Noname manuscript No.
(will be inserted by the editor)

Lagrangian relaxation based heuristics for a chance-constrained optimization model of a hybrid solar-battery storage system

Bismark Singh · Bernard Knueven

Received: date / Accepted: date

Abstract We develop a stochastic optimization model for scheduling a hybrid solar-battery storage system. Solar power in excess of the promise can be used to charge the battery, while power short of the promise is met by discharging the battery. We ensure reliable operations by using a joint chance constraint. Models with a few hundred scenarios are relatively tractable; for larger models, we demonstrate how a Lagrangian relaxation scheme provides improved results. To further accelerate the Lagrangian scheme, we embed the progressive hedging algorithm within the subgradient iterations of the Lagrangian relaxation. We investigate several enhancements of the progressive hedging algorithm, and find bundling of scenarios results in the best bounds.

Keywords Chance constraints · Stochastic optimization · Lagrangian decomposition · Progressive hedging · Solar power · Photovoltaic power station · Battery storage · Virtual power plant · Out of sample validation

1 Introduction

1.1 Motivation

A photovoltaic power station (PPS) is a system consisting of solar panels and inverters to convert light into electricity. Solar power is uncertain, and, of course, not available for all hours. Hence, coupling a sufficiently large storage device with a PPS can significantly increase its economic value by shifting energy to times when it is in higher demand, creating a hybrid energy system; see, e.g., [16] for different combinations of sources forming hybrid energy systems. Such energy storage systems find frequent applications in modern microgrids [5], especially as standalone systems in remote islands [26,33,34,63]. Standalone hybrid PPS systems have also been shown to be economically feasible and viable. Bhuiyan et al. [8] show the economical feasibility of a PPS system in a remote area of Bangladesh. Similarly, a large scale system of 100MW in the Gobi desert is shown to be economically feasible [22]. Ma et al. [34] study a hybrid wind-solar system with storage from a pumped-hydro system on a remote island in Hong Kong. Zhao et al. [63] study a hybrid system with wind, solar, diesel and battery storage on an island in China. Buildings can also be powered efficiently using micro-grids [18,61]. For an introduction to PPS systems and their characteristics, see, e.g., [38,39].

Due to the uncertainty in the availability of renewables, additional complications arise when there is a need to ensure highly reliable operations. To mitigate the effect of the uncertainty in renewables, often two

B. Singh
Discrete Mathematics
Friedrich-Alexander-Universität Erlangen-Nürnberg
Erlangen, 91058 Germany

B. Singh · B. Knueven
Discrete Math & Optimization
Sandia National Laboratories
Albuquerque, NM 87123 USA

or more sources of energy are coupled, in addition to the storage system. Solar-wind hybrid systems have been extensively studied—optimal sizing [59,64], feasibility of operations [7,34], and ensuring reliable operations [9,64]. In this work, we consider a problem where a standalone PPS, connected to a battery storage, commits (e.g., in forward energy markets) to deliver a fixed supply of energy *reliably* at every hour. Although, we consider a single battery storage with a single (unified) PPS, the approach we propose can be extended to multiple power sources if they can be viewed to interact externally as a single group. Thus, our system supplies power as a so-called virtual power plant; see, e.g., [28,35].The renewable source of power we consider is solar power alone, and we supply power into a self scheduling day-ahead market. Our aim is to ensure highly reliable operations, despite the fact that supply decisions are made a day-ahead (and, thus without observing the uncertain solar power availability). Ensuring reliability is the primary purpose, while maximizing revenues from supplying power is secondary.

The model we propose outputs a day-ahead promise of hourly power, using a forecast of the hourly day-ahead electricity prices and the hourly day-ahead solar power. We assume the electricity prices are completely known, and we have a sufficiently large number of solar power scenarios. A single solar power scenario consists of a vector of 24 values for the hourly day-ahead solar output; i.e., we assume that once a solar power scenario is realized, the solar power output for the entire day is known. Although having a dynamic hour-by-hour forecast of solar power would result in a higher fidelity model, this assumption significantly simplifies modeling and computations as we describe in Section 1.2. Next, at every hour, we dynamically decide whether to charge the battery (if solar power is available in surplus above our promise) or discharge the battery (if solar power is short of our promise). We assume the battery cannot be simultaneously charged and discharged. Similar self-scheduling models have been proposed before, although for different energy systems [13,50].

1.2 Joint Chance Constraints

To ensure highly reliable operations in the self-scheduling market, we make use of joint chance constraints (JCCs). The following is an example of a linear joint chance constraint:

$$\mathbb{P}(x_i \geq y_i(\xi) + a_i(\xi)), \forall i \in I) \geq 1 - \varepsilon.$$

Here, ξ is a random variable, x is a vector of decisions of size I, $a_i(\xi)$ is a vector of data, for each i, that is known only after the realization of ξ, and $y_i(\xi)$ is a vector of decisions, for each i, made after observing the realization of ξ. We require the joint probability of satisfying the above inequality to be no less than 1-ε, where ε is a predetermined threshold. Thus, if we "fail" in even a single i, we fail to satisfy the JCC. In this sense, the JCC represents a very stringent satisfaction criterion. Computationally, optimization models with JCCs are hard to solve [32], and several tailored decomposition approaches [29,31], as well as heuristics [54,6], are available for their solution. Theoretically, such optimization models are known to be NP-hard [32] and the feasible region is generally non-convex [42], with a few exceptions such as [21].

Nonetheless, optimization models with JCCs are an attractive modeling choice for ensuring reliability of a set of operations. Chance-constrained models find frequent applications— in power systems [24,56], in traffic flow [12,53], and in portfolio selection [1,27], among others. There exists a modeling choice with regards to the timing of observing the uncertainty. The first case is if the entire set of future scenarios is known after making the first-stage decision, x. In the second case, the uncertainty is revealed periodically and subsequent decisions need to be made in multiple stages. Such models are known as two-stage and multi-stage stochastic programs, respectively; see, e.g., [42]. Multi-stage models are computationally harder than two-stage models. Since optimizing over JCCs already poses a significant challenge, in this tradeoff between fidelity and computational effort, we use a two-stage model instead of a multi-stage model. In our model, we make a first-stage decision of 24 hourly day-ahead power promises. Then, we observe the previously uncertain solar power output and make all the second-stage decisions of charging/discharging the battery. We use a highly reliable regime of operations (ε at most 0.05). Our motivation comes from the fact that failure to satisfy the promise could lead to curtailment of future contracts with the supplier, and thus ensuring reliability is valued more than increasing profit. In Section 2 we describe our model in greater detail.

2 Mathematical Modeling

2.1 Notation

Indices and Sets:

$t \in T$	set of hours; $\{1, 2, \ldots, \lvert T\rvert\}$
$\omega \in \Omega$	set of scenarios; $\{\omega_1, \omega_2, \ldots, \omega_{\lvert\Omega\rvert}\}$

Parameters: First Stage

η	efficiency of battery $0 < \eta < 1$
$\overline{X}$	maximum energy that can be stored in the battery [kWh]
$\underline{X}$	minimum energy that must remain in the battery [kWh]
R_t	marginal revenue earned at hour t [\$/kWh]
C_c	operational cost of charging the battery [\$/kWh]
C_d	operational cost of discharging the battery [\$/kWh]
λ_t^q	selling price at hour t [\$/kWh]
$\overline{P}$	maximum charging rate of battery [kW]
$\overline{Q}$	maximum discharging rate of battery [kW]

Parameters: Second Stage

s_t^ω	solar power available at hour t under scenario ω [kWh]

Decision Variables: First Stage

y_t	power promised to deliver at hour t [kWh]

Decision Variables: Second Stage

p_t^ω	energy charged to battery during hour t in scenario ω [kWh]
q_t^ω	energy discharged from battery during hour t in scenario ω [kWh]
x_t^ω	energy stored in battery at hour t in scenario ω [kWh]
w_t^ω	binary variable; 1 if battery is charging at hour t and 0 if discharging

2.2 Optimization Model

$$\max \sum_{t\in T} R_t y_t - \mathbb{E}[C_c p_t^\omega + C_d q_t^\omega] \tag{1a}$$

$$\text{s.t. } \mathbb{P}\left(y_t \le s_t^\omega + q_t^\omega - p_t^\omega, \forall t \in |T|\right) \ge 1 - \varepsilon \tag{1b}$$

$$x_{t+1}^\omega = x_t^\omega + \eta p_t^\omega - \frac{1}{\eta} q_t^\omega, \qquad \forall t = 1, 2, \ldots |T| - 1, \forall \omega \in \Omega \tag{1c}$$

$$p_t^\omega \le \overline{P} w_t^\omega, \qquad \forall t \in T, \forall \omega \in \Omega \tag{1d}$$

$$q_t^\omega \le \overline{Q}(1 - w_t^\omega), \qquad \forall t \in T, \forall \omega \in \Omega \tag{1e}$$

$$\underline{X} \le x_t^\omega \le \overline{X}, \qquad \forall t \in T, \forall \omega \in \Omega \tag{1f}$$

$$w_t^\omega \in \{0, 1\}, \qquad \forall t \in T, \forall \omega \in \Omega \tag{1g}$$

$$y_t, p_t^\omega, q_t^\omega \ge 0 \qquad \forall t \in T \forall \omega \in \Omega. \tag{1h}$$

Boundary condition: $x_1^\omega = x_0, \forall \omega \in \Omega$.

The objective function in (1a) aims to maximize profit; i.e., revenue from the promise minus the expected cost of operating (charging or discharging) the battery. A more sophisticated model could include piecewise linear or quadratic costs; see, e.g., [11,57]. The JCC in equation (1b) requires that the joint probability of meeting the promised energy to sell, by the available solar power (via s^ω) and discharging the battery (via q^ω), for the entire time horizon meets a threshold; excess solar energy can be used to charge the battery (via p^ω). If the solar power exceeds the need, we might need to curtail it. However, in this article we do not penalize this curtailment, and hence equation (1b) is expressed as a

inequality. Here, ε is a positive quantity just greater than zero, such as 0.05. Constraint (1c) relates the energy stored by the battery in a subsequent hour with the previous hour; here, the quantity $\eta p_t^\omega - \frac{1}{\eta} q_t^\omega$ is the energy charged or discharged by the battery during hour t. Note that both p_t^ω or q_t^ω cannot simultaneously be positive, this is ensured by constraints (1d),(1e) and (1g). Further, since $\eta < 1$, a fraction of energy is lost while both charging and discharging; i.e., we consume more than 100% while charging and supply less than 100% while discharging. Further, constraint (1f) ensures the minimum and maximum amount of energy stored in the battery, and thus equations (1d)-(1h) restrict the amount the battery can be charged or discharged. Lower bounds on x are typically included for electric batteries to maintain long and healthy lifetimes; see, e.g., [35]. Constraints (1g)-(1h) ensure the binary and non-negativity restrictions on the relevant decision variables.

One potential criticism of optimization models with JCCs, such as model (2), is that in the ε percent of scenarios which are violated by equation (1b) the violations can be large. Although, equation (1b) restricts the number of violations to be a few, there is nothing in model (2) that restricts the magnitude of these violations. This well-known drawback is an artifact of chance-constrained programming in general, and is also reflected in our model. Several methods exist to bound the violations. We direct the interested reader to [45] and [58] for two remedies using risk measures and envelope constraints, respectively. However, models that only penalize energy violations (or, do not employ chance constraints) can yield solutions that violate the promise with large frequency when it is economically advantageous. Our aim is to decide whether it is feasible to couple a solar plant with an appropriately sized battery unit, and whether this coupling would enable the hybrid unit to participate in the day-ahead market.

Now, consider the following optimization model in which we drop the binary variable w_t^ω:

$$\max \sum_{t \in T} R_t y_t - \mathbb{E}[C_c p_t^\omega + C_d q_t^\omega] \tag{2a}$$

$$\text{s.t. } \mathbb{P}\left(y_t \leq s_t^\omega + q_t^\omega - p_t^\omega, \forall t \in |T|\right) \geq 1 - \varepsilon \tag{2b}$$

$$x_{t+1}^\omega = x_t^\omega + \eta p_t^\omega - \frac{1}{\eta} q_t^\omega, \qquad \forall t = 1, 2, \ldots |T| - 1, \forall \omega \in \Omega \tag{2c}$$

$$p_t^\omega \leq \overline{P}, \qquad \forall t \in T, \forall \omega \in \Omega \tag{2d}$$

$$q_t^\omega \leq \overline{Q}, \qquad \forall t \in T, \forall \omega \in \Omega \tag{2e}$$

$$\underline{X} \leq x_t^\omega \leq \overline{X}, \qquad \forall t \in T, \forall \omega \in \Omega \tag{2f}$$

$$y_t, p_t^\omega, q_t^\omega \geq 0 \qquad \forall t \in T, \forall \omega \in \Omega. \tag{2g}$$

Proposition 1 *In an optimal solution to model* (2)*, both* p_t^ω *and* q_t^ω *cannot be strictly positive together for any* $t \in T, \omega \in \Omega$.

Proof. Assume an optimal solution of model (2) is given by $y^* = [y_t]_{t=1,2,\ldots,|T|}, p^* = [p_t]^\omega_{t=1,2,\ldots,|T|}, q^* = [q_t]^\omega_{t=1,2,\ldots,|T|}, x^* = [x_t]^\omega_{t=1,2,\ldots,|T|}, \forall \omega \in \Omega$. Assume that p_t^ω and q_t^ω are strictly positive, for some $t \in T, \omega \in \Omega$. We show that this solution cannot be optimal by proving a better solution (one that gives a larger objective function value) always exists.

First, assume that $\eta^2 p_t^\omega > q_t^\omega$, for this t and ω pair. Then, consider a new solution $p_t'^\omega \leftarrow p_t^\omega - \frac{1}{\eta^2} q_t^\omega, q_t'^\omega \leftarrow 0, x_t'^\omega \leftarrow x_t^\omega, y_t' \leftarrow y_t$. We observe that this solution is feasible to model (2); further, it results in a strictly larger objective function than the optimal solution. This is a contradiction.

Next, assume that $\eta^2 p_t^\omega \leq q_t^\omega$, for this t and ω pair. Again, consider a new solution $p_t'^\omega \leftarrow 0, q_t'^\omega \leftarrow q_t^\omega - \eta^2 p_t^\omega, x_t'^\omega \leftarrow x_t^\omega, y_t' \leftarrow y_t$. This solution is feasible to model (2), and results in a strictly larger objective function than the optimal solution. This is a contradiction.

Thus, in any optimal solution to model (2), for any t and ω at most one of p_t^ω or q_t^ω is strictly positive. □

Proposition 1 offers the advantage that there are no binary $w = [w_t]^\omega_{t=1,2,\ldots,|T|}$ variables in the second-stage. Although, a reformulation of constraint (1b) would still entail second-stage binaries (see, equation (3)), the reformulation in model (2) could save computational effort by reducing $|T||\Omega|$ binary variables. There is some evidence, however, that having additional binary variables can actually speed up the computations by offering MIP solvers additional variables to branch and cut on; see, e.g. [36, 50]. However, we choose the reformulation without the binary w variables as it lends itself better to the

heuristics we describe in the later sections of this article. The probabilistic constraint in equation (2b) can be reformulated using a big-M approach as follows:

$$y_t - q_t^\omega + p_t^\omega \leq s_t^\omega + M_t^\omega z^\omega, \forall t \in T, \omega \in \Omega \tag{3a}$$

$$\sum_{\omega \in \Omega} z^\omega \leq \quad \lfloor N\varepsilon \rfloor \tag{3b}$$

$$z^\omega \in \quad \{0,1\}, \quad \forall \omega \in \Omega. \tag{3c}$$

Here, $\lfloor \cdot \rfloor$ rounds its argument down to the nearest integer. Henceforth, we use model (2) with equation (1b) reformulated as equation (3) as our proposed model. Below, we provide a sufficiently large value for the big-M of equation (3).

Proposition 2 *The following inequality is a valid inequality for model* (2)*:* $y_t \leq \min\{\overline{Q}, \eta(\overline{X} - \underline{X})\} + s_t^{\omega(\lfloor N\varepsilon \rfloor + 1, t)}, \forall t \in T$*, where* $s_t^{\omega(l,t)}$ *denotes the* l^{th} *largest solar power value at time* t*.*

Proof. We first seek an upper bound on the quantity $q_t^\omega - p_t^\omega$. If $q_t^\omega = 0$, a valid upper bound on this quantity is 0. If $p_t^\omega = 0$, a valid upper bound on this quantity is $\min\{\overline{Q}, \eta(\overline{X} - \underline{X})\}$; this is obtained from equation (2c), (2e) and (2f). Thus, from Proposition 1, $q_t^\omega - p_t^\omega \leq \min\{\overline{Q}, \eta(\overline{X} - \underline{X})\}, \forall t \in T, \omega \in \Omega$. Now, the proof replicates that of Proposition 5.2.1 of [49]. By constraint (3b) effectively at most $\lfloor N\varepsilon \rfloor$ of the z^ω variables can take a value of 1; i.e., for every t at most $\lfloor N\varepsilon \rfloor$ of the scenarios can be removed. Thus, the proposed inequality is valid. □

Corollary 1 *From Proposition 2, we have* $M_t^\omega \geq \min\{\overline{Q}, \eta(\overline{X} - \underline{X})\} + s_t^{\omega(\lfloor N\varepsilon \rfloor + 1, t)} - s_t^\omega, \forall t \in T, \omega \in \Omega$*.*

We denote the upper bound, presented in Proposition 2, on the y variables as $y_t^{\max} = \min\{\overline{Q}, \eta(\overline{X} - \underline{X})\} + s_t^{\omega(\lfloor N\varepsilon \rfloor + 1, t)}, \forall t \in T$. And, in our computational experiments of Section 6 we use the tightest big-M given from Corollary 1, namely $M_t^\omega = y_t^{\max} - s_t^\omega, \forall t \in T, \omega \in \Omega$. Despite this, as we show later in this article, model (2) is computationally challenging for relatively larger values of ε (such as 0.05) or, for a large number of scenarios. Model (2) is a classic joint chance-constrained two-stage stochastic program with recourse. The second stage has integer restrictions (the variable z is binary) as well, which makes it even more challenging [2]. To this end, we investigate a Lagrangian relaxation scheme as well as a heuristic, in addition to the traditional approach with the big-M constraints.

3 A Lagrangian Dual Procedure

First, we note that constraint (3b) links together the scenarios. This motivates a straightforward Lagrangian relaxation of model (2). Lagrangian relaxation schemes have been studied before in various settings; see, e.g., individual chance constraints for unit commitment [37], relaxing non-anticipativity constraints [3], and coupling scenario decomposition with Lagrangian decomposition [54].

The Lagrangian dual for a given value of $\lambda \geq 0$ for model (2), obtained by dualizing the constraint (3b), is given as follows:

$$L(\lambda) = \max \sum_{t \in T} \big(R_t y_t - \mathbb{E}[C_c p_t^\omega + C_d q_t^\omega]\big) + \lambda\big(\lfloor N\varepsilon \rfloor - \sum_{\omega \in \Omega} z^\omega\big) \tag{4}$$

$$\text{s.t. (2c)} - \text{(2g)}, \text{(3a)}, \text{(3c)}.$$

To solve model (4), we use the well-known subgradient method [40,48], with the step-size update rule from [19]. Algorithm 1 summarizes this scheme; we need a lower bound for model (2) for the step-size update rule in Step 4. In principle, this lower bound can be computed using any feasible solution; setting $\lfloor N\varepsilon \rfloor$ of the z variables to 1, and the rest to 0, provides a feasible solution. However, since the termination criteria of the Lagrangian relaxation procedure depends on this lower bound, good feasible solution values (those giving a larger value of the objective function) assist in convergence.

Lower Bounding Heuristic: To this end, we obtain a feasible solution using the following simple procedure described in [3]. We first solve model (2) separately for each ω with the corresponding $z^\omega \leftarrow 0$; i.e., a total of $|\Omega|$ problems. We rank the corresponding objective function values from lowest to highest and choose the first $\lfloor N\varepsilon \rfloor$ (i.e., the worst performing scenarios) of these to set $z^\omega \leftarrow 1$. Then, with $\lfloor N\varepsilon \rfloor$ of the z values fixed to 1, we solve model (2), to obtain a valid lower bound (LB) to the problem.

Algorithm 1 Lagrangian Relaxation of model (2). We use $M = 10, time = 1800, \psi = 0.001$.

Input: M, LB, $time$, θ, ψ.

Solve LP relaxation of model (2), set UB to optimal objective value of LP, set λ to the optimal dual of constraint (3b).
while iter $\leq M$ **do**
  Solve model (4), obtain optimal z^ω, $UB \leftarrow \min\{UB, L(\lambda)\}$.
  $\gamma \leftarrow \lfloor N\varepsilon \rfloor - \sum_{\omega\in\Omega} z^\omega$ and $\Delta \leftarrow \theta\frac{(UB-LB)}{\gamma^2}$.
  if $\gamma > 0$, then $\lambda \leftarrow \lambda - \min\{\Delta, \frac{\lambda}{\gamma}\}\gamma$.
  if $\gamma < 0$, then $\lambda \leftarrow \lambda - \Delta\gamma$.
  if no improvement in two iterations, $\theta \leftarrow \frac{\theta}{2}$.
  if $\frac{UB-LB}{UB} \leq \psi$ or time $\geq time$, $STOP$.
  iter $\leftarrow$ iter+1, update time to the cumulative wall-clock time.
end while

For an initial upper bound to the Lagrangian relaxation procedure, we use the linear programming (LP) relaxation of model (2). In the following proposition, we show that model (4) cannot be infeasible.

Proposition 3 *The feasible region of model* (4) *is non-empty.*

Proof. Set $p_t^\omega \leftarrow 0, q_t^\omega \leftarrow 0, x_t^\omega \leftarrow x_0, z^\omega \leftarrow 1, y_t \leftarrow 0, \forall t \in T, \forall\omega \in \Omega$. Clearly, this choice of $p_t^\omega, q_t^\omega, x_t^\omega, z^\omega, y_t$ is feasible to model (4) for all $t \in T, \omega \in \Omega$. □

4 A Progressive Hedging Heuristic for the Lagrangian Dual

Algorithm 2 Progressive Hedging Algorithm for model (4). We use $\kappa = 0.001, \rho_t = 1$.

Input: ρ_t, κ, $\overline{iter}, \lambda$.

Solve model (5), $\forall\omega \in \Omega$, with $u \leftarrow 0, \rho \leftarrow 0$, obtain optimal y_t^ω. Set $PH - UB \leftarrow \sum_{\omega\in\Omega} PH^\omega$.
$\overline{y_t} \leftarrow \sum_{\omega\in\Omega} y_t^\omega/|\Omega|$, and $u_t^\omega \leftarrow \rho_t(y_t^\omega - \overline{y}_t)$.
while iter $\leq \overline{iter}$ **do**
  Solve model (5), $\forall\omega \in \Omega$, obtain optimal y_t^ω.
  $\overline{y_t} \leftarrow \sum_{\omega\in\Omega} y_t^\omega/|\Omega|$, and $u_t^\omega \leftarrow u_t^\omega + \rho_t(y_t^\omega - \overline{y}_t)$.
  Solve model (5) without the quadratic term in objective, $\forall\omega \in \Omega$, set $PH - UB \leftarrow \min\{PH - UB, \sum_{\omega\in\Omega} PH^\omega\}$.
  if $\sum_{\omega\in\Omega}\sqrt{\sum_{t\in T}(\overline{y}_t - y_t^\omega)^2} \leq \kappa$, $STOP$; else iter $\leftarrow$ iter+1.
end while
Solve model (4) with $y_t \leftarrow \overline{y}_t$, obtain optimal z^ω.

Output: $z^\omega, UB \leftarrow PH - UB$.

The progressive hedging (PH) algorithm [43] can be used to solve stochastic programs that have easy-to-solve individual scenario problems. While PH cannot be directly applied to model (2) due to the interlinking of the scenarios via constraint (3b), the relaxed model (4) lends itself perfectly to the decomposition structure required for PH. In this section, we leverage this idea to develop computationally cheap heuristics for the Lagrangian dual. First, we convert our maximization problems to a minimization by reversing the signs on the objective coefficients (R, C_c, C_d), since most of the existing literature on PH considers minimization [15,54,55]. Then, model (5) is the PH sub-problem and the PH heuristic is summarized in Algorithm 2.

$$PH^\omega = -\min \sum_{t\in T}\big(-R_t y_t - [-C_c p_t^\omega - C_d q_t^\omega]\big) - \lambda\big(\frac{\lfloor N\varepsilon\rfloor}{\Omega} - z^\omega\big) + \sum_{t\in T} u_t^\omega y_t + 0.5\sum_{t\in T}\rho_t(y_t - \overline{y}_t)^2 \tag{5a}$$

s.t. (3a), (3c), (2c) − (2g).

Here, u_t^ω is the scenario-dependent weight, $\overline{y}_t$ is the scenario-averaged value for y_t, and PH^ω is the optimal value of each sub-problem. Solving the LR dual approximately via the PH algorithm offers several advantages and disadvantages compared to the naive solve. We discuss some of these in Section 6.5.

In the presence of discrete decision variables (as is the case with our model), the PH algorithm can only be used as a heuristic as its convergence is not guaranteed [30, 55]. Since we are interested in solving the Lagrangian dual, to obtain an upper bound to the true problem given by model (2), any upper bound to the Lagrangian dual is also an upper bound to the true problem. Consider Figure 1; here, LB is the lower bound obtained via the heuristic procedure described in Section 3, z^* is the true optimal value of model (2), and LD is the optimal value of model (4). Since convergence of the PH algorithm is not guaranteed, if we solve the Lagrangian dual via the PH algorithm, the PH optimal objective function value could be either larger or smaller than LD.

If the PH algorithm provides a upper bound (larger than LD) at any subgradient iteration of Algorithm 1, then the bound is still a valid upper bound to z^*. However, if the PH algorithm provides a lower bound (smaller than LD) at any subgradient iteration of Algorithm 1, then this bound could also be smaller than z^* (or, even smaller than LB). To prevent this case, we seek valid (i.e., guaranteed to be larger than LD) upper bounds to LD via the PH algorithm (which may or may not be optimal to PH). Such bounds can be obtained as described in [15]. This bound is obtained by solving model (5) without the quadratic term in the objective; then, the upper bound is given by $\sum_{\omega\in\Omega} PH^\omega$. This bound is represented in Figure 1 by PH-UB, and is computed in Algorithm 2 in Steps 1 and 6.

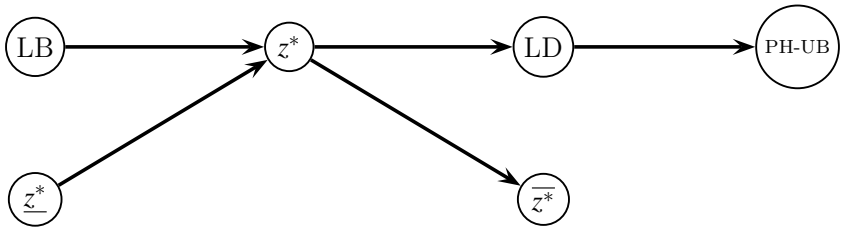

Fig. 1 Bounds for model (2). An arrow from a source node to a sink node indicates the optimal value of the source node is smaller than the sink node. Here, z^* is the true optimal value, LB is the lower bound from a heuristic, LD is the optimal value of the Lagrangian dual, PH-UB is an upper bound for the Lagrangian dual obtained via PH, and $\overline{z^*}$ and $\underline{z^*}$ are the upper and lower bounds for z^* if solved sub-optimally. An optimality gap of β between an upper bound U and a lower bound L indicates $\beta = \frac{U-L}{U}$. The optimal PH value is not shown and could be larger or smaller than z^*.

5 Data Sources

For the computational experiments in this article, we use $\eta = 0.9$ [25], and values of R_t from ERCOT (Electric Reliability Council of Texas) for the year 2012 [14, 50]. Table 1 presents the hourly marginal selling prices. Further, we use $\overline{X} = 960$kWh, with a 2V/1000Ah rating, from the system with 480 lead-acid batteries described in [62]. To compute the cost of charging or discharging the battery, we again use the formula given by [62]: $390\alpha QI$, where α is an effective weighting factor, I is the initial investment cost to purchase the battery, and $390Q$ is an approximation for the total Ah throughput of the battery [23]. For the system in [62], a $\overline{X}$ kWh sized battery is equivalent to $\frac{2000}{960}X$ Ah rating. We assume $\alpha = 0.5$, and use a lead-acid battery cost of \$200/kWh from [47]. An approximately similiar battery cost is available from [41]. This gives us, $C_c = C_d =$ \$0.0256/kWh. Further, we assume $\underline{X} = 0.2\overline{X}, \overline{P} = \overline{Q} = 0.5\overline{X}$, and the battery is 50% charged initially (i.e., $x_0 = 0.5\overline{X}$).

To generate solar power scenarios, we take hourly year-long historical solar power output from a site described in [17]. We use ARMA(p, q) models for the fourteen hours with sunlight, [06:00-20:00). For each hour, we verify the stationarity of the time series and test a number of ARMA(p, q) models to find the best. For details on the forecasting method we used, see [51]. Then, using Monte Carlo sampling

from the best ARMA models for each hour, we create a set of hourly scenarios. Fig. 2 plots a sampling of 300 scenarios, as well as the hourly medians and the ten percent quantile.

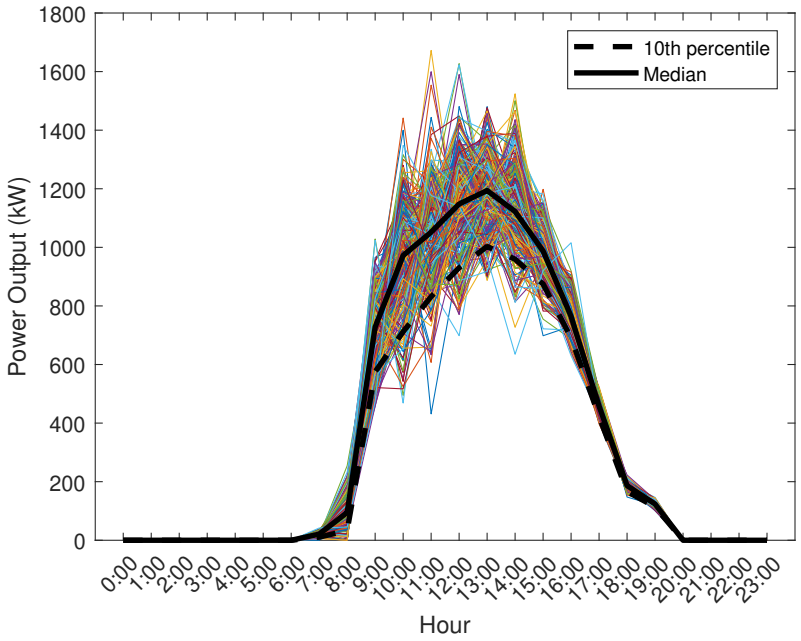

Fig. 2 300 hourly scenarios (unscaled) for solar power generated using the scheme described in [51]. The solid black line is the median hourly value, and the dashed black line is the 10 percentile solar power value.

Hour	Selling price ($/kWh)	Hour	Selling price ($/kWh)
0:00	0.0189	12:00	0.0250
1:00	0.0172	13:00	0.0261
2:00	0.0155	14:00	0.0285
3:00	0.0148	15:00	0.0353
4:00	0.0146	16:00	0.0531
5:00	0.0151	17:00	0.0671
6:00	0.0173	18:00	0.0438
7:00	0.0219	19:00	0.0333
8:00	0.0227	20:00	0.0287
9:00	0.0226	21:00	0.0268
10:00	0.0235	22:00	0.0240
11:00	0.0242	23:00	0.0211

Table 1 Marginal revenue, R_t, earned by selling energy at hour t [14,50].

6 Computational Experiments

6.1 Computational Setup

All tests in this article were carried out with GAMS 25.1.2 using CPLEX 12.8 [44] on an Intel Core i7 2.8 GHz processor with 16 GB of memory. We attempt to solve the original model (2) to a MIP gap of

0.1%, with a maximum time limit of 40 minutes. And, for the rest of the experiments as well we consider an optimality gap of 0.1%; i.e., if the relative gap between the best upper and lower bounds is less than or equal to 0.1% we consider the problem solved. We use multiple sets of scenarios, the smallest instance with 150 scenarios and the largest instance with 1200 scenarios. And, we use two reliability regimes with $\varepsilon = 0.01, 0.05$.

6.2 Analysis of Naive Solve

In Table 2 we present our first results obtained by solving model (2) naively. The "$\underline{z^*}$" column denotes the 24-hour profit obtained in dollars; if the problem could not be solved to the optimal tolerance, we present the best feasible solution. The $\varepsilon = 0.01$ instance is relatively easier (smaller computation time) than the $\varepsilon = 0.05$ instance due to the fewer number of combinatorics involved; the problem can be solved to the optimal tolerance for all instances except $|\Omega| = 1200$. The gaps increase consistently when $|\Omega|$ increases; beyond 600 scenarios the $\varepsilon = 0.05$ case cannot be solved to the optimal tolerance.

$\|\Omega\|$	ε	$\underline{z^*}$	Gap	Time
150	0.05	280.90	-	30
	0.01	273.01	-	2
300	0.05	283.79	-	870
	0.01	278.49	-	29
450	0.05	283.73	-	1932
	0.01	278.93	-	96
600	0.05	280.65	6.7%	2400
	0.01	277.58		234
900	0.05	278.66	7.6%	2400
	0.01	276.83	-	1341
1200	0.05	271.76	9.5%	2400
	0.01	270.95	4.6%	2400

Table 2 Computational results for model (2). We use a maximum time limit of 2400 seconds and a MIP gap of 0.1%. Entries with a "-" indicate that the problem was solved to the required gap within the time limit; i.e., $\underline{z^*} = \overline{z^*} = z^*$. For details, see Section 6.2 and Fig. 1.

6.2.1 Analysis: Solution Profile

Next, we provide a profile of an optimal solution. We choose the $\varepsilon = 0.05, |\Omega| = 300$ instance; i.e., the third row of Table 2. To this end, we first let the problem solve to a MIP gap of 0%, with an indefinite time limit. The selling price, R_t, exceeds the operational cost of the battery, $C_c = C_d$, for the nine hours [13:00—22:00), see Table 1. Solar power is non-zero, for at least one scenario, only for the fourteen hours [6:00—19:00), see Fig. 2. The optimal profit for the 24 hour horizon is $ 283.79 (thus, the value reported in Table 2 was indeed optimal to a MIP gap of 0%). We provide the characteristics of the optimal solution in Table 3. No power is promised during the seven night hours from midnight to the hour 6:00 (i.e., till 6:59 AM) and the three evening hours from 20:00 to the hour 22:00 (i.e., till 22:59P M). This is because the cost of operating the battery is higher than the reward for the seven night hours, and further there is no solar power available in these hours. In the three evening hours, although the cost of operation is (slightly) less than the reward, the model still chooses to not make a promise. This is because there is no solar power, and the model prefers to completely drain the battery to its lower limit in the last hour. Most of the hours meet the energy promise largely through the solar power, which is dictated by the economics of model (2). However, there is a good synergy between the charging and discharging times of the battery; i.e., (i) the promise made does not rely entirely on the solar power, and (ii) for the same hour the model chooses to charge or discharge based on the observed solar power scenario. An extreme example is the hour 17:00—17:59. In this hour, the battery is discharged (on average) to over 80% of its discharge limit to increase profit—solar power is decreasing beyond this hour, and this hour has the

largest marginal profit (i.e., $R_t - C_c$). Also, we note that excess solar power in the afternoon hours is used to charge the battery. Hence, we see that the model is successfully using the battery to shift solar power to times when energy commands a higher price. Finally, as we mentioned before, at the last hour of 23:00—23:59, the battery is completely discharged to its permissible limit. To counter this end-effect, the model can be run in a rolling-time horizon or a boundary condition could be imposed.

Hour	Promised energy (kWh)	Average energy charged (kWh)	Average energy discharged (kWh)	Average solar energy available (kWh)
0:00				
1:00				
2:00				
3:00				
4:00				
5:00				
6:00		0.01		0.19
7:00	12.78	0.77	0.57	20.70
8:00	39.26	5.28	2.79	97.76
9:00	620.80	26.89	11.08	725.45
10:00	837.18	60.05	22.53	959.92
11:00	911.42	68.25	17.20	1042.44
12:00	1004.14	64.00	16.99	1147.96
13:00	1077.48	49.11	15.26	1183.66
14:00	946.56	80.69	7.04	1127.01
15:00	913.47	19.66	9.97	986.78
16:00	807.63	2.09	47.01	767.36
17:00	877.07		405.65	457.87
18:00	185.30	0.08	5.73	186.90
19:00	123.15		2.66	123.77
20:00				
21:00				
22:00				
23:00	480.00		480.00	

Table 3 Profile of the optimal solution for model (2) with $\varepsilon = 0.05$ and $|\Omega| = 300$. The average energy charged for hour t is $\frac{\sum_{\omega\in\Omega} p_t^\omega}{|\Omega|}$, the average energy discharged is $\frac{\sum_{\omega\in\Omega} q_t^\omega}{|\Omega|}$, and the average solar energy available is $\frac{\sum_{\omega\in\Omega} s_t^\omega}{|\Omega|}$. A blank in the table indicates zero. For details, see Section 6.2.

6.2.2 Analysis: Storage Size

Next, we analyze the impact of the size of the battery storage on the optimal profit. Similar to Section 6.2, we choose the $\varepsilon = 0.05, |\Omega| = 300$ instance. We recompute the optimal profit from model (2) by varying the maximum storage capacity of the battery, $\overline{X}$, in increments of 10%. Since we set $x_0 = 0.5\overline{X}$ and $\underline{X} = 0.2\overline{X}$ in our computational experiments, the lowest feasible limit of $\overline{X}$ is 0.5 $\overline{X}$ corresponding to a 50% decrease. Figure 3 presents the pareto-optimal curve for the percentage changes in the optimal profit by varying the maximum storage capacity of the battery. The red dot presents the base case; i.e., a profit of $ for a battery size of $\overline{X} = 960$ kWh. There is no change in the optimal profit beyond a size of 1344 kWh; this suggests that for the given parameters of the model a battery of size $\overline{X} = 1344$ kWh is sufficiently large.

6.2.3 Analysis: Quality of the Solution

In this section, we perform an out-of-sample validation to analyze the performance of the optimal power promise. Since model (2) is solved using a given set of scenarios by a so-called sample average approximation (SAA) of the "true" model, this solution is naturally sub-optimal or infeasible to a new set of

scenarios [4]. Thus, we seek to answer the following question: to what minimum reliability level can we guarantee a given power promise profile to safeguard us against new scenarios? Intuitively, we expect this reliability level to be lower than the chosen $1 - \varepsilon$ level.

To this end, we first independently sample ten batches of 300 scenarios each. We fix the promised power output to the optimal power output of the base case instance with $|\Omega| = 300$ instance. And, we then find the minimum number of violations required to ensure feasible operations (i.e., constraints (2c)-(2g)) in each of the ten batches. We construct a 95% confidence interval (CI) around the number of violations using a t-distribution. For the SAA solved with a 95% reliability regime, the CI of the true reliability is (85.1%, 89.5%). Analogously for the 99% reliability regime, the CI is (92.9%, 95.4%).

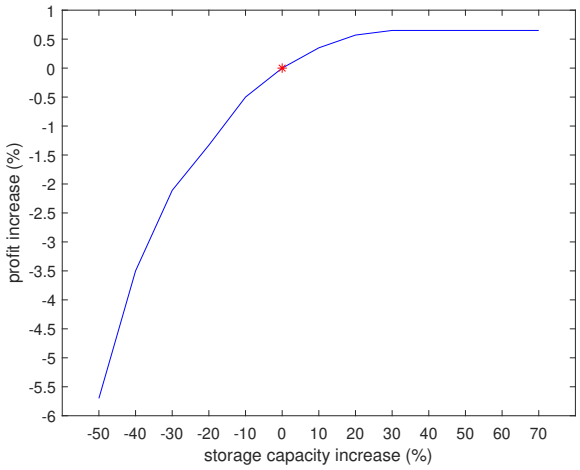

Fig. 3 Effect of varying the maximum storage capacity of the battery on the optimal profit.

6.3 Analysis of Lower Bound Heuristic

In this section, we briefly analyze the performance of the lower bounding heuristic mentioned in Section 3. Table 4 presents our results. The "LB" values are the lower bounds to model 2, see also Fig. 1. We solve all the scenario sub-problems using the Gather-Update-Solve-Scatter extension of GAMS [10]. This scheme is very efficient; 1200 scenario problems plus the fixed scenario problem can be solved in about half a minute. To compute the "Gap", we use the best known upper bound—this is available from either the naive solve (Table 3) or, from the Lagrangian relaxation that we present later (Table 5). For example, for the $|\Omega| = 1200, \varepsilon = 0.05$, the upper bound is $\min\{\frac{271.76}{1-0.095}, 284.77\}$. A "-" in the table indicates the gap is lesser than the optimality tolerance. Thus, six of the twelve problems had a gap under 1%, while two solved to optimality.

6.4 Analysis of the Lagrangian Relaxation Scheme

In this section, we attempt to solve the Lagrangian relaxation of model (2) given by model (4) using Algorithm 1. We use a maximum of 10 iterations (i.e., $\overline{iter} = 10$). We use the same maximum time limit and optimality tolerance of 2400 seconds and 0.1%, respectively, as for the results in Table 2. Theoretically, this gap might never be attained, even with a larger time limit as (i) different schemes for obtaining a lower bound would give different gaps, and (ii) even the tightest upper bound from the Lagrangian procedure might never be attained as our problem includes binary variables. However, as we show later in this section our empirical results demonstrate that in some instances an optimality gap smaller than the naive solve can be achieved. For the LB for Step 4 of Algorithm 1, we use the LB bound from Section 6.3. Further, we "warm-start" the z variables for the first LR iteration, using the optimal z values from the LB heuristic. Also, at each subgradient iteration, we warm-start the y variables with

$\|\Omega\|$	ε	LB	Gap	Time
150	0.05	280.28	0.2%	6
	0.01	273.01	-	3
300	0.05	283.33	0.2%	7
	0.01	278.17	-	4
450	0.05	273.32	3.7%	7
	0.01	276.40	0.9%	7
600	0.05	275.58	5.0%	17
	0.01	274.53	1.1%	16
900	0.05	277.79	4.1%	22
	0.01	275.69	0.4%	29
1200	0.05	277.05	2.7%	31
	0.01	267.98	2.0%	35

Table 4 Computational results for lower bounding heuristic. The "Gap" denotes the optimality gap of LB from $\min\{\text{LD}, \overline{z^*}\}$; see, Fig. 1 for the notation. For details, see Section 6.3.

their optimal value from the previous subgradient iteration. In CPLEX, this can be done by setting the `mipstart` parameter to 1.

Table 5 presents our results. All 10 subgradient iterations were completed in the required time limit, except for $|\Omega| = 900, 1200$. In our experiments, if ten LR iterations were performed within the time limit, the optimal Lagrangian dual (denoted by "LR") is typically achieved around the sixth iteration. The four instances ($\varepsilon = 0.05, |\Omega| = 600, 900, 1200$ and $\varepsilon = 0.01, |\Omega| = 1200$) that could not be solved to optimality in Table 3 all have smaller gaps in Table 5, suggesting the Lagrangian relaxation is better suited for larger problems. Also note, that for these four instances, only a few subgradient iterations could be completed. This again lends credence to the observation that good Lagrangian bounds are obtained fairly early in the iterations.

The optimal solution provided by the Lagrangian relaxation might not even be feasible to model (2). Thus, the algorithm is useful only for providing good bounds to model (2), and not for obtaining high quality feasible solutions. At best, the optimal y variables from the last iteration of Algorithm 1 could be used to warm-start model (2). This aspect of Lagrangian decomposition techniques is a very well-known drawback; researchers have addressed this issue in a number of ways [60, 65]. A similar issue is encountered in Step 9 of Algorithm 2 as well; model (4) could be infeasible. However, if $\overline{y}_t \leq \min\{\overline{Q}, \eta(\overline{X} - \underline{X})\} + s_t^{\omega(\lfloor N\varepsilon \rfloor + 1, t)}, \forall t \in T$, then it follows from Proposition 3 that model (4) is assuredly feasible.

$\|\Omega\|$	ε	Bounds			Performance	
		LB	LR	Gap	LR Time	LR iterations
150	0.05	280.28	284.80	1.6%	56	10
	0.01	273.01	273.01	-	8	3
300	0.05	283.33	291.59	2.8%	272	10
	0.01	278.17	279.61	0.5%	210	10
450	0.05	273.32	291.09	3.9%	980	10
	0.01	276.40	279.14	1.0%	660	10
600	0.05	275.58	290.15	5.0%	1936	10
	0.01	274.53	277.98	1.2%	1581	10
900	0.05	277.79	289.68	4.1%	2369	2
	0.01	275.69	276.83	0.4%	2362	4
1200	0.05	277.05	284.77	2.7%	2356	3
	0.01	267.98	273.37	2.0%	2351	4

Table 5 Computational results for model (2) using Algorithm 1. "LB" and "LR" refer to lower bound and Lagrangian relaxation, respectively. The "Gap" column is the relative gap between the LB and LR. We use a maximum time limit of 2400 seconds (cumulative sum of LP, LB and LR) and a gap of 0.1%. For details, see Section 6.4.

6.5 Analysis of the Progressive Hedging Heuristic

The performance of the Lagrangian relaxation algorithm in Section 6.4 suggests little room for improving the optimality gaps. In this section, we pursue a different approach, specifically aimed at achieving fast bounds albeit of lower quality. To this end, we also analyze the advantages and disadvantages of using the PH heuristic presented in Section 4 and Algorithm 2. As we describe in Section 4, PH offers theoretical difficulties, however often they are marred by the algorithm's practical performance. This makes PH a powerful heuristic for practical problems; similar trends have been reported earlier [46,55].

6.5.1 Choice of ρ

Algorithm 2 is known to be very sensitive to the choice of the parameter ρ [15,55]; no consensus seems to exist in the literature for deciding optimal values of ρ. We compare the sensitivity of our results using three values of ρ_t, in addition to the $\rho_t = 1$ results reported above. In the first two runs, we use constant ρ_t values of 0.1 and 10, and in the third run we use a coefficient specific ρ value described in [55]. This value is $\rho_t = \frac{R_t}{\max\{1, \sum_{\omega\in\Omega}(\overline{y}_t - y_t^\omega)/\Omega\}}$, where the y values are those obtained in Steps 1 and 2 of Algorithm 2. We observed the following trends: larger values of ρ result in weaker upper bounds of Step 6 of Algorithm 2, but close the gap in Step 7 faster. This is similar to the trend observed in [15]. However, $\rho_t = 0.1$ resulted in weaker upper bounds than $\rho_t = 1$. The variable specific ρ closes the gap fastest, however the improvements compared to $\rho_t = 10$ were minimal. Further, none of the ρ values could completely close the gap; and, $\rho_t = 1$ provided the best upper bounds.

6.5.2 Linearization

The quadratic proximal term in the PH objective can be easily linearized using piece-wise linear segments. While, there are many sophisticated techniques for this linearization [20,52], we attempted a basic first-order Taylor expansion. Dropping the t subscript, if we use $L-1$ segments, we have: $y^2 \approx a^2 + 2a(y-a)$, where $a = \frac{(l-1)y_{\max}}{L-1}, l = 1, 2, \ldots, L$. However, in our experiments, we observed that to achieve solutions comparable to that of the original mixed-integer quadratic constrained program (MIQCP), a large value for L—close to 100—is needed. Then, the time required to solve the larger MIPs is no significantly different from that of the MIQCP. The reason for this is as follows.

There is only a single binary variable for each subproblem in Step 6 of Algorithm 2; thus, each MIP can be solved in at most two LP solves (by branching on $z^\omega = 0$ and $z^\omega = 1$). When L increases, the time required for an LP solve increases due to the increased number of constraints in the LP. Poor values of y reported by PH cascade to poor values for z in Step 9 of the algorithm, resulting in poor values for γ, and ultimately poor performance for the Lagrangian dual. Two possible suggestions for the interested reader, that we do not investigate, are as follows. First, use a dynamic linearization as described in [52] that gradually increments the number of piecewise segments. Second, use a subgradient step-size update rule that does not depend on γ.

6.5.3 Convergence of Algorithm 2: Bundling

The number of PH iterations required for each subgradient iteration, $\overline{iter}$, can be several hundreds to force PH to converge. Although each iteration involves only a single binary variable, this can make the PH algorithm computationally inefficient. In our computational experiments, we did not find PH to converge, using the criteria in Step 7 of Algorithm 2, even with $\overline{iter} = 100$. As the number of scenarios grows, the time to solve the MIQCPs in Step 6 grows as well; solving MIQCPs can be significantly more challenging than solving MIPs. However, we noticed that nearly always the best upper bound, PH-UB, was obtained in the first iteration itself. To this end, we begin our discussion by reporting results obtained by setting $\overline{iter} = 1$; i.e., without even entering the `while` loop of Algorithm 2. Table 6 reports these results. The PH upper bounds are on average 10% (and, at most 11.7%) far from the optimal Lagrangian values (the fourth column of Table 5). For the instances that could be solved optimally in Table 2, the PH-UB values are on average 10.9% (and, at most 11.7%) far from the true optimal solutions. These bounds can be obtained very quickly, see column "Time" in Table 6; the 1200 scenario instances could be solved in under half a minute.

$\|\Omega\|$	ε	PH-UB	Time
150	0.05	317.24	3
	0.01	309.33	3
300	0.05	320.29	5
	0.01	311.36	5
450	0.05	319.24	8
	0.01	310.32	8
600	0.05	318.60	10
	0.01	309.93	10
900	0.05	317.96	5
	0.01	308.99	16
1200	0.05	317.37	15
	0.01	308.56	23

Table 6 Computational results for model (4) using Algorithm 2 with $\overline{iter} = 1$. "PH-UB" refers to the upper bound obtained using the progressive hedging algorithm. See Section 4 for details.

However, if Step 4 of Algorithm 1 is replaced with Algorithm 2 without the `while` loop, the quality of the Lagrangian relaxation scheme deteriorates significantly due to the cascading effect we describe in Section 6.5.2. Again, there is a tradeoff between computational effort required (a large value for $\overline{iter}$) versus the quality of the solution (a value close to zero for γ). To this end, we investigated the following enhancement in the PH algorithm.

Instead of solving individual scenario sub-problems, we bundle together several scenarios and solve this mini-extensive form problem as our new sub-problems. Bundling scenarios could offer the advantage of speeding convergence of PH, but with increased computational effort per PH iteration. For details on implementation of bundling in progressive hedging, see, e.g. [15]. Further, the bounds obtained in Step 6 of Algorithm 2 also extend to the case with bundling [15]. Finally, we emphasize again that since the PH algorithm is a heuristic for solving model (4), the bounds obtained would be worse than those obtained using a naive solve (Table 5); however, we are interested in the tradeoff between speed and quality.

$\|\Omega\|$	ε	size = 100				size = 150			
		LR	Gap	Time	LR iterations	LR	Gap	Time	LR iterations
150	0.05	289.32	3.1%	79	10	284.80	1.6%	68	10
	0.01	280.79	2.8%	115	6	273.01	0.0%	9	3
300	0.05	293.94	3.6%	369	10	293.55	3.5%	251	10
	0.01	284.18	2.1%	95	2	281.66	1.2%	242	6
450	0.05	293.39	4.6%	537	10	292.26	4.3%	487	10
	0.01	283.43	4.8%	229	2	280.31	3.7%	299	10
600	0.05	292.50	5.8%	576	10	291.47	5.5%	279	10
	0.01	282.23	2.7%	203	1	279.30	1.7%	409	10
900	0.05	295.06	5.9%	676	4	292.47	5.0%	870	4
	0.01	283.70	2.8%	611	2	281.14	1.9%	613	2
1200	0.05	290.78	4.7%	781	2	289.21	4.2%	896	2
	0.01	277.78	3.5%	887	2	277.61	3.5%	747	2

Table 7 Computational results for model (2) using Algorithm 2 with Algorithm 1 using bundling. Here, "size" indicates the size of a bundle, "LR" indicates the Lagrangian relaxation bound obtained using the PH algorithm, and "Gap" is the relative gap between the LB column of Table 5 and the "LR".

Table 7 compares the results of our experiments using two different bundles. We use Algorithm 1 with Step 3 substituted by Algorithm 2. Further, to see the tradeoff between the computational time and the value of the optimal solution, we set a maximum time limit of 900 seconds. When the completion of a LR iteration would have resulted in the total time being greater than a 900 seconds, we do not report

it. For example, for the 900 scenario, $\varepsilon = 0.01$ instance if the third iteration were to complete the total time would be greater than 900 seconds; thus, we report results only for two iterations. In this sense, our results are conservative. For the 150 and 450 scenario instances, when we use bundles of size 100, there are respectively one and four bundles of size 100 and one each of size 50. In all of the 12 instances, the 150 size bundles lead to better (lower) Lagrangian bounds than the 100 size bundles. However, there is no consistent trend in the time required for completion. For example, the 450 scenario instance with $\varepsilon = 0.05$ took lesser time for the 150 bundle case than the 100 bundle case; but, the effect was the opposite for the 450 scenario instance with $\varepsilon = 0.01$.

6.5.4 Summary of Bounds

$\|\Omega\|$	ε	LP		LR		LR-PH	
		Gap	Time	Gap	Time	Gap	Time
150	0.05	6.6%	1	1.4%	56	1.4%	68
	0.01	4.5%	1	-	8	-	9
300	0.05	6.8%	1	2.7%	272	3.3%	251
	0.01	4.4%	1	0.4%	210	1.1%	242
450	0.05	6.7%	2	2.5%	980	2.9%	487
	0.01	3.8%	2	-	660	0.5%	299
600	0.05	7.5%	3	3.3%	1936	3.7%	279
	0.01	4.2%	3	-	1581	0.6%	409
900	0.05	8.0%	9	3.8%	2369	4.7%	870
	0.01	4.4%	9	-	2362	1.5%	613
1200	0.05	8.2%	10	2.7%	2356	4.2%	896
	0.01	5.9%	9	0.9%	2351	2.4%	747

Table 8 Comparison of different upper bounds for model (2). "LP" denotes the linear programming relaxation, "LR" denotes the Lagrangian relaxation of model (4), and "LR-PH" denotes the Lagrangian relaxation using progressive hedging algorithm with bundling. The "Gap" columns denote the optimality gaps from $\max\{\underline{z}^*, LB\}$; see, Fig. 1 for the notation. The "LR-PH" has a maximum time limit of 900 seconds. For details, see Section 6.

From the preceding discussions, we have two valid upper bounds for model (2) available from plain LR and LR with PH. A third bound is obtained by solving the LP relaxation of model (2). Table 8, we summarize these bounds. The "Gap" here is the optimality gap of the bound from $\max\{\underline{z}^*, \text{LB}\}$. The LP has a gap of 5.9% on average. From the LR column, we note that four of the twelve instances solved to optimality—but this optimality could not be verified (see Table 5). Imposing a time limit and solving the LR with the bundled PH algorithm increases this optimality gap by at most 1.5% (with bundles of 150 scenarios). This again indicates that the PH algorithm achieves good bounds in fairly early iterations.

7 Conclusion

We developed a stochastic optimization model to schedule a standalone hybrid system consisting of battery storage with solar power. We used a chance-constrained formulation to ensure highly reliable operations under the uncertainty of solar power. The model makes charging/discharging decisions dictated by a mix of economics and the available solar power. For a few hundred scenarios, the model is relatively tractable benefiting from a strong LP relaxation. For larger number of scenarios, we demonstrate the applicability of a simple Lagrangian relaxation procedure. The procedure is remarkably effective, and tightens the optimality gap when a naive solve could not. Finally, we present a heuristic, based on the progressive hedging algorithm, that we couple with the Lagrangian relaxation. We find significant benefit when scenarios are bundled and solved as one in the progressive hedging iterations; however, when solved plainly the PH heuristic does not achieve good quality solutions. Future work could examine interconnected updates of the subgradient algorithm with the progressive hedging algorithm.

Acknowledgements The first author thanks David Pozo for discussions on the early stages of this manuscript and Frederik Fiand for assistance with the GUSS implementations. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government. SAND2019-5783 J.

References

1. Agnew, N., Agnew, R., Rasmussen, J., Smith, K.: An application of chance constrained programming to portfolio selection in a casualty insurance firm. Management Science **15**(10), B–512 (1969)
2. Ahmed, S.: Two-stage stochastic integer programming: A brief introduction. Wiley Encyclopedia of Operations Research and Management Science (2010)
3. Ahmed, S., Luedtke, J., Song, Y., Xie, W.: Nonanticipative duality, relaxations, and formulations for chance-constrained stochastic programs. Mathematical Programming **162**(1-2), 51–81 (2017)
4. Ahmed, S., Shapiro, A.: Solving chance-constrained stochastic programs via sampling and integer programming. In: State-of-the-art decision-making tools in the information-intensive age, pp. 261–269. Informs (2008)
5. Bahramirad, S., Reder, W., Khodaei, A.: Reliability-constrained optimal sizing of energy storage system in a microgrid. IEEE Transactions on Smart Grid **3**(4), 2056–2062 (2012)
6. Bai, X., Sun, J., Sun, X., Zheng, X.: An alternating direction method for chance-constrained optimization problems with discrete distributions. Technical report, School of Management, Fudan University (2012)
7. Bekele, G., Palm, B.: Feasibility study for a standalone solar–wind-based hybrid energy system for application in Ethiopia. Applied Energy **87**(2), 487–495 (2010)
8. Bhuiyan, M., Asgar, M.A., Mazumder, R., Hussain, M.: Economic evaluation of a stand-alone residential photovoltaic power system in Bangladesh. Renewable Energy **21**(3-4), 403–410 (2000)
9. Bilal, B.O., Sambou, V., Ndiaye, P., Kébé, C., Ndongo, M.: Optimal design of a hybrid solar–wind-battery system using the minimization of the annualized cost system and the minimization of the loss of power supply probability (LPSP). Renewable Energy **35**(10), 2388–2390 (2010)
10. Bussieck, M.R., Ferris, M.C., Lohmann, T.: GUSS: Solving collections of data related models within GAMS. In: Algebraic Modeling Systems, pp. 35–56. Springer (2012)
11. Carrión, M., Arroyo, J.M.: A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem. IEEE Transactions on Power Systems **21**(3), 1371–1378 (2006)
12. Clare, G., Richards, A.: Air traffic flow management under uncertainty: Application of chance constraints. In: Proceedings of the 2nd International Conference on Application and Theory of Automation in Command and Control Systems, pp. 20–26. IRIT Press (2012)
13. Conejo, A.J., Arroyo, J.M., Contreras, J., Villamor, F.A.: Self-scheduling of a hydro producer in a pool-based electricity market. IEEE Transactions on Power Systems **17**(4), 1265–1272 (2002)
14. Electric Reliability Council of Texas, Inc (ERCOT): Market prices. `http://www.ercot.com/mktinfo/prices` (2016). Accessed: Feb 17, 2019
15. Gade, D., Hackebeil, G., Ryan, S.M., Watson, J.P., Wets, R.J.B., Woodruff, D.L.: Obtaining lower bounds from the progressive hedging algorithm for stochastic mixed-integer programs. Mathematical Programming **157**(1), 47–67 (2016)
16. Giraud, F., Salameh, Z.M.: Steady-state performance of a grid-connected rooftop hybrid wind-photovoltaic power system with battery storage. IEEE Transactions on Energy Conversion **16**(1), 1–7 (2001)
17. Golestaneh, F., Gooi, H.B., Pinson, P.: Generation and evaluation of space–time trajectories of photovoltaic power. Applied Energy **176**, 80–91 (2016)
18. Guan, X., Xu, Z., Jia, Q.S.: Energy-efficient buildings facilitated by microgrid. IEEE Transactions on Smart Grid **1**(3), 243–252 (2010)
19. Held, M., Wolfe, P., Crowder, H.P.: Validation of subgradient optimization. Mathematical Programming **6**(1), 62–88 (1974)
20. Helseth, A.: Stochastic network constrained hydro-thermal scheduling using a linearized progressive hedging algorithm. Energy Systems **7**(4), 585–600 (2016)
21. Henrion, R., Strugarek, C.: Convexity of chance constraints with independent random variables. Computational Optimization and Applications **41**(2), 263–276 (2008)
22. Ito, M., Kato, K., Sugihara, H., Kichimi, T., Song, J., Kurokawa, K.: A preliminary study on potential for very large-scale photovoltaic power generation (VLS-PV) system in the Gobi desert from economic and environmental viewpoints. Solar Energy Materials and Solar Cells **75**(3-4), 507–517 (2003)
23. Jenkins, D., Fletcher, J., Kane, D.: Lifetime prediction and sizing of lead–acid batteries for microgeneration storage applications. IET Renewable Power Generation **2**(3), 191–200 (2008)
24. Kargarian, A., Fu, Y., Wu, H.: Chance-constrained system of systems based operation of power systems. IEEE Transactions on Power Systems **31**(5), 3404–3413 (2016)
25. Knueven, B., Ostrowski, J., Ollis, B., Irminger, P., Starke, M., Herron, A., King, D., Xiao, B., Xue, Y., Karlson, P., et al.: Economic feasibility analysis and operational testing of a community energy storage system. In: Energy Conversion Congress and Exposition (ECCE), 2016 IEEE, pp. 1–5. IEEE (2016)
26. Lasseter, R.H.: Microgrids. In: Power Engineering Society Winter Meeting, 2002. IEEE, vol. 1, pp. 305–308. IEEE (2002)
27. Li, S.: An insurance and investment portfolio model using chance constrained programming. Omega **23**(5), 577–585 (1995)

28. Lima, R.M., Conejo, A.J., Langodan, S., Hoteit, I., Knio, O.M.: Risk-averse formulations and methods for a virtual power plant. Computers & Operations Research **96**, 350–373 (2018)
29. Liu, X., Küçükyavuz, S., Luedtke, J.: Decomposition algorithms for two-stage chance-constrained programs. Mathematical Programming **157**(1), 219–243 (2016)
30. Løkketangen, A., Woodruff, D.L.: Progressive hedging and tabu search applied to mixed integer (0, 1) multistage stochastic programming. Journal of Heuristics **2**(2), 111–128 (1996)
31. Luedtke, J.: A branch-and-cut decomposition algorithm for solving chance-constrained mathematical programs with finite support. Mathematical Programming **146**(1-2), 219–244 (2014)
32. Luedtke, J., Ahmed, S., Nemhauser, G.L.: An integer programming approach for linear programs with probabilistic constraints. Mathematical Programming **122**(2), 247–272 (2010)
33. Ma, T., Yang, H., Lu, L.: A feasibility study of a stand-alone hybrid solar–wind–battery system for a remote island. Applied Energy **121**, 149–158 (2014)
34. Ma, T., Yang, H., Lu, L., Peng, J.: Technical feasibility study on a standalone hybrid solar-wind system with pumped hydro storage for a remote island in Hong Kong. Renewable Energy **69**, 7–15 (2014)
35. Morales, J.M., Conejo, A.J., Madsen, H., Pinson, P., Zugno, M.: Virtual power plants. In: Integrating Renewables in Electricity Markets, pp. 243–287. Springer (2014)
36. Ostrowski, J., Anjos, M.F., Vannelli, A.: Tight mixed integer linear programming formulations for the unit commitment problem. IEEE Transactions on Power Systems **27**(1), 39–46 (2012)
37. Ozturk, U.A., Mazumdar, M., Norman, B.A.: A solution to the stochastic unit commitment problem using chance constrained programming. IEEE Transactions on Power Systems **19**(3), 1589–1598 (2004)
38. Parida, B., Iniyan, S., Goic, R.: A review of solar photovoltaic technologies. Renewable and Sustainable Energy Reviews **15**(3), 1625–1636 (2011)
39. Patel, M.R.: Wind and solar power systems: design, analysis, and operation. CRC press (2005)
40. Polyak, B.T.: A general method for solving extremal problems. In: Doklady Akademii Nauk, vol. 174, pp. 33–36. Russian Academy of Sciences (1967)
41. PowerTech Systems: Lithium-ion vs lead-acid cost analysis. `https://www.powertechsystems.eu/home/tech-corner/lithium-ion-vs-lead-acid-cost-analysis/` (2015). Accessed: May 5, 2019
42. Prékopa, A.: Stochastic Programming. Kluwer Academic Publishers (1995)
43. Rockafellar, R.T., Wets, R.J.B.: Scenarios and policy aggregation in optimization under uncertainty. Mathematics of Operations Research **16**(1), 119–147 (1991)
44. Rosenthal, R.E.: GAMS–A User's Guide. ALS-NSCORT (2004)
45. Ruszczyński, A., Shapiro, A.: Optimization of convex risk functions. Mathematics of Operations Research **31**(3), 433–452 (2006)
46. dos Santos, M.L., da Silva, E.L., Finardi, E.C., Gonçalves, R.E.: Practical aspects in solving the medium-term operation planning problem of hydrothermal power systems by using the progressive hedging method. International Journal of Electrical Power & Energy Systems **31**(9), 546–552 (2009)
47. Saurorja = Solar Energy: Lead acid is the cheapest battery: Conditions apply. `https://saurorja.org/2011/08/30/lead-acid-is-the-cheapest-battery-conditions-apply/` (2011)
48. Shor, N.: The rate of convergence of the generalized gradient descent method. Cybernetics and Systems Analysis **4**(3), 79–80 (1968)
49. Singh, B.: Optimal spatiotemporal resource allocation in public health and renewable energy. Ph.D. thesis, The University of Texas at Austin (2016)
50. Singh, B., Morton, D.P., Santoso, S.: An adaptive model with joint chance constraints for a hybrid wind-conventional generator system. Computational Management Science pp. 1–20 (2017)
51. Singh, B., Pozo, D.: A guide to solar power forecasting using ARMA models. In: 2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe), pp. 1–4 (2019). DOI 10.1109/ISGTEurope.2019.8905430
52. Veliz, F.B., Watson, J.P., Weintraub, A., Wets, R.J.B., Woodruff, D.L.: Stochastic optimization models in forest planning: A progressive hedging solution approach. Annals of Operations Research **232**(1), 259–274 (2015)
53. Waller, S.T., Ziliaskopoulos, A.K.: A chance-constrained based stochastic dynamic traffic assignment model: Analysis, formulation and solution algorithms. Transportation Research Part C: Emerging Technologies **14**(6), 418–427 (2006)
54. Watson, J.P., Wets, R.J., Woodruff, D.L.: Scalable heuristics for a class of chance-constrained stochastic programs. INFORMS Journal on Computing **22**(4), 543–554 (2010)
55. Watson, J.P., Woodruff, D.L.: Progressive hedging innovations for a class of stochastic mixed-integer resource allocation problems. Computational Management Science **8**(4), 355–370 (2011)
56. Wu, H., Shahidehpour, M., Li, Z., Tian, W.: Chance-constrained day-ahead scheduling in stochastic power system operation. IEEE Transactions on Power Systems **29**(4), 1583–1591 (2014)
57. Wu, L.: A tighter piecewise linear approximation of quadratic cost curves for unit commitment problems. IEEE Transactions on Power Systems **26**(4), 2581–2583 (2011)
58. Xu, H., Caramanis, C., Mannor, S.: Optimization under probabilistic envelope constraints. Operations Research **60**(3), 682–699 (2012)
59. Yang, H., Zhou, W., Lu, L., Fang, Z.: Optimal sizing method for stand-alone hybrid solar–wind system with LPSP technology by using genetic algorithm. Solar Energy **82**(4), 354–367 (2008)
60. Yunjun, H., Xiangdong, Y., Dan, W.: Lagrangian relaxation based feasible solution algorithm. In: 2012 24th Chinese Control and Decision Conference (CCDC), pp. 875–878. IEEE (2012)
61. Zhang, D., Shah, N., Papageorgiou, L.G.: Efficient energy consumption and operation management in a smart building with microgrid. Energy Conversion and Management **74**, 209–222 (2013)
62. Zhao, B., Zhang, X., Chen, J., Wang, C., Guo, L.: Operation optimization of standalone microgrids considering lifetime characteristics of battery energy storage system. IEEE Transactions on Sustainable Energy **4**(4), 934–943 (2013)
63. Zhao, B., Zhang, X., Li, P., Wang, K., Xue, M., Wang, C.: Optimal sizing, operating strategy and operational experience of a stand-alone microgrid on Dongfushan island. Applied Energy **113**, 1656–1666 (2014)
64. Zhou, W., Lou, C., Li, Z., Lu, L., Yang, H.: Current status of research on optimum sizing of stand-alone hybrid solar–wind power generation systems. Applied Energy **87**(2), 380–389 (2010)
65. Zhuang, F., Galiana, F.D.: Towards a more rigorous and practical unit commitment by lagrangian relaxation. IEEE Transactions on Power Systems **3**(2), 763–773 (1988)