variation		**Larger variation**	
Uniform distribution			
Optimal cost	**Variance**	**Optimal cost**	**Variance**
12337.20	32.04	12640.97	150.24
12216.32	29.30	12587.71	143.50
12398.36	24.45	12836.32	159.98
12343.92	23.45	12786.89	153.57
12543.82	27.01	13055.08	113.92
Gaussian distribution			
16942.72	23.88	17222.81	111.41
16635.05	35.41	17056.74	146.53
15968.80	33.07	16265.13	155.53
16612.91	30.42	17046.62	135.45
17111.34	27.85	17367.92	108.90

uncertain scenarios. Mathematically, $VSS = \frac{EVC-SPC}{SPC}$. Here, EVC represents the total cost from applying the solution of EVP on the random scenarios, whereas SPC represents the total cost resulting from the stochastic programming solution. EVP is a deterministic problem where the expected value of the random demands over all the scenarios are used. If the solution of the EVP is applied to an uncertain environment, it is likely that the resulting cost will be larger than the cost from applying a stochastic programming solution. Due to explicit consideration of uncertainty, the stochastic programming based solution is robust to uncertain demands. The larger the value of VSS, the larger is the cost of ignoring uncertainty for a problem that is actually uncertain.

In Table 4, we present VSS values for five different problems with a 24-month horizon with base demand curves generated from a Gaussian distribution. We observe that the total cost from using the deterministic solution in an uncertain environment is larger than the cost resulting from a stochastic programming solution. We also observe that VSS increases as the range of uncertainty in demand increases. With large uncertainties, EVP gets erroneous and leads to purchase decisions with larger number of underutilized machines, or reliance on on-demand (spot pricing) with higher penalty costs. The VSS values obtained for the problem instances with based demand curves generated from a Uniform distribution also demonstrates the similar behavior as discussed above.

6.3 Comparison with a GA-based approach

As the last part of our evaluation, we compare the quality of our approach with respect to a genetic algorithm (GA) based approach of Friese *et al.* [6] We summarize the results in Table 5 for a problem with 36-month planning horizon and base demand curves with Gaussian distribution. We observe that the stochastic programming based approach significantly outperforms the GA based approach by up to **54%**. Since GA is a heuristic, there are no guarantees for the quality

Table 4: Value of stochastic solution for 24-months with Gaussian demand curves

Smaller variation			Larger variation		
SPC	**EVC**	**VSS(%)**	**SPC**	**EVC**	**VSS(%)**
16942.72	19992.45	18.02	17222.81	22326.62	**29.63**
16635.05	19562.82	17.60	17056.74	22148.01	**29.85**
15968.80	18518.86	15.97	16265.13	20196.25	24.17
16612.91	19486.94	17.29	17046.62	22013.15	**29.14**
17111.34	19986.04	16.80	17367.92	22170.15	27.65

of solutions. Further, the GA-based approach addresses uncertainties indirectly from solving all the scenarios independently and picking the best solution. In contrast, our approach improves the Master solution by systematically considering each scenario. We also observe that the total cost of the solution computed by GA gets larger with larger variation in the data. We are currently exploring methods to improve the overall quality of the GA-based approach.

Table 5: Comparison with the GA based approach for 36-months with Gaussian demand curves The cost reduction (%) is the percentage reduction in the total cost provided by the solution from SAA integrated L-shaped approach compared to the GA based approach

Smaller variation		
Total cost (SAA+L-sh)	**Total cost (GA)**	**Cost reduction(%)**
20390.94	31773.69	35.82
19947.72	34206.53	**41.68**
20529.12	30624.64	32.97
20225.67	27842.28	27.36
20120.93	34748.06	**42.09**
Larger variation		
Total cost (SAA+L-sh)	**Total cost (GA)**	**Cost reduction(%)**
20834.15	34157.65	39.00
20624.69	35600.94	42.07
20835.65	41864.19	**50.23**
20546.52	44433.39	**53.76**
20443.74	33946.73	39.78

7 Related work

Our work is motivated by a general lack of rigorous optimization approaches for workflow scheduling with uncertainties. Towards this end, we introduced demand uncertainty in computing the cost-efficient resource allocation of distributed resources for execution of high energy physics workflows. Our work

is closely related to resource allocation problem in cloud computing. Much of the existing literature in cloud computing ignores uncertainty in resource allocation problems [10, 22]. While a few studies consider uncertainty in demand for cloud computing resources, the uncertainties modeled are from a service provider's perspective. For example, fuzzy optimization is used by Johannes *et al.,* for resource allocation with uncertainty in demand to provide better service to consumers [12]. Similarly, Kusic *et al.,* consider uncertainty in workloads in an optimization framework to provide resources to customers [14]. Resource allocation problems in cloud computing are also explored by several other researchers [16, 17, 23].

In this paper, we build on our previous work, where we introduced a cost-efficient resource selection framework with demand uncertainties using Sample Average Approximation and Genetic Algorithms [6]. We provide a detailed comparison with this approach in §6. Our work is also inspired from the Unit Commitment problem in electric power grids [8]. Stochastic programming is widely used to provide resource allocation decisions under uncertainty in many areas including unit commitment problems [24], power generation and transmission line expansion problem [11], and cyber security [2]. Stochastic programming has also been utilized in cloud computing resource management problems [4], in which VM's are reserved or purchased on demand for a given time period, reservations that span more than a single time-period are not considered. However, stochastic programming has not been widely applied for cost-efficient resource selection problems from a user's perspective, and in the context of scientific workflows. To the best of our knowledge, this is the first work to develop a two-stage stochastic programming model and stochastic optimization algorithm for selection of geographically distributed resources under demand uncertainty for efficient execution of complex scientific workflows.

8 Conclusions and Future Work

Efficient utilization of geographically distributed resources in the context of large scientific workflows is a challenging problem. We presented a novel stochastic programming based approach for cost-efficient selection of resources under demand uncertainties. By integrating a sampling strategy, Sample Average Approximation with the L-shaped decomposition algorithm, we developed a solution for continuous distribution of the uncertain parameters for demand, capable of solving problems with a large number of scenarios. Using two case studies and two probability distribution functions, we demonstrated the efficacy of the proposed solution. We also demonstrated superior performance relative to a previously developed method using genetic algorithms.

In order to scale the proposed solution approaches to real-world problems, computational complexity needs to be addressed in a systematic manner. Solving a large number of mixed integer problems can be computationally infeasible. One approach can be to approximate this problem by using Lagrange relaxation, which leads to the solution of a large number of small problems. Another approach is to integrate the ideas from this work to develop efficient heuristics to

seed the genetic algorithm (GA) based method. Both these methods are part of our ongoing and future work.

In addition to uncertainties in demand, the focus of this work, there are uncertainties in the availability of resources. Further, network and file system congestion lead to uncertainties in system performance. Therefore, the proposed approaches need to be augmented to include these uncertainties without increasing the computational complexity due to the number of scenarios that need to be considered. Systematic analysis of historical demand and supply data can lead to accurate understanding of probability distribution functions, and in turn benefit SAA-based methods. We are collecting a large amount of historical data from the execution of Belle II jobs towards this end.

To the best of our knowledge, this is the first stochastic programming based approach to address the resource allocation problem with demand uncertainties for large-scale scientific workflows. We believe that our work will inspire the development of scheduling methods with explicit consideration of uncertainties – an important problem in distributed computing.

Acknowledgements

This work was supported by the Integrated End-to-end Performance Prediction and Diagnosis for Extreme Scientific Workflows (IPPD) Project. IPPD is funded by the U. S. Department of Energy Awards FWP-66406 and DE-SC0012630 at the Pacific Northwest National Laboratory. The work of Luis de la Torre was supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Visiting Faculty Program (VFP).

Bibliography

[1] Benders, J.F.: Partitioning procedures for solving mixed-variables programming problems. Numerische mathematik 4(1), 238–252 (1962) 7

[2] Bhuiyan, T.H., Nandi, A.K., Medal, H., Halappanavar, M.: Minimizing expected maximum risk from cyber-attacks with probabilistic attack success. In: Technologies for Homeland Security (HST), 2016 IEEE Symposium on. pp. 1–6. IEEE (2016) 7, 17

[3] Birge, J.R., Louveaux, F.: Introduction to Stochastic Programming. Springer Publishing Company, Incorporated, 2nd edn. (2011) 2

[4] Chaisiri, S., Lee, B.S., Niyato, D.: Optimization of resource provisioning cost in cloud computing. IEEE Transactions on Services Computing 5(2), 164–177 (2012) 17

[5] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE transactions on evolutionary computation 6(2), 182–197 (2002) 10

[6] Friese, R.D., Halappanavar, M., Sathanur, A.V., Schram, M., Kerbyson, D.J., de la Torre, L.: Towards efficient resource allocation for distributed workflows under demand uncertainties. In: Job Scheduling Strategies for Parallel Processing. Springer-Verlag (2017), lect. Notes Comput. Sci. To appear. 1, 10, 12, 15, 17

[7] Gurobi, O.: Gurobi optimizer reference manual. URL: http://www. gurobi. com (2015) 12

[8] Halappanavar, M., Schram, M., de la Torre, L., Barker, K., Tallent, N.R., Kerbyson, D.J.: Towards efficient scheduling of data intensive high energy physics workflows. In: Proceedings of the 10th Workshop on Workflows in Support of Large-Scale Science. pp. 3:1–3:9. WORKS '15, ACM, New York, NY, USA (2015) 2, 17

[9] Hara, T.: Belle II: Computing and network requirements. In: Proc. of the Asia-Pacific Advanced Network. pp. 115–122 (2014) 2

[10] Huang, Z.C., He, C., Gu, L., Wu, J.F.: On-demand service in grid: Architecture, design and implementation. In: Parallel and Distributed Systems, 2005. Proceedings. 11th International Conference on. vol. 2, pp. 674–678. IEEE (2005) 17

[11] Jirutitijaroen, P., Singh, C.: Reliability constrained multi-area adequacy planning using stochastic programming with sample-average approximations. IEEE Transactions on Power Systems 23(2), 504–513 (2008) 17

[12] Johannes, A., Borhan, N., Liu, C., Ranjan, R., Chen, J.: A user demand uncertainty based approach for cloud resource management. In: Computational Science and Engineering (CSE), 2013 IEEE 16th International Conference on. pp. 566–571. IEEE (2013) 17

[13] Kleywegt, A.J., Shapiro, A., Homem-de Mello, T.: The sample average approximation method for stochastic discrete optimization. SIAM Journal on Optimization 12(2), 479–502 (2002) 7

[14] Kusic, D., Kandasamy, N.: Risk-aware limited lookahead control for dynamic resource provisioning in enterprise computing systems. Cluster Computing 10(4), 395–408 (2007) 17
[15] Mak, W.K., Morton, D.P., Wood, R.K.: Monte carlo bounding techniques for determining solution quality in stochastic programs. Operations research letters 24(1), 47–56 (1999) 7
[16] Medernach, E., Sanlaville, E.: Fair resource allocation for different scenarios of demands. European Journal of Operational Research 218(2), 339–350 (2012) 17
[17] Rodriguez, M.A., Buyya, R.: Deadline based resource provisioningand scheduling algorithm for scientific workflows on clouds. IEEE transactions on cloud computing 2(2), 222–235 (2014) 17
[18] Santoso, T., Ahmed, S., Goetschalckx, M., Shapiro, A.: A stochastic programming approach for supply chain network design under uncertainty. European Journal of Operational Research 167(1), 96–115 (2005) 6, 7
[19] Saravanan, B., Das, S., Sikri, S., Kothari, D.: A solution to the unit commitment problem–a review. Frontiers in Energy 7(2), 223 (2013) 3
[20] Shapiro, A., Homem-de Mello, T.: A simulation-based approach to two-stage stochastic programming with recourse. Mathematical Programming 81(3), 301–325 (1998) 7
[21] Van Slyke, R.M., Wets, R.: L-shaped linear programs with applications to optimal control and stochastic programming. SIAM Journal on Applied Mathematics 17(4), 638–663 (1969) 7
[22] Yang, J., Qiu, J., Li, Y.: A profile-based approach to just-in-time scalability for cloud applications. In: Cloud Computing, 2009. CLOUD'09. IEEE International Conference on. pp. 9–16. IEEE (2009) 17
[23] Zhang, Q., Zhu, Q., Boutaba, R.: Dynamic resource allocation for spot markets in cloud computing environments. In: Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on. pp. 178–185. IEEE (2011) 17
[24] Zheng, Q.P., Wang, J., Pardalos, P.M., Guan, Y.: A decomposition approach to the two-stage stochastic unit commitment problem. Annals of Operations Research 210(1), 387–410 (2013) 17